An Arabic language resource for computational morphology based on the Semitic model

Alexis Neme

UNIVERSITÉ PARIS-EST

École doctorale MSTIC
Mathématiques — Sciences et Techniques de l'Information et de la Communication

Thèse de doctorat
Spécialité Informatique

An Arabic language resource for computational morphology based on the Semitic model

Alexis Amid Neme

soutenue le 1er juillet 2020

devant le jury composé de

Mourad Abbas	Président et Examinateur
Tita Kyriacopoulou	Examinatrice
Eric Laporte	Directeur de thèse
Denis Maurel	Rapporteur
Alexis Nasr	Rapporteur

Laboratoire d'informatique Gaspard-Monge
UMR 8049 LIGM

Contents

Presentation

'*The need for incorporating linguistic knowledge is a major challenge in Arabic Data-driven MT (Machine Translation). Recent attempts to build data-driven systems to translate from and to Arabic have demonstrated that the complexity of word and syntactic structure in this language prompts the need for integrating some linguistic knowledge and with a minimum cost since the amount of linguistic resources added has consequences for computational complexity and portability*' (Zbib, Soudi, 2012:2).

We do agree with this quotation. We are perceiving an advancement lately in software dedicated to Arabic language technology based on statistical or rule-based approaches; better accuracy in Arabic linguistic knowledge will improve the output of such software. The needs expressed by Zbib and Soudi are repeated periodically in papers, conferences, and presentations, and since 2000: *"Arabic spell checking is an active area of research since results are not satisfactory."* (Shaalan Kh. *et al*., 2003) and the state-of-the-art did not improve enough according to the author (Shaalan Kh. et al., 2012). Similar complaints about Hebrew are uttered by Wintner, (2008): "*However, when wide coverage morphological grammars are considered, finite-state technology does not scale up well*." Statistical approaches applied to Germanic and Romance languages yield a better output than Arabic. So, the problem of Semitic languages might be not in software development, but elsewhere, mainly in a misconception of lexical resources.

Still in 2016, the mainstream projects for Arabic lexicon are based on multi-stem approaches and more specifically on the BAMA (2002) lexicon or on resources derived from it, as the Arabic TreeBank (Maamouri and al, 2004) in Pennsylvania, or the resources for MADA+TOKAN (Habash and al., 2009)[1], at Columbia University. "*Any formal representation that is not adapted to Semitic morphology will be rejected by the majority of Arabic-speaking linguists. Many computational representations have been proposed based on the Semitic model, others were newly created. However, when linguists use a newly created formalism, they continue to work with the traditional root-and-pattern representation and subsequently, they unfold their descriptions for a specific formalism.*" (Neme, 2011). In fact, it is very challenging for a linguist or a computer scientist to update the BAMA lexicon.

Actually, a lot of software relies on the BAMA morphological tagging (or SAMA, 2004, its successor), which is complemented in the best case by *"a morphological backoff procedure"* (Habash and al., 2009). Habash's team at Columbia University undertook to create their own Arabic resource in the MAGEAD (2005) project based on finite-state technologies, but their last paper consistent with this attempt was published in 2011 (Altantawy and al., 2011), and the later papers of the same team use BAMA. The breakthrough of BAMA (2002) opened a range of possible applications for Arabic. Since then, the needs are expressed periodically for a better morphological analysis tagging, but no team was able to propose a more viable and operational

[1] In all metric aspects, the newer MADAMIRA (2014) represents a deterioration of accuracy compared to MADA.

solution than BAMA and able to deal with requirements of the Arabic Natural Language Processing, and more particularly of the implementation of the Semitic Model.

A natural path for Arabic morphology consists in adopting or adapting both the traditional Semitic model and finite-state technologies. On the one hand, we have to facilitate the linguist's tasks of lexical encoding by proposing a familiar formalism: the Semitic model for morphology[2]. On the other hand, computer scientists, in general, point to FSTs as standard devices for inflection; and FSTs have shown their simplicity and efficiency in inflectional morphology for European languages. Nevertheless, there are countless complexities in the implementation of this model with such a technique. This is due to the richness of Arabic morphology and to the actual details of the traditional root-and-pattern model. In fact, there is an opposition between the requirement to be faithful to the essence of the Semitic model, for the sake of lexicon encoders, and the necessity to curb the complexity of its traditional version. Yet, no trade-off has been found.

Indeed, we have achieved and created from scratch a lexical resource containing 76,000 lemmatized entries, fully vowelized and manually encoded for inflectional morphology, representing more than 6 million inflected forms based on Semitic morphology and using finite-state technologies. Our resources are comprehensive, straightforward, accurate, and easy to update for a native linguist.

The availability of such Arabic linguistic resources[3] is a significant advantage for data-driven or rule-based applications. For example, usual utilities for pattern matching typically apply regular expressions on texts; our resource offers more facilities. We are able to describe large classes of forms using simple patterns: for instance, the lexical entry of a particular adjective may locate all its variations, 54 forms partially or fully vowelized, or only the feminine plural ones, for instance.

Publications and applications

In 2011, we published "*A lexicon of Arabic verbs constructed on the basis of Semitic taxonomy and using finite-state transducers*", in which we explore the morphology of 15,400 verbs. In 2013, we generalized the model intended to verbs to broken plurals, and we published "*Pattern-and-root inflectional morphology: the Arabic broken plural*". In 2019, we published "*Restoring Arabic vowels through omission-tolerant dictionary lookup*", where we specify two dozen rules to deal with these optional orthographic symbols and an algorithm to recognize fully or partially vowelized forms and restoring them; we summarize Arabic-Unitex: entry counts by

[2] Even the encoding in the traditional root-and-pattern model is problematic: «chaotic» for broken plurals and to a lesser level for verbs (Neme and Laporte, 2013).

3 Our Arabic resources are not public; however, you may download from the Unitex site two tagged corpora as samples: one is dedicated to locating broken plurals (cf. Neme and Laporte, 2013) and contains three documents totalling 3,550 tokens (about 10 pages); and the other is dedicated to a prototype to exemplify a local grammar that identifies <Minister_Portfolio> and <Title_Name_Surname> in sentences beginning with "the minister said" (cf. appendix).

POS, lexical coverage (over 99%), parsing speed (5,000 tokens/second, and over 200,000 tokens/second if the dictionaries are preloaded in RAM); and we present the reasons why most previous technical attempts to create a comprehensive resource for Arabic were imprecise, whereas our approach based on the reversal of the Semitic model (see below, The traditional Semitic model of Arabic morphology) was successful in building comprehensive lexical resources, accurate and easy to update. Differing from the tradition, our model excludes derivational morphology from its representation; but similar to this tradition, it describes inflection independently from agglutination, which is sound for native-speaker intuitions.

In 2014, we also published in a Moroccan journal "*Why Microsoft Arabic Spell-checker is ineffective*" where we evidence an unsystematic and arbitrary lexical coverage of Arabic language resources in MS spell checker; this pinpoints the absence of a clear definition of a lexical entry and an inadequate design of the related agglutination rules in Microsoft Office 2007 Arabic resources. Noticing that a fully inflected dictionary will be useful for a spell checker application, we adjusted the resources of Neme (2011), and published in 2014 in an Algerian Journal "*A fully inflected Arabic verb resource constructed from a lexicon of lemmas by using finite-state*" where we describe a fully inflected lexicon of 2.5 million verbal forms generated by using finite-state transducers.

We have also set up a web site with an Arabic conjugation application and an Arabic spell checker (see Appendix, babelarab.univ-mlv.fr).

The Arabic resources in the UNITEX platform

In order to take into account all aspects of the rich morphology of Arabic, we have identified 1,000 inflectional paradigms or classes implemented with FSTs devices. These classes were divided into verbal taxonomies, nominal/adjectival broken-plural taxonomies, sound-plural taxonomies, and others.

The 76,000 encoded lemmata are inflected by using these 1,000 FSTs, producing a fully inflected lexicon with 6 million forms. The fully inflected resource is extended by agglutination grammars in order to identify words composed of up to 5 segments, agglutinated around a core inflected verb, noun, adjective, or particle. The agglutination grammars extend the recognition to more than 500 million valid delimited word forms (DWF). These resources are described in Neme and Paumier (2019). The contribution of Paumier was the implementation and the adjustment of tools in the C/C++ core engine of Unitex for Semitic inflection, the lookup algorithm to handle partial vowelization, and the algorithm for Semitic compression.

The flat file size of the encoded and fully inflected dictionary of 6 million forms is 340 megabytes (UTF-16). It is compressed then into 10 Mb before loading to memory for fast retrieval. The generation, compression, and minimization of the full-form lexicon take less than one minute on a MacBook. The speed of tagging is 5,000 words/second without any specific optimization for Arabic. We have tested our resource on unrestricted text extracted from standard Arabic online newspapers, and the lexical coverage rate is more than 99%.

Our original approach sheds new light on Arabic traditional morphology and brings new concepts in Semitic lexicography, lexicology, and morphology. As a contribution to computational morphology, we propose a new methodology in order to handle the rich and complex Semitic languages based on Finite-State good practices.

The traditional Semitic model of Arabic morphology

So far, there is no comprehensive and accurate Arabic resource for computational morphology. Since 1990, several teams of computer scientists have implemented the traditional model of Arabic morphology in systems of Natural Language Processing (NLP) without questioning its aims, assumptions, or the definitions of its key concepts. Medieval grammarians and lexicographers had designed Arabic morphology and lexicography for human minds tooled up with paper, whereas we should design Arabic computational morphology for humans equipped with processors and memory devices. This technological shift requires adapting the model of Arabic morphology.

The aim of forerunners of grammar in the eighth century was to discover the features of the Arabic language, and they had political and religious incentives. These pioneers accumulated knowledge in semantics, syntax, morphology, phonology, and lexicography, produced fabulous inventories in order to standardize the language, generating the massive grammatical production of that time. Teaching for native and non-native speakers probably soon became an urgent goal due to geographical expansion. Language teaching has always been focused on vocabulary, word meaning, and text understanding.

As for other Semitic languages, Arabic morphology was established around the ***abstract notion of root***, three consonants representing a meaning, whether precise or vague. The traditional derivational morphology based on the root-and-pattern model focuses on this abstract consonantal root. In this model, each word is represented by the combination of a root and a pattern, such as *kitaAb* = [ktb & 1i2aA3] *(kitaAb* "book" *كتاب)*. A pattern is a discontinuous affix (or transfix), made of vowels and non-radical consonants inserted around slots for the root consonants. To each pattern, traditional grammar associates a morphological category and/or inflectional features, and/or semantic features such as agent (*kaAtib* = [ktb & **1aA2i3**], *"*writer*",* *كاتب)*, patient *(makotuwb* = [ktb & **ma1o2uw3**], *"*letter*",* *مكتوب)*, instrument *(makotab* = [ktb & **ma1o2a3**], *"*office table*",* *مكتب)*, place *(makotabap* = [ktb & **ma1o2a3ap**] , *"*library*",* *مكتبة)*, etc. Such formalisation is used by traditional grammar to describe both derivational ("write"/"book"/"writer"/"letter"/"library") and inflectional ("write"/"wrote"/"written") morphology.

For pedagogical purposes, such approximate and "elegant" formalisation is fine to describe word formations and associated meanings. Including such an "elegant" description in a computational representation is tempting, but not fit for a systematic description. Assuming that we have the inventory of roots and patterns (around 5,000 roots and 400 patterns, according to Beesley, (2001), thus, a lemma in the vocabulary is defined by a root and a pattern. Theoretically, we may have up to 2 million lemmata, a maximum which is obviously not reached, and one cannot predict for a given root the subset of applicable patterns; even if a pattern exists, its meaning is

unpredictable and irregular. So, with the root-and-pattern model, Arabic tradition includes word formation in its representation, which is one more level of complexity in the linguistic description, and which is absent in the computational formalisation of European languages.

In the following, we analyze which concepts need to be changed/adjusted in the traditional model.

What should computational morphology keep or drop from the traditional model?

The primary goal of computational morphology is to formalize and manage forms, not meaning. Word derivation should entirely remain out of the scope of computational morphology (Neme and Laporte, 2013), at least in its present phase of development. When systems include a partial implementation of word derivation, this adds an unnecessary level of complexity. The first goal of Arabic computational morphology should be the inflectional morphology and the production of accurate inflectional resources, as it is for French or English.

Only reliable information can be used in computational morphology. The pattern and root concepts in the model should be reduced to reliable phonological and orthographical representations: sequences of consonants and vowels. The semantic and syntactic information traditionally attached to roots and patterns does not allow for reliably predicting the meaning and use of the resulting forms. A pattern should be a sequence of consonants and vowels occurring together around root consonant slots. A root should be a sequence of consonants. This delimitation of the objective and scope of a modern, realistic model for Arabic computational morphology is proposed in Neme and Laporte (2013). The contribution of Laporte is a collaboration in the wording of the article.

At the present stage, word derivation and semantics do not fit in a reliable, formalized account of form variations. Thus, computational morphology should formalize only inflection.

Arabic morphological analyzers designed by computer scientists often include in their formalization a partial description of derivational morphology and word semantics, taken directly from grammatical tradition. By doing so, computer scientists probably hope that the output of their systems will be more comprehensive and more useful for further steps in an NLP pipeline. But such additional information is too incomplete and disorderly for use in information technology. And these scholars miss computational morphology's first goal, *i.e.*, a formal, clean, updatable, and accurate account of form variations.

The notion of combining roots with patterns, which has been tested for over twelve centuries, is the backbone of Semitic morphology; it is directly applicable to information technology and should be kept in computational morphology. Moreover, it works equally well with derived words and when 'roots' have 4 consonants or even more: as far as inflection is concerned, the broken plural of *misokiyon-masaAkiyn (*"poor-poors", *مِسكين مساكين)* is well described by [mskn & 1a2aa3ii4], with the same plural pattern as *Eunoquwd-EanaAqiyd (cluster-clusters, عنقود عناقيد)*,

although *misokiyon* is a derived word (in traditional morphology from a 3-letter root, [skn & ma1o2ii3]) while *Eunoquwd* has a 4-letter root.

As more exploitable regularities lie in patterns than in rules related to roots, the key to Arabic computational morphology is to assign patterns to words first, and determine their roots in consequence, by subtracting the pattern from the word, thus reversing the traditional root-and-pattern precedence in favour of the pattern-and-root model (Neme and Laporte, 2013).

The priority given to pattern over root is also justified by two other reasons: patterns are less numerous than roots (10 times at least), thus defining larger classes; and weak root-letters[4] are subject to alternations, obfuscating root-based classification. In fact, the traditional model uses pattern-and-root precedence with success (almost with full-precision) for the classification of verbal inflection, by defining verbal classes on the basis of patterns and verbal subclasses according to roots (3- or 4-root-letters) and root alternations, thus handled as exceptions (see the 12 chapters dedicated to their description in Ryding, 2005: chap.22-33). The precision and practicality of the resulting classification gave me an insight into extending the same approach to broken plurals, traditionally classified using root-and-pattern precedence.

Excluding rare cases, computer scientists reiterate concepts of traditional morphology in papers and books without questioning this tradition. Beesley (1990-2002) reproduces the lexicographical tradition of paper dictionaries and the root-and-pattern model, in a system that encompasses derivational and inflectional morphology. In contrast with this approach, the Buckwalter Arabic Morphological Analysis (BAMA, first version: Buckwalter, 2002), a lexical resource dedicated to parsing text, individuates each entry by taking out word derivation from the representation of lexical entries. The BAMA parser, now a standard in Arabic NLP, was used extensively in the Penn Arabic treebank (Maamouri et al., 2004). But, as the stems with transfixation are specified directly in the BAMA lexicon, and not obtained by transfixation, this system does not take advantage of the Semitic representation. Updating the lexical resource is difficult due to these redundancies and to the dependencies in the compatibility tables that express the inflexional, agglutinative and orthographical variations and constraints. Finally, the BAMA algorithm ignores partial vowelization, which is helpful to filter ambiguities (Neme and Paumier, 2019).

Issues and solutions for Arabic computational morphology

Our goal is to craft a process for comprehensive and accurate morpho-syntactic annotation of Arabic texts. To do so, we need to create an inflectional resource for Arabic with broad and precise coverage of forms. In this task, Arabic computer scientists and linguists face the following issues, though they are somehow partly unaware of them or ignore them intentionally. In a classical database software application, software engineers take, in general, the required time and give great importance to formalizing the relational database, often before writing any line of code. By taking many notions of traditional morphology as granted, computer scientists go too fast to implementation issues and disregard the importance of rethinking, redefining, and

[4] In the terminology of Arabic grammar, weak letters are [j], [w] and long [a].

restructuring the underlying concepts of the Semitic morphology. In fact, they do not take the required time to examine Arabic linguistic data carefully and methodically. The challenges are the following:

a) Lack of substantial critical scrutiny of traditional morphology based on modern linguistics and compatibility with computational formalisms. For instance, the blurred boundary between derivational and inflexional morphology in the Semitic tradition must become more definite (Neme and Laporte, 2013).
b) Richness of inflectional morphology, with numerous irregularities, idiosyncrasies, and phonological and orthographical variations.
c) The non-concatenative part of Semitic morphology, a challenge to computational morphology (Neme and Laporte, 2013; Neme and Paumier, 2019).
d) Difficulty of formalizing and implementing rules of clitic agglutination depending on inflection and orthographical variation of forms (Neme, 2011; Neme and Paumier, 2019).
e) Omission of vowels and other diacritics in standard text, and especially partial vowelization (Neme and Paumier, 2019).

In the following, we will summarize our new approach to the issues above.

Redefining traditional Arabic morphology

We have proceeded to some revisions of traditional Arabic morphology by keeping well-defined notions, dropping useless ones and redefining fuzzy ones clearly. Compared to tradition, our view of morphology for computational inflection retains, as a backbone of Semitic morphology, the notions of pattern and root and the operation of interdigitating a pattern with a root, and it includes the following main improvements discussed thoroughly in Neme and Laporte (2013):

- Like in the grammatical tradition of European languages, we redefine clear boundaries between derivational and inflectional morphology and exclude the former from our representation.
- We reverse the *root-and-pattern* to a *pattern-and-root* Semitic model, in the sense that we assign first the pattern, then the root.
- Inflectional Semitic paradigms are denominated according to naming rules applied independently to pattern-based classes and root-based subclasses.
- We do not use roots to label underlying meanings or concepts.
- We assign a surfacy root directly based on observable morpho-phonological alternation, and we exclude the traditional/generative notion of "deep or underlying" root (McCarthy and Prince, 1990).
- Based on the pattern-and-root model, we simplify the "chaotic" classification of broken plurals into 300 classes, instead of the estimated 3,000 classes inventoried in the Arabic tradition (Tarabay, 2003).

Taxonomic approach

In grammar textbooks of Arabic-speaking countries, children are supposed to know by heart tables of conjugation and to compute all variations of nouns according to gender, number, definiteness and case. Irregularities are learned at school according to the lemma characteristics: its pattern and the nature of its root consonants; then, with a normalized form, depending on the syntactic context and on the possible presence of agglutinative pronouns, they learn to handle case ending, letter deletion, etc. In addition, those characteristics are ordered according to the grammar textbook. In our approach to computational morphology, such hierarchical rules learned at school are unfolded in an unequivocal, systematic, and straightforward taxonomy.

In our computational representation and tools, we embrace those habits and teaching methods, widely shared by Arabic native speakers, and consequently by most potential descriptors of Arabic. Moreover, our citation form or lemmatized entry is similar to traditional dictionaries: the perfect 3rd person masculine singular for a verb, and masculine or feminine singular for a noun or an adjective. We have adjusted tools in the Unitex platform in order to facilitate the encoding of paradigmatic variations. We have created two Semitic taxonomies relative to verb variations and broken plural variations; each is split into two large sub-taxonomies according to the number of root letters: triliteral or quadriliteral, which is compatible with traditional morphology. In the end, we have designed 1,000 inflectional classes based on the pattern-and-root model and on regular noun/adjective concatenative models.

As inflectional classes are numerous, the main challenge in our approach is to assign the right pattern-class and root-subclass to each lexical entry when manually building or updating the dictionary. The scheme must be intelligent and systematic so that for each entry, users should guess the associated class quickly. The main Semitic taxonomies are defined according to pattern classes and root subclasses; the regular nouns/adjectives and other POS taxonomies are for entries based on suffix values:

- A straightforward verbal taxonomy for conjugation models with 460 classes (Neme, 2011).
- A straightforward broken plural taxonomy with 300 classes for nouns and 50 classes for adjectives (Neme and Laporte, 2013).
- The remaining classes are for nouns and adjectives with suffixed plural and other POS classes.

FST best practices: paradigmatic approach

Numerous studies have shown the adequacy of automata for linguistic problems at descriptive levels in morphology and phonology for European languages. The morphology of these languages may generally be described by simple concatenative operations. In concatenative morphology, an FST neatly maps a surface form to its morphemic structure. In Unitex, an

inflexional grammar is a representation of linguistic phenomena[5] on the basis of recursive transition networks (RTN), a formalism closely related to finite state automata. A grammar created with Unitex carries the FST approach further by using a readable graphical formalism.

For Arabic, we take advantage of the readability of this formalism and extend it to cover Semitic morphology. We use inflectional grammars to represent variations within each paradigm. We use agglutination grammars to represent allowed combinations of morphemes that constitutes a delimited word form. These grammars are represented by graphs that users can easily create, correct, and update.

Compared to Beesley's XFST approach to Arabic, here are the main distinctive features of our approach with FST (discussed thoroughly in Neme and Paumier, 2019):

- Our FSTs are compact and strictly alphabetical. More specifically, our grammatical encoding does not include levels of abstraction such as feature-value pairs, [6] whereas XFST uses such level: `[POS]Noun[gender] Masc`.
- Our FSTs implement simple and readable rules and an FST is represented visually by a graph, whereas XFST uses complex and heterogeneous rules to define at the same time rule scope, morphological alternation and suffixation, in addition to substitutions in grammatical annotations.
- We use "blind" FSTs, *i.e.*, context-insensitive FSTs, whereas XFST uses massively context-sensitive FSTs. Each blind FSTs has a predefined scope, delimited by marking disjoint sets in the lexicon, i.e. the lexicon specifies which FST applies to which entry, whereas XFST uses a part-of-speech scope, i.e. each rule fires on all entries with a given POS.
- There is no need to order the rules since at most one FST applies to each entry, whereas XFST needs ordered rules.
- For modularity, FST-rules are independent, because their scopes are disjoint, which is not the case in Beesley's pool-of-rules approach.
- Our tagging of inflected forms follows the analysis-by-generation approach, with independent phases for generation and analysis: first, we generate a fully inflected

[5] In this section, grammar means a language-theoretic computational device.

[6] Such wordy practices in representing lexical resources are still common nowadays. In *The Power of Language Music: Arabic Lemmatization through Patterns* (Attia et al., 2016) in a workshop dedicated to the lexicon, the authors formalize patterns in 655 lines, with 11 attributes in each line/pattern (see below), in which 3,100 values out of 7,200 are "unspec"; moreover, much unnecessary redundancy occurs in the **pattern, vType, comment** attributes, like in the following example:

pattern:tafAEal **singularPattern**:unspec **type**:verbs أوزان_الفعل **nType**:unspec
vType:6 **isBrokenPlural**:unspec **hasBrokenPlural**:unspec **hasFem**:unspec
subOf:unspec **examples**: تصالح، تسابق، تعامل **comment**: reciprocal – intransitive.

The encoding in our lexicon of such pattern is: V-taFaaEaL-123, where 123 denotes a regular pattern, i.e., one that does not undergo morpho-phonological alternation. Therefore, our encoding is more compact and will use only 3 attributes instead of 11 listed above (underlined attributes).

resource, then we reuse it for analysis through a lookup procedure, whereas Beesley claims that his resources are symmetrical or reversible for analysis and generation (but he does not give evidence for generation uses)[7].

UNITEX adjustments to Semitic morphology

Our linguistic tools were adjusted to take into account Arabic morphological needs (Neme, 2011; Neme and Paumier, 2019):

- Our transliteration tools avoid the hassle of handling bidirectional text files: Right-To-Left Arabic script and Left-To-Right linguistic annotations. A transliteration Arabic/Latin was implemented in Unitex, which is mostly inspired by Buckwalter's encoding, used in Arabic Penn Treebank.
- The compiler of inflectional FSTs in Unitex was extended to support the interdigitation of a root with a pattern (Neme, 2011).
- We have also created other inflectional operators to support specific surface variations of paradigms, making our inflectional taxonomy more compact with fewer classes (Neme and Paumier, 2019).
- For agglutination, the linguist can describe word-internal grammars by defining the allowed sequences of morphemes with the appropriate feature values and the orthographic variant form with a mandatory pronoun or no (+pro, +nopro). These grammars are readable resources separate from the code of the lookup procedure.
- We reused first for Arabic verbs (Neme, 2011) and then for nouns the extension of the look-up procedure to morphological analysis with predefined word-internal grammars, implemented by Paumier in 2006 for Korean (Paumier, Nam, 2014).
- Vowels in Arabic are optional orthographic symbols written as diacritics above or under letters. For partial diacritization, since our resource is fully vowelized, the lookup procedure in the full form dictionary was adapted to retain only analyses compatible with the diacritic scripted, which speeds up the process. There are no needs for backtracking or filtering, as in other approaches (Neme et Paumier, 2019).

[7] Beesley denies the reversibility of two-level morphology devices in practice:" Various *diacritical features inserted into the lexical strings to insure proper analyses made this and other KIMMO-style systems awkward or impractical for generation*" (Beesley, 1996, Section 3). Given the complexity of XFST rules devices, we think that it would be difficult to adjust Beesley's resources for actual generation uses too, due to complex dependencies between levels of representation and other issues related to numerous idiosyncrasies and exceptions.

A full account of diacritics and variations

Our resources identify unvowelized words as well as partially or fully vowelized words. In most Arabic texts, some words are scripted with at least one vowel: they make up anywhere from 1% to 15% of words, depending on the author, genre and field. Our approach takes into account the presence or omission of vowels and diacritics by means of two dozen typographical rules defined in Neme and Paumier (2019) for written text. These rules are predefined as a configuration file in UNITEX.

The standard of pronunciation is loose regarding some vowel variations, mainly for the first vowel after the first consonant. It seems that such variations are often linked to the interference of a dialectal and regional pronunciation with the standard variant of Arabic in that region. To account for first-vowel variations, we have recorded such variation and prioritized formal representation, and also readability and lexicon compactness. Thus, all the inflected forms and related vowel variations were grouped under the same lemma (Neme et Paumier, 2019).

Future developments

Our approach is by far more efficient for lexical coverage than the exclusive corpus approach. Each added lemma in our lexicon covers a considerable variation of verbal forms. For instance, one added verb sums up more than 250 inflected forms, more than 10,000 agglutinated forms, and several million partially vowelized forms. For a very rich inflectional language, it is impossible for collected corpora to cover such form variations.

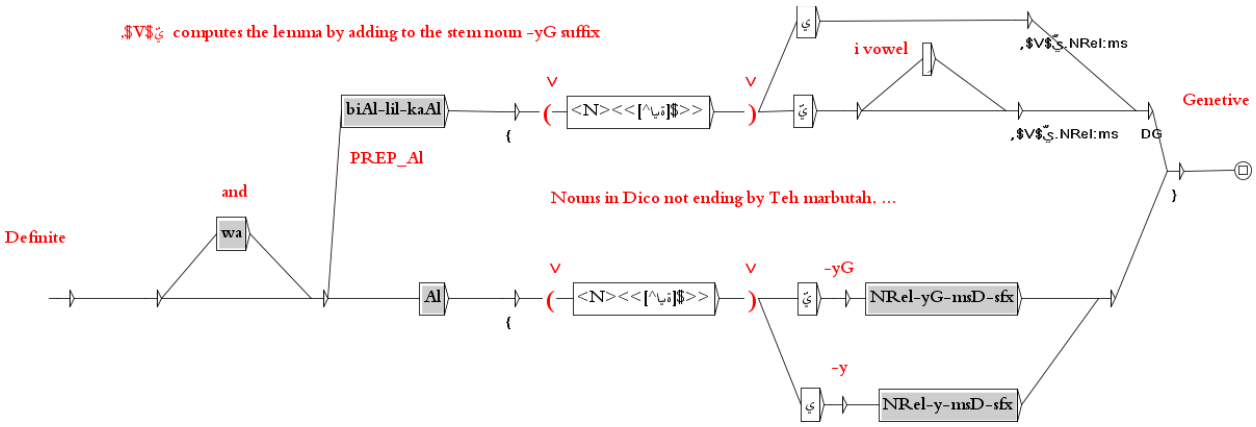
Our analyser's failure in recognition of a word form is often due to a missing lemma, or to a flaw in the inflectional rules or the agglutination grammars. In our approach, we spotted flaws in inflectional class rules (and agglutination grammars) in the early stage of development of our linguistic resources and such flaws disappeared quickly and almost entirely during the development, usage and testing on real texts, because spotting and fixing such flaws in our approach is straightforward and simple. So, the main residual cause of incorrect tagging of a word form in our approach is due to the absence of a lemma in the lexicon, and consequently of all its inflected and agglutinated forms, fully or partially vowelized. Some rule-based lexical analysers overanalyse tags by including an improper analysis due to the unexpected firing of a rule sequence. In Arabic, it may also be caused by a failure in ruling out an analysis involving a scripted diacritic, e.g. ignoring a spelling rule about hamza symbols above or under a letter, as in إعلام "media" *versus*. أعلام "flags".

Unknown forms are generally proper or common nouns and adjectives and require fallback procedures for tagging them with the right feature values. For verbal forms and according to our test set for verbal forms, containing 10,000 verbal forms vowelized (or not) and agglutinated (or not) extracted from the Nemlar corpus, our lexical coverage rate for verbs is 99.9 percent (Neme, 2011), which makes a fallback procedure for unknown verbs almost useless.

For unknown adjectives, we have identified a recurrent morphological pattern that represents relational adjectives like *IislaAm-iyG* "islam-ic", occurring in essays and literature (Neme and

Paumier, 2019, Section 5.3.1). This morphological derivation ending by *-iyG* is a productive pattern. Therefore, we crafted a morphological grammar for relational adjectives with ending *-yG* to address this gap in our lexical coverage.

We built for relational adjectives a graph invoking 18 subgraphs: 9 for the masculine and 9 for the feminine (singular, dual, plural; definite, indefinite, annexed). This morphological grammar identifies adjective forms ending in *-yG*, agglutinated or not, fully or partially vowelized. The graph checks if the stem noun is listed in the dictionary. Of course, this simplified graph does not take into account phonological alternations at the end of the stem noun and needs to be completed. However, it shows the potential of morphological grammars in Unitex.



A morphological grammar of relational adjectives (NRel-yG): the sub-graph (1/18) of the masculine singular definite (D is for 'definite')

1	أَسْمِدَةٌ زِرَاعِيَّةٌ وَمُخَلَّفَاتٌ/	{آدَمِيَّةٌ,آدَمِيّ.NRel:fsiN} وَمُلَوِّثَاتٌ صِنَ
2	لِعَوْدَةِ الْحَيَاةِ لِدَلْتَا النِّيلِ/	{الْ,.DET}{بَحْرِيَّةٍ,بَحْرِيّ.NRel:fsDG}، فَخِلَالَ
3	لَوِّثَاتِ نَقْصَ كَمِّيَّاتِ السَّمَادِ/	{الْ,.DET}{عُضْوِيٍّ,عُضْوِيّ.NRel:msDG} الطِّبِي
4	تَسْتَعِيدُ عَافِيَتَهَا وَثَرَوَاتِهَا/	{الـ,.DET}{سَّمَكِيَّةَ,سَّمَكِيّ.NRel:fsDA} الْمَفْقُ
5	هٰذَا حَسْبَمَا جَاءَ فِي دِرَاسَةٍ/	{عِلْمِيَّةٍ,عِلْمِيّ.NRel:fsiG}حَدِيثَةٍ نُشِرَت
6	مَوَادِّ الْعُضْوِيَّةِ وَالْهَائِمَاتِ/	{الـ,.DET}{نَّبَاتِيَّةِ,نَّبَاتِيّ.NRel:fsDG} وَالْأَ
7	الدَّلْتَا مَلَايِينَ الْأُطْنَانِ مِنَ/	{الـ,.DET}{رُّسُوبِيَّاتِ,رُّسُوبِيّ.NRel:fpDA:fpDG} ا
8	عْدَ ذٰلِكَ بَدَأَ الْوَضْعُ يَتَحَسَّنُ/	{تَدْرِيجِيًّا,تَدْرِيجِيّ.NRel:msiA} وَبَدَأَتْ تَظْ
9	أَيِ نِكْسُونْ- سَوَاحِلَ مِصْرَ/	{الْ,.DET}{مُتَوَسِّطِيَّةَ,مُتَوَسِّطِيّ.NRel:fsDA}
10	نِكْسُونْ فِي تِلْكَ الدِّرَاسَةِ مِنَ/	{الـ,.DET}{إِحْصَائِيَّاتِ,إِحْصَائِيّ.NRel:fpDA:fpDG}
11	كَمَا لُوحِظَ اسْتِعَادَةُ الْأَسْمَاكِ/	{الـ,.DET}{سَّطْحِيَّةِ,سَّطْحِيّ.NRel:fsDG} لِحَالَةِ
12	ي كَانَتْ عَلَيْهَا أَمَامَ الْأَسْمَاكِ/	{الْ,.DET}{قَاعِيَّةِ,قَاعِيّ.NRel:fsDG} قَبْلَ بِنَا
13	لْبُرُوتِينِ الْكُلِّيِّ وَالْبُرُوتِينِ/	{الـ,.DET}{حَيَوَانِيِّ,حَيَوَانِيّ.NRel:msDG} و

Part of a concordance obtained by applying the `NRel-yG.grf graph`[8] with the `Locate` program in Unitex.

For unknown proper nouns or person names and surnames, we built a morphological grammar representing patterns with the prefixes *Ebdul-*, *Abu-*, or *bu-*; in Algeria, names also use often the prefixes *bel-*, *bin-* (Riadh ***Bel***kebir, personal communication). Such grammars for names require to be completed with suffixes such as *-Allah* (*Nasr**Allah***) and *-Aldiyn* (*Nasr**Aldiyn***), etc. The same applies to place names with prefixes like *kafar-*, *bayt-*, etc…

However, not all proper nouns may undergo a "patternalization" with prefix or suffix and particularly foreign proper names transliterated in Arabic, like *Android, Trump, McDonald's* or *International* (as part of a company name). Here, quantitative approaches would be more appropriate and statistics on transliterated strings should be of great help. Character N-grams counts made on collected foreign proper nouns and on Arabic common and proper nouns might determine if a word is an Arabic word or a transcription from a foreign language, which is generally a proper noun.

For unknown nouns and adjectives, one may craft fallback procedures based on a statistical approach in order to guess the gender, number, definiteness and case. The fallback procedure may be based on the extraction of morphological features (word length, prefixes and suffixes) from our full-form lexicon. The advantage of this method is that the resource is comprehensive and takes into account all linguistic facts.

For future applications, our compact notation has an effortless interpretation in Arabic Semitic morphology; therefore, it promotes precise and swift communication between linguists, computational linguists and developers. We have reconciled compactness with readability. Our notation allows for encoding derived nouns in conformity with standard, traditional patterns: for example, derived nouns with root ending in *'n'* like *IstiHosAn* إستحسان can be encoded by combining the standard pattern `IisotiFoEaaL` إستفعال with the constraint on the 3rd root letter, `IisotiFoEaaL-12n`,[9] etc.

In fact, the bases of Arabic Semitic morphology, which are a predefined number of patterns and a set of morpho-phonological alternations rules, naturally translate to regular expressions and FSTs. This notation makes implementations easier to maintain and debug as it is more intelligible to all members of a development team.

Perspectives

Morpho-syntactic tagging is an operation that associates with each word grammatical information designated by a label or a list of potential labels. A word in Arabic may be composed of up to five morphemes. Therefore, a correct analysis should achieve the expected

[8] Examples 7 and 10 in this concordance denote "sediments" and "statistics". They are in the feminine plural, but they must be lemmata in lexical entries in the dictionary. The grammar should not relate these occurrences to the respective canonical singular forms, but the program cannot decide this automatically. We think that a frequency list of singular/plural forms and related concordances should be a great help to assist linguists in deciding whether to insert (or not) plural forms as lemmata in lexical entries.

[9] In Unitex, we create a graph named `IisotiFoEaaL-12n.grf` with this pattern and this constraint. This implementation uses Unitex morphological mode and the graph can match forms fully or partially vowelized.

morphemic segmentation and assign the correct labels to each segment. In multi-candidate tagging, a proper tagging of a segment is defined by the presence of the appropriate grammatical label in the candidate list. Contrariwise, an incorrect tagging is the lack of this correct label in the candidate list.

Multi-candidate tagging is fine, but it is usually insufficient for applications. In general, the rate of morphosyntactic ambiguity in Arabic is higher than in French due to morpheme agglutination, and mainly to diacritic omission. We think taking a hybrid approach by tagging morphemes first and then applying a language model based on supervised learning will be accurate enough to pick the correct solution from a list of candidates. We think this hybrid approach to one-solution tagging will give better accuracy than pure quantitative approaches for two main reasons:

- it exploits a comprehensive and accurate lexicon, which reduces the number of unknown words;
- the units to be labelled through this statistical classification are morphemes and not words made of agglutinated segments, which reduces the data sparsity of labels caused by agglutination.

Conclusions

Our PRIM model has redefined and simplified the traditional Arabic morphology and we have proceeded to a revision by keeping well-defined notions, dropping useless ones and redefining fuzzy ones clearly. Compared to tradition, our view of morphology for computational inflection retains, as a backbone of Semitic morphology, the notions of pattern and root and the operation of interdigitating a pattern with a root, and it includes the following main improvements discussed thoroughly in Neme and Laporte (2013):

- Like in the grammatical tradition of European languages, we redefine clear boundaries between derivational and inflectional morphology and exclude the former from our representation.
- We reverse the *root-and-pattern* to a *pattern-and-root* Semitic model, in the sense that we assign first the pattern, then the root.
- Inflectional Semitic paradigms are denominated according to naming rules applied independently to pattern-based classes and root-based subclasses.
- Root is a sequence of consonants. We do not use roots to label underlying meanings or concepts. Similarly, we do not use the pattern to label POS.
- We assign a surfacy root directly based on observable morpho-phonological alternation, and we exclude the traditional/generative notion of "deep or underlying" root (McCarthy and Prince, 1990).

Based on the pattern-and-root model, we simplify the "chaotic" classification of broken plurals into 300 classes, instead of the estimated 3,000 classes inventoried in the Arabic tradition (Tarabay, 2003).

For future developments, our compact notation possesses an interpretation in the Arabic Semitic morphology. Consequently, it enables accurate and swift communication between linguists, computational linguists and developers. Our encoding applies to derived nouns by using standard Semitic patterns with, if necessary, morpho-phonological alternations or constraints on the root. Such notation creates the condition to separate the design of Semitic morphological masks from its implementation (as a set of regular expressions, for instance). In this way, it contributes to building a high-level language for Arabic morphology, independently from device implementations.

Main publications

Alexis Amid Neme (2011). "A lexicon of Arabic verbs constructed on the basis of Semitic taxonomy and using finite-state transducers". In *Proceedings of the International Workshop on Lexical Resources WoLeR, ESSLLI International Workshop on Lexical Resources*, Ljubliana, Slovenia. halshs-01186723v1

Alexis Amid Neme, Éric Laporte (2013). "Pattern-and-root inflectional morphology: the Arabic broken plural." *Language Sciences*, Vol 40, November 2013, Pages 221-251. hal-00831338v1

Alexis Amid Neme, Sébastien Paumier (2019). Restoring Arabic vowels through omission-tolerant dictionary lookup. In *Language Resources and Evaluation*, 2019, Vol.53, pp.1-65 pages, Springer Verlag, https://doi.org/10.1007/s10579-019-09464-6

Other publications

Alexis Amid Neme (2014a). Why Microsoft Arabic Spell-checker is ineffective, *Linguistica Communicatio*, http://www.al-erfan.com/, 2014, Arabic Language in Information Technology, 16, pp.55. <http://www.al-erfan.com/> hal-01081965v1

Alexis Amid Neme (2014b). A fully inflected Arabic verb resource constructed constructed from a lexicon of lemmas by using finite-state transducers, *Revue RIST : revue de l'information scientifique et technique*, 2013, 20 (2), pp.13. <http://www.webreview.dz/> halshs-01186734v1

Bibliography

Altantawy, Mohamed; Habash, Nizar; Rambow, Owen; Saleh, Ibrahim (2010). Morphological Analysis and Generation of Arabic Nouns: A Morphemic Functional Approach. In Proceedings of the Language Resource and Evaluation Conference (LREC), Malta, pages 851-858.

Attia, Mohammed, Pavel Pacyna, Lamia Tounsi, Antonio Toral, Josef van Genabith. (2011). An Open-Source Finite State Morphological Transducer for Modern Standard Arabic. International Workshop on Finite-State Methods and Natural Language Processing (FSMNLP). Blois, France.

Attia Mohammed, Zirizkly Ayah, Mona Diab (2016). The Power of Language Music: Arabic Lemmatization through Patterns, CoLing 2016, Proceedings of the Workshop on Cognitive Aspects of the Lexicon, p. 40-50, Osaka, Japan, https://bit.ly/2PsJHNk.

Beesley, K. R. (1990). Finite-state description of Arabic morphology. In Proceedings of the Second Cambridge Conference on Bilingual Computing in Arabic and English. No pagination.

Beesley K. R.. 1991. Computer analysis of Arabic morphology: A two-level approach with detours. In Bernard Comrie and Mushira Eid, editors, Perspectives on Arabic Linguistics III: Papers from the Third Annual Symposium on Arabic Linguistics, pages 155-172. John Benjamins, Amsterdam. Read originally at the Third Annual Symposium on Arabic Linguistics, University of Utah, Salt Lake City, Utah, 3-4 March 1989.

Beesley, Kenneth R. (1996). Arabic finite-state morphological analysis and generation. In Proceedings of the International Conference on Computational Linguistics (COLING), Copenhagen, Center for Sprogteknologi, volume 1, pages 89-94.

Beesley, Kenneth R. (1998). Constraining separated morphotactic dependencies in finite-state grammars. In FSMNLP-98, Bilkent.

Beesley, Kenneth R. (2001). Finite-State Morphological Analysis and Generation of Arabic at Xerox Research: Status and Plans in 2001. In Proceedings of the ACL/EACL Workshop 'Arabic Language Processing: Status and Prospects', pages 1-8.

Beesley, Kenneth R., Lauri Karttunen, (2003). Finite State Morphology, CLSI Studies in Computational Linguistics, 509 pages.

Buckwalter, Timothy (2004). Arabic Morphological Analyzer Version 1.0. (BAMA, 2002). LDC Catalog No.: LDC2002349.

Maamouri, M., Bies, A. & Buckwalter, T. (2004). The Penn Arabic treebank: Building a large-scale annotated Arabic corpus. In NEMLAR Conference on Arabic Language Resources and Tools, Cairo, Egypt.

Habash, N., Rambow, O., and Roth, R. 2009. MADA+TOKAN: a toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization. In Proceedings of the Second International Conference on Arabic Language Resources and Tools, Cairo, Egypt, pp. 102-9.

McCarthy, J. J. & Prince, A. S. (1990). Foot and word in prosodic morphology: the Arabic broken plural. Natural Language and Linguistic Theory 8(2), 209–283.

Paumier, S., Nam, J.-S. 2014, Un système de dictionnaire de mots simples du coréen. In Fryni Kakoyianni-Doa, *Penser le Lexique-Grammaire. Perspectives actuelles*, Honoré Champion, pp. 481-490, Collection Colloques, congrès et conférences. Sciences du Langage, histoire de la langue et des dictionnaires. 30th International Conference on Lexis and Grammar (Nicosia, Cyprus, 2011).

Paumier, S. (2016). Unitex—User manuel 3.1RC. Champs-sur-Marne: University of Marne-la-Vallée.

Shuly Wintner (2008). Strengths and weaknesses of finite-state technology: a case study in morphological grammar development, Natural Language Engineering, Volume 14 Issue 4, October 2008, Page 457-469, Cambridge University Press New York, NY, USA.

Tarabay, Adma (2003). A dictionary of Arabic plurals. Librairie du Liban Publishers. 590 pages. In Arabic.

Zbib, Rabih; Soudi, Abdelhadi (2012). Introduction. Challenges for Arabic machine translation. In Soudi, Abdelhadi; Farghaly, Ali; Neumann, Günter; Zbib, Rabih (eds.), Challenges for Arabic Machine Translation, Natural Language Processing, 9, Amsterdam: Benjamins, p. 1-13.

Appendix A: Arabic conjugator on http://babelarab.univ-mlv.fr/

In this site, the full dictionary contains 15,400 verbal entries classified into 460 conjugation models. As in traditional grammar, the dictionary contains simple and augmented triliteral (14,500) and quadriliteral (900) root verbs: with regular root, geminate root, with hamza, with one or two weak consonants, or a combination of these features.

The user should simply type in the verb without caring about knowledge in Arabic morphology. No need to enter the pattern and the kind of root. The full dictionary does not allow the conjugation of an arbitrary sequence of letters, as in some published conjugators like https://qutrub.arabeyes.org

الضمائر	المضارع المنصوب	الأمر المؤكد	الماضي المعلوم	المضارع المؤكد الثقيل	المضارع المعلوم	المـ
أنا	أَكْبَكَ		كَبَكْتُ	أَكْبَكَنَّ	أَكْبَكُ	أكـ
نحن	نَكْبَكَ		كَبَكْنَا	نَكْبَكَنَّ	نَكْبَكُ	نكـ
أنت	تَكْبَكَ	اِكْبَكَنَّ	كَبَكْتَ	تَكْبَكَنَّ	تَكْبَكُ	تكـ
أنتِ	تَكْبَكِي	اِكْبَكَنَّ	كَبَكْتِ	تَكْبَكِنَّ	تَكْبَكِينَ	تكـ

Fig. A1: an example of an inexistent verb (here '*kbk*') conjugated by https://qutrub.arabeyes.org

Our conjugator displays four tabs: a basic conjugation (34 forms) with two tabs for Arabic script and Latin transliteration, and a complete conjugation (144 forms), in the Arabic script only, divided into three tabs for active with or without pronoun and passive. Short vowels are fully scripted in the displays.

Our 460 conjugation models are complete. Our lexicon is almost complete, with 15,400 entries. Each entry is inflected into 144 surface forms and on average, 158 forms if we include orthographic variations of core form due to agglutination. In 2014, a conjugator prototype with a sample of 300 verbs was implemented at http://tasrif.univ-mlv.fr/.

"The size of the full-form dictionary is 2.43 million surface forms. The size of the full-form dictionary in plain text is 132 Megabytes in Unicode UTF-8. It is compressed into 4 Megabytes before loading to memory for fast retrieval. The generation, compression, and minimization of the full-form lexicon take less than one minute on a common Windows laptop. The tagging of a 4-segment verbal form takes less than 0.5 milliseconds." (Neme, 2011)

In 2015, we extended the coverage of the prototype to 15400 verbs and since then we have been extending also the coverage of participles. We added two conjugation tables to our new site at http://babelarab.univ-mlv.fr/. As far as we know, these two tables do not appear in any conjugation software, as of 2019: a) variation of verbal forms with an agglutinated pronoun (Fig.A2); b) active and passive participles with more 1.7 million forms (Fig.A3).

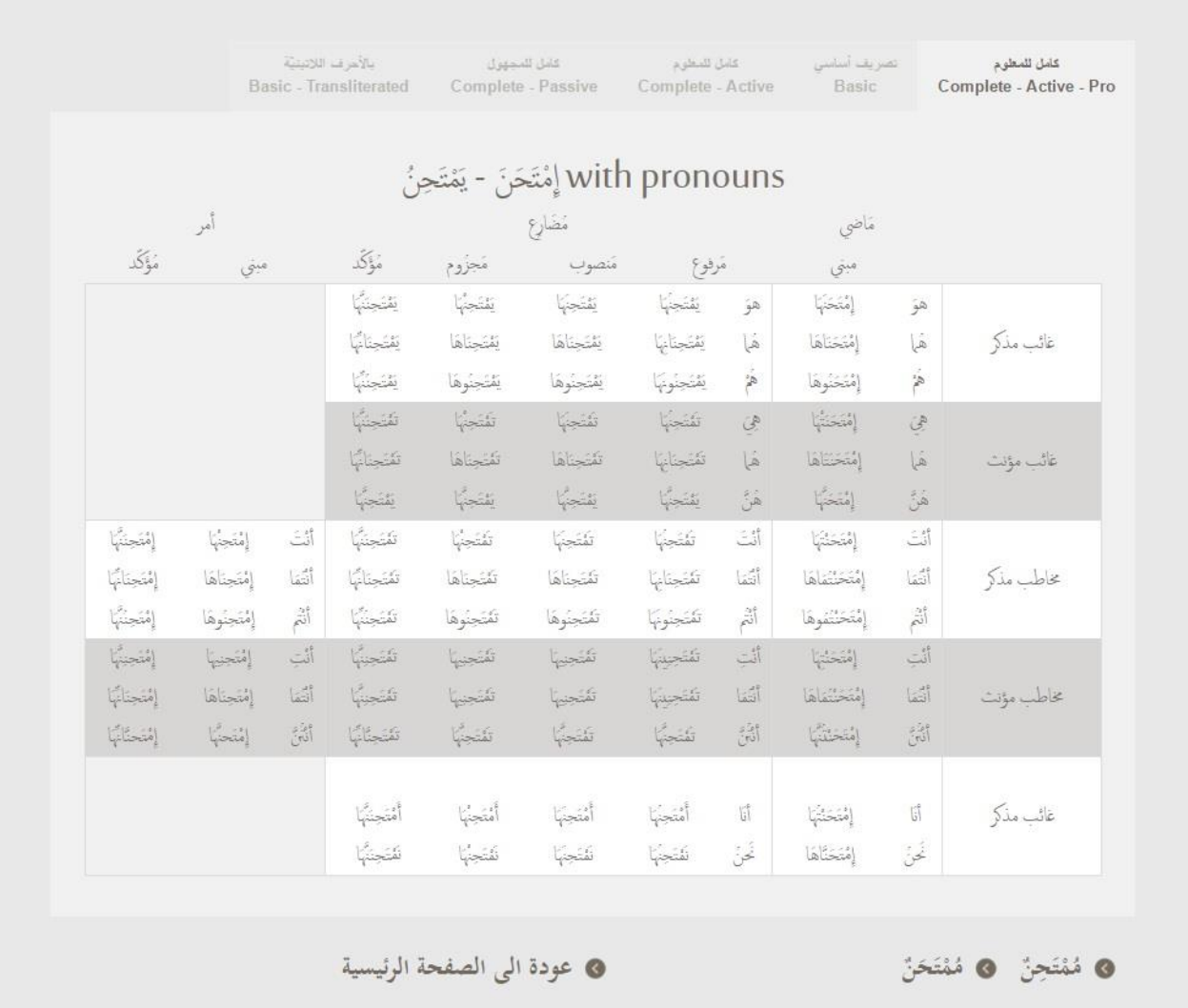


Fig. A2: The verb conjugated with an enclitic pronoun *IimotaHana_ha* "examine_it": column in the perfect, imperfect (indicative, subjunctive, jussive and energetic), and imperative (simple, energetic). Cf. http://babelarab.univ-mlv.fr/

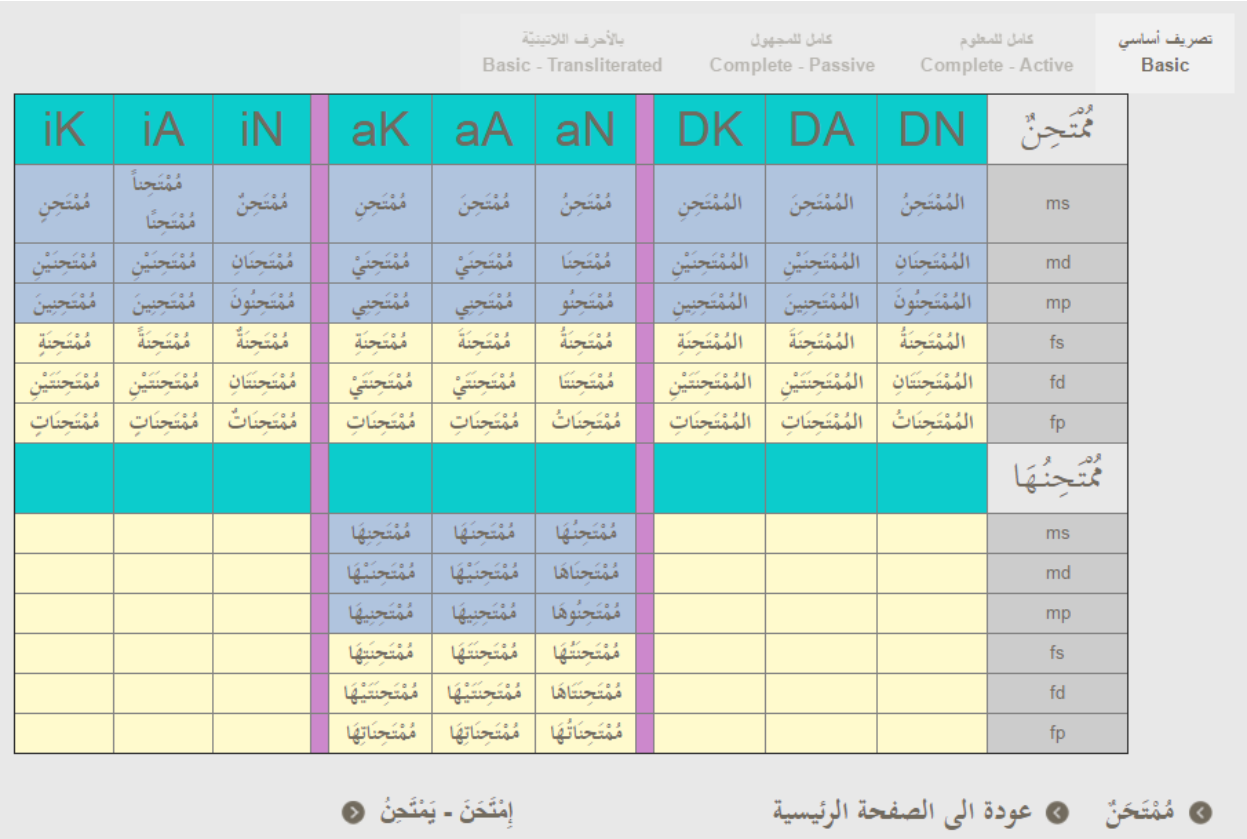


Fig. A3: Inflected forms of the active participle of the verb “to examine”, “examining”, with enclitic pronouns and variations in gender, number, definiteness and case; the lower part of the table represents the annexed forms (aN, aA, aK) with an enclitic pronoun. Cf. http://babelarab.univ-mlv.fr/

Appendix B: Arabic spell checker http://babelarab.univ-mlv.fr/

Fig B1: Unknown words are in blue. Most result from spelling mistakes or missing lexical entries. Diacritics' mistakes are detected as well.

Fig. B2: This is the same text as B1, but the spell checker finds 5 more unknown words resulting from the absence of a *hamza-below-the-alif* symbol. The corresponding rule was configured in the spell checker as mandatory for this experiment.

Restoring Arabic vowels through omission-tolerant dictionary lookup

Alexis Amid Neme, Sébastien Paumier

Restoring Arabic vowels through omission-tolerant dictionary lookup

تشْكيل الكَلِمات عَبْرَ مَوارد حاسوبيّة

Alexis Amid Neme and Sébastien Paumier
Université Paris-Est, LIGM, UPEM, CNRS, ENPC, ESIEE, 77454, Marne-la-Vallée, France

Abstract

Vowels in Arabic are optional orthographic symbols written as diacritics above or below letters. In Arabic texts, typically more than 97 percent of written words do not explicitly show any of the vowels they contain; that is to say, depending on the author, genre and field, less than 3 percent of words include any explicit vowel. Although numerous studies have been published on the issue of restoring the omitted vowels in speech technologies, little attention has been given to this problem in papers dedicated to written Arabic technologies.

In this research, we present Arabic-Unitex, an Arabic Language Resource, with emphasis on vowel representation and encoding. Specifically, we present two dozens of rules formalizing a detailed description of vowel omission in written text. They are typographical rules integrated into large-coverage resources for morphological annotation. For restoring vowels, our resources are capable of identifying words in which the vowels are not shown, as well as words in which the vowels are partially or fully included. By taking into account these rules, our resources are able to compute and restore for each word form a list of compatible fully vowelized candidates through omission-tolerant dictionary lookup.

In our previous studies, we have proposed a straightforward encoding of taxonomy for verbs (Neme, 2011) and broken plurals (Neme & Laporte, 2013). While traditional morphology is based on derivational rules, our description is based on inflectional ones. The breakthrough lies in the reversal of the traditional root-and-pattern Semitic model into pattern-and-root, giving precedence to patterns over roots.

The lexicon is built and updated manually and contains 76,000 fully vowelized lemmas. It is then inflected by means of finite-state transducers (FSTs), generating 6 million forms. The coverage of these inflected forms is extended by formalized grammars, which accurately describe agglutinations around a core verb, noun, adjective or preposition.

A laptop needs one minute to generate the 6 million inflected forms in a 340-Megabyte flat file, which is compressed in two minutes into 11 Megabytes for fast retrieval. Our program performs the analysis of 5,000 words/second for running text (20 pages/second).

Based on these comprehensive linguistic resources, we created a spell checker that detects any invalid/misplaced vowel in a fully or partially vowelized form. Finally, our resources provide a lexical coverage of more than 99 percent of the words used in popular newspapers, and restore vowels in words (out of context) simply and efficiently.

Abstract in Arabic

الحركاتُ رموزٌ إختياريٌ كتابتها في اللّغة العربية، وتُكتب كل حركةٍ فوق أو تحت الحرف المُناطة إليه. تشمل معظم النصوص العربية على كلمات مُشكَلة جزئيًا ولا يتعدى عامةً نسبتها 3٪ من الكلمات وهذه النسبة تتوقف على الناشر والكاتب والميدان المتخصِص. على الرغم من أن العديد من الدراسات العلمية قد تمَّ نشرها في مسألة حذف الحركات في تقنيات الكلام، فقد أولِيَ اهتمام لا يذكر لنفس المشكلة في الدراسات المخصصة لتقنيات العربية المكتوبة.

في هذا البحث، نقدم وصفاً مفصلاً لحذف الحركات في النصوص المكتوبة والقواعد المطبعية ذات الصلة وقواعد حذفهما في الموارد الحاسوبيَّة. مواردنا قادرة على التعرُّف على الكلمات المشكلة كلياً أو جزئيا أو غير المشكلة كما وإعادة الحركات لكلٍ منها،

في دراسات سابقة، اقترحنا تصنيفات للأفعال (Neme, 20011) وتصنيفات لجموع لتكسير (Neme & Laporte, 2013) مبنية على أُسُس علم الصرف التقليدي. ففي حين يحتوي علم الصرف التقليدي على توصيف القواعد الاشتقاقية وغير الاشتقاقية، يستند وصفنا على الصرف غير الإشقاقي حصرياً. والجديد في مقاربتنا يكمن في عكس مقاربة علم الصرف التقليديّة التي هي معادلة (الجذر–الوزن) إلى (الوزن–الجذر) مع إعطاء الأولوية للوزن على حساب الجذر. هذا التغيير سمح لنا التعرّف على الفعل كمدخل معجمي بشكل أسرع وأدقّ وبالتالي التعرّف على جذره ووزنه، كما قلّص تحديد وبرمجة مئات القواعد الصرفيّة والإملائية التي تربط أشكال الفعل بجذره ووزنه.

وقد تم بناء المورد اللغوي يدويًا ويحتوي على 76000 مدخل معجمي محرّك بأكمله. تمّ تصريف هذا المورد ليحتوي على 6 ملايين شكل محرّك أيضاً. وقد تمّ إضافة السوابق واللواحق لهذه الأشكال عن طريق قواعد تلاصقيّة نحويّة دقيقة حول فعل أساس، إسم، أو صفة. هذه القواعد تحدِّد تتابع الشرائح المسموح بها من سوابق ولواحق حول شريحة أساسية.

يحتاج حاسوب محمول إلى دقيقة واحدة لتوليد 6 ملايين شكل محرّك وحجم الملف 340 ميغابايت، قد تمّ ضغطه إلى 11 ميغابايت للبحث السريع. يقوم برنامجنا بتحليل 5000 كلمة في الثانية (20 صفحات/ثانية). والتغطية المعجمية لمواردنا تضاهي 99٪ لنصوص من الصحف العامة.

في هذه الدراسة، نركز على توصيف قواعد حذف الحركات والشدّة والهمزة. ونعرض حلاً بسيطاً فعالاً وأنيقاً يتعرّف على كلمات غير مشكلة أو مشكلة جزئياً أو كلياً وإعادة الحركات لكلٍ منها في برنامج للتحليل الصرفي.

1 Introduction

Writing conventions in Arabic are characterized by being based on consonants and also underspecified—they usually lack short vowels and other diacritics. This is indirectly connected to the historical legacy of the first consonantal Phoenician alphabet, as is the case with other Semitic languages. In practice, speakers and readers do restore these essential lacking pieces based on their memory and knowledge of Arabic. Therefore, it is a legitimate goal that computers should be able to compute and restore these missing vowels and diacritics in written texts.

Big institutions were unsuccessful in dealing with the issue of missing vowels in written texts. Googlelabs withdrew its software to restore vowels in Arabic text in 2012, just a year after its release, while in May 2012 an Arabic spell checker for Gmail was released only to be withdrawn the same year. One of the problems users encountered using Gmail's spell checker was that it erroneously flagged as mistakes fully or partially vowelized words which happened to be correct. Microsoft Office 2016 suffers from the opposite problem: its Arabic spell checker ignores fully or partially vowelized words - erroneous vowels are not flagged as mistakes and neither are typographical mistakes such as the *'-bF'* and *'-AN'* endings in كتابَ كتاباً *ktAbF* or *ktAbAN*.[1]

Lately, maybe in 2016, Google released an Arabic spell checker with a low coverage of inflection and of affixed and agglutinated words. This time, like Microsoft, it ignores partially vowelized words; even worse, it does not flag a wrong word if it contains one vowel. In average, Google's spell checker flags around 10% of valid words erroneously.

These problems highlight the difficulties in building accurate Arabic computational and morphological resources. There are a number of reasons for this:

- Arabic has a rich morphology, containing six attributes for verbs and four for nouns and adjectives
- its inflection uses prefixes, suffixes, and mostly infixes described by the root-and-pattern traditional model
- words may have agglutinated clitics (from a set of around 30 clitics)
- vowels in words are generally omitted or partially represented.

If the first three issues have been handled in Arabic Language Technologies with some degree of attention, the last issue is less studied in computational morphology and has not been given the correct rank of importance, as Maamouri et al. (2006) state[2]: *In general, the role of diacritics in a NLP pipeline that includes parsing is very much an open question*.

[1] The TB++ transliteration used in this paper is derived from the Buckwalter encoding and adopted in Unitex to map Arabic <=> Latin: ء, c; آ, C; أ, O; ؤ, W; إ, I; ئ, e; ا, A; ب, B; ة, p; ت, T; ث, V; ج, J; ح, H; خ, x; د, d; ذ, J; ر, r; ز, z; س, s; ش, M; ص, S; ض, D; ط, T; ظ, Z; ع, E; غ, g; ف, f; ق, q; ك, k; ل, l; م, m; ن, n; ه, h; و, w; ى, Y; ي, y; ً, F; ٌ, N; ٍ, K; َ, a; ُ, u; ِ, i; ّ, G; ْ, o.

[2] There are optional typographical signs in another Semitic language. "The Hebrew script [has two variants]: one in which vocalization diacritics, known as niqqud "dots", decorate the words, and another in which the dots are missing, and other characters represent some, but not all of the vowels. Most of the texts in Hebrew are of the latter kind; unfortunately, different authors use different conventions for the undotted script. Thus, the same word can be written in more than one way, sometimes even within the same document, again adding to the ambiguity." (Wintner, 2008)

Many Arabic lexical resources lack information about vowels, an absence often explained by the rarity of vowels in written texts. This is a view that is becoming widespread with the expansion of corpus linguistics.

However, spelling out vowels in words is a convenient way to distinguish lemmas with different meanings: *Eaqod/Eiqod/Eaqid* "contract/necklace/thickening (for a liquid)" عَقْد/عِقْد/عَقِدor *giloyaAn/galayaAn* "is boiling (adjective)/the boiling (noun)" غَلَيان غِلْيان. Vowels and other diacritics are part of the message, even if they are not represented as graphical symbols. Language is foremost an oral form of communication and the selection of writing conventions is subsequent. Vowels are an essential part of Arabic, even if they lack in its written form. Why would such an essential part of the language be irrelevant to NLP, or less relevant than POS?

Creating Arabic lexical resources is not a simple task. Making them accurate without vowels is impossible. For example, in some words, the short vowel after the first consonant alternates with a variant: *n**u**faAyap* vs. *n**i**faAyap* "rubbish" (whereas **n**a**faAyap* is inacceptable), and the prevalence of a choice in a text may indicate a regional pronunciation or a register of language: formal or colloquial. In all Arabic dictionaries, both old and modern, diacritical information is available and inventoried thoroughly. For speech technologies, vowels are required.

By 'accurate' ALR, we mean both recall (high lexical coverage) and precision (rejection of invalid forms), at three levels:

- inflection: if a verb or noun is in the ALR, then all the inflected forms of its lemma and no invalid inflected forms must be taken into account;

- agglutination: if an inflected form is in the ALR, then all of its valid agglutinated forms, and no invalid forms, must be taken into account;

- vowelization: if an inflected form, agglutinated or not, is in the ALR, then all of its vowelized forms, whether it is partial or total vowelization, must be taken into account, as well as forms not containing vowels, and no invalid forms. [3]

Devices (involving programs, extensive lists, FSTs, etc.) recognizing and/or generating such forms should not over- or under-generate.

The orthographic system of Arabic includes 34 'bare letters', which are always transcribed, and nine diacritical marks optionally written:

- Three short vowels (*a, i, u*) and the zero-vowel diacritic or *sukoon* (*o*), for the absence of a vowel; all four occur in all positions except word-initial, although *o* occurs very rarely between the first and second consonants;
- Three nunation marks (*-N, -F, -K*, phonetically equivalent to *-un, -an, -in*) used as noun case and definiteness (indefinite) suffixes, and therefore only in ending positions;
- the gemination mark ّ or *shadda* (*G*), which is used for the derivation of new words or broken plural inflection and occurs after the second consonant of the main morphological element of the word;
- the superscript long 'a' or superscript *alif* ٰ (*R*), a rarely scripted, archaic form usable in some frequent words such as the pronoun هذا *haRJaA* 'this' and in some archaic spellings still used in modern Arabic such as *raHomaRn* 'merciful'.

[3] One may add a typographical consistency at document(s) level, in terms of the so-called editing style of a publication. In French for instance, this requirement includes using the same symbol in words such as *oeuvre* or *œuvre* throughout one or a set of documents; in Arabic, it will be a mandatory transcription of a hamza-above-alif.

Moreover, an initial glottal stop or *hamza* can be omitted. In a word initial position, it is represented by two characters: *O* for *hamza* above *A* أ; and *I* for *hamza* under *A* إ. Omitting the glottal stop consists of writing the *A* ا instead of *O* or *I*; therefore, these two characters belong to our topic in this paper. In non-initial position, the *hamza* diacritic appears in five different characters (*c,* ء; *W,*ؤ ; *e,* ئ; *O,* أ; *I,* إ); but it is not an optional diacritic and cannot be omitted. Consequently, these characters with *hamza* in non-initial position do not belong to our topic.

For simplicity, we use interchangeably 'vowel' and 'diacritic' throughout the rest of the paper and we mean by both terms all nine diacritical marks, and the initial *hamza* diacritics carried by *A*.

Diacritization/vowelization is the operation to assign/restore a diacritic/vowel to a undiacritized/unvowelized consonant in a word. It is a typical knowledge test in Arabic vocabulary and grammar. Words with at least one written vowel are said to be partially vowelized; and fully vowelized, when all are written. A word form delimited by two spaces may include one or two vowels (three in rare cases). "*In the Penn Arabic Treebank (part 3), 1.6 percent of all words have at least one diacritic indicated by their author*" (Habash, 2010, p.11). In most newspapers, only about 2-3 percent of words are partially vowelized, although this can reach 12-15 percent in well-edited articles. Some reference books are almost completely vowelized, such as *Kitab fasl al-maqal* by Averroes, the Andalusian philosopher of the XIIth century; while other books including dictionaries, teaching textbooks and holy texts are fully vowelized.

"Arabic NLP research faces two major challenges, not necessarily shared with many other natural languages: the first is its complex linguistic structure and the second, the specific features of its orthographic system" (Maamouri et al. 2006, Introduction). In the next sub-section, we present the main consequence of under-representation of vowels on morphological analysis: it increases tagging ambiguities.

DIACRITICAL AMBIGUITY

Word-level ambiguity is common to all natural languages, including Arabic; even the full representation of vowels does not prevent ambiguity in Arabic, as in *EaAmil*, "worker/agent" عامل. However, the under-specification of Arabic script – the loss of vowels – causes written Arabic to have more ambiguities, called *diacritical ambiguities*. We restrict the definition of diacritical ambiguity to the case where the omission of one or more vowels generates additional ambiguity.

To illustrate diacritical ambiguity in Arabic, let us draw a parallel with French examples with or without accent(s). In French, poor and rich typography refers respectively to non-accented and accented typography. In order to make a parallel with vowel omission in Arabic, we extend the use of the term 'poor typography' to the case where at least one accent is omitted, and at least another is present. The rich word form *chantées* has only one possible poor typographical representation *chantees*, whereas *déjà* also has two possible partial accentuations *déja* and *dejà*. A word form such as *déjà* has four possible typographical representations: fully, partially accentuated or not accentuated.

How to retrieve the fully vowelized form from a partially vowelized one? An index is the simplest way to access stored information through a keyword. Thus, in order to access a fully

accented word in a French lexicon, one may build an auxiliary index on the `poor:rich` pattern by replacing each accented letter by its non-accented counterpart:

```
chantees:chantées
chantées:chantées
deja:déjà
déja:déjà
dejà:déjà
déjà:déjà
```

Conversely, the form *chantéés* would be inexistent in such an index since only the omission of an accent is valid, not the addition (as in *katab**aa**tu* in Arabic); the form *chanteees* would also be inexistent, since it has no corresponding valid rich form.

If an index for word forms like *chantées* is simple to construct, the index for *déjà* exhibits more complexity. Arabic word forms are more complex than *déjà* because in the full representation of a word form, a diacritic occurs after each consonant. Building such an index for Arabic would not be a viable solution because it would contain several billions of partially vowelized forms.

There is no diacritical ambiguity in the words *deja* and *déja* since they refer to a single fully accented form: *déjà*. A complex diacritical ambiguity would be the poor typographical representation of *pêche, péché, pèche, péché, pêche, pêché,* (resp. "peach", "sin", "(he) sins", "sinned", "fishing", "fished"). All six are represented in poor typography by *peche*. So, the under-representation of accents in *peche* is the origin of an ambiguity between 6 candidates. But partial representation of diacritics, as in *pêche*, reduces them from six to three. It is a pity not to take advantage of such information in a parser (cf. Sections 2.3 and 2.4).

Serbian exhibits similar features; only 5% of words in ordinary text contain at least one accentuated letter, and many of them have no diacritical ambiguity since they stand for a single fully accented form like *déjà*. In "*Knowledge and Rule-Based Diacritic Restoration in Serbian*", Krstev et al. (2018) propose a solution for Serbian based entirely on linguistic resources. They present *"a procedure for the restoration of diacritics in Serbian texts written using the degraded Latin alphabet. The procedure relies on the comprehensive lexical resources for Serbian: the morphological electronic dictionaries, the Corpus of Contemporary Serbian (processed for uni-, bi- and tri-gram frequencies) and local grammars. Dictionaries are used to identify (in 5 modular steps) possible candidates for the restoration, while the data obtained from SrpKor and local grammars assists in making a decision (defined by 7 steps) between several candidates in cases of ambiguity*". They conclude, "*The diacritic restoration can be successfully solved by using a rule-based approach that relies on the lexical resources. […]. This solution exhibits the advantage of transparency (and modularity) which is usually characteristic of such methods*."

Fig. 1 illustrates partial diacritization with some statistical data about a 200-word excerpt of a newspaper text about "the rising price of gold".

لا تقتصر أهمية الذهب وقيمته على كونه أداة للتزيّن، بل على الدور الذي يضطلع به في تهدئة مخاوف المستثمرين في الأيام الأكثر تشاؤماً، باعتباره ملاذاً آمناً يقيهم شر التراجع في الأسواق المالية. وفي ظل تضافر العوامل التي يمكن أن تضغط على أسعاره صعوداً أو هبوطاً، يبقى مصير المعدن الأصفر رهن التطورات المقبلة، تماشياً مع تراجع الاقتصاد الصيني والتوقعات باحتمال رفع الفيدرالي الأميركي أسعار الفائدة مرتين هذه السنة.

وتجتمع عوامل عدّة للتأثير إيجاباً على أسعار الذهب، كالتحفيز النقدي مثلاً الذي يعتمده حالياً كلٌّ من المصرف المركزي الأوروبي والمصرف المركزي الصيني، وتراجع أسعار النفط الذي ينشط عملية اللجوء إلى الملاذات الآمنة، لأنَّه يحضّ المستثمرين في العقود الآجلة للنفط على وضع حد لهذا الإستثمار، فضلاً عن أنَّ ارتفاع نسب الفوائد في الولايات المتحدة الأميركية وتالياً تعزيز الطلب على الدولار، يدفع أسعار المعدن الأصفر نزولاً. وفي هذا السياق، يشير بو سليمان إلى أنَّ حالات التضخم كفيلة هي أيضاً برفع أسعار الذهب، في حين أنَّ الانكماش يهبط بها. أخيراً، يتوقع بو سليمان أن تراوح أسعار الذهب خلال السنة الجارية بين 950 و1200 دولار نتيجة التضارب الحاصل في الأسواق العالمية، إذ إنَّ الإقتصاد الأميركي أظهر بوادر تعاف، في حين أنَّ نظيره الصيني تراجع، بما يضمن عدم سلك الأسعار مساراً انحدارياً وتالياً تحقيق التوازن في السوق .

Diacritics are included by authors to facilitate reading.

Among the 404 words, 50 (in red above) are partially vowelized: 38 with one diacritic and 12 with two vowels. The 50 diacritics are: 26 *-AF*, 23 *G*, 10 *a*, 2 *u*, 1 *-N*.

In *Annahar* (Beirut) and *Al-Hayat* (Saudi Arabia), which are reference newspapers in Arab countries, the percentage of partially vowelized words is often estimated to 2-3 percent[4], but this rate also depends on the journalist and the field, as articles on special topics tend to include more diacritics.

The *-AF* ending is used to mark the accusative or the adverbial POS that may be confused with the dual if the *F* is omitted.

The *-Ga-* sequence is often used to disambiguate between conjunctions: *InGa, OnGa, Ono*

The *-G-* gemination diacritic is often used in 2 or 3-letter words, such as in quantifiers or bi-literal verbs, but also to avoid confusion between simple tri-literal and derived tri-literal verbs.

Fig. 1. An extract from *Annahar* of 13 January 2016 with partial vowelization (http://www.annahar.com/article/301388)

In Section 2, we present previous work about building ALR and the (un)reliability of these resources for diacritic restoration. In Section 3, we make a general presentation of Arabic-Unitex as a full-form diacritized ALR. In Section 4, we detail our solutions in Arabic-Unitex for diacritic omission rules and related typographical issues. In section 5, we present the Arabic-Unitex tagset, lexicon figures and performance. In section 6, we detail our compression algorithm for Semitic languages and our algorithm for restoring Arabic vowels for words (out of context) through omission-tolerant dictionary lookup.

2 Previous Work

Studies focusing on diacritics in Arabic Speech Technologies, and especially in Text-to-Speech (TTS), are numerous since restoring omitted vowels is critical for syllabification. TTS systems *inevitably* contain such functionality for restoring vowels; whereas this functionality is *optionally* included in systems processing written text. Zitouni et al. (2006)[5] report Word Error

[4] According to our corpus study of 6930 words from the *Annahar* newspaper, 209 words (3%) include at least a diacritic (Neme, 2011, Section 4.2).

[5] "The lack of diacritics may lead to considerable lexical ambiguity that must be resolved by contextual information, which in turn presupposes knowledge of the language. It was observed in (Debili et al., 2002) that a

Rates (WER) in diacritization ranging from 10 percent for lexical diacritics to 25 percent where case endings are included.

Contrariwise, in Arabic Natural Language Processing, few papers are dedicated to Arabic vowelization, *"still largely understudied in the current NLP literature"* (Maamouri et al , 2006). There are many reasons: *"since non-diacritized text prevails, the Arabic NLP community seems to have accepted using it as the de facto 'real world' information material without feeling an obligation to question its choice/use, even espousing the idea sometimes that the robustness of software algorithms can deal with the problem and reduce the negative effect of the missing information on their research." […] "The prohibitive cost and the usually unequal and questionable quality of human/manual diacritization have led the scientific Arabic NLP community and its sponsors to focus more on volume of unvowelized data so far"* (Maamouri et al , 2006).

One may wonder if Arabic Speech Technologies Speech-To-Text (STT) and Text-To-Speech (TTS) approaches to diacritization might be adapted to written text technologies. But TTS and written text processing approaches to restoring diacritics use similar techniques: rule-based, statistical, and hybrid approaches; and they face the same challenges: sparseness of data since Arabic is morphologically rich and agglutinated, Out-Of-Vocabulary tokens, scarcity of modern Arabic vowelized resources, etc. Thus, there is no reason to speculate that adaptation of current TTS technologies might bring about any key innovation in diacritization of written text.

Alternatively, STT might be used to overcome the present scarcity of diacritized corpora in Modern Standard Arabic, by implementing an ambitious programme of accurate transcription of audio recordings of formal news. However, such an undertaking would involve post-edition, and even with massive investment, would probably not remedy more than *partially* the lack of training data. Therefore, the availability of more training data will not dispense from exploiting large coverage lexicon and accurate grammatical rules. *"Hybrid approaches in many surveyed systems perform better as these techniques are guided by language-dependent rules […] Inflection property of Arabic may cause many words to be unseen in learning phase.[…] Pure statistical approaches usually give unsatisfactory performance with unseen data, especially in complex languages that suffer from sparseness as is the case with Arabic, a highly inflected language. This sparseness may cause training data to be insufficient."*.(Azmi and Almajed, 2015, Section 5)

2.1 ARACOMLEX (2006-2015)

Not only have commercial packages failed in handling vowels but also research groups have omitted vowels in ALR, such as AraComLex 1.0. *"The decision to ignore diacritics was taken after examining a corpus of 4.5 million Arabic words, where only 54 (sic) words were found to carry meaningful diacritic marks, which is statistically insignificant."* (Attia A. Mohammed, 2006).

non-diacritized dictionary word form has 2.9 possible diacritized forms on average and that an Arabic text containing 23,000 word forms showed an average ratio of 1:11.6."

In this sub-section, we discuss the extended version of *AraComLex* (Attia et al., 2011, 2015) because of its representativeness: recently created, available publicly, well documented and based on a sound methodology, it may be considered to represent the current state of the art in the domain of ALR; and a new trend attempting to build a full coverage of an ALR. We also mention some resources derived from *AraComLex*.

AraComLex 1.0 (Attia, 2006) has 10,800 lemmas; Attia et al. (2011) have increased semi-automatically their resource to reach 30,587 lemmas, arguing that creating a lexicon is time-consuming: *"Creating a lexicon is usually a labour-intensive task. For instance, Attia took* ***three years*** *in the development of his morphology, while SAMA and its predecessor, Buckwalter's morphology, were developed over* ***more than a decade****, and at least seven people were involved in updating and maintaining the morphology. […] and [we have built] a large-scale open-source finite-state morphological transducer for Arabic, AraComLex, that contains 30,587 lemmas. AraComLex generates 12,951,042 words."* According to the authors, the lexical coverage rate for general news or semi-literary text is around 86%. They add, *"The quality and coverage of the lexical database determines the quality and coverage of the morphological analyser, and limitations in the lexicon will cascade through to higher levels of processing […]"*

A common method to create a reliable reference list of words for a language is inspired from corpus linguistics: it consists in collecting corpora of several gigabytes, removing duplicate words, and validating the unique words semi-automatically. But, as Attia et al. (2015) notice: *"due to the richness and complexity of Arabic morphology, there is no corpus, no matter how large, that contains all possible word forms. Given a word in Arabic, one can change its form by adding or removing yet another prefix, suffix, proclitic or enclitic. This is why a morphological generator is essential in creating an adequate list of possible words."* (Attia et al., 2015).

Generation of word forms with affixes and clitics is required, indeed. However, it does not resolve another shortcoming of the corpus-based approach: this approach limits the coverage of the dictionary to that of the corpus.

2.2 BAMA (2002)

The well-known Buckwalter Arabic Morphological Analyzer (BAMA) is one of the best Arabic morphological analyzers and is available as open source. The BAMA lexicon is considered the baseline of Arabic computational processing. The BAMA uses a concatenative lexicon-driven approach (Buckwalter, 2002) based on three lexica, labelled A, B and C, where B is a multi-stem lexicon, and on a lookup algorithm based on compatibility constraints within the string ABC. In order to match a surface form, the parsing algorithm uses the lexicon's unvowelized stem field and the corresponding ad-hoc category provided in the lexicon: it selects compatible (proclitics and) prefixes and suffixes (and enclitics) in two precompiled lists (cf. Neme, 2011, Section 2).

Buckwalter (2007, 3.6) explains the advantage of BAMA (2004) compared to the Beesley-Xerox solution (Beesley, 1989-2001). The latter is an intricate solution based on twelve lexica, the traditional root-and-pattern model, two-level FST morphology, a large pool of rules formalized to be used with XFST and a lookup algorithm slowed down mainly by the pool of rules. We do agree on Buckwalter's critics to the Beesley-Xerox solution. Even with an

important team and support, it is not viable (see Neme & Laporte, 2013 section 2, and "*On the Misuse of Finite State Technology in Semitic Languages: Hebrew and Arabic*", 30 pages, to be published).

The Buckwalter stem-lexicon is constituted by 92,814 stem lines representing 41,178 lemmas, which amounts to a ratio of 2.27 stem/lemma. As an example, Table 2.2 shows the encoding of the lemma '*>aSiyl'* 'authentic' أصيل with its broken plural which admits three orthographic variants determined by case and agglutinated enclitics: *'>uSalA&-u_hu'* (nominative) أصلاؤه , *'>uSalA'-a_hu'* (accusative) أصلاءه , *'>uSalA&-i_hi'* (genitive) أصلائه . Inflectional attributes values are assigned through values attached to affixes.

Table 2.2. Stem-based representation of the adjective >aSiyl in the BAMA lexicon

LEMMA_ID	**unvowelized stem**	vowelized stem	**Morphological category**	**parser Output form/POS**		Line #	Feature values	initial alif grapheme	in Arabic
>aSiyl_1	**>Syl**	>aSiyl	**N/ap**	**>aSiyl/ADJ**		1	sing; sing+pro	hamza-above (O)	أصيل
>aSiyl_1	**ASyl**	>aSiyl	**N/ap**	**>aSiyl/ADJ**		2	sing; sing+pro	bare-alif (A)	اصيل
>aSiyl_1	**>SlA'**	>uSalA'	**Ndip**	**>uSalA'/ADJ**		3	plu; plu-acc+pro	0	أصلاء ه
>aSiyl_1	**ASlA'**	>uSalA'	**Ndip**	**>uSalA'/ADJ**		4	plu; plu-acc+pro	A	اصلاء ه
>aSiyl_1	**>SlA&**	>uSalA&	**Nuh**	**>uSalA&/ADJ**		5	plu-nom+pro	O	أصلاؤ ه
>aSiyl_1	**ASlA&**	>uSalA&	**Nuh**	**>uSalA&/ADJ**		6	plu-nom+pro	A	اصلاؤ ه
>aSiyl_1	**>SlA}**	>uSalA}	**Nihy**	**>uSalA}/ADJ**		7	plu-gen+pro	O	أصلائـه
>aSiyl_1	**ASlA}**	>uSalA}	**Nihy**	**>uSalA}/ADJ**		8	plu-gen+pro	A	اصلائـه

In Table 2.2, only the fields in **bold** are used directly by the BAMA parser, the other fields are for managing the lexicon and the last three columns are notes by the authors of this paper. The +pro feature indicates a variant with a mandatory pronoun and its absence a form used without a pronoun: the third and fourth lines represent variants in the plural without pronoun in whatever case, or in the accusative with a pronoun. Note the redundancy between unvowelized/vowelized stem fields. There are duplicates, for example the fifth and sixth lines: both of them represent plural nominative forms with a mandatory pronoun, the only difference being the omission (*A*, bare-alif) or not (*O*, hamza-above) of the initial glottal stop.

In the stem-based approach to the lexicon, a noun with broken plural (BP) and ending glottal stop normally requires four stem forms: one for the singular form and three for the BP. The three BP forms are the stem variants depending on the noun case and the occurrence of a

pronoun. But since the word '*>aSiyl'* may begin either with *bare-alif 'A'* or with *alif-with-hamza-above '>'*, it requires a duplication of stems in the lexicon, i.e. four more stem entries are necessary to handle the possible orthographies[6].

We have calculated the number of cases of initial *alif* spelling variation which require stem duplications in BAMA, which is the number of orthographic stem duplications related to an initial *O* (*alif-with-hamza-above*) or *I* (*below*) with the *A* (*bare-alif*) variant. The amount of added stems is 12,204 stems out of 92,814 (13%). This solution for initial glottal-stop diacritics is unsatisfactory. The redundancy of these additional stem-entries and of other duplicated fields (vowelized/unvowelized stem) is error-prone, and very unnatural to Arabic linguists, making the maintenance of the dictionary unnecessarily tricky. Duplication of entries in a manually maintained dictionary has the same drawbacks as code duplication in software engineering: it duplicates the effort required to detect errors, correct them and construct new items.

2.3 MADA (2007) and partial diacritization

Hamdi A. (2012) notes that almost all the morpho-syntactic taggers such as Buckwalter (Buckwalter, 2004), Xerox (Beesley, 2005) or MADA (Habash and Rambow, 2007) take as input texts with words partially diacritized, and remove all diacritics, and therefore do not exploit diacritics to disambiguate words. He implements for the MADA analyser (see Table 2.3.b) a solution which takes into account partial vowelization by excluding candidate analyses. The solution is built on the incompatibility between the partially vowelized surface forms and their lexical representation by means of the intersection of two Finite-State-Automata.

To assess performance, Hamdi A. (2012) uses six test sets derived from a single corpus of 25,000 words. The six test sets (in Table 2.3.b) differ as regards the percentage of partially vowelized words: 0%, 1.3%, 10%, 40%, 70% and 100%. The set with 1.3 percent of words is the original corpus, partially vowelized naturally by its authors; the set with 100% is a fully vowelized version, created manually; the other three partially vowelized sets are generated randomly from the fully vowelized set. The baseline of MADA, on the artificially de-vowelized set, is 84.25 percent (Table 2.3.a) of correct morphological analysis. On the set with 1.3 percent of vowelized words, the analysis improves to 84.91 percent. The improvement by 0.66 percentage point reflects the authors' intuitive partial vowelizing of difficult words to make reading easier.

Table 2.3.a. *MADA performance on a corpus of 25K words (from Hamdi, 2012)*

Criteria	Diacritization	Grammatical tagging	Morph. Analysis
Performance (read Accuracy)	**86.38%**	96.09%	84.25%

[6] In the HAMSAH Hebrew project (Wintner, 2008), an XML encoded lexicon, similar redundancies are observed: dotted/undotted. An example with the lexical entry of *bli* "without":

```
<item id="4917" translit="bli" dotted="xxd" undotted="xxu">
      <conjunction type="coord"/>
</item>
```

Table 2.3.b. Performance of MADA taking into account diacritics (from Hamdi, 2012)

Diacritization Rate	MADA Performances		
	Diacritization	Grammatical tagging	Morph. Analysis
1.3%	**86.97%**	96.41%	84.91%
10%	88.47%	96.79%	86.28%
40%	91.74%	97.12%	89.48%
70%	94.85%	97.33%	92.51%
100%	98.01%	97.49%	95.59%

The MADA research group also created the MAGEAD system (Habash, Rambow, 2006; Altantawy *et al*., 2010, 2011), implemented with FST technologies and a formalism that mixes inflexional classes and rule-based morphology.

The MAGEAD lexical data are borrowed from Buckwalter (2002): 8 960 verbs (Altantawy *et al.*, 2011:122) and 32 000 nouns and adjectives, admitting broken and suffixed plural (Altantawy *et al.*, 2010:854), but the coverage of broken plural nouns includes only a formalization of triliteral entries: 'we are not evaluating our lexicon coverage (…) Our evaluation aims at measuring performance on words which are in our lexicon, not the lexicon itself. Future work will address the crucial issue of creating and evaluating a comprehensive lexicon' (Altantawy *et al.*, 2010:856; see Neme & Laporte, 2013, Section 2.4.2, for more details). MAGEAD project's latest publication was in 2011.

2.4 MADAMIRA (2014)

MADA uses the BAMA lexicon and is based on the native algorithm of BAMA written in PERL. MADAMIRA (2014) is a new version of MADA also offering a coverage of the Egyptian dialect, and implemented in Java: "MADAMIRA follows the same general design as MADA with some additional components inspired from AMIRA"; it is thus "a system for morphological analysis and disambiguation of Arabic that combines some of the best aspects of two previously commonly used systems for Arabic processing". MADAMIRA is "implemented in Java, which provides substantially greater speed than Perl and allows new features to be quickly integrated with the existing code." The reference to Perl alludes to the lexicon and algorithm of BAMA (2002): any implementation using the BAMA lexicon is dependent of the BAMA native algorithm, so MADAMIRA had to reimplement this algorithm in Java.

MADAMIRA uses SAMA 3.1 (2010, https://catalog.ldc.upenn.edu/LDC2010L01), an enhanced version of BAMA involved in the Arabic Treebank. Proclitics/prefixes and suffixes/enclitics in SAMA were extended compared to BAMA, but the lexical coverage remains almost the same with lemmas, instead of the 38,600 lemmas in BAMA (2002). The

goal of MADAMIRA is apparently the implementation with Java of the disambiguation with statistical approaches.

Table 2.4.a Evaluation of MADAMIRA accuracy (From Table 3, MADAMIRA, 2014)

Evaluation Metric	MADA	MADAMIRA	NOTES
EVALDIAC	86.4	86.3	EVALDIAC: Percentage of words where the analysis chosen by MADAMIRA has the correct fully diacritized form and an exact spelling
EVALLEX	96.2	96.0	EVALLEX: Percentage of words where the chosen analysis has the correct lemma
EVALPOS	96.1	95.9	EVALPOS: Percentage of words where the chosen analysis has the correct part-of-speech
EVALFULL	84.3	84.1	EVALFULL: Percentage of words where the analysis chosen by MADAMIRA has all the features above [EVALDIAC + EVALLEX + EVALPOS].

In all metric aspects, MADAMIRA represents a deterioration of accuracy compared to MADA for Standard Arabic. Moreover, MADAMIRA does not take into account Hamdi's critics of MADA (2005).

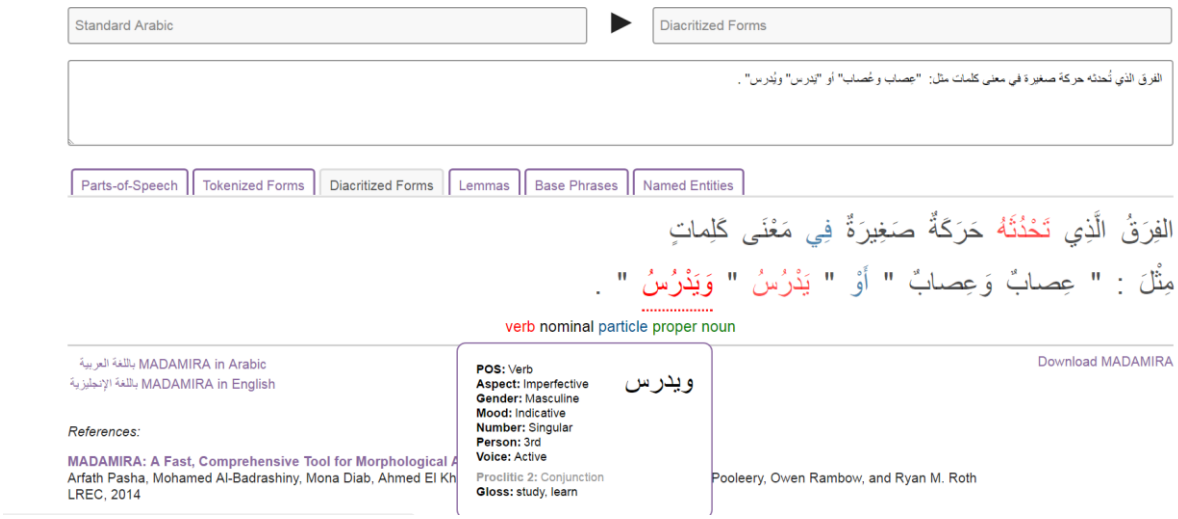


Fig. 2.4. Screenshot of MADAMIRA with an input sentence (translation in English: *That difference, a small vowel makes it happen in the meaning of words such as 'of ligature' vs. 'of neurosis' or 'studies' vs. 'is studied'*) and diacritized output. The popup window is the tagging of the verb *wa_yadorusu* "and _learn". Source: https://camel.abudhabi.nyu.edu/madamira/?locale=en

Fig 2.4 is a screenshot of a 14-word sentence tested with MADAMIRA. Tables 2.4.b and 2.4.b-bis detail the tagging of this sentence and the output for 5 of its explicit vowels (underlined); vowels are **bold underlined** if explicit in the input, but removed and wrongly recomputed by MADAMIRA; they are and **bold** if omitted in the input and wrongly computed by MADAMIRA. The grey-background columns display MADAMIRA outputs.

Table 2.4.b MADAMIRA vowelization and tagging output details for sentence in Fig. 2.4

Line	Transliteration	Input Text	MADAMIRA Output: Diacriticized Text	(should be)	MADAMIRA Output: Diacriticized Text	Meaning	Meaning selected by MADAMIRA
1	Alfrq	الفرق	الفِرَقُ	Alfaroqu	Alfiraqu	the_difference	**the_groups**
2	AlJy	الذي	الَّذِي	AlGaJiy	AlGaJiy	that(masc-sing)	that (masc-sing)
3	t**u**Hdvh	تُحدثه	تَحْدُثُهُ	t**u**Hodivahu	t**a**Hoduvuhu	(she)makes-happen_it	**happens_it**
4	Hrkp	حركة	حَرَكَةُ	HarakapN	HarakapN	(a) vowel	(a) vowel, motion
5	sgyrp	صغيرة	صَغِيرَةٌ	sagiyrapN	sagiyrapN	small	small
6	fy	في	فِي	fiy	fiy	in	in
7	mEnY	معنى	مَعْنَى	maEonaY	maEonaY	meaning	meaning
8	klimAt	كلمات	كَلِماتٌ	kalimAt**K**	kalimAt**N**	words (nominative)	of words (genitive)
9	mvl	مثل:	مِثْلَ	mivola	mivola	like	like
10	E**i**SAb	عِصاب	عِصابٌ	EiSAb**K**	EiSAb**N**	ligature	of ligature
11	wE**u**SAb	وعُصاب	وَعِصابٌ	waE**u**SAb**K**	waE**i**SAb**N**	and_neurosis	**and_of_ligature**
12	Ow	أو	أو	Oawo	Oawo	or	or
13	ya_drs	يَدرس	يَدْرُسُ	ya_dorusu	ya_dorusu	studies	studies
14	wy**u**drs	ويُدرس	وَيَدْرُسُ	way**u**dorasu	way**a**dorasu	and_is_studied	**and_studies**

Table 2.4.b-bis Complementary notes on MADAMIRA output. The line numbers refer to the lines of Table 2.4.b

Line	**Notes on the diacritics computed by MADAMIRA (wrong/correct)**	**Notes on agreement mismatch and other discrepancies**
1	Selection of a wrong lemma *firaq /faroq*	*firaq*: broken plural of *firqap*. In this situation, words in grammatical agreement with this one are in the feminine singular
2		PRONOUN: agreement mismatch with the noun selected as coreferent (line 1): masc_sing/fem_sing
3	After the removal of *u*, selection of the wrong verbal lemma *Hdv/Ohdv*, "happen/makes_happen"	"happen" is an intransitive verb, the agglutination of a clitic pronoun (here, object pronoun) is wrong.
8	Wrong case ending *N* instead of *K* (nominative/genitive)	
9	Wrong value of definiteness: 'construct state' (*mudaf*)[7], *mivola*	Mismatch between the features and the case-marking diacritic: if in the construct state, *mivol**a*** should be

[7] The three values of definiteness in Arabic are definite, indefinite and construct state. A noun is in the construct state if it has an adjunct in the genitive.

	is correct	in the genitive case *mivol**i***
10	*EisaAbN/EisaAbK*	case ending must be genitive instead of nominative
11	After the removal of *u*, selection the wrong lemma, although the other entry exists in BAMA	Case ending must be genitive (*-K*) instead of nominative (*-N*)
14	After the removal of *u*, selection of the wrong voice of the verb: active instead of passive	

MADAMIRA removes all diacritics, recomputes them according to the BAMA lexicon and algorithm, and finally selects a solution from the available candidates: "*Input text enters the Preprocessor, which cleans the text and converts it to the Buckwalter representation used within MADAMIRA. The text is then passed to the Morphological Analysis component, which develops a list of all possible analyses (independent of context) for each word. The text and analyses are then passed to a Feature Modelling component, which applies SVM and language models to derive predictions for the word's morphological features*" (Section 3, Pasha et al., 2014).

In the example, four meanings (in **Bold** in Table 2.4.b) are wrongly selected by MADAMIRA. The agreement between the relative pronoun and the BP is incorrect (Table 2.4.b, line 2). The correct grammatical agreement between a broken plural and an adjective sets the adjective in the feminine singular. MADAMIRA finds correctly the related singular form, but systematically selects the masculine-singular form of an adjective following a broken plural instead of the feminine-singular form.

According to the authors, MADAMIRA has 86.3 % of words well diacritized, an improvement compared to 82.7%, which is the precision of Zitouni et al. (2006). On the other side, it has 84% of precision in disambiguation (EVALFULL). This means about two tagging errors per line in a text. In a pipeline of NLP, we estimate MADAMIRA useless with such an error rate.

To sum up, MADAMIRA computes erroneous vowels, omitted in the input; and it removes correct ones written in the input and replaces them by erroneous ones, which is more shocking since such errors are obviously evitable. Finally, its language model fails to capture some dependencies between adjacent words.

Like Madamira, Farasa (Mubarak and Darwish, 2014) removes first the presumably valid diacritics from the source text and recomputes autocorrected words according to its processing pipeline. It seems that the autocorrected words are recalculated based on "common typographical mistakes", such as the final *h/p* (Table 2.5, line 1) or y/Y (line 2), very likely combined with a rough frequency of tokens without taking into account word segmentation. In Table 2.5, we show three examples submitted to Farasa (http://qatsdemo.cloudapp.net/farasa/demo.html):

Table 2.5 FARASA: Three examples with *G* diacritics deletion and auto-correction

Line	Input Text	FARASA autocorrected text	Transliteration	FARASA Transliteration	Meaning	FARASA Meaning selected
1	سيّده	سيدة	sy**G**dh	sydp	master_his	(a) lady
2	التقيّ	التقى	AltqyG	AltqY	the_devot	(he) meets

3	يحدّثونها	يحددنها	yhd**G**vwnhA	yhddnhA	talk(they-masc)_her	Defines (they-fem)_her

In the words in lines 1, 3, the reader must restore a gemination diacritic in syGdp and yhdGdnhA, not explicitly given by Farasa resources; and in line 3, besides removing the valid *G* diacritic, the processing removed 2 other letters, replacing the masculine plural form by the feminine plural of another verb lemma.

Hamed et Torsten (2017) compare Farasa to Madamira: their Table 11 (annotated WER subcategories) shows that errors for both systems are **mainly** related to diacritics, 13/16 errors for Farasa and 14/18 for Madamira. The paper concludes: "*We find that FARASA is outperforming MADAMIRA in both evaluation modes, but that* ***in relaxed mode the simple dictionary lookup baseline is surprisingly strong****. In general, our error rates are much higher than the ones reported in the literature and we currently have no satisfying explanation for the difference*".

Zalmout & Habash (2017) present a model for Arabic morphological disambiguation based on Recurrent Neural Networks (RNN); "*adding learning features from a morphological analyzer to model the space of possible analyses provides additional improvement.*". Compared to MADAMIRA, the accuracy of the system with RNN improves from 85,6% to 90%. They evaluate the accuracy for out-of-vocabulary words separately, as 7,9%: globally, the accuracy is in fact 77%; therefore, the accuracy is almost 96% for words in the vocabulary. They conclude *"that enriching the input word embedding with additional morphological features increases the morphological tagging accuracy* ***drastically***". Nonetheless, a better coverage would increase even more the accuracy of the whole system.

"When considering full analyses, we observe that our system still makes some errors in words where MADAMIRA is correct. However, the number of times our system is correct and MADAMIRA is not is over twice as the reverse (MADAMIRA is correct and our system is not)". Explanations of why and how such dissimilarities and differences happen would be speculative. It seems the SVM approach of 2014 cannot benefit from the RNN approach in 2017, and reciprocally. ***This is a serious limitation for scientific improvements***.

2.5 Automatic diacritization with RNN (2015)

Abandah et al. (2015) present an Arabic diacritizer based on Recurrent Neural Network (RNN-LSTM). The processing is divided in two stages: the RNN transcribes the input into a fully diacritized sequence; then post-processing corrections are applied to overcome some transcription errors.

Since our purpose in this article is to propose linguistic resources with rich encoding that can be used in symbolic or statistical NLP pipelines, we describe below the related "light" linguistic operations in the post-processing stage.

The post-processing includes:

- *Sukun correction*: *o* (zero-vowel) diacritics are removed from the transcribed sequence. For example, the output *Al**o**tGaAlibu* is corrected to *AltGaAlibu*[8].

[8] Abandah et al. (2015) does not respect the orthographic representation in his examples, so we have transcribed the examples given according to TB++ encoding which is a mapping one-to-one (cf. footnote 1).

- *Fatha correction*: The letter that precedes *A, Y, p* always has the short vowel *a* or *Ga*. If such a letter in the output sequence has a short vowel other than *a*, it is corrected to *a*. For example, the output *AltG**u**Alibu* is corrected to *AltG**a**Alibu.*
- *Dictionary-based correction*: "A dictionary is consulted to check whether the output word is in this dictionary. This dictionary is built from the training data and is indexed by the non-diacritized version of the word." The dictionary is 3 million words (or twelve thousand pages) – see Table 1, mainly from the "Tashkila collection of Islamic religious heritage Books". Such an index is rudimentary for diacritization, because of its low coverage.

Table 2.5. From Table 7 of Abandah et al. (2015)

	Target	**Output**	**Notes target**	**Notes on Output**
3	yaSonaE-a	yaS**onG**aEa	Fabricates (he)-Subjunctive	Invalid word: invalid phonological sequence 'onG'
5	la tar-uwanGa haA	litarawonihaA	to_see-(you-mas-plu-Energetic) her	invalid token
6	wal**a**A	walA**a**	and_not	invalid typography: *A* is never with vowel, *a* must precede

Table 2.5 shows 3 sample sequences that have errors, out of six in Table 7 of Abandah et al. (2015). We show that the use of linguistic resources allows for avoiding such errors:

- *yaS**o**n**G**aEu* is an invalid token that may be detected if a dictionary offers the valid vowelized candidates to *ySnE*. Moreover, this word form ***breaks*** a major phonological rule: the diacritic *o* cannot precede a geminated consonant as in *onGa*.
- *li_tarawoni_haA* is ungrammatical[9] with an impossible verbal suffix *–awo**ni*** instead of *–awon**a***. The vowelized output for *ltrwnhA*, لترونها should be *la_taruwanGa_haA*[10]. The imperfect in the energetic mode is a rare form in Arabic. Here, it is the inflected form of a frequent verb meaning "to see"; but the two agglutinations make this form even more rare in current corpora. This token occurs in the Koran, and we have found only one occurrence in the ArabicCorpus, occurring in a quotation of the same Koranic verse. However, our resources predict this rare agglutinated form.
- Finally, it outputs *wa_lAa* instead of *wa_laA*, which is a typographical error.

Abandah et al. (2015) is one of the very few experimentations that makes almost no use of Arabic linguistic knowledge. Such extreme usage of Machine Learning techniques in Arabic NLP shows bluntly its flaws and its limits. Statistical techniques are able to learn from aligned data made of character strings such as (*ySnE*, *yaSonaEa*), but they are unable to learn that *yaSonaEa* is a verb and its lemma is *SanaEa* with such data. It is no surprise that without comprehensive linguistic knowledge, such technology generates invalid word forms, even worse, it generates strings that are phonologically and typographically invalid. In addition,

[9] If the subordinate conjunction *li* is retained, *li_tar-awona_haA* is ungrammatical too, because of the presence of *na*.

[10] This token is validated by our resources (agglutination grammars and full-form dictionary): our parser restores the vowels, recognizes three agglutinated segments and relates the stem with the verbal lemma "to_see": لَتَرُوَنَّهَا {لَ.PART_la} {تَرُوَنَّ,رأى.V:aI2mpE} {هَا,ه.PRO+Ppers+Acc:3fs}. (see, Neme, 2011). For all these 3 examples, if our resources are applied upstream in an NLP pipeline, they provide the right candidates; if downstream, they reject the ungrammatical output forms.

building a lexical resource is a better investment than dedicating an equivalent effort to manually annotating a corpus, because a comprehensive dictionary is valid for long and for many domains. The existing entries of the dictionary need not be edited as long as the behaviour of the words don't change, whereas a new corpus must be annotated every time you change domains.

Finally, Abandah et al. (2015) admit they *"expect that providing the* ***morphological analysis*** *of such words to the RNN (Recurrent Neural Networks) would provide it with better information to achieve higher accuracy".*

2.6 AlKhalil-2 resources (2016)

Boudchiche et al. (2016) present AlKhalil-2, a second version of AlKhalil-1 (Boudlal et al., 2010), a morpho-syntactic analyser for words taken out of context. AlKhalil-2 recognizes successfully partially or fully vowelized forms and eliminates incompatible analyses. The output provides for each word: a lemma field (inexistent in AlKhalil-1), rich inflectional attributes, traditional derivational POS labels, and some semantic labels proper to traditional Arabic morphology[11], such as temporal-locative nouns, associated usually to some derivational patterns. Finally, output labels are wordy (and in Arabic), which hinders integration in a NLP pipeline, as compared to mnemonic abbreviations.

The lexicon is in XML format and based on a root-and-pattern approach similar to SARF (Al-Bawab et al., 1994). Like SARF, the AlKhalil-1 algorithm for identifying forms is based on root-and-pattern morpho-phonological rules that apply to all the entries of its lexicon; whereas AlKhalil-2 operates on the basis of a multi-stem approach similar to BAMA (proclitics-stem-enclitics). AlKhalil-2 is written in Java and evaluated on a vowelized corpus containing mainly Islamic religious heritage and old classical books, with a relatively small amount of diacritized Modern Arabic texts.

Compared to AlKhalil-1 (cf. Neme & Laporte, 2013, Section 2.4.3), AlKhalil-2 improved its lexical coverage and its speed also improved seriously to 632 word/second[12]. AlKhalil-2 is even quicker when analysing fully vowelized text since the text is less ambiguous.

AlKhalil-2 segments agglutinated morphemes correctly and associates generally accurate inflectional attributes to words. The singular form (lemma field) is associated to its broken plural (BP) form, which was not the case in AlKhalil-1. Some of the awkward surface patterns in AlKhalil-1, such as *FaALa* فال associated to قال *qaAla*, were standardized to *FaEaLa* to correspond to the traditional patterns, but many awkward others still remain. For some difficult cases, more accuracy and improvements are necessary in computing the associated pattern. For

[11] The derivational tradition that associates semantic features to patterns is not reliable. As Al-Khalil-2 takes for granted this traditional morphology, it inherits the same flaws: for instance, it labels *muxaTGaT*, "plan, plot" مخطط as a temporal-locative noun.

[12] AlKhalil-2 performance is calculated on the basis of word types in texts not word occurrences. Words in a text are sorted; then the sorted list of word types (agglutinated or not) are labelled and presented to the user. However, the standard in NLP is to associate to each word occurrence the adequate labels, to keep the pair occurrence/labels text order. The output presentation is not standard in NLP.

example, with some more difficult BP[13] forms involving two or more morpho-phonological alternations, the association of singular form fails, for example in *barobariyG* (singular), *baraAobirap* (BP) "barbar(s)".

The lexicon contains 215,508 lemmas: 42,656 for verbs and 172,852 nouns. The lexicon contains two root files for verbs and nouns with 7,500 roots each. These root bases generate 2,197,962 stems related to nouns and 1,903,541 stems related to verbs. Even if the authors standardized the patterns in the result presentation, behind the scene the concept of "surface pattern" remains in Al-Khalil-2. The lexical database contains a `VoweledStemCanonicPatternVerb` file with 1,756 vowelized patterns for (surface) stems of verbs. The `VoweledStemCanonicPatternNoun` file contains 8,042 vowelized patterns for (surface) stems of nouns (Boudchiche et al., 2014, Tableau 1, Boudchiche et al., 2016). There are two files for clitics: proclitics (67 compound elements, see Boudlal, 2010; Section 4.2) and enclitics (68 elements), sub-categorized by POS for nouns, verbs and common to both, as in BAMA.

The procedure for lookup into the lexical resource is complex with more than 20 steps: removing the diacritic but keeping a copy for checking incompatibility; operating a segmentation based on clitic compatibilities; analysing the stem for each valid segmentation:

- scanning non-derived word first (proper nouns);
- then scanning the stem of nouns (in five steps);
- then the stem of verbs (in five steps);

excluding invalid analyses via clitic compatibilities; excluding other analyses by using typographical rules. The result restores for each word the vowelized surface form with a rich tagging including root, pattern, POS and feature values, presented as CSV or XML format.

AlKhalil-2 is a new version of the lexicon of SARF and our remarks (Neme, 2011) still apply to it: "*The SARF project (Al-Bawab et al., 1994, http://sourceforge.net/projects/sarf/) is based on root-and-pattern representation. Starting from three-and four-consonant roots, it can generate Arabic verbs, derivative nouns, and gerunds, and inflect them. . […]. The project uses conventional programming techniques with the Java language and roots encoded in XML files. […]. The patterns are hard-coded in the form of Java code. […]; in addition, updating and correcting the language resource included in source code is complex since it involves two expertise: an Arabic linguist and a programmer; updating data and updating source code obey to different professional practices.*"

Besides, the number of 'voweled stem canonic patterns' for verbs and nouns is nearly 10,000. One may wonder how so many "stem patterns" are obtained and managed, and if there is a consensus in the team (linguists and computer scientists) around the (automatic maybe) attribution of such a "meta-morpheme" to each surface form. Moreover, many auxiliary fields are added to AlKhalil-2 databases, which makes it more complex.

[13] The coordinator of AlKhalil-1, Mansour Al-Ghamdi asked Alexis Neme during a conference in Beirut to evaluate AlKhalil-1. In May 2012, Alexis sent him an evaluation report (4 pages of technical report with annotated output from Al-Khalil1 in an Excel sheet). In this report, Alexis formulated such critics: awkward patterns, absence of the lemma field, etc. It seems that such critics were partially taken into account in AlKhalil-2.

Boudchiche et al. (2016, Section 5) claims "*AlKhalil-2 analyzer achieves a speed close to that of the fastest analyzer (632 words per second against 685 for BAMA analyzer). However, the speed coverage ratio is largely in favor of Alkhalil2 analyzer*". However, the difference in speed is rather due to the fact that the BAMA lookup algorithm is written in PERL, an interpreted language (rather slow); whereas AlKhalil-2 is written in Java, a compiled language.

In 2012, in order to compare our verbal lexicon, we tested Al-Khalil-1 on the first 553 occurrences of verbs of the same test collection extracted from the Nemlar corpus (Neme, 2011). 42 occurrences of verbs were unrecognized, which represents an error rate of 7,6 % in the lexical coverage of verbs. With Al-Khalil2, our evaluation noted a strong improvement in the verbal coverage with a fault rate down to 0.5%.

For global coverage, we evaluated Al-Khalil-2 lexical coverage with the same corpus (11,950 words) used for evaluating Arabic-Unitex (cf. 5.3.1). Before running the test, we changed all *I* to *A*. The coverage is less than 88% for Modern Standard Arabic texts. We repeated the experience with other MSA texts and found coverages ranging between 87% and 93%. Many common relational adjectives are missing such as "terrorist", "colonial" "Zionist"; singular forms are covered but not broken plural forms as common as "turtles" and "bishops". Moreover, although the University of Oujda is in Morocco, the words *Amazigh, Amazighian* are not in the lexicon.

2.7 Automatic diacritization with AlKhalil-2

Using AlKhalil-2, Chennoufi & Mazraoui (2016) present a diacritizer that uses *"a hybrid system for automatic diacritization of Arabic sentences combining linguistic rules and statistical treatments"*. The processing is divided in 4 stages:

- for each word, AlKhalil-2 outputs diacritized candidate form/tag pairs, out of context;
- phonological/syntactic rules are used to eliminate invalid surface diacritized forms and/or morpho-syntactic analyses of a word;
- HMM algorithms determine the most probable diacritized sentence;
- finally, the system deals with words not analysed by AlKhalil-2.

Examples of rules of step 2:

- Phonological rules: two *o* (zero-vowel) diacritics in two consecutive syllables are not allowed in Arabic, so that *min**o** (A)l**o**kitaAbi* (from the book) becomes *min**a** (A)l**o**kitaAbi*. This rule is in cross-word diacritization, where a word ends with *o* and the following word begins with the determiner *Al-*. Thus, this rule relies not only on phonology but on segmentation and tagging, as well.
- Syntactic rules: <PREP><NOUN:genitive>, meaning that after a preposition only the genitive case ending is allowed; for example, *mina Alomadorasat**i*** (from the school) is a valid utterance while *mina Alomadorasat**a*** is not valid. Similar rules are implemented for <CONJ-SUBORDINATION> <VERB>, …

The system also includes a typographical standardization[14] of diacritics (Section 4.2.1): "*The tanween fatha sign with the letter Alif " ا"/A/ has two forms of writing: one before the letter (سَلاَمًا salaAm**FA** (peace)) and the other after the letter سَلاَمًا salaAm**AF**). The second form has been adopted"* [15]. In addition, the point 1) in the same section includes 3 occurrences of *AlomAlyziywna* 'the-Malaysians', instead of the correct form *AlomAlyz**G**iywna*, missing the gemination mark *G*. Such repeated errors indicate carelessness for linguistic data. Nonetheless, this does not lessen the value of the experiments and evaluations of the HMM in diacritization with or without rules.

Table 2.7. Comparison between Arabic automatic diacritization systems[16] (Chennoufi, Mazroui, 2016, from Table 3, 4). WER1/2 = Word Error Rate with or without case ending diacritics

System	WER1	WER2
1st assessment		
AlKhalil-2-HMM	8.29	4.10
AlKhalil-2-rules-HMM	6.28	2.58
2nd assessment		
MADAMIRA-SAMA-SVM	27.29	16.14
AlKhalil-2-rules-HMM	6.22	2.53
3rd assessment		
Abandah et al. (2015)-RNN (Tashkeela corpus[17])	5.82	3.54
AlKhalil-2-rules-HMM (Tashkeela corpus)	4.45	1.86

Each assessment in the Table 2.7 reproduces the same evaluation metrics. The first comparison is between AlKhalil-2-HMM with or without rules and shows a better result (+2%) with rules.

[14] In newspapers, the most frequent variant is *–AF*; literature magazines (such as http://al-adab.com/, Evaluation *Section 5.3.1*) and reference books adopt the normative variant *–FA*, since the variant *–AF* is considered by normative grammarians as erroneous. In this case, the choice of variant (or typography) depends on editorial practices in a printing industry.

[15] Default rules for diacritics in Al-Khalil-2 are similar to Neme (2011, section 4.2), implemented but documented in the Unitex User Manual.

[16] The paper includes also an evaluation of the MS-Office plug-in *Arabic Authoring services*, with word error rates (WER1and WER2) of 20.56 and 11.18, better than MADAMIRA. We do not have access to the description of the *Arabic Authoring services*; nonetheless, the better performance of the plug-in is partly due to the lexical coverage of the Arabic resources of MS-Office, better than the embedded SAMA in MADAMIRA.

[17] http://sourceforge.net/projects/tashkeela/

About the comparison with MADAMIRA and Abandah et al. (2015), Chennoufi & Mazroui (2016) conclude that the good performances of the system are consequences of *"combining morphological analysis, syntactic and diacritic rules and* [of the] *large size of the corpus (used in statistical processing)".*

2.8 Conclusions and perspectives

As Attia et al. (2011) underline, *"The quality and coverage of the lexical database determines the quality and coverage of the morphological analyser, and limitations in the lexicon will cascade through to higher levels of processing*". This is true for diacritics too. The accusative suffix *-F* (pronounced [an]) is likely to help in the disambiguation of words, the gemination diacritic in selecting the right lemma of a verb (causative, for instance) or a noun, and the presence of a *u* after the first root consonant in the detection of a passive. Such inconspicuous information is valuable for disambiguation.

AlKhalil-2 eliminates analyses incompatible with the partially vowelized word but through lookups in several XML databases. Chennoufi & Mazroui (2016) demonstrate that *"combining morphological analysis, syntactic and diacritic rules used in a pipeline with statistical processing produces better performance than other systems",* including the RNN approach. No matter the approach, symbolic or statistical, one may expect a better result in disambiguization or vowelization with a better lexical resource in an Arabic NLP pipeline.

Hamdi (2012) demonstrates that statistical approaches were unable to give a satisfactory solution for partially vowelized words, whereas symbolic approaches propose a solution with disarming simplicity.

Our solution, which was implemented in November 2010, is similar to Hamdi's (2012). Nonetheless, Arabic-Unitex was built on a more radical basis: from the beginning, the lookup procedure ***retains only the candidates compatible*** with a partially diacritized word. The procedure uses a compressed finite-state automaton (FSA) and accesses the fully vowelized resource to discard the paths incompatible with the diacritics present in the text.

Arabic-Unitex uses FSTs intensively for inflection and takes into account all morphological and orthographical alternations to achieve a large lexical coverage of Arabic. The lexicon has been built and encoded manually. Arabic-Unitex consists of 76,000 lemmas and is inflected into 6 million fully vowelized forms, which are stored in an FSA data structure for fast retrieval through a lookup procedure. We evaluate the potential of recognizable agglutinated forms to more than 500 million valid forms if we count only fully vowelized forms, and to several billions of recognizable and valid partially vowelized forms.

In what follows, we will present briefly the overall architecture of Arabic-Unitex.

3 General presentation of Arabic-Unitex

Arabic-Unitex is a lemma-based, fully vowelized language resource with straightforward inflectional encoding based on the Semitic grammatical tradition and extended by independent agglutination grammars. In 2010, being aware of the four complications (cf. Section 1) facing the Arabic computational morphology, we adapted Unitex programs and tools to Arabic

traditional representation so that the resources may be more easily read and maintained by Arabic linguists. We have adjusted Unitex programs to deal with:

- inflection with Semitic patterns or infixes;
- agglutination of proclitics/enclitics;
- partial vowelization.

3.1 The PRIM Model

Inspired by the Semitic traditional root-and-pattern model, our model for Arabic morphology requires detailed lexical representation as well, but uses at the same time up-to-date algorithmic techniques (FSTs). Neme & Laporte (2013) introduce the ***pattern-and-root inflectional model*** (PRIM) for Arabic morphology. We define a *pattern* as a template of characters surrounding the slots (place-holders) for the *root* letters. Around the slots, patterns contain short vowels, and sometimes consonants or long vowels.

The breakthrough lies in the reversal of the traditional root-and-pattern Semitic model into pattern-and-root, giving precedence to patterns over roots. Traditionally, the analysis of an Arabic word begins by assigning it an **etymological** root, and the rest is the pattern[18]. We begin by instead recognizing the **inflectional** pattern of the word, and the remainder is the root. In the traditional analysis, the pattern combines derivational and inflectional information, including all the derivation of the word from its remotest root. With our innovation, it is purely inflectional. This change keeps the expressiveness of the traditional model, which has been tested and validated during ten centuries; additionally, it enables faster identification of the verbal entry, its root and its pattern, with a smaller margin of error; moreover, it avoids the definition of several hundred interdependent morphological, phonological and orthographic rules.

Pattern-and-root inflectional morphology is adequate to Arabic morphology. We keep inflection apart from derivational morphology. The PRIM inflectional sub-taxonomies for verbs, suffixed plural and BP are simple, methodical and detailed; they avoid shortcuts or over-simplifications. The PRIM model complies with the conventions of the Semitic traditional morphology and is understood quickly by Arabic-speaking linguists. The lexicon is organized in fully vowelized lexical entries, like traditional dictionaries; and not in stem entries, as in the multi-stem approach. A lexical entry in traditional dictionaries is a lemmatized entry as well, but entries with the same etymological root are indexed under this root, and roots are ordered alphabetically.

In the PRIM model, a pattern is a simple sequence of consonant slots, consonants and vowels (short or long), but is not used to represent a meaning or morpho-syntactic features attached to patterns. In PRIM, a root is merely a sequence of letters (usually consonants). Orthographical variations of the glottal stop are encoded in the same way. Root letter substitutions and insertions are restricted to *w*, *y*, *A*, and to glottal stop allographs. We deal with morpho-

[18] Smrz (2007) converges with us on the definition of root and diverges on the definition of pattern: "*The 'root' should not be understood in the sense of Semitic linguistics. Rather, it is the core lexical information associated with the lexeme and available to the inflectional rules.*" (p.31). Smrz creates the concept of morphophonemic pattern (surface pattern) which creates numerous patterns awkward to native speakers: "*Morphophonemic patterns and their significance for the simplification of the model of morphological alternations*" (p.13).

phonological alternations in a factual way: inflected forms are generated from their observable surface lemma, and not from a "deep" or "underlying" root.

An inflectional transducer is associated with each inflectional class in the taxonomy, and it generates all the inflected vowelized forms of any lemma in the class. Each lexical tag is accurate and informative and its format consists of a lemma followed by a set of feature-value pairs. Agglutinated clitics are analysed without the generation of artificial ambiguity. Clitic-agglutination grammars are described independently from inflection, in separate grammars. Morphological analysis of Arabic text is performed directly with a dictionary of words and without morphological rules: all orthographical variants are registered in the dictionary, which simplifies and speeds up the process.

The main challenge was to elaborate the inflectional model of pattern-and-root morphology based on Semitic grammatical tradition and our critical reading of Beesley's work (1991-2001), a generativist forerunner in Arabic computational morphology. If one can find attempts to build a systematic taxonomy for verbs in the Arabic morphological tradition already in the 10th century, it is the first time that the broken plural gets a straightforward and elegant representation based on three new principles crafted for encoding Semitic morphology. Moreover, they were complemented by concatenative encoding for regular suffixation to depict all aspects of morphological representation.

3.2 A full-form inflected dictionary

A line encodes one lexical entry in our lemmatized lexicon. The encoding contains a lemma followed by grammatical codes, and optionally comments. In order to facilitate direct human reading of the entry, the lemma is separated from the code by a simple comma, and the code from the comments by a slash. For regular plural, also known as sound plural, the inflectional transducer is designed to be used by the generator of inflected forms in the concatenative mode, which is the default mode.

The grammatical code contains sub-fields for singular, gender and plural, separated by hyphens:

`nufaAyap,N00ap-f-At/ نفاية 'rubbish'`

/ singular ending in *-ap* ("teh marbutah" in Arabic); feminine; plural suffix in *-At*

`manaAx,N0000-m-At/ مَنَاخ 'climate'`

/ singular with no particular suffix; masculine; plural suffix in *-At*

Our lemmatized lexicon produces fully vowelized forms by using FSTs based on a Semitic-style taxonomy for verbs (Neme, 2011) and nouns (Neme & Laporte, 2013).

The output format of an FST is `surface-form,lemma.V:feature-values` as in:

`takotubu,ktb.V:aI3fsN /active-Imperfect-3rd_Pers-fem-sing-iNdicative`

The '/' character comments out the text that follows it up to the end of the line.

For verbs, the *feature values* are detailed as in traditional morphology and in the following order:

- Voice: active (a), passive (b);
- Tense: Perfect, Imperfect, Imperative (Y);
- Person: 1, 2, 3;
- Gender: masculine, feminine;
- Number: singular, dual, plural;
- Mode: indicative (N), Subjunctive, Jussive, Energetic.

For nouns and adjectives, the *feature values* are in the following order:

- Gender: masculine, feminine.
- Number: singular, dual (d), sound plural (p), broken plural (q).
- Definiteness: definite (D), indefinite (i), and construct state (a).
- Case: Nominative, Accusative, Genitive.

The order between features is not significant, but our resources respect a fixed order, in order to facilitate human reading and therefore checking.

'*Distinct codes are required for broken plural (q) and suffixed plural (p) because rules of agreement between a plural noun and an adjective, a participle or a verb depend on whether the noun is a BP or a suffixed plural (Neme & Laporte 2013, pages 243-245).*'

3.3 Delimited Word Forms (DWF) grammars

A word delimited by spaces or punctuation symbols (DWF) is composed of a sequence of segments. A word or DWF is described in our resource of Arabic as the undelimited concatenation of clitics around an inflected form. Agglutination of morphemes in a word is represented by grammars. Each segment in a word will be called a morpheme[19]. The combination of a sequence of morphemes obeys a number of constraints which are expressed by a POS agglutination grammar. For instance, a verbal word is composed by one morpheme <*V*> or the concatenation of up to 4 morphemes as in:

<*CONJC*> <*CONJS*> <*V:inflected*> <*PRO+accusative*>

where <CONJC> is a coordinating conjunction, <CONJS> is a subordinating conjunction and <*PRO+accusative*> an agglutinated object pronoun.

<*CONJC*> combines freely with any inflected verb. The <*CONJS*> constraints the verb to the imperfect subjunctive or to the jussive. Finally, an inflected verb is often insensitive to the agglutinated pronoun (i.e. its form is not affected) but some forms are sensitive: for example, forms with a glottal stop as the third root consonant (for verbs, see Neme, 2011, Section 4.1; for nouns, see Neme & Laporte, 2013, Section 8).

In BAMA, agglutination of verbs is formalized by the following:

```
[<CONJC>][<CONJS>]<inflexional-prefix><V-stem><inflexional-suffix>[<PRO+accusative>]
```

where <V-stem> is the string common to a subset of inflected forms vis-à-vis the concatenative operations and where the morphemes between [] are optional.

Both Arabic-Unitex and BAMA provide a segmented and tagged morphemic representation of a text. However, there are 2 essential differences: (1) Arabic-Unitex segmentation is closer to tradition and (2) Arabic-Unitex lemma grouping is closer to intuition: for example, singular and broken plural are grouped under the singular canonical form in Unitex, but under two stems (at least) in BAMA. With a better grouping of lemmas, lemma counts in a text are closer to the distribution of meanings. Therefore, we obtain a better representation of a document for applications such as automatic summarization and topic extraction.

[19]The morphemic status of some segments is controversial. The pattern, the lemma, the case ending may also be analysed as morphemes or morphs (find a detailed discussion in Smrz, 2007, morph *versus* morpheme). However, calling each segment a morpheme simplifies the description.

3.4 Building the dictionary based on a paradigmatic and taxonomic approach

In elementary and middle schools of Arabic-speaking countries, children are supposed to know by heart tables of conjugation and to compute all variations of a noun according to gender, number, definiteness and case. Irregularities are learned at school and related with two characteristics of the lemma: its pattern and the nature of its root consonants; then, once pupils have identified the lemma and the 'weak' root consonants (*A, w, y* and glottal stop), they learn to handle case endings, letter deletion, etc. according to syntactic context or the presence of an agglutinated pronoun. In addition, rules belong to a hierarchy of priority, but the hierarchy adopted by grammar textbooks is not always explicit, and sometimes fuzzy or messy. In our approach to computational morphology, the ordered and hierarchical rules learned at school were replaced by a formalized, operational grammar and a straightforward taxonomy. Each inflexional class in our taxonomy is provided with all the corresponding paradigmatic variations of forms, similar to the conjugation tables learned at school by children[20].

In our computational representation and tools, we have respected most of those habits and teaching methods, because they are widely shared by Arabic native speakers, and consequently by most potential descriptors of Arabic. For example, our citation form or lemmatized entry is similar to traditional dictionaries: the perfect 3rd person masculine singular for a verb, and the masculine or feminine singular for a noun or an adjective; and the description of inflection is similar to the traditional one.

We have adjusted Unitex tools in order to facilitate the encoding of paradigmatic variations. We have created two Semitic sub-taxonomies relative to verb variations and broken plural variations; each was split in two large sub-taxonomies related to the number of root letters: triliteral or quadriliteral, which is compatible with the traditional morphology. At the end, we have designed more than 1,150 inflectional classes; those for verbs and broken plurals are based on the pattern-and-root model, and those for suffix inflexion of noun and adjectives on the concatenative model.

As inflectional classes are numerous, the main challenge in our approach was to guess and assign the right pattern-class and root-subclass to each lexical entry when manually building or updating the dictionary. In order to facilitate this task, we designed the scheme to be straightforward and systematic, so that, for a given entry, linguists guess the associated class quickly. The sub-taxonomies are defined according to POS first, then to pattern classes and root subclasses:

- A straightforward verbal taxonomy for conjugation models with 460 classes (Neme, 2011).
- A straightforward broken plural taxonomy with 400 classes[21] for nouns and 50 classes for adjectives.
- The 250 remaining classes are dedicated to nouns and adjectives with suffixed plural and other POS classes. This number is comparable to the number of classes for French

[20] See also http://babelarab.univ-mlv.fr, site in Arabic, for displaying tables of conjugation of 15 400 verbs including a table with an agglutinated pronoun, two tables for active and passive participles, and an Arabic spell checker with a unique feature for detecting invalid/misplaced diacritics.

[21] Neme and Laporte (2013) inventoried 300 inflexional classes for BP; this inventory increased with the lexicon extension to 4200 lemmas with BP instead of 3200 in 2013.

resources in Unitex.

The manual effort[22] towards the building of the lexicon may be schematically split into the following tasks:

- Typing-in the list of lemmas based on reference lists and dictionaries (checked mainly in Abdel Nour, 2006, as a reference dictionary).

- Encoding each lexical entry: POS and inflectional class.

- Hand crafting the 1,150 main graphs representing the inflexional classes and correcting each of them by checking the generated output, in part manually and in part automatically.

- Adding active and passive participles to the 460 graphs of the verbal inflection: 54 forms for active and 54 for passive.

- Generating automatically regular deverbal nouns (almost 10,000) and the related relative adjectives (almost 10,000) based on verbal lemma (V61-V70, V41-V42), taking into account 'weak' root consonant (*A, w, y* and glottal stop) alternations. These lists were filtered semi-automatically and checked manually.

- Validating codes, correcting typo errors, adding more classes….

- Enhancing the lexical coverage by processing corpora and by encoding valid words not found by Unitex.

3.5 Enhancing Lexical coverage

Fig 3.4 exemplifies the work involved to deal with a neologism: داعشيّ, the denomination of ISIS members in Arabic, in order to illustrate the task of extending the lexical coverage. This lemma has millions of hits in Google search with its masculine, feminine and broken plural forms[23]: `daAoEiMiyG:ms`, `daAoEiMiyGap:fs`, `dawaAEiM:q` (broken plural), `daAoEiMiyGaAt:fp`. An inflexional class for this neologism does not exist in our lexicon; however, we have found similar classes for (a) a triliteral noun ending in *–yG* '`kurodiyG,$N3yy-g-FvEvL-OaFoEaaL-123/ kurd`', admitting gender inflection, and for (b) triliteral nouns with the same pattern for broken plural. We made an inflectional transducer for (c) by combining parts of (b) for the masculine plural, and parts of (a) for the rest of the paradigm (Fig. 3.4). We named the new transducer and class with a similar combination.

a) `kurodiyG,$N3yy-g-FvEvL-OaFoEaaL-123/ kurd` كُرْدِيّ كُرْدِيَ أَكْرَاد
b) `taAobiE,$N300-g-FvvEvL-FaEaaLiB-1w23/ dependent` تَـأْبِـع تَـابِـع تَـوَابِعَ
c) `daAoEiMiyG,$N3yy-g-FvvEvL-FaEaaLiB-1w23/` داعشيّ دواعش داعشيّات

[22] The manual effort cannot be quantified with precision in man-years; however it was a part time (with ups and downs) occupation of 1 person from 2010 to 2016.

[23] Note that the suffixed sound plural form داعشيّون, *dAEMiyG-uwn* (33 500 hits, Google search in May 2018) looks somehow awkward to native speakers as compared to the broken plural (2 930 000 hits). BP is preferred for most new nouns and suffixed plural for most new adjectives (Neme & Laporte, 2013). Note also the BP diptotic case ending, Fig. 3.4 "`N:sfx:uaiuaa`", where the nunation is not allowed for indefinite; and the genitive case is with *–a* ending.

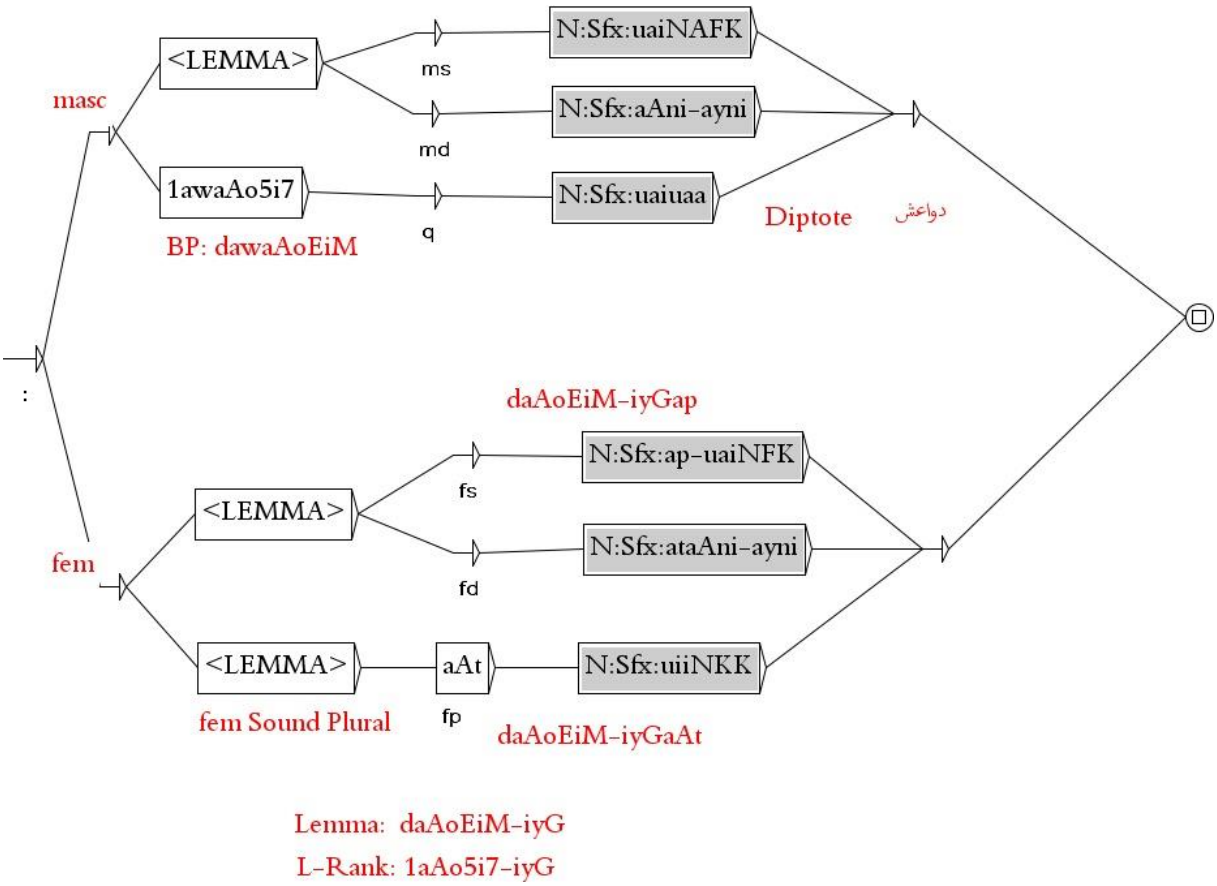


Fig.3.4. A new inflectional class for `daAoEiMiyG,$N3yy-g-FvvEvL-FaEaaLiB-1w23`

Even if many inflectional classes are replications with minor changes, creating 1,150 inflexional graphs (and 4000 sub-graphs, mainly for tenses and suffixed paradigms) was time consuming; besides, we have checked one by one the outputs of each inflexional graph. Summing up, the manual effort towards the building of the lexicon was to collect and type in each lemma, based on existing references dictionaries, verb lists, and results of corpus processing.

4 Vowel and vowel omission in Arabic-Unitex

4.1 Rules of vowel omission

Words in Arabic are often unvowelized and our system relies on our full-form inflected lexicon and agglutination grammars to restore the missing vowels. When Unitex uses a compressed Arabic lexicon that includes vowels, it is able to deal with unvowelized and with partially or fully vowelized words. If a word includes one or many diacritics, the lookup procedure extracts from the dictionary only the string candidates with the same diacritic(s) at the same position(s) as in the word, taking into account at the same time the predefined rules of diacritic omission.

A set of rules specifies in which conditions the lookup procedure tolerates vowel omission. In the Unitex folder for Arabic, the configuration file *Arabic-typo-rules.txt* defines rules for diacritic omission and other typography-related rules. The data distributed with Unitex contains

this file with predefined rules suitable for usual printed text (see appendix); you can enable or disable each rule to cope with more restrictive or less restrictive standards. The predefined rules are designed to be used with a fully vowelized dictionary. The analysis restores the corresponding form(s) stored in the dictionary.

Each rule has the form `RULE=YES/NO`. Here are examples of rules:

- Rules of omission of one vowel/diacritic:

```
                                / <dictionary_form> => <allowed_form>
                                / <E> stands for the empty string
fatha omission=YES                    / a =>  <E>
dammatan omission at end=YES          / N =>  <E> (N is pronounced [un])
                           / the kasra rule below is not in
                           /the predefined rules in the distributed data
kasra omission=NO                     / i => <E> rule disallowed
```

With the rules above, if *kitaAbN* is in the dictionary, *kitaAbN* matches it; *kitAb* and *kitaAb* also do; but **ktaAbN* doesn't, because *i* may not be omitted. [24]

- Rules of omission of two diacritics: When the word is fully vowelized, *G* is always followed by a short vowel (including *o* or a nunation). The following rules allow omitting *G*, but only if the vowel just after it is omitted too. Rules of Arabic script forbid to omit a *G* and write the vowel just after it:

```
shadda fatha omission=YES       /   Ga => <E>
```
*kat**Ga**ba => katba*
```
shadda dammatan omission at end=YES    /   GN  => <E>
```
*ruwsiy**GN** => ruwsiy* / رُوسِيٌّ

- Accusative marker inversion at the end of a word (*F* is pronounced [an]):

```
fathatan alef equiv alef fathatan=YES /at the end -FA => -AF
```
*kitab**FA** => kitab**AF***

```
fathatan alef maqsura equiv alef maqsura fathatan=YES
```
*fata**YF** => fata**FY*** /FY =>YF

- Substitution of initial *O* or *I (alif hamza)* by *A (bare alif)*:

```
   alef hamza above O to A =YES       / O => A
```
***O**akala => **A**akala*
```
alef hamza below I to A=YES           / I => A
```
***I**ikotub => **A**ikotub*

- Rare diacritics:

The presence or omission of the *R* superscripted variant of *alif* is handled by Unitex as well, e.g. in demonstrative pronouns.

```
superscript alef omission=YES    / R => <E>, R superscript alif
```
*h**R**JaA => hJaA* / هذا

*AllG**R**hu => AllGh* /الله

[24] An asterisk '*' indicates that a form is not in use in standard modern Arabic.

- Solar assimilation of *Al*: the assimilation of *l* to a coronal consonant (15 consonants/30) may be marked through the insertion of *G* after *Al<coronal-consonant>*: [25]

```
solar assimilation=YES
```

	/taAniy is in the dictionary
AltaAniy	/ allowed, assimilation not graphically marked
*Alt**G**aAniy*	/ allowed too, assimilation graphically marked

The coronal consonants, which admit assimilation, are the following:

ت, t; ث, v; ج, j; د, d; ذ, J; ر, r; ز, z; س, s; ش, M;
ص, S; ض, D; ط, T; ظ, l; ن, n; ه, h;

- Non-assimilation of *Al*: the assimilation of *l* to a non-coronal consonant (15 consonants/30) is disallowed in *Al<non-coronal-consonant>*:

```
lunar assimilation=NO   /check disallowed lunar consonant assimilation
```

	/qamaru is in the dictionary
Alqamaru	*/ allowed,*
*Alq**G**amaru*	*/ **NOT** an allowed form*

The non-coronal consonants do not admit assimilation and are the following:

ء, c; آ, C; أ, O; ؤ, W; إ, I; ئ, e; (all glottal stop variants)
ب, B; ح, H; خ, x; Z; ع, E; غ, g; ف, f;
ق, q; ك, k; م, m; و, w; ي, y; ا, A;

Table 4.1 illustrates the operation of the predefined Arabic typographical rules by giving the output of Unitex restoration. Each line in this table presents only one analysis, but in lines 3 and 4 Unitex produces several analyses.

Table 4.1. Restoration of vowels with the predefined rules. The TB++ and AR columns show the input

	TB++		AR	U N I T E X O u t p u t			
1	**Input**	**Notes**	Input	**Word with restored vowels**		**Lemma**	**POS:feats**
2	k**a**t**a**ba	All diacritics scripted	كَتَبَ	kataba	كَتَبَ	كتب	V:aP3ms
3	k**a**tb	2 diacritics omitted	كَتب	kat**a**b**a**	كَتَبَ	كتب	V:aP3ms
4	ktub	2 omitted	كتُب	k**u**tub**a**	كُتُبَ	كِتَاْب	N:qaA
5	kt**i**b	2 omitted	كتِب	k**u**tib**a**	كُتِبَ	كتب	V:bP3ms
6	kat**G**b	2 omitted	كتَب	katG**a**b**a**	كَتَّبَ	كتَّب	V:aP3ms

[25] The letter *l* of the determiner is still written, but pronounced in the form of the following consonant.

7	kt**aG**b	Ga -> *aG	كـتّب		Unknown		
8	Al**qG**mru	wrong 'Al-' assimilation	القـمرُ		Unknown		
9	Alqmru	no 'Al-' assimilation	القـمرُ	Alq**a**m**a**ru	قـمَرُ	قـمَر	N:msDN
10	Al**MG**msu	assimilation scripted	الـشّمسُ	AlM**G**amosu	شَمْسُ	شَمْس	N:fsDN
11	**A**ErAbN	allowed variant of I (hamza-under-alif)	اعرابٌ	**Ii**EoraAbN	إعْرَابٌ	إعْرَاب	N:msiN
12	**O**ErAbN	wrong variant of I (hamza-under-alif)	أعرابٌ		Unknown		
13	kitaAb**FA**	accusative marker, normative form	كِـتـابـا	kitaAb**FA**	كِـتَـابـا	كِـتَـاب	N:msiA
14	kitAb**AF**	allowed inversion	كِـتـابـاً	kitaAb**FA**	كِـتَـابـا	كِـتَـاب	N:msiA

Line 6 in Table 4.1 shows the form *katGb* where two vowels are omitted. Unitex dictionary lookup restores the vowelized full form *katGaba*, the related lemma *ktGb* and the morpho-syntactic tag V:aP3ms which means Verb in the active Perfect 3rd person masculine singular.

4.2 Inflected forms with short vowel variations

Arabic-Unitex takes into account short vowel variation in surface forms. This free variation affects the first vowel of some nouns. Three situations are common: *u/i/*a*, *a/u/*i* and *a/i/*u*; thus one may say ***nu****faAyap* or ***ni****faAyap* "rubbish" نفاية but not ****na****faAyap*. The lexicon could record the two allowed vowelized forms in two lemmas, but we have chosen to encode this information in the inflectional transducers. This is less redundant and we avoid an artificial ambiguity between two lemmas in morphological annotations. Moreover, we also have the same allowed variations in the dual and in the plural: ***nu****faAyataAn/****ni****faAyataAn* "two pieces of rubbish" نفايتان; ***nu****faAyaAt/****ni****faAyaAt* نفايات "pieces of rubbish" for sound plural. The encoding of such variations was achieved for almost a hundred of lexical entries and needs to be completed.

In this section, we describe how we encoded lexical entries and inflectional transducers for nouns without vowel variant; then for nouns with vowel variant; finally, we present the special case of broken plurals and a similar variation observed in the suffixed plural of some feminine nouns.

4.2.1 Inflection without variant

The following three lexical entries undergo the short vowel variation in question, but here is an encoding that overlooks the vowel variation:

```
nufaAyap,N00ap-f-At/ نـفـايـة       'rubbish'
manaAx,N0000-m-At/   مَـنـاخ        'climate'
HaDaAnap,N00ap-f-At/ حَضَانَـة      'kindergarten'
```

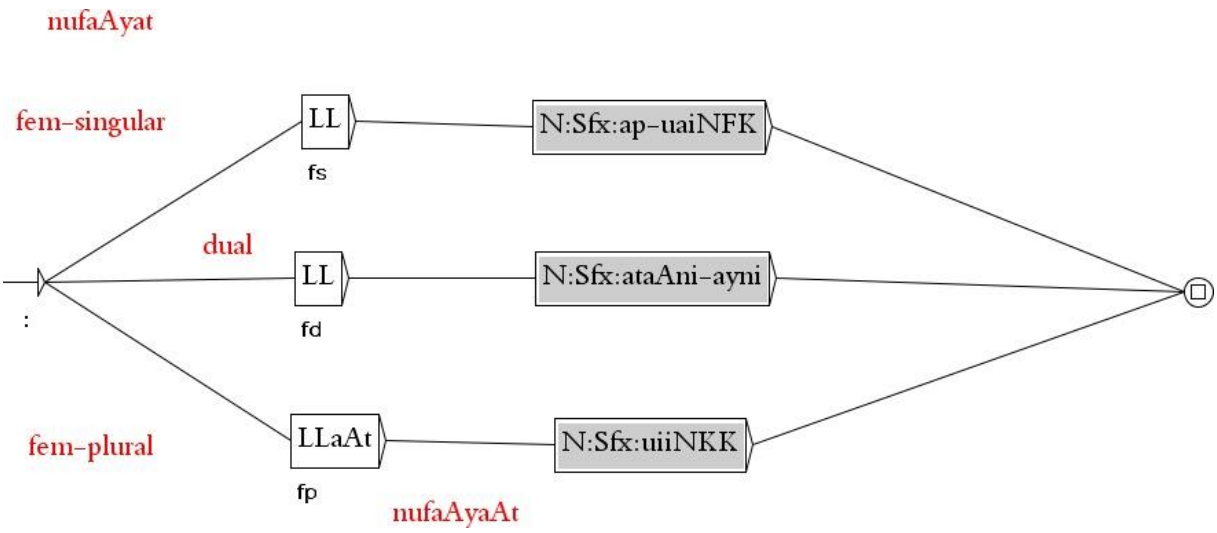


Fig. 4.2.1.a. An inflectional transducer in the concatenative mode for nufaAyap

Fig. 4.2.1.a shows the inflectional transducer for *nufaAyap* "rubbish"[26]. It contains three paths to produce singular, dual, and plural forms. The paths describe the suffixes to be added or removed to get an inflected form from a canonical form. The LL box (L is for Left shift) removes two letters from the end, here *ap*. The outputs (displayed under the boxes) are the inflectional codes to add to a dictionary entry[27]. A box not connected to another one is a comment or an explanation included in the transducer. A grey box is a call to a subgraph. In this graph, the subgraphs concatenate the suffixes of definiteness and case. For instance, the "`N:Sfx:uiiNKK`" subgraph (Fig. 4.2.1.b) represents the endings for the regular feminine plural.

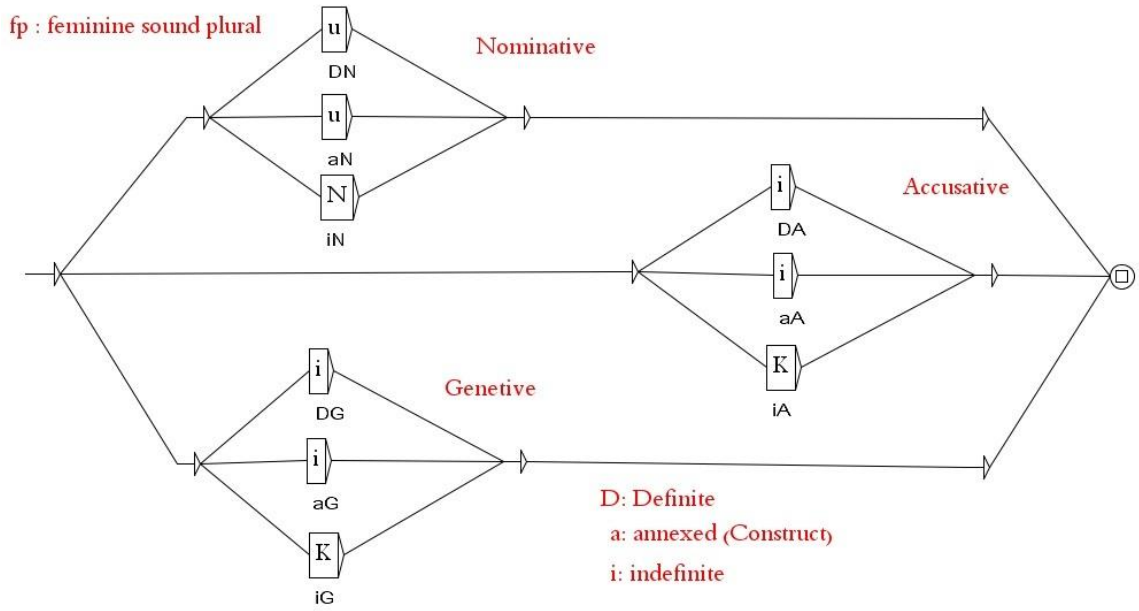


Fig. 4.2.1.b. The N:Sfx:uiiNKK subgraph relative to the 9 variations of feminine plural

4.2.2 Inflection with vowel variant

Here we describe our representation of short vowel variation. We use the generator of inflected forms in the Semitic mode, which is specified by the "$" symbol in the encodings below. We encode the vowel variation by inserting "`_v_`" in the grammatical code, where *v* indicates the alternate value of the first vowel. Below, the encoding of the same three entries as above, but with vowel variation.

```
nufaAyap,$N0_i_0ap-f-At/      نفـايـة      'rubbish'
manaAx,$N0_u_000-m-At/        مَنـاخ       'climate'
HaDaAnap,$N0_i_0ap-f-At/      حَضـانـة     'kindergarten'
```

[26] In this paper, we do not cover other free variations of short vowels such as the permutation of the vowels *a-i in mi**no**Ta**q**ap* and *ma**no**Ti**q**ap* "area" منطقة This variation may be written in the inflectional class as <1a2o3i4ap>.
[27] For a detailed description of inflectional transducers, see Unitex User Manual 3.1, Chap. 3.5, for concatenative and Semitic mode.

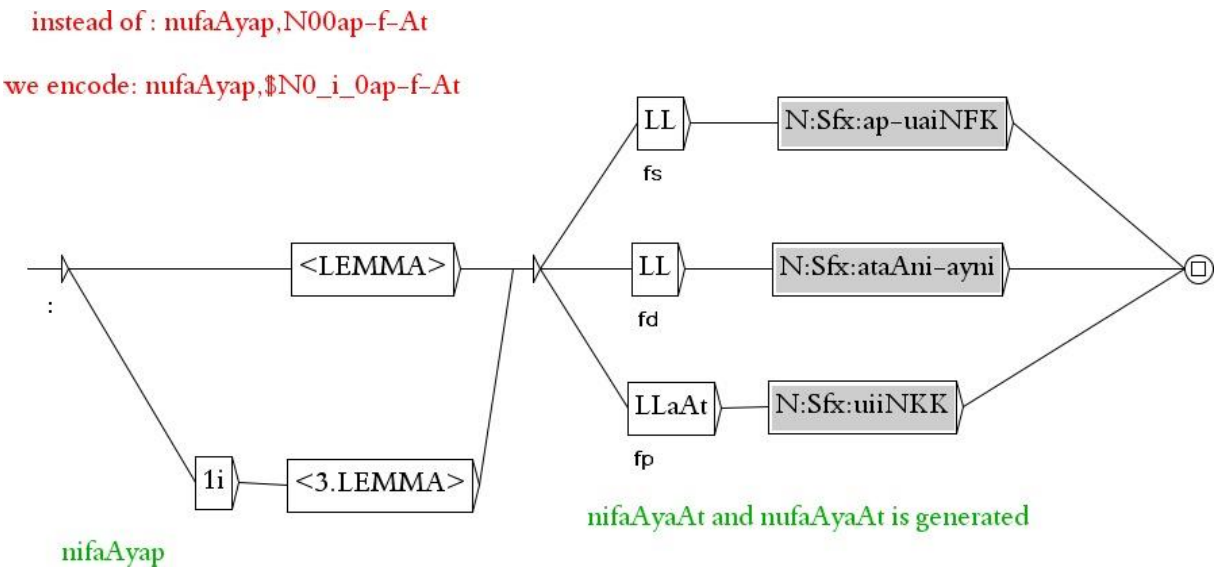


Fig. 4.2.2. An inflectional transducer in the Semitic mode for *n**u**faAyaAt/n**i**faAyaAt*

In the example (Fig.4.2.2), we have 6 paths: 3 paths inflect *nufaAyap* in the singular/dual/plural; they begin with the `<LEMMA>` operator, which retrieves *nufaAyap*, the lemma of the entry; the other 3 paths inflect *nifaAyap*, and they begin with the box 1i, which copies the first letter of the lemma, followed by the `<3.LEMMA>` operator, which copies the lemma from the third letter until the end. The `<n.LEMMA>` operator copies the lemma field from the n^{th} position to the end of the field. The same three subgraphs representing suffixes are used in Figs. 4.2.1.a and 4.2.2, and in many other graphs.

The inflectional transducer produces both variants with *u* and with *i* as inflected forms of the lemma *nufaAyap* (in **bold** the example below). The inflectional transducer produces 54 inflected forms and associates them to the same lemma: 27 "standard" forms with *u*, plus 27 "variant" forms with *i*. The plural forms are the following output:

/standard with u	/ variant with i
n**u**faAyAatu,**nufaAyap**.N:fpDN	n**i**faAyAatu,**nufaAyap**.N:fpDN
nufaAyAatu,nufaAyap.N:fpaN	nifaAyAatu,nufaAyap.N:fpaN
nufaAyAatN,nufaAyap.N:fpiN	nifaAyAatN,nufaAyap.N:fpiN
nufaAyAati,nufaAyap.N:fpDA	nifaAyAati,nufaAyap.N:fpDA
nufaAyAati,nufaAyap.N:fpaA	nifaAyAati,nufaAyap.N:fpaA
nufaAyAatK,nufaAyap.N:fpiA	nifaAyAatK,nufaAyap.N:fpiA

The `<LEMMA>` operator copies the complete lemma field, no matter the number of letters in the field, and is useful for Arabic nouns and adjectives where masculine forms are generated by inserting vowels in the consonantal skeleton, whereas feminine forms are obtained by appending suffixes (Fig. 4.2.3.a). [28]

4.2.3 Vowel variant with broken plural

[28] These inflectional operators are useful also for an Austronesian language (cf. Unitex User manual Section 3.5.4 Inflection of Semitic languages): *In Tagalog, an Austronesian language that uses commonly infixes and reduplication for inflection, <LEMMA> and <n.LEMMA> may be used to produce verb tenses. The toy inflection grammar of Fig. 3.18 produces the perfect* kumain, *future* kakain *and imperfect* kumakain *of the verb* kain *"eat".*

We have noticed this variation for the nouns *Euqodap/Eiqodap* "knot" عقدة, *gurofap/girofap* "room" غرفة, in the singular and dual, but also in the broken plural: *Euqad/Eiqad* "knots", *guraf/giraf* "rooms".

In the transducer for these entries (Fig.4.2.3.a), we use the `<LEMMA>` operator to copy the complete lemma field. The digits 1, 3, and 5 in the two boxes `1u3a5`, `1i3a5` stand for the rank of the letter in the lemma in order to generate the broken plural (Neme & Laporte, 2013).

غُرْفَة, غرف

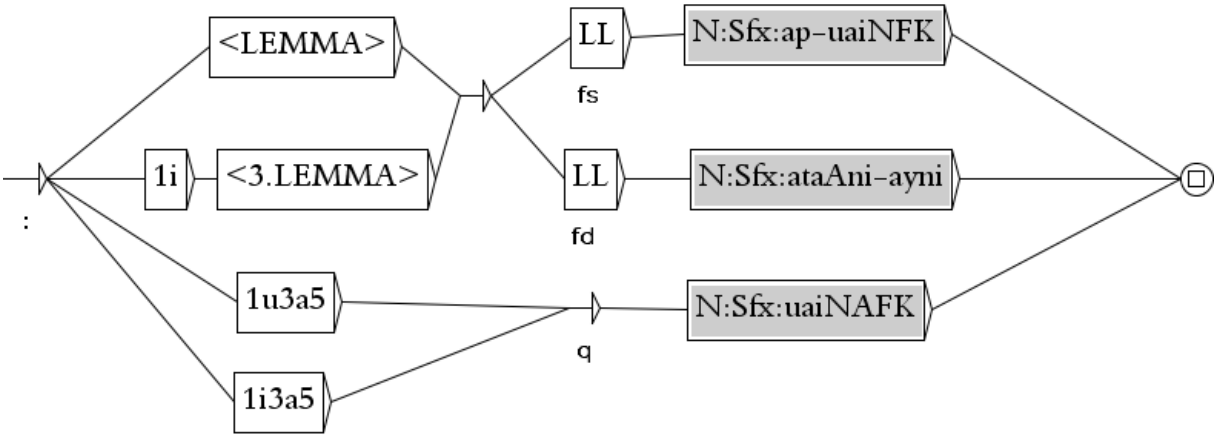


Fig. 4.2.3.a. Inflectional transducer generating forms with vowel variation in the singular, dual and broken plural forms (in red, example in Arabic)

Another case with a broken plural variant is *Saliyob* 'cross' صليب : we may say for the broken plural either *SilobaAn* or *SulobaAn* (Fig. 4.2.3.b) , but not **SalobaAn* صلبان . This pattern variation FuEolaan/FiEoLaan is frequent for broken plurals; still, not all nouns with the same pattern in the singular admit such variations: one may say *fusotaAn* "dress" but not **fisotaAn* or **fasotaAn* فستان.

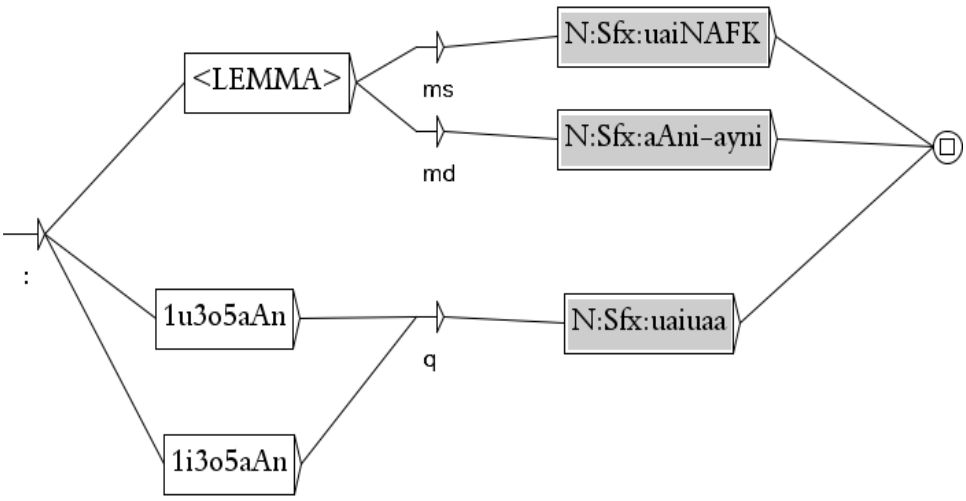


Fig. 4.2.3.b. Inflectional transducer for broken plural variation for Saliyob *'cross'* صليب*, we may say for the broken plural either* SulobaAn *(normative usage) or* SilobaAn

4.2.4 Suffixed feminine plural with *a/o*

Some feminine singular nouns such as *laSoqap* 'scotch tape' لصقة admit a variation in the plural (cf. Al-Ghalāyini, 2007, Vol 2, p.26): *laS**a**qaAt* vs. *laS**o**qaAt* لصقات (Fig. 4.2.4), or *Oaz**o**map* 'crisis'أزمات , in the plural *Oaz**a**maAt* or *Oaz**o**maAt* أزمات . The sequence of operators `LLLLaRaAt` deletes from the end four letters, inserts *a*, copies a letter (here *q*) and adds *aAt* to produce *laSaqaAt* (`L,R` for Left, Right shift). Also note *sul**o**Tap/sul**u**TaAt* 'authority' سُلُطات, and more examples in Arabic in footnote [29] .

Instead of 27 forms, the transducer of Fig. 4.2.4 generates 54 surface forms (9x2 singular + 9x2 dual + 9x2 broken plural forms) and associates them to the same lemma.

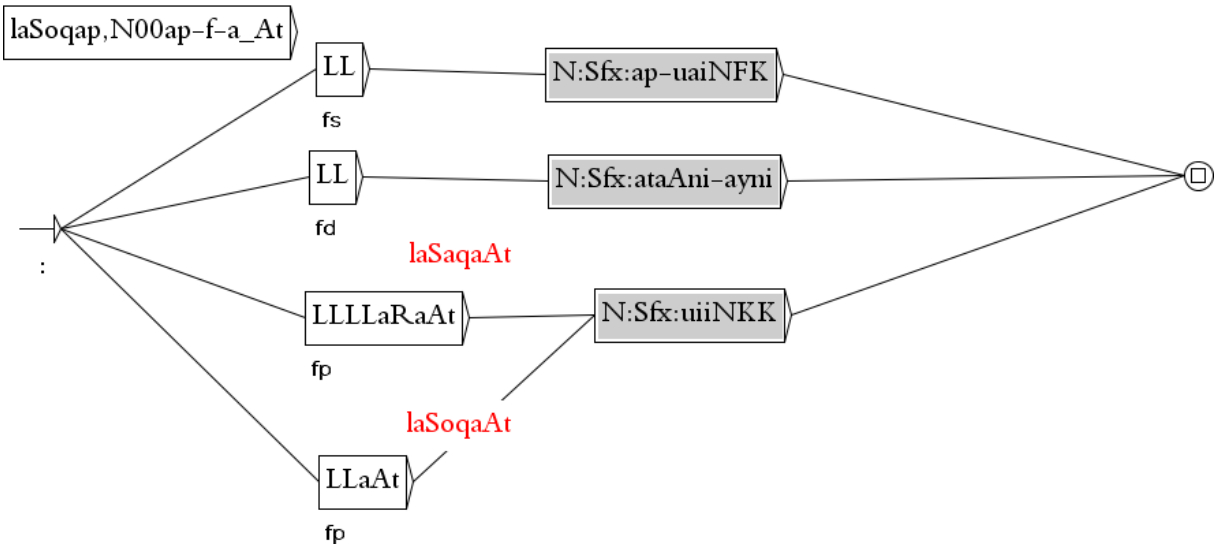


Fig. 4.2.4. Inflectional transducer for variation of the plural with the suffix -aAt

Tamazight, a Moroccan and Algerian language from the Hamito-Semitic family, has a similar phenomenon: the substitution of *e* (mute or pronounced schwa) by *a* before the plural suffix -*en* as in *izg**e**r/izg**a**r**en*** "ox/oxes". This plural formation is called external plural[30] in this grammatical tradition.

5 Unitex - Arabic Lexicon

5.1 Tagset

[29] We identified many examples in our corpora: جُرعَاتٍ خُطُوَاتٍ سُلُطَاتُ عُمُلَاتٍ نَدَبَاتٍ نَشَرَاتٍ نَشَرَاتٍ نَفَحَاتٍ هَجَمَاتٍ صَدَمَاتٍ صَفَحَاتٍ طَفَرَاتٍ حَلَقَاتٍ

[30] Nabil Chebieb, personal communication.

The following tables give an overview of the different codes used in the Arabic-Unitex dictionaries. These codes are meant to cover the morpho-syntax of Arabic simple inflected forms. For the open grammatical categories such as verbs, nouns and adjectives, all the inflectional values are detailed in appendix. They are consistent with traditional morphology, so that Arabic specialists can become quickly familiar with the tag set. The encoding is divided in three tables: POS (Table 5.1a), inflectional features (Table 5.1.b in appendix, with 360 combinations of inflectional features), and semantic-syntactic features (Table 5.1.c in appendix, with 30 syntactic and semantic features).

Table 5.1a. Part Of Speech codes used in Arabic-Unitex

Code	**POS in English**	**Encoded example**	POS in Arabic	Arabic examples
<V>	Verb	<V:aI3msN>	فِعل	تـتـهـمكـون
<N>	Noun	<N:fsiG>	إسم	تُـفَـاحـةٍ ، إمـرأةٍ
<NPr>	Proper noun	<NPr+Loc:fsDN>	إسم عَلم	دمـشقُ
<A>	Adjective	<A:msiN>	صفـة	صغـيـرٌ
<EL>	Elative, i.e. comparative and superlative		أفـعـل الـتـفـضـيـل	
<ADV>	Adverb (indefinite accusative)	<ADV> or <V:FmsiA>	ظرف	واقـفـاً ، مـعـاً
<PREP>	Preposition	<PREP+gen>	حـرف (جـر)	بَـيْـنَ
<PRO+Pdem>	Demonstrative pronoun	<PRO+Pdem:s>	اسـم إشـارة	هـذان، هـؤلاء، هـنـاكَ، ذَاكَ
<PRO+Prel>	Relative pronoun	<PRO+Prel-Hum:s>	ضـمـيـر	مِـمَّـاً
<PRO+Pinterrog>	Interrogative pronoun	<PRO+Pinterrog +Hum:s>	إسـم إسـتـفـهـام	مَـنْ ؟
<CONJC>	Coordinating conjunction		حـرف عـطـف	أمَّـا ، أوْ، وَ، فـَ
<CONJS>	Subordinating conjunction for verbs		حـروف الـنـصـب والـجـزم لـلـفـعـل	لَـنْ، لَـمْ
<INTJ>	Interjection			واللهِ،
<DET>	Determiner Al-		ال الـتـعـريـف	الـــــ
<INNA>	Governs accusative nouns	<INNA>	إنَّ وأخـوتـهـا	إنَّ أنَّ كـأن
<PRTCL+Part_la>	Confirmation particle		لام الـتـوكـيـد	لـــــــَـيـضـربُ
<PRTCL+Part_sa>	Future particle before imperfect indicative		سـيـن الـمـسـتـقـبـل	ســــــــَـيـضـربُ
<PRTCL>	Any particle	<PRTCL+vocative>	أداة أو حـرف	يـا

5.2 Size and parsing speed

The Arabic-Unitex lexicon of lemmas has been built and encoded manually and checked semi-manually. Its format consists of a simple line for each lemmatized lexical entry:

```
lemma,inflectional-code        / Notes
ktb,$V3au-123              / '$' indicates the Semitic mode
            / The encoding details are in Neme (2011)
kitaAob,$N300-m-FiEaaL-FuEuL-123
            / Broken plural      (See Neme & Laporte, 2013)
jamiyol,A0000-g-uwna
```

```
/A regular adjective admitting masculine and feminine inflection
/ with masculine plural in –uwna and feminine in –At
/ The inflectional transducer is in the concatenative mode
```

- The lexicon includes 76,000 lemmas and the full form language resource includes 6 million fully vowelized inflected forms.
- The lexicon has nearly 1,000 inflectional classes encoded in FSTs: 1,000 main graphs and 4,000 subgraphs
    - **15,400** verb lemmas
        - 4.1 million inflected forms including active and passive participles
            - including 550,000 inflected orthographic variants marked with +pro or +nopro for compatibility with enclitic pronouns
    - **41,500** noun lemmas including 4,200 with broken plural
        - 1.17 million inflected forms
            - including 125,000 inflected orthographic variants obligatorily with or without enclitic pronoun
    - **13,000** adjectives including 200 BP adjectives, and 200 elatives (such as "bigger")
        - 635,000 inflected forms
    - **6,000** proper nouns
        - 53,000 inflected forms (case and definiteness)
    - Several hundreds of entries with residual categories such as adverbs, pronouns, particles…
- For each POS, agglutination grammars are formalized in graphs restricting the combinatorics by using the inflectional attributes
- These resources potentially recognize at least **500 million** valid agglutinated words.

COUNTING PARTIALLY VOWELIZED FORMS

Equipped with our vowel-omission-tolerant lookup, the dictionary can store and identify a huge, theoretically infinite number of forms. Moreover, the presence of partially or fully vowelized words does not affect the speed of the analyser (section 6.2). In other words, *our data structure/algorithm is scalable*.

The lookup algorithm recognizes the form ***ya****s****o****t****a****q****o****b****i****l****u***, for instance, and all partially vowelized variants with the omission of any number of vowels ***ya****stqbl****u,*** *ystqb****i****l*, etc. and rejects as unknown incompatible forms such as ******ya****s****a****t****a****q****o****b****i****l****u, ****y****i****s****o****t****q****o****b****l*.

We created a program to estimate the number of these potential partially vowelized forms by counting the occurrences of short vowels, *G* (gemination), *O* and *I* (*hamza* above and under *alif*) in each form in the inflected dictionary (6 million forms) and by computing the number of possibilities. Given that each vowel may appear or not, a fully vowelized form with 4 diacritics admits 16 possibilities of partial vowelization (2^4); a form with 5 vowels admits 32; and a form with 10 vowels admits 1024. The addition of such possibilities for the 6 million forms totals almost 250 billion partially vowelized forms. Moreover, if we include in the estimate the agglutination grammars (i.e. the agglutinated clitics which may have 1 to 4 vowels), this number can easily reach several trillion forms.

In addition, the system is able to discriminate between a huge set of correct forms and an even huger set of incorrect forms**.** The number of rejected forms is a theoretical, not an experimental, issue: in practice, the words that occur in real texts, either correct or incorrect, are much less numerous than the theoretical possibilities, either accepted or rejected. However, consider only the 4 short vowels *a, u, i, o*: one vowel is allowed at a given word position and the other 3 are incompatible with the fully vowelized form. The forms rejected by the algorithm for a word with 4 vowels are more than 81 (3^4); [31] with 5 vowels, they exceed 243; and with 10 vowels, they exceed 59 049 (3^{10}).

That is to say that ***an FSA is adapted to store and retrieve an infinity of string forms*** in a compressed file of about 10 Megabytes (see below about compression).

5.3 Evaluation

5.3.1 With a corpus with a high rate of vowelization

From *Al-adab* (http://al-adab.com/), a literature and critical essay magazine edited in Beirut since 1953, we have chosen three texts [32] (published in May, 2017, 60 pages): the first two are a political essay on democracy and an essay on the Syrian Civil War (2011-2017), written by Levantine writers from Lebanon and Syria, and representing together 15 pages; the remaining 45 pages are a discussion about Moroccan identity between six university professors and intellectuals from Morocco. Our choice of this corpus is motivated by the quality of its vocabulary, richer than in common newspaper texts, and the density of its authentic partial vowelization, which exceeds 33%, indicating a high level of editing process[33], achieved, we guess, by the writers, and controlled and enriched by the editor. This corpus allows us to test the Arabic-Unitex lexical resources and our lookup algorithm against partial vowelization that occurs spontaneously, independently from our lexical encoding. A carefully edited corpus with a high rate of vowelization provides a stricter evaluation than a corpus with a standard rate (3%), since each vowel written in the corpus is compared with vowels specified in the dictionary.

Our corpus is constituted of 11,950 words, 4,225 of them (*versus* 350 with a standard rate) with partial vowelization: 7,725 with no diacritics (64,6%), 3,886 with one diacritic (32.5%), 328 with two diacritics (2,74%) and 11 with three diacritics (0.1%). Table 5.3.1.a details the distribution of the diacritics in the tested corpus.

Table 5.3.1.a. Distribution of 4,576 diacritics in 4,225 words in the corpus (11,950 words)

Vowels	without G	in endings		G and vowel		G without vowel
a	468	284		Ga	53	

[31] The 3^4 forms don't include the rejected forms with omitted vowels.
[32] The three texts are: http://bit.ly/2fNxD9T, http://bit.ly/2wSk7Wx, http://bit.ly/2vFQbyl.
[33] Texts with such a high rate of vowelization are not rare, particularly in opinion journalism, and even in articles in common newspapers such as in al-Hayat *http://bit.ly/2t10OuQ*, where we found 146 words with diacritic(s) out of 468 words: 156 diacritics are used; 136 words have one diacritic and 10 words have two. *G* is used in 114 words; *–AF*, for indefinite accusative case ending, is used in 31; *–u* is used in 9 occurrences, to mark the active/passive in a verbal form, such as *t**u**Hrj/y**u**stHsn*.

u	414	245		Gu	29	
i	120	55		Gi	43	
F (F, **FA**, FY)	440	440 (95,**339**,6)		GF	58	
N	97	97		GN	5	
K	210	210		GK	8	
o	139	84		Go	0	
Total	**1888**	1057			196	2492

The gemination marker *G* (2,688 occurrences, 59%) is more frequent than all short vowels, nunations and *o* together occurring without *G* (1,888 occurrences, 41%), because it represents a duplication of a bare consonant, thus often referring to another lemma. The most frequent diacritic ending is *-FA* with 339 occurrences, it distinguishes the indefinite accusative from the dual construct state (*-A*, called *mudaf*) form of a noun. The magazine uses exclusively the normative variant of the indefinite accusative *-FA*, as opposed to *-AF*, often used in the *Al-Hayat* or *Annahar* newspapers. Our typographical rules (`fathatan alef equiv alef fathatan=YES, Section 4.1`) accept both variants. The *o* is a frequent ending because it indicates the dual for nouns or adjectives in order to disambiguate it from plural forms.

Table 5.3.1.b. Lexical coverage of the corpus (11,950 words/5,950 types)

Missing	Occurrences	Types		Occurrences (%)	Types %
Proper nouns	80	38		0.7	0.6
Other valid forms	71	26		0.6	0.4
Total	**151**	64		**1.3**	1.1

Our algorithm detected in the corpus only one typo error: a bare letter substitution (المغزى/المغزى; ز/ر ; z/r), which indicates an excellent editing quality. The first 15 pages (Syria-Lebanon) were totally covered by our resources except one verb نكّل (*nkGl*,$V62-123) "to torture". The other 167 uncovered occurrences (90 types/5,600) are in the 45 pages from Morocco and may be classified in three categories:

(i) Typo errors, diacritics and glottal stop (16 occurrences): The 4,225 words with one, two or even three diacritics were all validated by our algorithm except 16 words not found in the resources. 11 of them are misplaced occurrences of *G*. Three are true typo errors: the *G* occurs on the wrong bare letter (*tq**GS**y* instead of *tq**SG**y*). The other 8 flagged words are cases of inversion vowel-*G* / *G*-vowel. Our typo rules state that *G* must be followed by the vowel. In fact, the two sequences *Ga* and *aG* appear as two glyphs superposed in the same order; they are visually identical, and cannot be distinguished by the editors of *Al-adab*. The rule is observed in 196 cases and there are 8 inversions (*aG/Ga* or *FGA/GFA*).

The 5 remaining flagged "errors" are related to different standards for glottal stop scripting in Morocco and the Levant:

(a) بدؤوا / بدأوا ; *bd**O**wA/bd**W**wA* (2 occ.) <bdO:aP3mp>
; Morocco/Levant glottal stop rules

(b) بمبدأ / بمبدإ ; *bmbd**O**/bmbd**I*** (3 occ.)
; <PREP><mbdO:NmsaG>

(a) In Morocco, the suffix *–wA* (Perfect 3rd person masc-plural) at the end of a form is considered as external to the core verb; therefore, the glottal stop rule for the end of a word applies; whereas in the Levant, the suffix is considered part of the core verb; therefore, the glottal stop rule for the middle of a word applies.

(b) Our agglutination grammar rules select the genitive case ending (*-i, -K*) and in both cases (construct state or indefinite) the glottal stop diacritic followed by *i/K* should be written *preferably* as *I* (below *alif*), not as *O* (above *alif*).

(ii) Proper names (80):
Many proper names were recognized. However, the test collection shows that 80 occurrences (38 types) of proper names were not recognized, representing first names, surnames or place names, that are not included in our lexicon.

(iii) Other forms missing in the lexical resources (71):
The test collection shows that 71 other occurrences were missing in our lexicon, representing 26 types:

- The word *Amazigh* agglutinated or not occurs 27 times.
- The two orthographic variants *tfnAq* or *tfynAq*, denoting the Amazigh alphabet, occurs 12 times.
- The word الهويّاتيّ "identitarian" occurs 16 times as a noun or adjective in the masculine or feminine, agglutinated or not. This word is a derivative with the ending suffixes *-yG* or *-yGap*. 11 other occurrences of derived adjectives ending with *-yG* or *_-yGap*: إسلامويّ، التلازميّة، الحضريّة، الرغبويّ، الفلّاحيّة، القاعديّة، الموحّديّة، اللسنيّة، راهنيّته، قدحيّة، لاعقلاني;
- 4 nouns (المستفتين، الملالي، شيعًا، الوندال);
- 1 verb (ودستَرَها and_*dstara_hA*, "and_put-in-the-constitution_it") .

Morphosyntactic tagging is generally part of a pipeline of written text processing. Unknown words may jeopardize a subsequent deep syntactic parsing of a sentence. Thus, fallback procedures (not implemented) are required to assign a POS to unknown words, such as relational adjectives ending with *–yG* and typical Arabic proper nouns starting with *Ebd-* or ending with *-Allh* or *-Aldyn*, which are common prefixes and suffixes in Arabic proper nouns.

Summing up, our resource (see our Arabic spell checker http://babelarab.univ-mlv.fr/) has flagged 11 words with partial vowelization: 3 with true errors, and 8 with discrepancies regarding Morocco/Levant standards for glottal stop rules. The fault rate of coverage (Table 5.3.1) in Arabic-Unitex is 1.3%, proper nouns included (0.5%, if excluded), and the fault rate is 1% (0.4 % if proper nouns excluded). Finally, our lexical resources have a better coverage of Levantine usage.

5.3.2 An extrinsic evaluation through a local grammar

In the preceding experiment, the system uses information provided in the dictionary: inflected form, POS and inflectional features, and the results are therefore an indirect evaluation of these fields. However, it does not use the lemma field also provided in the dictionary. In this section, we report an extrinsic evaluation experiment devised to assess the system's ability to recognize lemmas.

We made an experiment similar to Traboulsi (2009) and Ben Mesmia et al. (2015) but with our resources. Traboulsi (2009) underlines that *"Despite the fact that the probabilistic approach (the supervised machine learning) and the symbolic approach (the rule based) have been successful in recognizing Arabic person names in news texts, these approaches require large tagged corpora, dictionaries or gazetteers, lists of proper names, which could have been avoided if the local grammar approach was used the way they do." (Section 2).* Traboulsi recognizes the structure <Reporting_verb><Noun+Human> which is frequent in newspapers. He takes advantage of the frequency of verbs such as *said*, *declared*, *indicated*, … and the predictable occurrence of a subsequent proper noun. To implement his local grammar, Traboulsi uses a cascade of FSTs that apply in a strict order. Ben Mesmia et al. (2015) presented many local grammars for recognizing Arabic Named Entities (ANE) based on a transducer cascade as well. They established word lists, a set of extraction rules based on trigger words and a set of transducers allowing the recognition of several ANE categories.

The advantage of these two implementations is that they dispense with annotated corpora; the drawbacks are: agglutinations are not handled properly, as each possible agglutinated form should appear explicitly in the local grammar, making it unnecessarily overloaded; the word lists are constructed on the fly from the corpora.

Consequently, we expected that, with a rich morpho-syntactic representation, the local grammar approach of these two methods could be adapted to have a better recall/precision. Moreover, it is easier to conceive a local grammar based on a pre-processed, segmented and annotated text. Our rich annotation with lemma, POS and inflexional attribute values helps to craft a more concise and readable grammar. For instance, checking agreement and disagreement between words helps to identify syntactic structures and boundaries, and consequently, semantic slots. Such checks result in more precision in capturing Named Entities.

We built a local grammar (Fig 5.3.2.a) that identifies the verb "to say" in the perfect or imperfect 3rd person masculine singular, followed by a chunk with the noun "minister". The local grammar outputs braces delimiting this pattern, as in:

وقال {وزير المال الفرنسي جان ارتوي} في اثناء الجلسة ان الدوام الجديد

"and_said {minister of_finance French Jean Artuis } (in) during the session

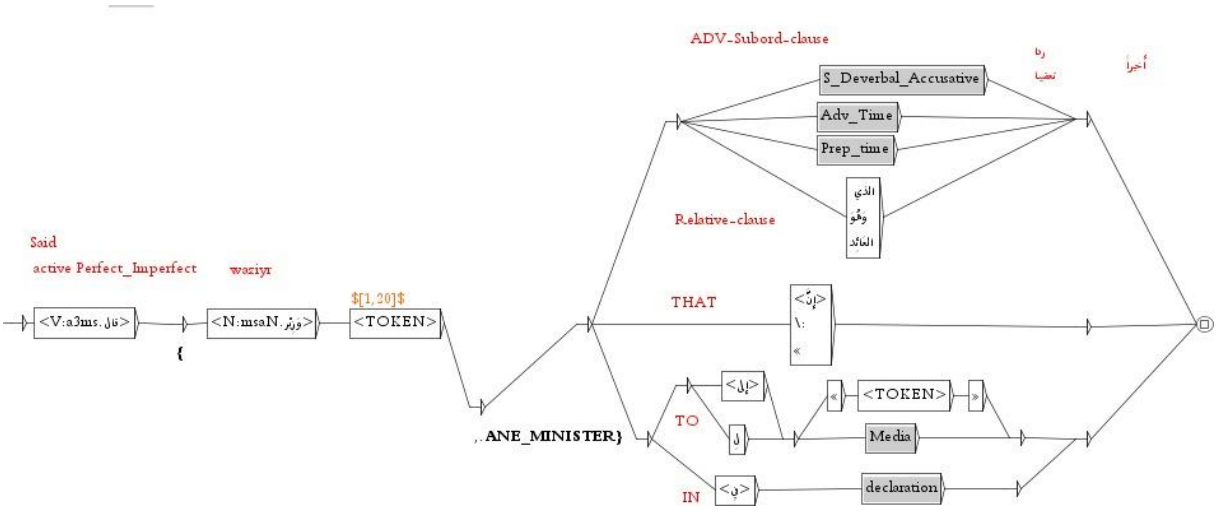


Figure 5.3.2.a. Local grammar identifying ministers

In Fig 5.3.2.a, the box <TOKEN>/$1,20$ defines a window of 20 words in which a pattern indicating the end of the chunk is searched. The local grammar contains 6 graphs and 55 boxes in total. The pattern belongs to one of three types:

- <THAT>: *IinGa* "that" introduces an embedded sentence beginning with a noun. The sentence may also be introduced by a colon or a double quotation mark.
- TO <MEDIA> or IN <DECLARATION>: *IilaY* or *li* "to" may introduce a media slot: journal(ists), Al-Hayat, (press) agency, radio. The preposition *fiy* "in" may be followed by a declaration slot such as conversation(s), conference(s), meeting(s), book(s) as: <مُؤْتَمَر.N:mG>+<جَلْسَة.N:fG>+<حَدِيث.N:mG>+<مُقَابَلَة.N:fG>+<تَصْرِيْح.N:mG>+<لِقَاء.N:mG>+<بَيَان.N:mG>+<نَدْوَة.N:fG>+<كَلِمَة.N:fG>+<إتِّصَال.N:mG>+<إجْتِمَاْع.N:mG>+<عَرْض.N:fG>+<رِسَالَة.N:fG>+<كِتَاْب.N:mG>+<مُدَاخَلَة.N:fG>+<تَعْلِيق.N:mG>+<خِطَاْب.N:mG>
  in the genitive case and either definite, construct state or indefinite, prefixed (or not) by *Al* and agglutinated (or not) to a pronoun such as in "intervention_his" (line 17 in the concordance below)
- ADVERBIAL or SUBORDINATE CLAUSE: It can be "yesterday", "Tuesday" or any date. It can be a relative clause introduced by a relative pronoun or an active participle such as "travelling" or a deverbal noun such as "commenting".

Table 5.3.2. Part of a concordance with 971 matches identified by the local grammar

1	قال {وزيرالعدل السوداني السيد عبدالباسط سبدرات,.ANE_MINISTER} ان
2	قال {وزيراقتصاد نيكاراغوا نويل ساكاسا,.ANE_MINISTER} انه
3	قال {وزيرخارجية كوبا روبرتو روبينا,.ANE_MINISTER} ان
4	وقال {وزيرالاعلام,.ANE_MINISTER} عقب ال
5	وقال {وزيرالشؤون الاجتماعية أيوب حميد,.ANE_MINISTER} بعد
6	وقال {وزيرالاعمار الوطني اتيان مبايا,.ANE_MINISTER} في بيان
7	وقال {وزيرالزراعة **ردا على اسئلة**,.ANE_MINISTER} «
8	وقال {وزيرالدفاع الاسرائيلي اسحق موردخاي,.ANE_MINISTER} قبل
9	وقال {وزيرالخارجية الكندي لويد اكسويرثي,.ANE_MINISTER} في كلمة
10	وقال {وزيرالخارجية البولندي داريوس روزاتي,.ANE_MINISTER} أمس الثلثاء
11	وقال {وزيرالعدل السوداني عبدالباسط سبدرات,.ANE_MINISTER} خلال
12	وقال {وزيرالخارجية الايراني كمال خرازي,.ANE_MINISTER} في مؤتمر
13	وقال {وزيرالخارجية البريطاني روبن كوك,.ANE_MINISTER} بعد
14	وقال {وزيرالخارجية البريطاني روبن كوك ,.ANE_MINISTER} أمس انه
15	وقال {وزيرالمال ليم تشانغ يول,.ANE_MINISTER} في مؤتمر
16	قال {وزيرالخارجية القبرصي اليكوس ميخاليدس,.ANE_MINISTER} أمس
17	وقال {وزير الشؤون الاوروبية بيار موسكوفيسي,.ANE_MINISTER} في **مداخلته**

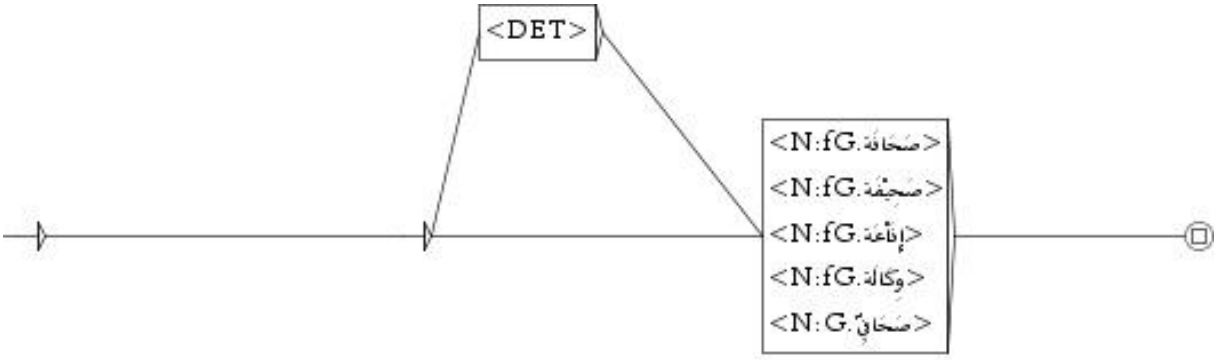
We evaluated the recall of the graph on part of ArabiCorpus http://arabicorpus.byu.edu/, an online set of untagged Arabic corpora that contains portions of textual documents from different sources. We have used *Al-Hayat* 1997 (Saudi Arabia).

We launched the search query *qAl wzyr* ("said minister") as a string and we obtained a concordance of 985 occurrences (Table 5.3.2). We discarded the 10 occurrences where *qAl* is a

substring of another verb such as *IEt**qAl*** "arrested" or *Ist**qAl*** "resigned". The remaining 975 are the target of our local grammar.

The local grammar identifies 971 occurrences (see Table 5.3.2) of the entity {MINISTER} out of 975 (99,6% recall). The 4 missing occurrences contain:

- One occurrence of *O* (instead of *I* or *A*) in *IinGa*, which is a spelling mistake since reporting verbs should be followed exclusively by *IinGa*. Our grammar identifies vowelization variants of the lemma <IinGa> (such as *In, Iin, InG, An, AnG, AnGa*, etc) but not of the lemma <OanGa>.
- One occurrence of *radGAF* "responding", tagged as unknown word. The lemma of this deverbal noun is missing in our dictionary (see concordance, line 7): *radGAF* is a deverbal noun based on a simple verb ($V31 to $V36 in our encoding, Neme 2011); these deverbal nouns are irregular.
- One occurrence of the pattern *Ily Al-SHAfyGyn* "journalists". This noun has two pronunciation variants *SuHaAfiyG* and *SaHaAfiyG* (cf. Section 4). In our lexicon, we opted for *S**u**HaAfiyG* and did not encode the variation, whereas in our grammar (cf. Fig 5.3.2.b), we used <S**a**HaAfiyG> as lemma to identify the inflected forms.
- One occurrence without any of the patterns recognized by the grammar to locate the end of the chunk. The contents of the declaration are before the verb "say" and the sentence does not mention the media: "Will they find it…, as said the previous American minister of foreign affairs Warren Christopher?"



For instance <صحافيّ.N:G> represent 18 forms, <journalist,N:G> represent the variations in:

gender (masc, fem)

number (sing, dual, plural)

definitness (Definite, annexed or indefinite)

but must be in Genitive case

Fig. 5.3.2.b. The subgraph <Media> in the local grammar of Fig. 5.3.2.a

The use of very informative lexical resources also facilitates the manual construction of local grammars. In the lexical resources, the lemma <journalist> has 54 inflectional variations. In a local grammar, <journalist.N:G> recognizes 18 forms in the genitive case and excludes the 36 other variations. This representation identifies standalone forms, but also agglutinated forms with *Al* or with 12 potential pronouns. Furthermore, it is useless to represent in the local grammar (Fig. 5.3.2.b) the agglutinated pronoun <PRO>, since the result of morphological analysis represents any variation of <journalist.N:G> separately, even before a possible

agglutinated pronoun (see line 17 in the concordance). For computational linguists, such simple and natural formalization of the local grammar represents an enormous gain and simplification.

Likewise, all the inflection of a verb may be covered by a lemma followed by inflectional features such as <say:a3ms>, with unspecified tense, and thus referring to both active perfect and imperfect. Moreover, since the segmentation of words is handled by our agglutination grammars, agglutinated forms with proclitics such as "and said" and optional enclitic pronouns may be detected simply by the formal representation <say:a3ms> (<E> + <PRO:3s>) which retrieves "said" and "said it". This turns local grammars more readable.

As we have said above, an adverbial clause may constitute the pattern that indicates the end of the chunk. An adverbial clause may begin with a deverbal noun such as "commenting", generated automatically (with 10 000 other deverbal nouns) from an augmented verb (classes $V61-$V70). From `EqGb,$V62-123`, "to comment", we have generated a dictionary entry for the lemma `taEoqiyb`, from which the indefinite accusative form تعقيباً is generated as an inflected form and encoded as a line in our lexicon as `taEoqiyba`**`AF`**`,taEoqiyb.N+`**`Masdar=EqGb`**`:msiA`. But the native linguist may extend this vocabulary in the local grammar by adding synonyms of "commenting" such as منتقداً "criticizing", based on ***introspection***, even if the synonyms do not appear in the corpus.

To conclude, the ability to recognize lemmas and their variations is tested successfully. Our resources allow for helpful conciseness in the detection of inflected forms by local grammars. Moreover, they make it unnecessary to tag corpora since we tag texts automatically using a dictionary which covers more than 76 000 lexical entries[34]. Besides, they allow to annotate corpora semi-automatically as an input for supervised learning.

5.4 Arabic-Unitex versus BAMA lexicon

Many features distinguish Arabic-Unitex from the BAMA lexicons. Here is a survey of the main differences and similarities.

a) Usage in the Levant

Arabic-Unitex is mainly based on the Levantine usage of Arabic language. The Levant defines *de facto* the Modern Standard Arabic usage. This tradition dates back to when the Umayyad caliph Abd Al-Malik made Arabic the official language during his reign (685-705) in Damascus. In Arabic-Unitex, most lexical entries are citation forms attested in paper modern dictionaries printed in Beirut after 1970: Abd-Nour (2006), Khalil Al-Jar (1973, Larousse) and others; we used https://www.almaany.com/ to double-check meaning and usage. We also included terms and neologisms found in the Arabic Wikipedia, the Nemlar corpus, and the *Annahar* (Beirut) and *AlHayat* (KSA-Beirut-London) newspapers.

[34] The list of proper nouns (around 6000) includes name of countries and important cities, Arabic and foreign forenames and family names such as celebrities: Ronaldo, Rif(v)aldo, B(P)edro Almodovar, and George Bush, etc. This list was created first by extracting the proper nouns from the Nemlar corpus. Secondly, we processed many newspaper corpora, short novels and other modern fictions with our Arabic-Unitex resources. From the unknown words list output by the Unitex tagger, we extracted the simple and agglutinated forms of proper nouns. The proper nouns represent often more than half the unknown-word list. We encode them manually and such encoding enables recognizing agglutinated proper noun forms such as <CONJC><PREP><NPr>.

The BAMA lexicon is derived from the ALPNET project, and based mainly on Hans Wehr's bilingual dictionary[35] (1952). BAMA includes Egyptian variants such as *kabuwriyA* كبوريا"crab", *mavGaAl* "sculptor" مَثَّال; *miMolawoz* مشلوز "apricot", excluded from Unitex, which contains instead *saloTaEuwon (*and *saloTaEAon)* سلطعون; *naHGaAt*, نحّات; *miMomiM* مِشْمِش, which are all in BAMA, as well. BAMA also includes old terms such as *jazuwr, niyb, ZaEuwn,* (resp. "fat camel for butcher meat", "old female camel", "load camel").

b) Loan words

Both BAMA and Unitex include the standard Babylonian naming of months such as *Oayoluwol أيلول,* current in the Levant and the Gulf, and both lexicons also include the names borrowed from English such as *September,* current in Egypt, Sudan and Libya. Neither lexicon includes the denominations of French origin such as *Janvier,* in use in Tunisia, Algeria and Morocco[36]. The month names used in the Islamic lunar calendar for religious events and ceremonies are included in both lexicons.

Both BAMA and Unitex include loan words: *dakotuwr, bruwfisuwr, bruwtiyn* "doctor, professor, protein". BAMA lists both variants *bri**i**doj* and *bri**y**doj* "bridge (game)", while Unitex inventories only the second representation. We *preferably* represent the vowel with the bare letter *y*, in keeping with the current tendency to write loan word vowels with bare letters.

c) Verbal inflection

In BAMA, we counted 415 perfect passive stems, 2845 imperfect ones, 116 stems for the imperative mode, and no energetic mode. Active and passive participles are described in the BAMA lexicon not as inflections of verbs but as adjectives and nouns. In Unitex, we have covered them as inflected forms for 15400 verbs. Note that the passive mode is possible for intransitive verbs such as *niyma bi_Al-firaAMi* "(it) was-slept in-the-bed". Contrariwise, Unitex covers some adjectives in the form of participles, e.g. *MaAeiE* شائع "current". This flaw needs to be fixed, at least for common adjectives.

d) Lemmas with suffixed plurals

In paper dictionaries, some lexical entries are in the plural, because the correspondent singular form exists with another meaning. In our inflectional approach, the lexicographer may encode a citation form in the plural. In Unitex, some lexical entries are lemmatized in the plural, e.g. `qalawiyGAt,N0aAt-p-0`, "alkali (chemistry)". The singular is an adjective. The noun *DaruwriyGaAt* ضروريّات means "necessities"; its singular counterpart is used only as an adjective meaning "necessary". They are encoded as independent lemmas in Unitex:

```
DaruwriyGaAt,N0aAt-p-0/      ضَرُورِيَّات
DaruwriyG,A0000-g-uwna/        ضَرُورِيّ
```

In BAMA, both lexical entries are encoded with the same lemma `Daruwriy~_1` but with different POS:

```
Daruwriy~_1    Drwry   Daruwriy~      N-ap    necessary/requisite    Daruwriy~/ADJ
Daruwriy~_1    Drwry   Daruwriy~      NAt     necessities            Daruwriy~/NOUN
```

[35] The original edition is in Arabic-German "Arabisches Wörterbuch (1952)", published later in bilingual Arabic-English edition as "A Dictionary of Modern Written Arabic".

[36] Unitex should include all these month denominations with features indicating the region of usage +Levant, +EgSuLy, +Maghreb, respectively for the names of Babylonian, English, French origin.

The singular of *mudaAEafaAt* "consequences" is *mudaAEafap* "the doubling", encoded in Unitex by:

```
muDaAEafap,N00ap-f-At/   doubling    masdar+DAEf  مُضَاعَفَة
muDaAEafaAt,N0aAt-p-0/  Consequences  مُضَاعَفَات
```

In BAMA, both lexical entries are encoded with the same lemma `muDAEafap_11` and the same POS:

```
muDAEafap_1    mDAEf   muDAEaf NapAt   doubling/compounding   muDAEaf/NOUN
muDAEafap_1    mDAEf   muDAEaf NAt     complications          muDAEaf/NOUN
```

BAMA contains two variants of "sixties", encoded in two lemmas whereas Unitex contains only the first variant:

```
(a) sitGiyonaAt,N0aAt-p-0/       سِتِّيْنَات sixties (in BAMA)
(b) sitGiyoniyaAt,Not-in-Unitex /سِتِّيْنَات  sixties variant (in BAMA)
```

We checked the usage of both variants through the Arab countries (Egypt, Syria, Kuwait, Jordan, Morocco) in a corpus of newspapers taken from arabiccorpus.byu.edu. The corpus has 3046 occurrences of (a) and 2093 of (b). We did not identify any difference in meaning or usage between the variants. Both are used almost at the same frequency in these newspapers, except for *AlHayat* 1997 (1031 a, 18 b) and 1996 (1198 a 23 b). It seems that *AlHayat* has a strict editorial policy and uses almost exclusively the (a) variant. Since there is no difference, we decided to create a new inflectional transducer that generates the *–yaAt* variant beside *-aAt* but attaches both to the same lemma (a) `sitGiyonaAt,N0_y_aAt-p-0`. We re-encoded similarly all this family of words: "twenties, thirties, …".

We have almost 200 lexical entries with *–aAt* suffixed plurals; this list need to be completed.

e) Broken plurals

BAMA includes two lemmas for *xaTar/OaxTaAor/maxaATir*:

```
xaTar_1     >xTAr       >axoTAr     N     dangers     >axoTAr/NOUN
maxATir_1   mxATr       maxATir     Ndip  dangers     maxATir/NOUN
```

whereas Unitex considers both BP forms as inflections of the same lemma (Neme & Laporte, examples 149-151):

```
xaTar,$N300-m-FvEvL-OaFoEaaL-123/   خَطَر    أخطار
xaTar,$N300-m-FvEvL-FaEaaLiB-m123/  خَطَر    مخاطر
```

5.5 Drawbacks and possible improvements

Since the breakthrough of the BAMA lexicon (Buckwalter, 2002), the majority of new scientific papers on Arabic NLP relies on this lexicon and on its related algorithm, *"a de facto standard tool which is widely used in the Arabic NLP research community"* (Attia et al, 2011).

Attia et al. (2011, Section 2.1) also point out the drawbacks of BAMA; nevertheless, no viable and better alternative has been proposed so far. "*After all aspects of morphological analysis have been adequately addressed, the only way to improve the quality of the analysis is by improving the lexicon.*"(Buckwalter, 2007, 3.6 Lexicon Design and Maintenance). ***Improving the lexicon for Buckwalter may be done by enhancing the lexical coverage and by increasing the level of grammatical detail***. He advocates an enhancement of BAMA (2004) by inserting traditional labels (Buckwalter, 2007, section 8):

- **gender, number**, humanness (for noun)
- **active and passive participles** and verbal nouns, deverbal noun (masdar from simple form or **augmented form**) **(**cf. Section 3.4**)**
- elative such as "bigger/the-biggest"
- instance noun, unit/collective noun
- verb features such as transitive, intransitive, grammatical colocations.

We do agree with the mentioned improvements. Our proposal of a new approach to Arabic morphology involves the *pattern-and-root model*, and a large and contemporary lexicon. Our alternative to BAMA is entirely based on the Semitic tradition, one fully inflected lexicon (lemma-based), the pattern-and-root model, and a look-up procedure in the fully inflected lexicon. Most of the enhancements recommended by Buckwalter (in **bold**, cf. 5.2) are included in Arabic-Unitex from its inception. The elative such as "bigger/the-biggest" was encoded for almost 200 adjectives and needs to be extended. Instance nouns, also called cognate nouns, such as ضربة *darb_ap* "hit_one", and unit/collective nouns such as نمل/نملة *namlap/namol*, "aunt_one/aunt_collective" are part of the lexicon and need a systematic encoding in Arabic-Unitex. Arabic-Unitex needs exposure and more testing by applications in order to be further validated.

6 Compression

The Unitex programs were adjusted in 2010 to Arabic morphology in order to handle:

- Semitic inflection and infixes,
- proclitic and enclitic agglutination,
- partial vowelization.

In the standard Unitex process, an inflected-form dictionary is compressed into a minimal acyclic deterministic finite automaton data structure in order to be stored in RAM for fast retrieval (Revuz, 1992).

6.1 The compression algorithm

The input of the Unitex dictionary to the compression algorithm is a text file whose lines are of the form:

```
<inflected form>,<lemma>.<grammatical:inflectional-codes>
```

like, for example:

```
takotubu,ktb.V:aI3fsN   / compact tag:  __246.V:aI3fsN
xawanapN,xaAoein.N:qiN  / compact tag:  __01Aoei4.N:qiN
                        / خونـةٌ BP of خَـائِـن
```

The compressed version of the dictionary is a finite state transducer that associates each inflected form with its lemma and codes. The algorithm spares space to store the inflected forms by representing the transducer in the form of a Minimal Acyclic Deterministic Finite Automaton. In order to minimize the space needed to represent the lemmas and codes, it replaces them with a compact tag that contains enough information to restore the complete entry from the inflected form. The standard version of the algorithm, applied to the entry `looks,look.V:P3s`, for example, produces the compact tag `1.V:P3s`. At lookup time, the inflected form `looks` is known, and the lookup program can rebuild `look.V:P3s` from the compact tag `1.V:P3s` by interpreting it as "remove 1 letter from the end of the inflected form and add `.V:P3s`". This strategy is very effective for many languages because it takes advantage of the regularities of the language's inflection system. For English, almost all entries for the third person of the present share the same compressed code "`1.V:P3s`" since the third person of the present of almost all verbs is the infinitive form plus *s* at the end.

However, the nature of Semitic languages makes this suffix-based approach very ineffective. The strategy of our Semitic-oriented version of the algorithm consists instead in indicating which letters from the inflected form should be kept to restore the lemma. Given the inflected form `takotubu`, the `246` substring in the compact tag (above) means that we need to keep the letters `#2 (k), #4 (t) and #6 (b)` from the inflected form to obtain `ktb`. In case some letters are missing from the inflected form they are added in the compressed form. For instance, if we have the inflected form `xawanapN` and the lemma `xaAoein`, we compress it as `01Aoei4` which means: letter `#0 (x),` letter `#1(a),` followed by the substring `Aoei` and the letter `#4 (n)` from the inflected form to obtain `xaAoein`.

In order to produce compact tags that are more likely to be shared by other entries and thus improve the compression rate, the algorithm tries all possible compact tags and keeps one that maximizes the number of letters copied from the inflected form. For instance, if we have the infected form `abcdefgh` and the lemma `hbc`, we could represent it with several codes: `hbc` (no letter copied from the inflected form), `7bc` (h copied from the end of the inflected form and adding `bc`) and `h12` (adding h and then the 2 letters `bc` copied from the inflected form). Our heuristic will select `h12` because it reuses two letters from the inflected form.

ADJUSTMENTS TO DICTIONARY LOOKUP IMPLEMENTATION

We adapted the Unitex dictionary lookup procedure to this Semitic-oriented compression strategy. Moreover, we adapted the lookup procedure so that it is *tolerant to partial vowelization and other Arabic typographical rules* (cf. Section 6.3). Our version finds for each input word (without vowels, partially or fully vowelized) those candidate forms compatible with the input word. When a diacritic is present in a surface form, the lookup procedure retains the candidates with the same diacritic at the same position in the compressed dictionary.

We also equipped the lookup procedure with *a hash table data structure stored in RAM memory*, which avoids to repeatedly search the minimal acyclic deterministic Finite State Automaton (MADFA) for occurrences of the same word. The procedure looks up the word in the hash table first; if it does not find it, it searches the MADFA and stores the entry in the hash table, in anticipation of other occurrences in the text. This speeds up the lookup by almost 50

times. This feature is independent from the compression strategy and has been adopted as the standard Unitex lookup. [37]

In addition, we pass the agglutination grammars to the lookup procedure in the form of a *flattened* FST. Each agglutination grammar is manually produced in the form of a network of graphs and subgraphs, which are compact, readable and reusable. Flattening replaces calls to subgraphs by copies, taking advantage of the fact that the network is not recursive. The global flattened grammar (grouping verbs, noun/adjectives and particle agglutination grammars) consists of 1 graph with 60 states and 286 transitions, instead of 25 graphs and subgraphs, totalling 175 states and 369 transitions. As a result, the flattened FST makes lookup approximately 2 times faster for the price of a simple compilation[38].

6.2 Two compression experiments

The full-form dictionary has 6 million surface forms. It is 340 Megabytes in plain text in Unicode UTF-8.

With the Semitic-adjusted version, we compress it into 13.5 Megabytes. The compilation of the 1,150 inflection graphs and 4,000 subgraphs takes one minute. The generation of the 6 million forms takes 10 seconds; the compression and minimization of the full-form lexicon takes one minute on a Windows laptop[39]. The morphological analysis processes almost 1000 words/second or 3 pages/second for vowelized or unvowelized text alike.

The compression ratio is better (see Table 5.4), and the lookup much quicker, if we compress separately the entries inflected in the Semitic mode. We have split into two parts the dictionary of 76,000 lemmas: 19,600 ones with inflection in the Semitic mode and 56,400 ones with inflection in the concatenative mode or no inflection.

From the 19,600 lemmas with Semitic inflection, we have generated 4,280,000 forms and a 228-Megabytes flat file. The Semitic-oriented version of the compression algorithm produces a 10.5-Megabyte compressed file.

From the 56,400 lemmas with concatenative inflection, we have generated 1,805,000 forms and 114 Megabytes flat file; the standard compression algorithm produces a 0.5-Megabytes file.

---

[37] Wintner's morphological analyser of Hebrew implemented in Java also stores the Hebrew lexicon in a lookup table (Wintner, 2008, Section 2.2): *"contemporary computers can efficiently store and retrieve millions of inflected forms. Of course, this method would break in the face of an infinite lexicon (which can easily be represented with FST), but for most practical purposes, it is safe to assume that natural language lexicons are finite."* Indeed, if the hash table approach were applied to an Arabic lexicon with all partially vowelized forms, the list would grow to an estimated tens (or hundreds) of billions of forms, almost unmanageable for a lookup table.

[38] Unitex includes a "compile and flatten" variant of the compiler for transducers. The output of Unitex transducer compilation is in the FST2 format. The basic version of the compiler "*conserves the architecture in subgraphs of the grammars, which is what makes them different from strict finite state transducers. The Flatten program allows you to turn a FST2 grammar* into *a [single] finite state transducer whenever this is possible, and to construct an approximation if not. This function thus permits to obtain objects that are easier to manipulate and to which all classical algorithms on automata can be applied.*" (Paumier, 2016, UNITEX-User manual 3.1RC, Section 6.2.2)

[39] Windows 7, HP Zbook 15 G2, i7- 250GHz x64, Memory: 16 GB.

Table 6.2. Comparing the two experiments of compression

	Together	Separately	
Compression algorithm	Semitic	Semitic	Concatenative
Number of entries	**6 082 374**	**4 280 000**	**1 805 000**
Flat File Size (Megabytes)	341	228	114
Bin file size (Megabytes)	**13.5**	**10.5**	**0.5**
INF entries	83 858	65 337	2 859
States	252 774	200 450	30 746
Transitions	586 103	427 027	68 305

With these two compressed files, the analysis speeds up to 1,800 words/second on a 2014 Windows laptop (5000 words/second on MacBook Pro i7, 2,0 GHz, 8 GB RAM), which is almost three times the speed of AlKhalil-2 (632 word/s) or BAMA (685 words/s). Compared with the compression with the Semitic compression only, the split speeds up the analysis by 80%.

In Neme and Laporte (2013), we compare the performance of our parser and MAGEAD-Express (both analysers cover verbal inflection and use FST technologies):

- *The resources of MAGEAD-Express (8700 verbs) compile in 48 h, and the analysis of a verb takes 6.8 ms* (Altantawy *et al.*, 2011- Octobre:123) (Section 2.4.2)
- *Neme (2011-August) describes a morphological analyser for Arabic verbs with a comprehensive lexical coverage:15 400 verbs. The dictionary compiles in 2 minutes and the analysis of a verb takes 0.5 ms on a 2009 Windows laptop, outperforming MAGEAD-Express* (Section 2.4.5)

With Hebrew resources (21,000 lemmas/0.5 million forms), Wintner (2008) reports the following numbers when using an FST lookup procedure and compression: 25 minutes to compile and compress the resources; and the analysis speed is 83 words/second. On the other hand, with the same Hebrew resources, when using a lookup with a hash table and a Java classical programming platform, the compilation of the resources takes few seconds and the analysis speeds up to 1500 words/second.

Our lookup is fast because the design is simple. Our inflectional ALR has a solid, straightforward Arabic morphological basis which made it possible to generate a comprehensive, detailed, accurate full-form dictionary, including literal morpho-phonological variants and with vowels fully represented. No on-the-fly computation of morphological changes in agglutinated forms is required during the analysis. The agglutination grammars in the ALR specify literal orthographical variants, which also speeds up the process.

6.3 Algorithm for restoring vowels

As explained before, the compressed dictionary consists of a transducer containing all possible fully vowelized forms. The lookup procedure explores in parallel the transducer and the text to find matches. Once a match is found, the transducer gives access to a compact tag that can be used to reconstruct a full dictionary entry.

The transducer/text matching takes into account partial vowelization and other Arabic typographical rules. The rules enabled by the user in the configuration file (see Section 4) affect this matching process. The code that explores the transducer looks first for an exact match but also looks for alternate matches depending on the rules that have been activated[40].

For instance, with the predefined rules, if the dictionary contains the form *kitaAbFA*, the lookup procedure matches *ktAbFA* in the text and restores the missing vowels from the dictionary. It also matches the input forms *ktAbAF* and *kitAbAF*, if the rule about the inversion between *A* and *F* is active. Then it uses the compact tag associated to *kitaAbFA* to get the lemma *kitaAb* and the POS/inflectional codes `N:msiA`. In the end, the output (cf. Fig.5.3) contains the following line with the fully diacritized form retrieved from the dictionary:

```
kitaAbAF,kitaAb.N:msiA
```

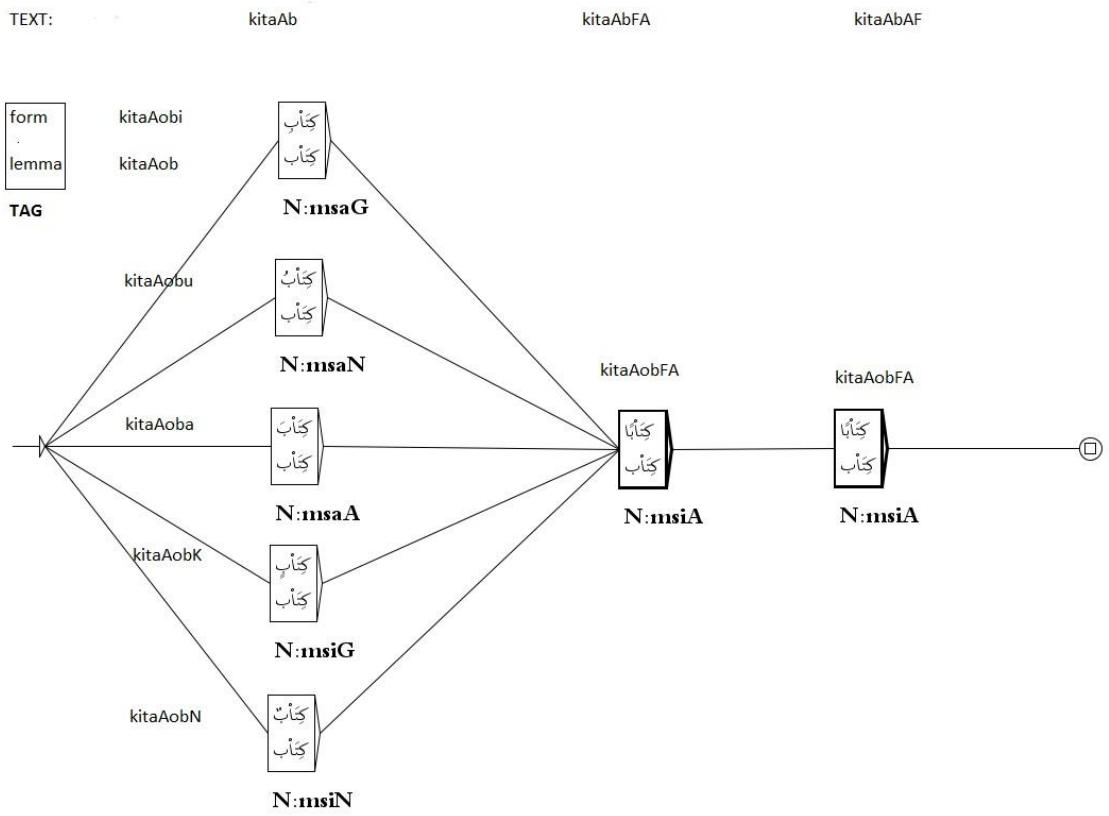


Fig. 6.3. Restoring the vowels. Parsing outputs of the sequences kitAb, kitaAbFA, kitaAbAF

[40] As a preprocessing, we normalize the text by keeping one space between words and trimming the *tatweel* character from words. This character is used for text justification and to extend the horizontal connexion line between two connected letters, as in *kt__Ab* كتـــاب instead of كتاب. Obviously, the *tatweel* is not used in the dictionary.

6.4 Exploiting lattice output in an NLP pipeline

As opposed to most taggers, who output a single analysis for each word, our tagging outputs several analyses, forming a lattice. In this section, we show how such a labeled word lattice can be exploited, either with Unitex or by including it into a processing chain with other systems.

First, Unitex itself can search a labeled lattice for a user-defined query, as exemplified in Section 5.3.2.. The presence of several analyses in parallel in the lattice might theoretically reduce the precision of the search results. However, this kind of lattice search is probably the most popular case use of Unitex in academia and NLP companies, since the Unitex default pre-processing looks up a compressed dictionary and provides the list of possible tags for each word; and, with typical queries, precision is not significantly lower than with a search performed on classical, single-analysis tagged text. Fairon, Paumier et Watrin (2005) quantify the difference in precision on the recognition of French syntactic structures. They formalize the syntactic structure of French verbs in order to generate "*parametrized graphs (Unitex, User Manual 3.2 Chap. 9), drawn with the help of Recursive Transition Network (RTN) formalism. Such graphs describe linguistic constructions […]. [The] method does not distinguish between pattern matching and parsing. Once we have generated graphs, we consider them as patterns. We use the pattern matching function of Unitex to find all matching sequences in a text. If sequences are matched by a graph, then we can say that we have parsed these sequences*". They make an evaluation of the identification of the syntactic structures for the most common five verbs in a corpus of 1.5 Million tokens. They demonstrate that the ambiguities present in the tagged lattice output do not prevent the syntactic parsing of verbal constructions and reach a comparable precision whether applied to an input with lattice ambiguity or without by using a statistical approach, like the one in TreeTagger.

Second, the labeled lattice can be turned to tagged text by selecting a path. Krstev et al. (2018) do that with Unitex for Serbian text without diacritics. We summarize their pipeline by the following:

1. For each word W_b they retrieve all possible Serbian words that use diacritics.
2. For each word W_b they rank all the possible candidates (W_{b1},W_{b2}, . . . ,W_{bn}) according to the possibility of their occurrence in a text.
3. For each word W_b that has more than one possible candidate W_{bi}, their procedure uses heuristics (based on the Corpus of Contemporary Serbian and processed for uni-, bi- and tri-gram frequencies), lexicons and rules (local grammars) to choose one.

"The evaluation results reveal that, depending on the text, accuracy ranges from 95.03% to 99.36%, while the precision (average 98.93%) is always higher than the recall (average 94.94%)" (Krstev et al., 2018:41)

Similar experiments have already been tried with success with a discriminant model or a hidden Markov model on lattices obtained with dictionaries and other tools than Unitex, in Turkish (Sak et al., 2011) and in Arabic (Chennoufi, Mazroui, 2017).

Sak et al. (2011) select the most likely analysis via a discriminative algorithm by exploiting the morphological tags associated to agglutinated morphemes in a Turkish token, "*The problem of finding the most likely morphological analyses of the words in a sentence can be solved by*

estimating some statistics over the parts of the morphological analyses on a training set and then choosing the most likely parse output using the estimated parameters. For parameter estimation, we use the averaged perceptron algorithm."

They conclude that *"Morphology is a very important knowledge source for morphologically complex languages like Turkish. Using these resources and tools, one can parse a text corpus and obtain the morphological analyses of the words as well as their probabilities, disambiguate the parse outputs, train statistical models using the web corpus, and build applications that fully exploit the information hidden in the morphological structure of words."*

Chennoufi & Mazraoui (2016) present a solution with HMM modeling for a diacritizer that uses *"a hybrid system for automatic diacritization of Arabic sentences combining linguistic rules and statistical treatments"*. The processing is divided into 4 stages, and the 4th stage is a fallback procedure for unknown words:

"After morphological analysis step that gives for each word all its possible diacritizations, and following the validation step of transitions between pairs of diacritized words and the application of diacritic rules, we present the third stage of diacritization process. It consists of a statistical treatment based on the hidden Markov models and the Viterbi algorithm (Neuhoff, 1975), which provides the most likely diacritized sentence (Fig. 2). The representation of observed states of HMM are the Arabic words without diacritics (eg " فهمتم " /fhmtm/) and the hidden states are diacritized word forms (eg " فَهِمْتُمْ " /fahimotumo/) (Elshafei et al., 2006; Bebah et al., 2014). This model states provided the best scores of automatic diacritization compared to other hidden states like lists of diacritical marks (Bebah et al., 2014).

They conclude, *"The good performances of our system are consequences of:*

- *The robustness of the* ***second version*** (with a large improvement of lexical coverage compared to the first one) *of AlKhalil analyzer used by our system in the morphological stage;*
- *The use of syntactic and diacritic rules;*
- *The strong representation of the corpus used in the training phase given its large size."*

Summing up, even if they output a labeled lattice with several analyses in parallel, our linguistic resources will improve downstream Arabic NLP pipelines, because the lexicon has comprehensive coverage and *unknown word*s may easily be added to the lexicon with their inflexional variations; moreover, specific symbolic grammar rules or statistical approaches may be also applied to remove paths from the lattice outputs, and with its fine-grained grammatical tags, our approach can enhance further the accuracy of statistical algorithm processing in the future.

Our resources-centered approach to Arabic NLP with Unitex reinforces the readability and maintainability of lexica and grammars for Arabic speakers and linguists; combined with machine learning, it can improve upon the best hybrid solutions in the current state-of-the-art in Arabic NLP.

7 Conclusions

Why do computer scientists ignore vowels in their Arabic-processing systems? As Maamouri et al. (2006) note, *"Since non-diacritized text prevails, the Arabic NLP community seems to have accepted using it as the de facto 'real world' information material without feeling an obligation to question its choice/use, even espousing the idea sometimes that the robustness of software algorithms can deal with the problem and reduce the negative effect of the missing information on their research." […] "The prohibitive cost and the usually unequal and questionable quality of human/manual diacritization have led the scientific Arabic NLP community and its sponsors to focus more on volume of un-vowelized data so far."*

Also note their excellent later discussions presented in *Diacritization: A challenge to Arabic treebank annotation and parsing* (Maamouri et al. 2008): "*Much parsing work with the ATB has used the unvocalized form, on the basis that it more closely represents the "real-world" situation. We point out some problems with this usage of the unvocalized data and explain why the unvocalized form does not in fact represent 'real-world' data*". The fact that vowels are largely absent from written text does not prevent us from taking advantage of them in applications.

Contrariwise, our system presents two dozen rules handling short vowels and gemination omission and glottal stop variations, each of which may be enabled or disabled according to the goal of the application. As in traditional dictionaries, we also provide lexicographers with a simple means to represent short vowel variations in inflected forms, grouping more forms under the same lemma. We have implemented as well adequate and specific inflectional operators that can be used easily by native linguists in Arabic (and Austronesian languages).

Our approach to Arabic morphology redefines and reuses standard concepts from the Semitic tradition (Neme & Laporte, 2013). Our lemmatized representation and implementation of morphology is similar to the grammatical tradition in that prefixes and suffixes of verbs are included in the inflectional representation and we account for clitics independently in agglutination grammars; whereas in the implementation of the stem-based approach, the boundaries between such affixes and clitics are ambiguous and fuzzy. Our distinctive approach to morphological analysis is integrated in a ***one-step processing***. This processing is defined by the application of agglutination grammars that validate the delimited word forms (DWF), which includes checking a core POS represented by a diacritized full form, and selecting only compatible solutions when the DWF is partially vowelized.

The supervised machine learning approach requires a large tagged dataset in order to be successful (for instance in Named Entity Recognition). Such resources are scarce for Arabic, or at least difficult (repetitive and "tedious") to tailor to specific needs. Contrariwise, with our lexical resources (once validated thoroughly) and a local grammar approach, such dataset resources are unnecessary or can be produced semi-automatically.

The excitement (2000-2018) for exclusive Machine Learning and statistical approaches comes mainly from the fact that the market needs quick development of viable solutions. Such solutions in simple applications, such as spell checking, indexation…, have satisfactory accuracy for English and even French, but not for Arabic. Previous experiences with ML (till 2017) show that these approaches were not able to propose satisfactory and accurate solutions, even in simple applications. Statistical approaches reached their limits for Arabic NLP, as is demonstrated by the superiority of the Microsoft Arabic spell checker, based on lexical

resources, over the one in GoogleDocs. Without Arabic lexical resources, the output of an NLP pipeline is disappointing.

Even with the latest RNN-LSTM technologies, recent publications show that using a rich morphological analyser with large coverage will improve *drastically* the accuracy of morphological tagging. In the case of Arabic NLP, it is time to take the best from all fields of NLP and linguistics: lexicography, morpho-syntactic rules, FST technologies, semantic methodologies, and statistical approaches.

The Arabic-Unitex resources provide a lexical coverage of 99 percent of the words used in online news media, and they offer an integrated, simple and efficient way of restoring vowels in partially vowelized or unvowelized words, by using almost standard finite-state technologies and algorithms. Moreover, we have tested our encoding scheme with native linguists, without noticing any strain in the learning process. Arabic-Unitex complies at the same time with the Semitic tradition, lexicographic tradition, a straightforward legibility and incrementability of the resources.

8 References

Abandah, G.A., Graves, A., Al-Shagoor, B., Arabiyat, A., Jamour, F., Al-Taee, M., 2015. Automatic diacritization of Arabic text using recurrent neural networks. Int. J. Doc. Anal. Recogn. 18, 183–197. http://dx.doi.org/10.1007/s10032-015-0242-2.

Abdel-Nour, Jabbour (2006). Dictionnaire Abdel-Nour al-Mufassal Arabe-Français. Dar El-Ilm Lil-Malayin. 10th edition.2034 pages, 3 columns.

Al-Bawab, M., Mrayati, M., Alam, Y.M., Al-Tayyan, M.H. (1994). A computerized morpho-syntactic system of Arabic. In The Arabian Journal of Science and Engineering, 19, 461-480. Published by KFUPM, Saudi Arabia.

Al-Ghalāyini, Mustafa (2007). "Jāmi3 al-durūs al-'arabiyah" (A university grammar textbook). 1st edition 1912. Dar El Fikr Printers-Publishers, Beirut. 3 volumes, 570 pages. In Arabic.

Al-Jar, Khalil (1973).Al-mu'jam al-'arabiy al-Hadith. Larousse. 1973, 53,500 lexical entries.1300 pages, 2 columns, in Arabic.

Altantawy, Mohamed; Habash, Nizar; Rambow, Owen (2011). Fast Yet Rich Morphological Analysis. In Proceedings of the 9th International Workshop on Finite State Methods and Natural Language Processing (FSMNLP), pages 116-124.

Alqahtani, Sawsan; Diab, Mona; Zaghouani, Wajdi (2018). ARLEX: A Large Scale Comprehensive Lexical Inventory for Modern Standard Arabic in the 3rd Workshop on Open-Source Arabic Corpora and Processing Tools, Miyazaki, Japan (OSACT 3, May-2018) pages 1-7.

Altantawy, Mohamed; Habash, Nizar; Rambow, Owen; Saleh, Ibrahim (2010). Morphological Analysis and Generation of Arabic Nouns: A Morphemic Functional Approach. In Proceedings of the Language Resource and Evaluation Conference (LREC), Malta, pages 851-858.

Attia, M. 2006. An Ambiguity-Controlled Morphological Analyzer for Modern Standard Arabic Modelling Finite State Networks. In: Challenges of Arabic for NLP/MT Conference, The British Computer Society, London, UK.

Attia, Mohammed, Pavel Pecina, Lamia Tounsi, Antonio Toral, Josef van Genabith. (2011). An Open-Source Finite State Morphological Transducer for Modern Standard Arabic.

International Workshop on Finite State Methods and Natural Language Processing (FSMNLP). Blois, France.

Attia., M., Yaseen., M., Choukri., K. (2005). Specifications of the Arabic Written Corpus produced within the NEMLAR project, www.NEMLAR.org.

Azmi, Aqil, Reham S Almajed. 2015. A survey of automatic Arabic diacritization techniques. Natural Language Engineering, 21, pp 477–495. doi:10.1017/S1351324913000284

Beesley, Kenneth. 1990. Finite-state description of Arabic morphology. In Proceedings of the Second Cambridge Conference on Bilingual Computing in Arabic and English, September 5-7. No pagination.

Beesley, Kenneth R. 1991. Computer analysis of Arabic morphology: A two-level approach with detours. In Bernard Comrie and Mushira Eid, editors, Perspectives on Arabic Linguistics III: Papers from the Third Annual Symposium on Arabic Linguistics, pages 155-172. John Benjamins, Amsterdam. Read originally at the Third Annual Symposium on Arabic Linguistics, University of Utah, Salt Lake City, Utah, 3-4 March 1989.

Beesley, Kenneth R. (1996). Arabic finite state morphological analysis and generation. In Proceedings of the International Conference on Computational Linguistics (COLING), Copenhagen, Center for Sprogteknologi, volume 1, pages 89-94.

Beesley, Kenneth R.. 1998a. Arabic morphological analysis on the Internet. In ICEMCO-g8, Cambridge, April 17-18. Centre for Middle Eastern Studies. Proceedings of the 6th International Conference and Exhibition on Multi-56 lingual Computing. Paper number 3.1.1; no pagination.

Beesley, Kenneth R. 1998b. Arabic stem morphotactics via finite-state intersection. Paper presented at the 12th Symposium on Arabic Linguistics, Arabic Linguistic Society, 6-7 March, 1998, Champaign, IL.

Beesley, Kenneth R. 1998c. Consonant spreading in Arabic stems. In COLING'98.

Beesley, Kenneth R. 1998d. Constraining separated morphotactic dependencies in finitestate grammars. In FSMNLP-98, Bilkent.

Beesley, Kenneth R. (2001). Finite-State Morphological Analysis and Generation of Arabic at Xerox Research: Status and Plans in 2001. In Proceedings of the ACL/EACL Workshop 'Arabic Language Processing: Status and Prospects', pages 1-8.

Beesley, Kenneth R., Lauri Karttunen, (2003). Finite State Morphology, CLSI Studies in Computional Linguistics, 509 pages.

Ben Mesmia F., Friburger N., Haddar K. and Maurel D. 2015. Arabic Named Entity Recognition Process using Transducer Cascade and Arabic Wikipedia. Proceedings of Recent Advances in Natural Language Processing, pp 48–54, Hissar, Bulgaria.

Boudchiche, M., Mazroui, A., Ould Abdallahi Ould Bebah, M., Lakhouaja, A., Boudlal, A., 2016. AlKhalil Morpho Sys 2: A robust Arabic morpho-syntactic analyzer. J. King Saud Univ. - Comput. Inf. Sci. http://dx.doi.org/10.1016/j.jksuci.2016.05.002

Boudchiche, M., Mazroui, A., Ould Abdallahi Ould Bebah, M., Lakhouaja, A., Boudlal, A., 2014, L'Analyseur Morphosyntaxique AlKhalil, Morpho Sys 2, DOI: 10.13140/RG.2.1.4280.0085

Boudlal, A., Lakhouaja, A., Mazroui, A., Meziane, A., Ould Abdallahi Ould Bebah, M., Shoul, M., 2010. Alkhalil Morpho SYS1: a morphosyntactic analysis system for arabic texts. In: International Arab Conference on Information Technology. Benghazi, Libya, pp. 1–6.

Buckwalter Arabic Morphological Analyzer Version 1.0. (2002). LDC Catalog No.: LDC2002349.

Buckwalter, T. (2004). Buckwalter Arabic Morphological Analyser Version 2.0. Linguistic Data Consortium (LDC) Catalog Number LDC2004L02, ISBN 1-58563-324-0.

Chennoufi, A., Mazroui, A., 2016. Morphological, syntactic and diacritics rules for automatic diacritization of Arabic sentences. Journal of King Saud University - Computer and Information Sciences.

Debili, F., Achour, H. (1998). Voyellation automatique de l'arabe. Actes du Workshop on Computational Approaches To Semitic Languages, Université de Montréal.

Debili, F., Souissi, E. (1998). Etiquetage grammatical de l'arabe voyellé ou non. In Proceedings of the Workshop on Computational Approaches to Semitic Languages, Stroudsburg.

Debili, F., Achour, H., Souissi, E. (2002). La langue arabe et l'ordinateur: de l'étiquetage grammatical à la voyellation automatique. Correspondances de l'IRMC, N°71, Tunis.

Gal Y., (2002). An HMM approach to vowel restoration in Arabic and Hebrew. In ACL-02 Workshop on Computational Approaches to Semitic Languages.

Fairon, C., Paumier, S., Watrin, P. (2005). Can we parse without tagging? 2nd Language & Technology Conference (LTC'05), 2005, Poznan, Poland. 2nd Language & Technology Conference (LTC'05), pp.473-477, 2005

Habash, N. and Rambow O. (2005). Arabic Tokenization, Part-of-speech Tagging and Morphological Disambiguation in One Fell Swoop. In Proceedings of the Conference of the American Association for Computational Linguistics, New York.

Habash, Nizar; Rambow, Owen (2006). MAGEAD: A Morphological Analyzer and Generator for the Arabic Dialects. In Proceedings of the International Conference on Computational Linguistics and Annual Meeting of the Association for Computational Linguistics (COLING-ACL), Sydney, Australia, pages 681–688.

Habash N. and Rambow O., (2007). Arabic Diacritization through Full Morphological Tagging, In Proceedings of the North American chapter of the Association for Computational Linguistics (NAACL), Rochester, New York.

Habash N. (2010) Introduction to Arabic natural language processing. Synthesis lectures on human language technologies. Morgan & Claypool, San Rafael. doi:http://dx.doi.org/10.2200/S00277ED1V01Y201008HLT010

Hamdi, Ahmed (2012), Apport de la diacritisation dans l'analyse morphosyntaxique de l'arabe; 247–254.

Hamed Osama, Torsten Zesch, (2017). A Survey and Comparative Study of Arabic Diacritization Tools. In JLCL, volume 32, Number 1, http://jlcl.org/content/5-allissues/1-Heft1-2017/Heft1-2017.pdf.

Kirchhoff K. and Vergyri D., (2005). Cross-dialectal data sharing for acoustic modeling in Arabic speech recognition. Speech Communication, 46(1):37–51, May.

Krstev, Cvetana, Stanković Ranka, Vitas Duško. 2018. Knowledge and Rule-Based Diacritic Restoration in Serbian. In Proceedings of the Third International Conference Computational Linguistics in Bulgaria, CLIB-2018:41-51.

Maamouri Mohamed, Ann Bies, and Seth Kulick. 2006. Diacritization: A challenge to Arabic treebank annotation and parsing. In Proceedings of the British Computer Society Arabic NLP/MT Conference, London, UK, October.

Maamouri Mohamed, Seth Kulick, Ann Bies 2008. Diacritic Annotation in the Arabic Treebank and its Impact on Parser Evaluation. . In Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC 2008), Marrakech, Morocco, May 28-30, 2008.

Maamouri, Mohamed, et al. LDC Standard Arabic Morphological Analyzer (SAMA) Version 3.1 LDC2010L01. Web Download. Philadelphia: Linguistic Data Consortium, 2010.

Maamouri, M., Bies, A. & Buckwalter, T. (2004). The Penn Arabic treebank: Building a largescale annotated Arabic corpus. In NEMLAR Conference on Arabic Language Resources and Tools, Cairo, Egypt.

Maamouri, Mohamed, Ann Bies, Tim Buckwalter, and Wigdan Mekki. The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus, 2004.

Mubarak, Hamdy, Kareem Darwish. 2014. "Automatic Correction of Arabic Text: a Cascaded Approach". Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (NLP).

Neme, Alexis, Laporte Éric (2013). Pattern-and-root inflectional morphology: the Arabic broken plural. Language Sciences. http://dx.doi.org/10.1016/j.langsci.2013.06.002

Neme, Alexis (2011). A lexicon of Arabic verbs constructed on the basis of Semitic taxonomy and using finite-state transducers. In Proceedings of the International Workshop on Lexical Resources (WoLeR) at ESSLLI.

Neme, Alexis Amid (2014). Why Microsoft Arabic Spell checker is ineffective, Linguistica Communicatio, http://www.al-erfan.com/, 2014, Arabic Language in Information Technology, 16, pp.55. <http://www.al-erfan.com/>

Paumier, Sébastien. (2016). Unitex – User manuel 3.1RC, University of Marne-la-Vallée.

Pasha Arfath, Mohamed Al-Badrashiny, Mona Diab, Ahmed El Kholy, Ramy Eskander, Nizar Habash, Manoj Pooleery, Owen Rambow, Ryan Roth (2014). MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic, Proceedings of the Ninth International Conference on Language Resources and Evaluation LREC, May 2014.

Revuz, Dominique (1992): Minimization of acyclic deterministic automata in linear time. Theoretical Computer Science 92:1, Elsevier 181-189

Sak, H., Güngör, T. & Saraçlar M. (2011). Resources for Turkish morphological processing. Lang Resources & Evaluation (2011) 45: 249. https://doi.org/10.1007/s10579-010-9128-6

Shaalan, K., Allam A., Gomah A. (2003). Towards automatic spell checking for Arabic. Conference on Language Engineering.

Shaalan, Khaled, Samih, Younes, Attia, Mohammed, Pecina, Pavel, & van Genabith, Josef (2012). Shaalan, Khaled,Younes Samih, Mohammed Attia, Pavel Pecina, and Josef van Genabith. (2012). Arabic Word Generation and Modelling for Spell Checking. Language Resources and Evaluation (LREC). Istanbul, Turkey. Pages: 719-725

Smrz, Otakar. (2007). ElixirFM — Implementation of Functional Arabic Morphology. In Computational Approaches to Semitic Languages, ACL 2007, Prague.

Traboulsi, Hayssam. 2009. Arabic named entity extraction: A local grammar-based approach. In Proceedings of the International Multi-conference on Computer Science and Information Technology (IMCSIT 2009), pages 139–143, Mragowo.

Wintner S. (2008). Strengths and weaknesses of finite-state technology: a case study morphological grammar development. Nat Lang Eng 14(4):457–469. doi:http://dx.doi.org/10.1017/S1351324907004676

Nasser Zalmout and Nizar Habash (2017). Don't Throw Those Morphological Analyzers Away Just Yet: Neural Morphological Disambiguation for Arabic. Conference on Empirical Methods in Natural Language Processing, Proceedings pages 715–724 Copenhagen, September, 2017

Zitouni I., Sorensen J. S., and Sarikaya R., (2006), “Maximum entropy based restoration of Arabic diacritics”. In Proceedings of ACL’06

9 Appendixes

The Unitex predefined Arabic typographical rules are the following:

```
fatha omission=YES              /a
damma omission=YES              /u
kasra omission=YES              /i
sukun omission=YES              /o silent vowel
superscript alef omission=YES          /R superscript  alif
fathatan omission at end=YES           /F
dammatan omission at end=YES           /N
kasratan omission at end=YES           /K
shadda fatha omission at end=YES       /Ga
shadda damma omission at end=YES       /Gu
shadda kasra omission at end=YES       /Gi
shadda fathatan omission at end=YES          /GF
shadda dammatan omission at end=YES          /GN
shadda kasratan omission at end=YES          /GK
shadda fatha omission=YES
shadda damma omission=YES
shadda kasra omission=YES
shadda superscript alef omission=YES   /R in AllGRhu = Allaah
solar assimilation=YES          /insertion a gemination after consonant
lunar assimilation=NO           /no  assimilation  exclude  assimilation
                                /after non-coronal consonnant
Al with wasla=YES                           /L   Al =>Ll
alef hamza above O=YES                      / O => A
alef hamza below I to A=YES                 / I => A
alef hamza below I to L=YES                 / I => L
fathatan alef equiv alef fathatan=YES              /at the end FA => AF
fathatan alef maqsura equiv alef maqsura fathatan=YES   /FY =>YF
```

Table 5.1.b. Inflectional features and values carried by POS used in Arabic-Unitex

POS carrying the value	FEAT:VALUE	In English	Encoded example	In Arabic	Arabic examples
<V>,<N>,<A>,<PRO>	Gender				
	:m	masculine	<PREP><N:fsDA><PRO:Gen:3fs>	مذكّر	لرجلِها
	:f	feminine	<DET><N:fsDA>	مؤنّث	الشمسَ
<V>,<N>,<A>,<PRO>	Number				
	:s	singular	<N:msiN>, <N:msiG>	مفرد	قائدٌ، خائنِ
	:d	dual	<N:fdiN>, <N:mdiA> or <N:mdiG>	مثنّى	طاولتان، مراقبَين
	:p	suffixed plural	<N:fpiN>, <N:mpiN>	جمع سالم	طاولاتٌ، مراقبون
<N>,<A>	:q	broken plural (non-suffixal)	<N:qiN>, <A:qiG>	جمع تكسير	قادةُ خوَنةٍ
<N>,<A>,<V:F>,<V:M>	Definitenes s				
	:D	Definite	<DET><N:fsD>	معرّف	الرسالةِ
	:a	construct state	<N:msaN><DET><N:msDG>	مضاف	مِقعدُ الرجلِ
ADV	:i	indefinite	<N:fsiN>	نكرة	مِقعدٌ
<N>,<A>	Case				
	:N	Nominative		مرفوع	رجلٌ
<ADV>	:A	Accusative		منصوب	رجلاً
	:G	Genitive		مجرور	رجلٍ
<V>	Voice, Aspect	Mode			
	:a	active		معلوم	يَكتبُ
	:b	passive		مجهول	يُكتَبُ
	:P	Perfect	<CONJC><V+nopro:aP3ms>	ماضٍ	وضربوا
	:I	Imperfect	<CONJS+subjunc><V+pro:aI3mp><PRO+acc:3fs>	مضارع	ليضربوها
	:Y	Imperative	<V+pro:Y3mp><PRO+acc:3mp>	أمر	إضربوهم
	:F	Active Participle	<V:FmsiA>	إسم فاعل	ضارباً
	:M	Passive Participle	<V:MfsiA>	إسم مفعول	مضروبةً
	:N	iNdicative		مرفوع	
	:S	Subjunctive		منصوب	
	:J	Jussive		مجزوم	
	:E	Energetic		مؤكد	
<V>,<PRO>	Person				
	:1	1st person		متكلّم	
	:2	2nd person		مخاطب	
	:3	3rd person		غائب	

Table 5.1.c. Semantic and other syntactic features and values in Arabic-Unitex. Semantic encodings *in italics* in the table are not encoded systematically in the dictionary and depend on the requirements of a domain

POS carrying the feature	Code	In English	Encoded examples	In Arabic	Arabic examples
<N><PREP> <PRO><PRTCL>	Case				
	+Nom	Nominative	<PRO+Ppers+Nom:1s>	مرفوع	أنَـا
	+Acc	Accusative	<PRO+Ppers+Acc:3d>	منصوب	ضربـهما
	+Gen	Genitive	<PREP+pro> <PRO+Ppers+Gen:3d>	مجزوم	بـهِمَاْ
<CONJS><PRO> <PRTCL>	Mode				
	+indic	Governs indicative	<CONJS+indic+nopro>	مرفوع	قَدْ
	+subjunc	Governs subjunctive	<CONJS+subjunc+nopro>	منصوب	لَنْ
	+juss	Governs jussive	<CONJS+juss+nopro>	مجزوم	لَمْ
<PREP><V> <N><A>	+pro	form with mandatory enclitic	<PREP+pro> <PRO+Ppers+Gen:3fs>		بـهـا
<PREP><V> <N><A>	+nopro	form incompatible with enclitic	<V+nopro:aP3mp>		كـتبـوا
<N><A>	*+Hum*	Human			طبيب
	-Hum	non-Human			دفـتـر
<N><PREP><PRO>	*+Loc*	Locative	<PRO+Pinterrog+Loc>		أيـن؟
<N><PREP><PRO>	*+Temp*	Temporal	<PREP+nopro+Temp>		طِيلَة
<PRTCL>	*+Vocative*	PRTCL	<PRTCL+Vocative >		يـا أيُـهـا
<N>	*+Abst*	Abstract	<N+Abst:ms>		حصول
			<N+Instance:fs> such as shippment	إسم مـرة	شحنة
	+generic		<N+generic:ms> such as shipping		شحن
<N>	*+Anml*	Animal	<N+Anml:ms>		حصان
<N>	*+AnmlColl*	collective animal	<N+AnmlColl:fs>		مـاشية
<N>	*+Conc*	Concrete	<N+Conc:fs>		طاولـة
<N>	*+ConcColl*	collective concrete	<N+ConcColl:p>		بـهـارات
<N>	*+HumColl*	Collective	<N+HumColl:msiN:ms>	إسم جمع	شعب
	+ species	Species	<N+AnmlColl+species:ms>	اسم جنس جمعي	**بقر**
	+count	countable species	<N+Anml+count:fs>	إسم الـواحد	**بقرة**
	+uncount	Uncountable	<N+Anml+uncount:fs>	**اسم الـجنس**	**لـبن**
<V>	*+t*	Transitive	<V+t>	مـتعدّي	ضرب
<V>	*+i*	Intransitive	<V+i>	لازم	جـاء
<V>, <N>, <A>,<ADV>	*+z1*	General vocab.	<N+z1>	مفـردات عـامة	دفـتـر
<V>, <N>, <A>,<ADV>	*+z2*	Specialized vocab.	<N+z2>	مفـردات مـتخصصة	بـرنـت - بـربـون
<V>, <N>, <A>,<ADV>	*+z3*	very specialized	<N+z3>	مـتخصصة جداً	الإيـكسيـتون

Pattern-and-root inflectional morphology: the Arabic broken plural

Alexis Amid Neme, Eric Laporte

Pattern-and-root inflectional morphology: the Arabic broken plural

Alexis Amid Neme [1] - Éric Laporte [2]

Abstract

We present a substantially implemented model of description of the inflectional morphology of Arabic nouns, with special attention to the management of dictionaries and other language resources by Arabic-speaking linguists. Our model includes broken plurals (BPs), i.e. plurals formed by modifying the stem.

It is based on the traditional notions of root and pattern of Semitic morphology. However, as compared to traditional Arabic morphology, it keeps the formal description of inflection separate from that of derivation and semantics. As traditional Arabic dictionaries, the updatable dictionary is structured in lexical entries for lemmas, and the reference spelling is fully diacritized. In our model, morphological analysis of Arabic text is performed directly with a dictionary of words and without morphophonological rules.

Our taxonomy for noun inflection is simple, orderly and detailed. We simplify the taxonomy of singular patterns by specifying vowel quantity as v *or* vv*, and ignoring vowel quality. Root alternations and orthographical variations are encoded independently from patterns and in a factual way, without deep roots or morphophonological or orthographical rules. Nouns with a triliteral BP are classified according to 22 patterns subdivided into 90 classes, and nouns with a quadriliteral BP according to 3 patterns subdivided into 70 classes. These 160 classes become 300 inflectional classes when we take into account inflectional variations that affect only the singular.*

We provide a straightforward encoding scheme that we applied to 3 200 entries of BP nouns.

1. Objective

We present a model of description of the inflectional morphology of Arabic nouns. Our purpose is to generate comprehensive dictionaries for Arabic natural language processing (NLP), and to equip them with easy procedures of manual, computer-aided updating. No such dictionary is currently available for Arabic NLP (cf. Section 2.4). Noun inflection is a crucial part of the inflectional system of Arabic: it regards a large part of the lexicon and 'nouns turn out to be far more complex than verbs' (Altantawy *et al*., 2010:851).[3]

Our approach, inspired from Neme's work on verbs (2011), is to generate plurals from fully diacritized singular forms. The input of the system is a noun lemma with an inflectional code. The output is a list of inflected forms with their morpho-syntactic features. We take fully diacritized spelling as reference, and we deal with partially diacritized or undiacritized spelling through the concept of optional information.

We focus on broken plurals (BPs), defined as those Arabic plurals formed by modifying the stem, as in *Euqodap* 'knot' vs. *Euqad* 'knots'. BPs contrast with suffixal plurals, which are formed by substituting suffixes, as in *Halaqap* 'ring' vs. *HalaqaAt* 'rings'. A large proportion of nouns, e.g. most nouns of concrete objects and animals and many technical terms, have only a BP. 'For the lexicon as a whole, then, broken plural formation is by far the norm rather than the exception' (McCarthy, Prince, 1990:213).

In this paper, examples displayed in the Latin alphabet are transliterated according to Buckwalter-Neme (BN) code, a variant (Neme, 2011, p. 6, note 4) of Tim Buckwalter's transliteration that avoids the use of special characters.[4] The diacritics for short vowels are noted *a, u, i*. A position between two basic letters without any

[1] LIGM, Université Paris-Est - alexis.neme@gmail.com
[2] LIGM, Université Paris-Est; DLL, Universidade Federal do Espírito Santo - eric.laporte@univ-paris-est.fr
[3] We thank Tim Buckwalter for helpful comments and discussions on an earlier version of this article.
[4] In this transliteration, upper-case and lower-case letters, e.g. *E* and *e*, denote distinct, independent consonants : ء, c; آ, C; أ, O; ؤ, W; إ, I; ئ, e; ا, A; ب, b; ة, p; ت, t; ث, v; ج, j; ح, H; خ, x; د, d; ذ, J; ر, r; ز, z; س, s; ش, M; ص, S; ض, D; ط, T; ظ, Z; ع, E; غ, g; ف, f; ق, q; ك, k; ل, l; م, m; ن, n; ه, h; و, w; ى, Y; ي, y; ً , F; ٌ , N; ٍ, K; َ, a; ُ, u; ِ , i; ّ , G; ْ, o. The BN transliteration is implemented in the Unitex

vowel is noted *o*, as in *Euqodap* [ʕuqdap]. In other words, *o* does not note the [o] vowel, but is a silent diacritic: when it is noted, it rules out the hypothesis of a non-scripted short vowel. This transliteration system is entirely based on the digital encoding of text, as defined by the Unicode standard, and does not necessarily reflect its graphic display on the screen (e.g. ligatures) nor its pronunciation.

2. Previous work

2.1. Root-and-pattern morphology

Among the possible formal representations of Arabic morphology, root-and-pattern morphology is a natural representation, as well as for other Semitic languages. It is so widely used that this model is also known as 'Semitic morphology'. A (surface) **root** is a morphemic abstraction, a sequence of letters, which can only be consonants or long vowels,[5] like *Eqd*, where *E* notes the pharyngeal or epiglottal consonant [ʕ], or *swr*, where *w* notes a long vowel in certain conditions. A **pattern** is a template of characters surrounding the slots for the root letters. These slots are shown in the pattern by indices, like in *1u2a3*. Between and around the slots, patterns contain short vowels, and sometimes consonants or long vowels. Once affixes are stripped off the surface form of a word, the remaining stem is analysed as the 'interdigitation' (Beesley, 1996) of a root with a pattern. For example, the stems *Euqodap* 'knot' and the BP *Euqad* 'knots' are represented by the root *Eqd* and, respectively, by the singular pattern *1u2o3ap* and BP pattern *1u2a3* :

Stem	Euqodap	Euqad	عُقْـدَة عُقَـد
Root	E q d	E q d	
Pattern	1u2o3ap	1u2a3	

A root is usually stable across all the forms in a lexical item; grammatical distinctions between these forms correspond to different patterns. Thus, lexical items are classified in biliteral, triliteral, quadriliteral, quinqueliteral depending on the number of letters in their root. The general principles of root-and-pattern morphology are ubiquitous in the Arabic-speaking world and are taught in school. This representation is well established in Arabic morphology and seems well founded.[6]

There is a parallel between this model and Arabic script. Arabic script distinguishes 'basic letters', which are obligatorily written, and diacritics, which are usually omitted. All basic letters are consonants or long vowels, just as all root letters also are; roots are written with basic letters only. This is an additional reason why root-and-pattern morphology is so intuitive for users of Arabic script. Between and around the slots, patterns comprise diacritics, and sometimes basic letters.

The slots for root letters in a pattern are traditionally noted by the consonants *f, E, l, l,* instead of the digits *1, 2, 3, 4*. For instance, *1u2o3ap* and *1u2a3* are noted *fuEolap* and *fuEal* (فُـعَـل ، فُـعْـلَـة). This makes the representation of the pattern pronounceable, and thus easier to remember. We adopted this convention and adjusted it in several ways. We modified the consonant for the 4th slot, so as to have four different consonants *f, E, l, b*. When we script patterns in Buckwalter transliteration, we type these consonants in upper case: *F, E, L, B*, so that the slots are visually salient: *FuEoLap* and *FuEaL*. We note the long vowels *aa ii uu* instead of *aA iy uw*, which would be the fully diacritized BN transliteration. With this convention, adopted by several authors, the slots for the root consonants are easier to identify visually. They appear in capitals, while most other letters in patterns appear in lower case. When *aA* is written in BN transliteration, the upper case letter tends to confuse the recognition of the slots.

---

system (Paumier, 2002).

[5] As a simplification, we introduce here the surface root corresponding to a set of actually pronounced segments, and not the underlying root postulated by traditional Arabic grammar and by generative grammar.

[6] Prosodic morphology uses a close variant of this model (McCarthy, 1981) in which a pattern such as *1i2a3* is replaced by two abstractions: a 'CV skeleton' for the position of vowels, here *1v2v3*, and a 'melody' for their values, here *ia*. This variant is used in some implementations (Kiraz, 1994). We use the traditional form of patterns, which is simpler (Smrž, 2007:33) and more usual to Arabic speakers.

2.2. Traditional morphology

A large part of traditional Arabic morphology (TM), including the description of BPs, dates back to Sibawayh, a grammarian of the VIIIth century (Sibawayh, ed. Haarun, 1977). Since then, his representation has been generally approved and transmitted by grammarians without major improvements. It is largely used at school in Arab countries.

This traditional view describes how BPs are produced from singular nouns. The path from a singular form to a BP passes through a root. The essential steps in this operation are:
- analysing the singular into a root and an existing singular pattern, e.g. *Euqodap* 'knot' = *[Eqd* & *FuEoLap]*,
- selecting a BP pattern, here *FiEaL*,
- combining the root with the BP pattern.

In the first step, we shift from a surface form to the root and pattern level; then, we shift back to surface. The steps listed above present four technical obstacles.

- The analysis of an Arabic word into a root and a pattern is not a deterministic operation and can a priori produce several results (cf. Section 4.1), even after discarding those results that violate any constraints about roots or patterns.

- TM's notions of root and pattern are not exactly the surface root and pattern introduced above, but a 'deep' root, e.g., in the case of *baAb* 'door' باب, *bwb* instead of *bAb*, and a 'deep' pattern. Rules modify these underlying forms to produce surface forms. Thus, the path from a singular form to a BP, in fact, passes through a deep root. To find the deep root, the rules have to be 'unapplied', i.e. applied regressively;[7] then, to generate the BP form, the same rules are applied back in the normal way.

- The BP pattern is generally unpredictable from the singular pattern.

- Once the root is combined with the BP pattern, rules apply and modify the deep forms.

Reliable dictionaries (Abdel-Nour, 2006) and excellent inventories of classes and nouns (Tarabay 2003) can be found. Sure, numerous entries in Tarabay are disused in Modern Standard Arabic, and some classes are missing, for example the human nouns with the *FaEaaLiBap* pattern in the BP, as *barobariyG* 'barbar' بربريّ or *malaAk* 'angel' ملاك. But the system is essentially unchanged since Sibawayh, and has incorporated loanwords harmoniously.

The TM model of BPs is precise enough to define taxonomies: two nouns are assigned the same class if they produce their BP in the same way. However, TM does not explicitly enumerate classes. The notion of taxonomy is also naturally connected with that of codes: two nouns belong to the same class if they are assigned the same BP codes. TM produces BPs from singular nouns through two 'codes': the first is either the singular pattern (*FiEoLap* in the example above) or the deep root (*Eqd*), and the second is the BP pattern (*FiEaL*).

Since Sibawayh, most lexicologists and linguists have contributed in the form of comments, rather than revisions. The accumulated comments tend to make the model seem more complex, not to simplify it. Among modern linguists, those who have adopted the root-and-pattern model have rarely questioned historical authors and practices either.

TM's model of BPs is complex. Tarabay's (2003) book about plural in Arabic, which is almost entirely dedicated to BPs, has 470 pages on 2 columns, plus 100 pages of glossaries representing more than 12 000 entries (not exhaustive, common words are lacking). BPs in themselves give an 'initial impression of chaos' (McCarthy, 1983:292) and are 'highly allomorphic' (Soudi *et al.*, 2002); grammatical and lexical traditions and practices along centuries do not give the impression of an effort towards a simpler and more orderly taxonomy, with fewer classes. Arabic specialists disagree about the deep root of some nouns, e.g. *xanoziyr* 'pig' خنزير is indexed under the roots *xnzr* and *xzr* in Ibn Manzur (1290) and under the root *xzr* in Al-Fairuzabadi (c. 1400). Descriptions of rules are often scattered in reference books, and their conditions of application are not formalized

[7] In case of doubt, lexicons provide the deep root directly.

and not always fully specified. In a typical example, Tarabay (2003:92, footnote 2) mentions a metathesis rule that substitutes *o*<*cons*>*i*<*cons*> by *i*<*cons*>*o*<*cons*>, as in the underlying form **OaxoMiMap* 'vermin' أخشِشَة (the '*' symbol signals a reconstructed, not directly observed form) which takes the form **OaxiMoMap* أخِشْشَة., which in turn is correctly spelt as *OaxiMGap* أخِشَّة , where the *G* diacritic notes the gemination of the preceding consonant. She words the conditions of application as follows: '[The nouns] that pluralize only with the *OaFoEiLap* pattern, that have the *FaEaAL* pattern in the singular and that have identical 2nd and 3rd root letters, apply an *i* shift which is substituted by *o*.' In this footnote, 'pluralize only' means that the noun does not have another BP: if it has a suffixal plural, the rule can apply. Thus, the conditions of applications of this rule are and not fully specified.[8] There are dozens of such rules. Their order of application matters for their final output, but it is not systematically specified. Good traditional dictionaries explicitly provide BPs in surface form, bypassing the pattern and the rules.

The number of classes in a BP taxonomy measures the complexity of the BP system. Since TM does not count classes, let us compute estimations from numbers of patterns. Tarabay (2003) distinguishes 56 BP patterns. This number can be viewed as a measure of the complexity of BP: 'The defining characteristic of fixed-pattern morphology is that consistency in such systems is found not in a consistent proportion or relationship between two forms (a base and a derivative, an input and an output) but in a consistent pattern (of syllable structure and vocalism) imposed on all derived forms of a particular class regardless of the form of the source word' (Ratcliffe, 2001:153). However, the number of BP patterns underestimates the complexity of deducing a BP from a singular, because it overlooks the problem of finding the root. We should then take into account the number of singular patterns. The BP pattern is unpredictable from a given singular pattern, and vice versa, but not all singular pattern/BP pattern pairs are represented in the lexicon. Estimates of the number of singular pattern/BP pattern pairs vary from 105 (Murtonen 1964, survey based on the dictionary of Lane 1893) down to 55 (Soudi *et al*., 2002, citing Levy 1971, based on Wehr 1960) or 44 (El-Dahdah, 2002), but they are limited to the common types. Again, the number of pattern pairs does not take into account the additional complexity brought about by morphological variations. Such variations affect the consonants *w*, *y* and [ʔ] (the glottal stop), and forms with reduplicated or geminated consonants. Tarabay (2003) dedicates 30 pages to the latter type of variations. We estimate that her inventory is equivalent to more than 2 000 classes.

For TM, the description of BPs is required to be consistent with other constraints. For example, roots are also used for the practical purpose of indexing dictionaries. *'The lexical root provides a semantic field within which actual vocabulary items can be located'* (Ryding, 2005:677). Derived nouns such as *miEowal* 'mattock' معول are listed in dictionaries under the root of their base, here *Ewl*, a root that also occurs in words meaning 'howl', 'raise (a family)', 'rely on'... Therefore, the consonants of derivational prefixes, here *m*, are not analysed as being part of the root, even when they are common to the singular and BP of the derived noun, as is the case for this noun.

In a similar vein, the roots and patterns relevant for inflectional morphology are also 'reused' for semantic description. *'A root is a relatively invariable discontinuous bound morpheme, (...) which has a lexical meaning'* (Ryding, 2005:47). TM associates some patterns with semantic features, e.g. the *miFoEaL* pattern with the notion of instrument, as in *miEowal* 'mattock'. However, such associations are never perfectly regular. The *miFoEaL* pattern could not be used as a semantic label for instrument nouns. Some instrument nouns do not have it, e.g. *qalam* 'pen' قلم . The broken plural of *miEowal* 'mattock' معول itself, *maEaAwil* 'mattocks' معاول , is still an instrument noun, and has another pattern.

TM also integrates inflection with derivational morphology, which also involves roots and patterns. When a word is the output of a derivational process and the input of an inflectional process, as *miEowal* 'mattock', it is traditionally implied that its root-and-pattern analysis is the same with respect with the two morphological processes.

[8] Probably because it is relatively intuitive for Arabic speakers: *o*<*cons*>*i*<*cons*> sequences are rare in Arabic, and where they are expected, *i*<*cons*>*o*<*cons*> sequences are often observed.

Thus, notions relevant to production of BPs from singular nouns are reused for three other purposes: dictionary indexing, semantic description or derivational morphology. This integration makes sense in a context of Arabic teaching, in that it facilitates memorization. However, if we consider each of these four objectives separately, the reuse may lead to conflicting constraints, if the best definition of roots and patterns for the different purposes do not coincide exactly, as in the examples above. In addition, this integration makes the assignment of a word to a BP class depend on semantic and derivational information, and not only on inflectional morphology.

Summing up, the TM's account of BPs produces the correct forms, it has been tested and validated over centuries, and it is familiar to the Arabic speakers that are likely to encode and update lexical resources. Dictionaries have a readable layout and provide reliable information. However, there might be room for simplification:
- of the taxonomy,
- of the morphophonological rules,
- of the procedure of assignment of a noun to a class.

2.3. BP in generative grammar

Generative grammar gives several formal models of BP generation, some of them well documented, taking into account large portions of the Arabic lexicon, and based on interesting analyses. McCarthy & Prince (1990) propose a computation of BP stem from singular stem, a 'rule for forming the broken plural' (p. 263); Kihm (2006) formalizes other rules in a rival trend within generative grammar.

As compared to traditional morphology, these models hypothesize underlying forms and rules for surface realisation too, but they endeavour to lower the number of inflectional classes for BP. McCarthy & Prince (1990:210 and 217) view Wright's (1971) account of BP, with 31 plural types, corresponding to 11 singular types, as a 'poorly understood or perhaps even chaotic process', and they try to 'substantiate the informal notion that a single pattern unites all the classes grouped under the iambic rubric'. The price for reducing this 'apparent complexity' are more abstract underlying forms, i.e. more distance between underlying forms and surface forms, and therefore a more complex system of rules. The rules perform, for example, metathesis, after Levy (1971), and glide realisation, after TM and Brame (1970). The complexity of the systems comes from relations between rules, such as order of application, and from the existence of exceptions to them.

In conformity with the generative paradigm, these authors assume that the underlying roots exist in native speakers' minds and are activated during the production of BPs. We are not committed to this assumption, for lack of evidence; in addition, when several underlying roots are a priori possible, as in *qabow/Oaqobiyap* (see Section 3.3), we lack evidence about whether hypothetical underlying roots would be identical or different in respective speakers' minds. Our approach focuses on verifiable facts as much as possible.

The generativist models are not directly exploitable for computational purposes, for two reasons:

- The rules are only partially specified. McCarthy and Prince's (1990) rules rely on a metathesis (Levy, 1971) observed in *OakotaAf* for **kataAf* 'shoulders' أكتاف , but they leave undefined the conditions of application of the metathesis, not because they are easy to describe, but because they are 'not wonderfully transparent'. Instead of this metathesis, Kihm (2006:83) uses an 'augment of obscure origin', but does not specify the conditions of its insertion either. He also sketches rules according to which the 2nd root letter does not count as such when it is a glide, and another that integrates into the root some inflectional affixes of the indefinite singular during the generation of the BP (p. 86), but he does not explain in which conditions. As for the lexical information required to generate BPs, he 'leave[s] the precise formalization of this information to future work' (p. 81). Similarly, McCarthy & Prince do not enter into details to the point that they would tell how many inflectional classes for BP should be distinguished with their model.

- Nouns showing exceptional behaviour are mentioned, but not dealt with in the models. For example, McCarthy & Prince's (1990:273-274) rules with left-to-right association give the correct BP in many quinqueliteral nouns, but they do not propose any device for exceptions, since generative grammar is not committed to describing

lexical items beyond those that 'reflect a regular grammatical process of the language' (p. 267). Generative grammar aims to model a specifically linguistic mental process, and is traditionally not interested in general-purpose mnemonic processes that are supposed to handle exceptions when they are not too numerous. This is an important difference with our objectives, since a comprehensive morpho-syntactic lexicon is required to deal with all cases.

Anyway, the generative models of BP, even incompletely specified, seem already too complex to be the best choice for our practical objective of a system easy to update. Complex relations between rules, such as order of application, and the existence of exceptions to them, obfuscate these systems.

In addition, this additional complexity of the rules (as compared to TM) does not always contribute to simplify the taxonomy of BPs. For example, McCarthy & Prince (1990) predict the quantity of last *i* in quadriliteral BP patterns when the first syllable of singular is bimoraic in the generative sense. This allows for merging the *FaEaaLiB* and *FaEaaLiiB* patterns for some nouns. With reference to the goals of generative grammar, such a prediction makes sense, since it models a linguistic process by a rule which is assumed to comply with a universal format. However, if we now take in mind our goal of simplifying the encoding of lexical items, the prediction tends to complicate the generation of the BP, without lowering the number of patterns, since *FaEaaLiB* and *FaEaaLiiB* must still be distinguished for the BP nouns whose first syllable is not bimoraic.

Kihm (2006:81) claims that his model simplifies dramatically the taxonomy of BPs: 'such a wild variety of forms actually results from one process and from the interplay of a few well defined factors', namely the timbre of an element inserted between the 2nd and 3rd root letters, which is chosen between *i*, *a* and *u*, and the category of the insertion: consonant or vowel. However, this claim overstates the simplicity of Kihm's taxonomy. In his model, the variety of forms also depends, for example, on the value of the vowel inserted between the 1st and 2nd root letters of the BP (p. 82).

2.4. Analysers and generators of Arabic inflected words

Because of the rich morphology of Arabic, NLP for this language requires dictionaries: 'we need to be able to relate irregular forms to their lexemes, and this can only be done with a lexicon' (Altantawy *et al.*, 2010:851). This need also applies to the statistic methods which are widely expoited almost without dictionaries for other inflectional languages: 'the need for incorporating linguistic knowledge is a major challenge in Arabic data-driven MT. Recent attempts to build data-driven systems to translate from and to Arabic have demonstrated that the complexity of word and syntatic structure in this language prompts the need for integrating some linguistic knowledge' (Zbib, Soudi, 2012:2).

Still, no comprehensive dictionaries equipped with easy procedures of updating are currently available for Arabic NLP. In the last 20 years, a number of computer systems for the morphological analysis and generation of Arabic words have been implemented. They can be classified into two approaches.

- The root/pattern/rule approach is based on traditional morphology. During analysis, a stem is analysed into a deep root and a deep pattern which are looked up among the roots and patterns stored in the system. The distance between deep level and surface level is covered with the aid of rules. This approach has a variant where patterns are closer to the surface, reducing the distance and simplifying the rules.

- The multi-stem approach seeks to avoid heavy computation during analysis. A stem is looked up among the stems stored in a dictionary. The term 'multi-stem' alludes to the fact that a lexical entry for a BP noun or a verb has at least two stems, e.g. *miEowal* 'mattock' معول and *maEaAwil* 'mattocks'. This approach has a variant in which the stems are generated from roots and patterns during a dictionary compilation phase.

2.4.1. Beesley (1996, 2001)

This system for Arabic inflection formalizes the traditional version of the root-and-pattern model and classifies in the root/pattern/rule approach. Its rules deal with root alternations, morphophonological alternations and spelling adjustments. They are encoded in the form of finite automata and compiled with the dictionary into a

finite transducer. For morphological analysis, these rules are applied regressively, i.e. they take surface forms as input and they output deep forms.

The system has a medium lexical coverage: 4 930 roots producing 90 000 stems [9] (Beesley, 2001:7), and it includes BPs. The lexical data originate from work at ALPNET (Buckwalter, 1990).

This system faces several challenges. One of them is that of analysis speed: 'the finite-state transducers (FSTs) tend to become extremely large, causing a significant deterioration in response time' (Altantawy *et al.*, 2011:116). This was, by the way, the main motivation for devising the multi-stem approach.

A second problem is the complexity of the rules that produce surface forms from underlying forms. The deep roots are borrowed from traditional morphology. For example, *baAeiE* 'seller' بائع, with surface root *beE*, and *baAEap* 'sellers' باعة (cf example 79 below), with surface root *bAE*, are analysed with deep root *byE*, which requires that the rules change *y* into *e* in the singular and into *A* in the plural. Each difference between surface forms and deep forms increases the complexity of the rule system. This complexity does not bring about any identifiable benefit. Once the roots are output by the analyser, they are to be essentially used as morpheme labels: the deep root borrowed from traditional morphology is not better for that than, say, the surface root of the singular. This additional complexity is inherited from traditional morphology, where it is meant to contribute to the semantic indexing of dictionaries, and to the consistency between inflection and derivation (Section 2.2 above). A morphological analyser of Arabic does not need to take into account these constraints: semantic indexing has no relation with morphological analysis; nobody finds it necessary to integrate inflection and derivation, for example, in English, in spite of obvious regularities between derivational suffixes and inflectional properties. "Dictionary maintenance need not require a thorough knowledge of Arabic derivational morphology, which few native speakers learn" (Buckwalter, 2007:37). And the useless complexity induced by the deepness of the underlying level has a cost: the rules are encoded and updated manually, 'a tedious task that often influences the linguist to simplify the rules by postulating a rather surfacy lexical level' (Beesley, 1996:91).

A third problem with this system is that the model lacks the notion of inflectional class. Two nouns belong to the same inflectional class if they inflect in the same way, and in particular if they pluralize in the same way. In lexicology for language processing, this notion allows for devising a common process shared by all the entries of a class, making the complexity depend on the number of classes (typically a few hundred) rather than on the number of lexical entries (in the dozens of thousands). Take for example root alternations: the surface root of *baAeiE* 'seller' بائع is *beE* in the singular and *bAE* in the BP, whereas for *HaAeir* 'indecisive' حائر,[10] it is *Her* in the singular and *Hyr* in the BP (cf example 78 below). Considering that there are no inflectional classes amounts to considering that both entries pluralize in the same way. This imposes to design and implement a **single** set of rules that outputs the correct alternation for both — and for **all** entries of all classes, in addition to the fact that for each entry, it should produce both the correct singular and the correct BP. In practice, this is a real challenge: 'Not surprisingly, to anyone who has studied Arabic, the rules controlling the realization of **w**, **y** and the hamza (the glottal stop)[11] are particularly complicated' (Beesley, 2001:5). Checking, correcting and updating such a set of rules are also heavy tasks: a typical rule affects several kinds of lexical entries, and there is no index of the entries or classes affected by each rule, or of the rules affecting each entry or class; the order of application of the rules is significant and must be decided and encoded. A separate, simpler set of rules for each class is more convenient to handle, even if at the cost of some redundancy between classes.

The solution adopted to specify BP patterns is diametrically opposed to the one for root alternations: patterns are manually specified separately for each root (Beesley, 2001:7), without sharing information at the level of inflectional classes.

[9] It is not measured as a number of entries because the formal model of the system does not include the notion of lexical entry.
[10] The BP system is essentially the same for nouns and for adjectives, except that BP is stylistically preferred for nouns, and suffixal plural for adjectives. We will exemplify some facts with adjectives.
[11] The consonants *w*, *y* and [ʔ] mentioned here are precisely those involved in root alternations.

The final shortcoming of this system is the format of the output of analysis, at the 'abstract lexical' level. It identifies the POS, root and pattern of the analysed words and their inflectional features, but not their lexical entries. Lexical entries of words are used to store, for example, their syntactic and semantic features, or, in the case of multilingual systems, an index to a lexical entry in another language. For example, *EawaAeil* 'families' is analysed by the system as a noun with root *Ewl* and pattern *FaEaAeiL*, and *maEaAwil* 'mattocks' as a noun with the same root and pattern *maFaAEiL*, but this is insufficient to identify lexical entries for them: since both words share the same root *Ewl*, nothing specifies whether one of them is the plural of *EaAeilap* 'family' or of *miEowal* 'mattock'. This is a difference with traditional dictionaries, which have a level for lexical entries in addition to the level for roots.

2.4.2. MAGEAD

The MAGEAD system (Habash, Rambow, 2006; Altantawy *et al*., 2010, 2011) is close to Beesley's (2001) in its design: 'We use "deep" morphemes throughout, i.e., our system includes both a model of roots, patterns, and morphophonemic/orthographic rules, and a complete functional account of morphology' (Altantawy *et al*., 2010:851); the rules are also compiled with the lexicon into a finite transducer. The lexicon is derived from Buckwalter's (Habash, Rambow, 2006:686; Altantawy *et al.*, 2010:853) through Smrž's (2007). The project has an on-going part for nouns, including BPs (Altantawy *et al*., 2010).

MAGEAD improves upon Beesley (2001) in several ways. The notion of lexical entry is represented. The output of morphological analysis of a noun comprises sufficient information to identify a lexical entry in the same way for the singular and the plural (Altantawy *et al.*, 2010:853): for *mawaAziyn* 'balances', the lexical entry of the noun is identified by the root *wzn* and the '*noun-I-M-mi12A3-ma1A2iy3*' codes, which specify the part-of-speech, the non-human feature, the gender and the compatibility with patterns. This makes the results of morphological analysis more easily usable in other tools. The notion of inflectional class is adopted for patterns, but not for root alternations (Habash, Rambow, 2006:683): each lexical entry is assigned a code that identifies the patterns it admits, e.g. '*mi12A3-ma1A2iy3*' (Altantawy *et al*., 2010:853). There are 41 classes for verbs (Habash, Rambow, 2006:684). Thus, inflectional information is shared at class level, reducing redundancy between entries. This facilitates dictionary checking, update and extension, reducing the cost of management of the dictionary: when an error is detected in the patterns of a class, the correction of the error affects all the class; when a new class is found and encoded, it can be shared by all the future members of the class through a simple code assignment.

However, MAGEAD still faces the other problems that we mentioned above about Beesley (2001).

- The resources of MAGEAD-Express compile in 48 h, and the analysis of a verb takes 6.8 ms (Altantawy *et al.*, 2011:123).

- The analysis opts for deep roots, complexifying the computation of the root from the surface form.

- Root alternations are not taken into account in inflectional classes, but controlled by a single set of rules for all entries. Encoding such rules is a challenge: 'we also exclude all analyses involving non-triliteral roots and non-templatic word stems since we do not even attempt to handle them in the current version of our rules' (Altantawy *et al.*, 2010:856).

In addition, the lexical coverage is still limited. The lexical data are borrowed from Buckwalter (2002): 8 960 verbs (Altantawy *et al.*, 2011:122) and 32 000 nouns, including those with suffixal plural (Altantawy *et al.*, 2010:854), but the rules are compatible only with triliteral nouns: 'we are not evaluating our lexicon coverage (…) Our evaluation aims at measuring performance on words which are in our lexicon, not the lexicon itself. Future work will address the crucial issue of creating and evaluating a comprehensive lexicon' (Altantawy *et al.*, 2010:856).

2.4.3. Systems with root alternations encoded in patterns

The Elixir system (Smrž, 2007) has a medium lexical coverage and includes BP. The lexical data are adapted from Buckwalter (2002). It is slow, but could be quicker if implemented in another language than Haskell. The results include a representation of lexical entries, as in MAGEAD.

Elixir follows the root/pattern/rules approach, but, as compared to the systems described above, patterns are closer to the surface level. In case of root alternation, surface forms of root letters are specified in patterns. For example, *baAEap* 'sellers' is analysed with root *byE* and pattern *FaaLap*, whereas traditional morphology taught at school analyses it with root *byE* and deep pattern *FaEoLap*, with *A* as the surface realisation of the second deep root letter *y*. Traditional morphology represents patterns with a fixed number of slots, even in case of root alternations. Elixir's option of encoding root alternations in patterns is shared by Ryding (2005:149): *FaaLap, FuEaap*…(*فَعَالة, فُعَالة*...) This option simplifies the rules and their application, but introduces numerous new patterns, which look odd to Arabic speakers because traditional inflectional taxonomy is entirely based on deep patterns. This difference makes some of the Elixir patterns difficult to read and handle. In NLP companies, management of Arabic language resources tends to involve native Arabic speakers, because of their wider knowledge of the language.

The open-source Alkhalil morphological analyser [12] (Boudlal *et al*., 2010) is used in various projects and won the first prize at a competition by the Arab League Educational, Cultural Scientific Organization (ALESCO) in 2010. We counted that Alkhalil's lexical resources cover 97% of the verb occurrences of a sample text, which is comparable to the coverage of Buckwalter (2002). The system includes BP. The patterns are scripted in Arabic. As in Beesley (2001), the output of the analyser does not identify lexical entries: nothing connects a noun in the BP to its singular. The general approach is close to that of Elixir, patterns are used in the same way, and the example of *baAEap* 'sellers' gets the same analysis.

Another difference with traditional morphology is that Alkhalil includes case and definiteness suffixes in the patterns. For example, in the noun *daAra* دَارَ 'home', Alkhalil assigns final *-a* to the pattern *FaaLa* فَالَ, whereas for traditional morphology, the stem is *daAr*, with root *dwr* and deep pattern *FaEaL* فعل (with *A* as the surface realisation of the second root letter *w*), and *-a* is an inflectional suffix of the accusative case and the construct-state definiteness. Traditional morphology has a systematic delimitation between stem and such suffixes; these suffixes have very little variation depending on lexical entries; most analysers comply with this distinction and exclude the suffixes from the pattern. The Alkhalil option introduces numerous such new patterns which are alien to familiar pattern taxonomy.

2.4.4. The multi-stem approach

Buckwalter's (2002) open source morphological analyser of Arabic, BAMA, is a well-known example of the multi-stem approach. It is slow, but could be quicker if implemented in another language than Perl. It has a medium lexical coverage: approximately 32 000 nouns and 9 000 verbs. The lexical data originate probably from work at ALPNET, as can be seen by the common morpheme labels (Buckwalter, 1990:3-5). All stems are stored in the resources, including most spelling variants, bypassing almost all morphophonological rules. This option simplifies dramatically the lookup algorithm. 'The BAMA uses a concatenative lexicon-driven approach where morphotactics and orthographic adjustment rules are partially applied into the lexicon itself instead of being specified in terms of general rules that interact to realize the output' (Buckwalter, 2002). Thus, 9 stems are stored for the verb *qara>a* 'read' قرأ (in Buckwalter transliteration), due to the orthographic variants of the 3rd root letter, here [ʔ], determined by the presence of an inflectional suffix or of an agglutinated pronoun. The form *qora>* appears in 3 items, with different compatibility codes:

[12] http://sourceforge.net/projects/alkhalil/

Stem	Compatibility code	Stem	Compatibility code
qara>	PV->	*qora|*	IV-|
qara|	PV-|	*qora&*	IV_wn
qara&	PV_w	*qora}*	IV_yn
qora>	IV	*qora>*	IV_Pass
qora>	IV_wn		

The information provided in morpheme labels includes the part of speech, the voice and aspect of verbs, and other relevant information.

Independent work by Soudi *et al*. (2002) shares the same design: 'Such an approach dispenses with truncating/deleting rules and other complex rules that are required to account for the highly allomorphic broken plural system' (Soudi *et al*., 2002). The main difference is that in case of purely orthographic variations, variants of stems are not stored in the lexicon, but the paper does not explain how they are recognised.

To date, the systems implementing the multi-stem approach have several common shortcomings. The multi-stem model lacks the notion of inflectional class: stems are manually specified separately for each root. For example, if a verb conjugates like *qara>a*, its 9 stems are listed independently of those of *qara>a*, without sharing information at the level of inflectional classes.

In addition, for a BP noun without root alternations, such as *EaAeilap* 'family' عائلة, *EawaAeil* 'families' عوائل , the stems stored in the lexicon include redundancy. The same root appears in each stem. Duplicated manual encoding of the same piece of information leads to errors. This flaw is connected to the preceding: multi-stem systems do not encode regularities.

Both have practical consequences. Human operations required to encode, check, correct and update the dictionaries are unnecessarily repetitive and costly. Fallback procedures for words not found in the dictionary are difficult to devise.

2.4.5. Neme (2011)

Neme (2011) describes a morphological analyser for Arabic verbs with a comprehensive lexical coverage: 15 400 verbs. The dictionary compiles in 2 mn and the analysis of a verb takes 0.5 ms on a 2009 Windows laptop,[13] outperforming MAGEAD-Express (cf. Section 2.4.2).

This system shows a concern with the comfort and efficiency of human encoding, checking and update of dictionaries. NLP companies need easy procedures for dictionary management, because most projects involve a specific domain with a particular vocabulary, and terminology evolves constantly; in addition, dialects show lexical differences, which are relevant to speech processing if not for written text processing; finally, the main advantage of dictionary-based analysers is that they provide a way of controlling the evolution of their accuracy by updating the dictionaries. None of the other authors surveyed above mentions the objective of facilitating manual dictionary management, and we reported the weak points of their analysers in this regard. Neme (2011) identifies the problem as belonging not only to computation and morphology, but also to NLP dictionary management, and considers language resources as the key point, as Huh & Laporte (2005). His dictionaries are constructed and managed with the dictionary tools of the open-source Unitex system (Paumier, 2002).

All forms are stored in the resources, including spelling variants; roots and patterns are handled at surface level. The main difference with previous multi-stem systems is that the full-form dictionary is automatically precompiled from another dictionary, which is specifically dedicated to manual construction, check and update. The dictionary is compiled by finite transducers that combine roots, patterns and inflectional suffixes. Each of the 480 inflectional classes is assigned one of the transducers, which ensures that the management of classes is mutually independent. The encoding of a new verb amounts to assigning it an inflectional code. Thus, the redundancy problems of the mainstream multi-stem approach are solved.

[13] Memory: 16 GB DDR3 1600 MHz; hard disks: 750 GB (7 200 rpm, Hybrid 4 GB Serial ATA) and 1TB (5 400 rpm, Serial ATA).

Pattern taxonomy is kept simple and close to that taught in school to Arabic speakers, by maintaining it separate from the encoding of root alternations and of tense, person, gender and number suffixes. This keeps codes readable and facilitates the encoding, improving upon the pattern labels of Smrž (2007) and Boudlal *et al*. (2010).

Such technology reduces the computational skills required for the linguistic part of dictionary management: these skills shift from software development to software use. Such a shift opens the perspective that Arabic language resources can be managed directly by native Arabic linguists. In current practice, management of resources typically requires a high-wage specialist of computation and an Arabic informant: a configuration which is more costly and inserts an intermediary between the source of linguistic knowledge and the formalization.

The results with verbs incited us to undertake the encoding of the BP system on the same bases. We called our project Pattern-and-Root Inflectional Morphology (PRIM), inverting the traditional 'root-and-pattern' phrase, because we capitalized on traditions about patterns, rather than about roots, to make our taxonomy intuitive to Arabic speakers.

3. General organization of PRIM

We decided to take advantage of the validation of traditional morphology over centuries, and we took it as a basis for our computerized model of BPs, formalizing and simplifying it. We gave priority to this objective of simplification in order to make easier and more comfortable the manual part of the encoding of Arabic dictionaries. Consistency with semantic features or derivational analyses was only a secondary objective. The most successful projects of morpho-syntactic codification are usually those that focus, in practice, on manual descriptors' ease and comfort. They produce long-lasting morpho-syntactic dictionaries which are actually updated over time by linguists, as has been the case of the Dela dictionaries since the 1980s (Courtois, 1990; Daille *et al*., 2002).

3.1. Inflectional codes

Arabic grammarians usually display the analysis of a singular stem/BP stem pair, e.g. *Euqodap* 'knot'/*Euqad* 'knots', in the form of a compact formula:

(a) `Eqd FuEoLap FuEaL`

where *Eqd* is the deep root, *FuEoLap* the singular pattern and *FuEaL* the BP pattern. By combining *Eqd* with *FuEoLap* and applying morpho-phonological and orthographical rules, one obtains the singular stem. The same operation with *Eqd* and *FuEaL* yields the BP stem.

Pattern pairs such as *FuEoLap*/*FuEaL* make up a taxonomy of BP noun entries, by crossing the two taxonomies based, respectively, on singular patterns and BP patterns. A given singular pattern is compatible with several BP patterns, but not with all, and vice-versa.

The PRIM format of a lexical entry is similar to (a), with the lemma in Arabic script and the codes in the Latin alphabet:

(b) `Euqodap, $N3ap-f-FvEvL-FuEaL-123`

In this entry, *Euqodap* is the lemma of the noun, which is the singular of the noun, stripped off of its case and definiteness suffix, and written in fully diacritized script. The remainder is the inflectional code provided by the dictionary. In this code, *FvEvL* and *FuEaL* are the PRIM counterparts of the two patterns *FuEoLap* and *FuEaL* in (a), and the root code *123* is comparable to the deep root *Eqd* in (a). Our encoding of nominal entries is also similar to that of verbal entries (Neme, 2011), with two patterns and a root code:

```
Euqodap,  $N3ap-f-FvEvL-FuEaL-123            / knot
kaAotib,  $N300-g-FvvEvL-FuEEaaL-123         / author, employee
kaAotib,  $N300-g-FvvEvL-FaEaLap-123         / employee
katiyobap,$N3ap-f-FvEvvL-FaEaaLiB-12h3       / brigade of soldiers
```

```
kitaAob,   $N300-m-FvEvvL-FuEuL-123          / book
ktb,       $V3-FaEaLa-yaFoEuLu-123           / write
Inktb,     $V3-IinoFaEaLa-yanoFaEiLu-123     / be written
tkAtb,     $V3-taFaaEaLa-yataFaaEaLu-123     / write each other

عِقْدة,      $N3ap-f-FvEvL-FiEaL-123           / knot
كَـأتِب,     $N300-g-FvvEvL-FuEEaaL-123        / author, employee
كَـأتِب,     $N300-g-FvvEvL-FaEaLap-123        / employee
كَـتِيْبَة,  $N3ap-f-FvEvvL-FaEaaLiB-12h3      / brigade of soldiers
كِـتَـأب,    $N300-m-FvEvvL-FuEuL-123          / book
كـتب,        $V3-FaEaLa-yaFoEuLu-123           / write
إنـكـتب,     $V3-IinoFaEaLa-yanoFaEiLu-123     / be written
تـكـاتـب,    $V3-taFaaEaLa-yataFaaEaLu-123     / write each other
```

In verbal entries, the two patterns are for the perfect and the imperfect. Verb lemmas are encoded without diacritics; the diacritics are specified in the perfect pattern.

3.2. Special plurals

As a simplification, our model does not take into account the traditional marking of a few BP forms as 'plurals of paucity'. Sibawayh (VIIIth century) states that in an older stage of Arabic, plural of paucity had been restricted to collections of 3 to 10 entities, and other plural forms to collections of more than 10; however, at his time, both constraints were commonly overlooked, and many nouns lacked a plural of paucity (Ferrando, 2002:5). Native speakers accept a 'non-paucity' BP after cardinal numbers from 3 to 10, even when the noun also has a plural of paucity:

عليه أن يختار ثلاثة (أمكنة + أماكن)
Ealayhi Ono yaxotaAr valaAvapu (Oamokinapin + OamaAkina)
on him to choose three (places+pauc + places)
'He must choose three places'

عليه أن يختار أربعَ (أيدي + أيادي)
Ealayhi Ono yaxotaAr OarobaEa (Oayodin + OayaAdi)
on him to choose four (hands+pauc + hands)
'He must choose four hands'

In addition, the delimitation of plural of paucity is fuzzy. Four BP patterns are associated to plurals of paucity, but they also generate non-paucity BPs. Grammars give examples of plurals of paucity, but never exhaustive inventories.

We do not mark 'plurals of plurals' either. Plurals of plurals in TM, as *OamaAkin* 'places', are supposedly obtained by morphologically pluralizing a BP, here *Oamokinap* 'places', which is re-pluralized on the same model as *zawobaEap* 'tornado'/*zawaAbiE* 'tornados'. In our model, *OamaAkin* 'places' is directly related to the singular *makaAn* 'place'.

As a rule, the PRIM taxonomy gives only one plural of a given lexical entry: when several plurals are observed, they are assigned to distinct entries, no matter whether they are equivalent or not, as in examples (86), (97) and (119). Neme (2011:7) discusses the same problem for verbs. When several entries generate identical singular forms, the Unitex system removes duplicates.

3.3. Interpretation of codes

The main 3 codes in a PRIM entry for a BP noun, as *FvEvL-FiEaL-123* in (b), correspond to 3 independent taxonomies which, crossed together, are sufficient to identify the generation of a broken plural.

The linguistic interpretation of these codes correspond to three conceptual steps in generating a BP from a lemma such as *Euqodap* 'knot': extract the surface root of the lemma, here *Eqd*; find out the surface root for the BP, which is unchanged here; and combine it with the BP pattern, which gives *Euqad* 'knots'.

The first step matches the singular-pattern code, here *FvEvL*, with *Euqodap*, to obtain *Eqd*:

Stem	Euqodap ‘knot’	qabow ‘cave’	قَبْو
Singular-pattern code	FvEvL	FvEvL	
Surface root of singular	E q d	q b w	

The second step applies root alternations [14] encoded in the root code, if any, as is the case with *12y*, the root code of *qabow* ‘cave’:

Surface root of singular	Eqd	qbw
Root code	123	12y
Surface root of BP	Eqd	qby

The third step combines the surface root with the BP pattern:

Surface root of BP	E q d	q b y	
BP pattern	FuEaL	OaFoEiLap	
BP stem	Euqad ‘knots’	Oaqobiyap ‘caves’	أقبية

Lemmas with a geminated consonant are a little more complex. In Arabic script, the *G* diacritic notes the gemination of the preceding consonant. For example, *MidGap* ‘trouble’ is to be read as if it were spelt **Midodap*. The silent diacritic *o*, which marks the absence of vowel (cf. Section 2), is not used when *G* is used. In this word, the singular-pattern code *FvEvL* implies that the geminated consonant corresponds to two slots in the root. The gemination is assigned to the root:

	gloss	sg. stem	PRIM codes	sg. root	in Arabic
1	trouble	MidGap	FvEvL-FaEaaLiB-12h2	Mdd	شدّة شدائد
2	luck	HaZG	FvEvL-FuEuuL-122	HZZ	حظّ حظوظ

In *sulGam* ‘ladder’, the geminated consonant corresponds to a single slot in the root, which is represented by a repeated letter in *FvEEvL*. The gemination is assigned to the pattern:

3	ladder	sulGam	FvEEvL-FaEaaLiB-1223	slm	سُلّم سَلالِم

The choice between the two analyses is determined by observing other forms and specified in the singular-pattern code.

In the Unitex implementation of PRIM, the three conceptual steps described above are performed simultaneously by inflectional transducers, as in Silberztein (1998). For example, in the transducer for inflectional class *N3ow-m-FvEvL-OaFoEiLap-12y*, which is the class of *qabow* ‘cave’, they are performed by formula *Oa1o2iy*, where *1* and *2* refer to the positions of root letters in the lemma, *y* is the value of the other root letter in the plural, and the remaining symbols correspond to the BP pattern; the *-ap* suffix in the pattern is specified in another part of the transducer, because it undergoes spelling variations in the presence of a clitic pronoun.

3.4. Encoding nouns

Encoding a noun consists of writing the stem of its lemma in fully diacritized form, and assigning it a code as in (b) (with the lemma in Arabic script), so as to generate the correct forms of the plural:[15]

(b) Euqodap, $N3ap-f-FvEvL-FuEaL-123

It is important that the stem is fully diacritized, since digits in inflectional transducers refer to the position of root letters. Each basic letter, except the last of the stem, is followed by a single diacritic, which is either a short vowel: *a*, *u*, *i*, or the void diacritic *o*. Thus, all root letters correspond to odd positions. The only exceptions are after a geminate consonant, which is transcribed as in example (3): the 3rd root letter, *m*, is in position 6.

[14] We term as ‘root alternations’ any changes in the surface value of root letters, as in *qabow* ‘cave’ قَبْو and *Oaqobiyap* ‘caves’ أقبية, or in the number of root letters, as in *TaAbiE* ‘stamp’ طابع and *TawaAbiE* ‘stamps’ طوابع (cf. Section 5).

[15] Computer aiding could be devised to assist encoders, but might have perverse effects, e.g. inciting them to systematically accept suggestions, even if they are inconsistent with previously encoded entries.

The choice of a code is not a deterministic process, because analysis in root and pattern is in general not deterministic (cf. (1)-(3) above, and Section 4.1). Traditional morphology provides rules for reducing indeterminacy. Our taxonomy complies with rules which are widely known by Arabic speakers: for example, triliteral roots take precedence over biliteral roots. However, we disregard rules that depend on scholarly or diachronic knowledge, when this reduces the number of classes or simplifies the task of assigning a class to a lexical item.

4. Conflation of patterns

In order to make the PRIM taxonomy of BPs simpler than the traditional one, we merged classes by conflating patterns without loss of information. We illustrate this in the following examples.

4.1. Singular patterns

The PRIM models substitutes singular-pattern codes, e.g. *FvEvL*, to the traditionally used singular patterns, e.g. *FiEoLap*. The PRIM singular-pattern codes are less numerous than singular patterns because they dispense with unnecessary information. Their only purpose is to be matched with lemmas, e.g. *Euqodap*, to obtain their surface roots, here *Eqd*:

Stem	`Euqodap` 'knot'
Singular-pattern code	`FvEvL`
Surface root of singular	`E q d`

The singular-pattern code cannot be dispensed of completely. Some nouns have more than three root consonants: the singular-pattern code *FvEvLvB*, matched with *diroham* 'dirham' درهم , extracts the root *drhm*. The difference between the two surface forms *Euqodap* and *diroham* would not be easy to tell without these codes.

Similarly, some noun lemmas have a long vowel, which is assigned either to the root or to the pattern. In *Miyomap* 'honour' شِيْمَة, the *iy* sequence [16] notes the long vowel [i:]; the *FvEvL* code implies that the root is *Mym*. The root letter *y* is realised as a long vowel. In contrast, in *sabiyol* 'road' سَبِيل , the *FvEvvL* code points to the root *sbl*. The long vowel belongs to the pattern.

Thus, simplified singular patterns such as *FvEvL*, *FvEvLvB*, *FvEEvL* or *FvEvvL* specify the number of root letters, the position of pattern-assigned long vowels, and the position of pattern-assigned geminations of root letters. They are sufficient to deduce the singular root.

Representing *o*, the silent diacritic, by *v*, a symbol for a short vowel, might seem paradoxical, but it is natural to Arabic speakers.

4.1.1. Omission of vowel quality

The quality of the vowels is not specified because it is not necessary. This reduces the number of classes in the singular-pattern taxonomy, without loss of generative power. In the following examples, 6 singular patterns distinguished by TM are conflated into a single code in the PRIM model:

[16] In Arabic script, the letters *y* and *w* code the semivowels [j w] or the long vowels [i: u:], depending on context. When *y* is preceded by *a* or *u*, it codes [j]; when *w* is preceded by *a* or *i*, it codes [w]. The long vowels [i: u:] are coded *iy* and *uw*. This system codes alternations between [i: u:] and [j w]. The silent diacritic *o*, which notes the absence of vowel between two basic letters (cf. Section 2), is usually omitted after long vowels (*iy*, *uw*, *aA*), even when writers intend to fully diacritize their text. However, the PRIM model requires that it be present in lemmas, so that the convention given in Section 3.4 is respected, and roots with semivowels do not require separate classes. For the sake of consistency, from here on, this diacritic will be explicitly scripted in our examples.

	gloss	sing.	plural	TM patterns	PRIM codes	Arabic
4	spirit	nafos	nufuwos	FaEoL-FuEuuL	FvEvL-FuEuuL-123	نفس نفوس
5	luck	HaZG	HuZuwoZ	FaEo**L**-FuEuuL	FvEv**L**-FuEuuL-122	حظّ حظوظ
6	stem	jiJoE	juJuwoE	FiEoL-FuEuuL	FvEvL-FuEuuL-123	جَذع جذوع
7	load	Humol	Humuwolap	FuEoL-FuEuuLap	FvEvL-FuEuuLap-123	حُمْل حمولة
8	mountain	jabal	jibaAol	FaEaL-FiEaaL	FvEvL-FiEaaL-123	جَبل جبال
9	shoulder	katif	OakotaAof	FaEiL-OaFoEaaL	FvEvL-OaFoEaaL-123	كتف أكتاف
10	man	rajul	rijaAol	FaEuL-FiEaaL	FvEvL-FiEaaL-123	رجل رجال

When triliteral nouns have a long vowel in the singular pattern, it may occur in any of the two positions between root letters:

	gloss	sing.	plural	TM patterns	PRIM codes	Arabic
11	friend	SaAoHib	OaSoHaAob	FaaEiL-OaFoEaaL	FvvEvL-OaFoEaaL-123	صاحب أصحاب
12	film	fiyolom	OafolaAom	N/A -OaFoEaaL	FvvEvL-OaFoEaaL-123	فيلم أفلام
13	book	kitaAob	kutub	FiEaaL-FuEuL	FvEvvL-FuEuL-123	كتاب كتب
14	messenger	rasuwol	rusul	FaEuuL-FuEuL	FvEvvL-FuEuL-123	رسول رسل
15	road	sabiyol	subul	FaEiiL-FuEuL	FvEvvL-FuEuL-123	سبيل سبل

The Arabic word for ‘film’ (12) is a loan world, so the pattern of the singular is anomalous and not listed in TM. The 5 cases are conflated to 2 singular-pattern codes.

In quadriliteral nouns, a long vowel may occur after the third root letter of the singular, or sometimes after the second:

16	statue	timovaAol	tamaAoviyol	FaEoLaaB-FaEaaLiiB	FvEvLvvB-FaEaaLiiB-1234	تمثال تماثيل
17	bird	EaSofuwor	EaSaAofiyor	FaEoLuuB-FaEaaLiiB	FvEvLvvB-FaEaaLiiB-1234	عصفور عصافير
18	light	qanodiyol	qanaAodiyol	FaEoLiiB-FaEaaLiiB	FvEvLvvB-FaEaaLiiB-1234	قنديل قناديل
19	bishop	muToraAon	mataAorinap	FuEoL**aa**B-FaEaaLiBap	FvEvLvvB-FaEaaLiBap-1234	مطران مطارنة
20	sample	namuwoJaj	namaAoJij	FaEuuLaB-FaEaaLiB	FvEvvLvB-FaEaaLiB-1234	نموذج نماذج

4.1.2. Omission of suffixes

Some singular nouns have a suffix which disappears in the plural. Traditional morphology includes this singular suffix in the singular pattern:

21	knot	Euqod**ap**	Euqad	FuEoL**ap**-FuEaL	FvEvL-FuEaL-123	عقدة عِقد
22	bomb	qunobul**ap**	qanaAobil	FuEoLuB**ap**-FaEaaLib	FvEvLvB-FaEaaLiB-1234	قنبلة قنابل
23	school	madoras**ap**	madaAoris	maFoEaL**ap**-maFaaEiL	FvEvLvB-FaEaaLiB-1234	مدرسة مدارس
24	whore	MaromuwoT**ap**	MaraAomiyoT	FaEoLuuB**ap**-FaEaaLiib	FvEvLvvB-FaEaaLiB-1234	شرموطة شراميط

Such information is unnecessary for producing the broken plural, since the suffix is absent from it. Our model does not specify the suffix in the singular-pattern code, which is generally conflated with a code for nouns without suffix in the singular. This simplification of the BP taxonomy affects many lexical items. The suffix *-ap* is generally the singular suffix for feminine forms (21-24).

The suffix *-iyG* and its feminine counterpart *-iyGap* are typical singular suffixes for human nouns (and adjectives) derived from nouns. Most of such nouns and adjectives pluralize with a sound plural suffix such as *-uwona* or *-aAoT*, but others take a BP:

25	soldier	junod**iyG**	junuwod	FuEoL**iyy**-FuEuuL	FvEvL-FuEuuL-123	جنديّ جنود
26	copt	quboT**iyG**	OaqobaAoT	FuEoL**iyy**-OaFoEaaL	FvEvL-OaFoEaaL-123	قبطيّ أقباط
27	foreigner	Oajonab**iyG**	OajaAonib	FaEoLaB**iyy**-FaEaaLiB	FvEvLvB-FaEaaLiB-1234	أجنبيّ أجانب
28	barbar	barobar**iyG**	baraAobirap	FaEoLaB**iyy**-FaEaaLiBap	FvEvLvB-FaEaaLiBap-1234	بربريّ برابرة
29	zionist	Sahoyuwon**iyG**	SahaAoyinap	FaEoLuuB**iyy**-FaEaaLiBap	FvEvLvvB-FaEaaLiBap-1234	صهيونيّ صهاينة

The following non-derived nouns illustrate the same situation:

30	rifle	bunoduq**iyGap**	banaAodiq	FuEoLuB**iyyap**-FaEaaLiB	FvEvLvB-FaEaaLiB-1234	بُنْدُقِيَّة بَنادِق
31	turtle	suloHaf**aAop**	salaAoHif	FuEoLaB**aap**-FaEaaLiB	FvEvLvB-FaEaaLiB-1234	سُلَحْفاة سَلاحِف

4.2. Broken-plural patterns

Most BP patterns in the PRIM taxonomy are the same as in traditional morphology. However, a few differences come from our choice to handle patterns and roots at the surface level.

The BP of *miqaSG* ‘scissors’ has two occurrences of the same consonant separated by *i*:

	gloss	sing.	plural	TM patterns	PRIM codes	Arabic
32	scissors	miqaSG	maqaAoSiS	miFaEoL-maFaaEiL	FvEvLvB-FaEaaLiB-1233	مَقصّ مَقـاصص

Some nouns behave in the same way, except that the two occurrences of the consonant are optionally (33-34) or obligatorily (35-36) replaced by a geminated consonant:

33	porcupine	lutunGap	lataAonin	FuEuLLap-FaEaaLiL	FvEvL-FaEaaLiB-1233	لـتـنّة لـتـانـن
34	porcupine	lutunGap	lataAonG	FuEuLLap-FaEaaLiL	FvEvL-FaEaaLiB-123G	لـتـنّة لـتـانْ
35	mission	muhimGap	mahaAomG	muFoEiLap-maFaaEiL	FvEvLvB-FaEaaLiB-123G	مـهـمّة مَـهـامْ
36	substance	maAod**G**ap	mawaAodG	FaaEiLap-FawaaEiL	FvvEvL-FaEaaLiB-1w22	مـادّة مـوادْ

Traditional morphology views these forms as the result of the application of a rule that erases *i* between two occurrences of the same consonant: “The plural *mawaadd* is the form that the plural pattern *fawaaEil* takes in geminate nouns because of the phonological restriction on sequences that include a vowel between identical consonants: **mawaadid* –> *mawaadd*. It is diptote (CaCaaCiC pattern)” (Ryding, 2005:471). In fact, the conditions of application of the rule are also lexical: it does not apply in (e), while it applies optionally in (33-34) and obligatorily in (35-36). Therefore, we account for this morphophonological variation through inflectional classes.

In the BP of (34)-(35), the surface pattern actually handled by the PRIM transducers, *FaEaaLoB*, differs from the traditional deep patterns which contain *i*. In this case, our option for surface patterns tends to increase the number of distinct patterns, and to separate (33)-(35) from (32) in the pattern taxonomy. In order to avoid this effect, we included the deep pattern **label** *FaEaaLiB* in the PRIM inflectional codes. Thus, they sound more familiar to Arabic speakers, because they comply with the deep patterns of traditional morphology taught in school. The forbidden, optional or obligatory geminated consonant is encoded by the respective root codes *1233*, *123G* and *1w22*.

When the BP surface patterns differ from traditional deep patterns, because of morphophonological constraints or variations, the deep pattern **label** is used in the inflectional code, and the surface pattern in the transducer associated to it. Thus, the pattern labels used in inflectional codes are relatively intuitive.

In the following case, we use the same method to conflate BP patterns labels. Some triliteral lemmas have the suffix *-iy* appended to the root in the BP:

37	night	layolap	layaAol**iy**	FaEoLap-FaEaaLiy	FvEvL-FaEaaLiB-123y	لـيـلـة لـيـالـي

The following nouns are similar, except for a free variation between *-iy* and *-aY*, where *Y* is an allograph of final *A*:

38	desert	SaHoraAoc	SaHaAor**iy**	FaEoLaac-FaEaaLiB	FvEvL-FaEaaLiB-123y	صحراء صحاري
39	desert	SaHoraAoc	SaHaAor**aY**	FaEoLaac-FaEaaLaY	FvEvL-FaEaaLiB-123Y	صحراء صحارى
40	complaint	MakowaY	MakaAow**iy**	FaEoLaY-FaEaaLaY	FvEvL-FaEaaLiB-123y	شكوى شكاوي
41	complaint	MakowaY	MakaAow**aY**	FaEoLaY-FaEaaLaY	FvEvL-FaEaaLiB-123Y	شكوى شكاوى

The noun *EaJoraAoc* ‘virgin’ has obligatorily *-aY*:

42	virgin	EaJoraAoc	EaJaAor**aY**	FaEoLaac-FaEaaLaY	FvEvL-FaEaaLiB-123Y	عذراء عذارى

The BP surface pattern actually handled in the implementation details of the PRIM transducers for (39), (41) and (42) is *FaEaaLaB*. However, it is natural to Arabic speakers to consider it as a superficial allomorph of *FaEaaLiB*, which is a regular BP pattern: the fact that the sequence *iY* cannot occur in Arabic explains the surface forms in *aY*. We adopted the pattern label *FaEaaLiB* in the inflectional code, in order to reduce the number of pattern labels and to keep the encoding of these nouns intuitive. The quality of the long vowel in the suffix is encoded in the root code *123Y*.

The same situation occurs in the following examples, with the suffixes *-aAon* in the singular and *-aY* in the BP:

43	drunk	sakor**aAon**	sakaAor**aY**	FaEoLaan-FaEaaLaY	FvEvL-FaEaaLiB-123Y	سكران سكارى
44	Christian	naSor**aAoniyG**	naSaAor**aY**	FaEoLaaniyy-FaEaaLaY	FvEvL-FaEaaLiB-123Y	نـصـرانـيّ نـصـارى

and in the following BPs with the *-aA* ending:

45	corner	zaAowiyap	zawaAoyaA	FaaEiLap-FaEaayaA	FvvEvL-FaEaaLiB-12yA	زاوية زوايا
46	mirror	miroCp	maraAoyaA	miFoEaLap-maFaayaA	FvEvL-FaEaaLiB-12yA	مرآة مرايا
47	intention	niyGap	nawaAoyaA	FiEoLap-FaEaayaA	FvEvL-FaEaaLiB-1wyA	نيّة نوايا
48	feature	miyozap	mazaAoyaA	FiEoLap-FaEaayaA	FvEvL-FaEaaLiB-13yA	ميزة مزايا

The sequence *iA* cannot occur in Arabic, which explains the surface forms in *aA*. The quality of the long vowel in the suffix is encoded in the root codes. In example (46), the character *C* (آ) is an obligatory substitute for the sequence *OaAo*.

Example (37) poses a segmentation problem. Recall that TM, most analysers and PRIM exclude from the pattern the case and definiteness suffixes. PRIM appends these suffixes to the root/pattern combination during the generation of inflected forms (cf. Section 8.2). In general, these suffixes have little variation depending on lexical entries, and little interaction with the end of the root and pattern. In the case of (37) *layaAoliy* ‘nights’, the *iy* ending is removed in the indefinite nominative and genitive *layaAolK*. We consider the *iy* ending as a part of the pattern; this ending is removed when the case and definiteness suffixes are appended. Our segmentation is conforted by the fact that in other nouns, *iy* is actually part of the root, as in *qaAoDiy* ‘judge’ which declines as *qaAoDK* in the indefinite nominative. Our analysis deviates slightly from tradition and simplifies it. According to TM, *iy* is present in underlying forms **layaAoliyN* and **layaAoliyK*, which are both rewritten as the surface form *layaAolK*, and the ‘citation form’ used to refer to the word is *layaAolK*, a form without *iy*.

4.3. Simultaneous conflation of singular and broken-plural patterns

In the framework of traditional morphology, the analysis of broken plurals is systematically consistent with the roots traditionally used for the practical purpose of indexing dictionaries. For instance, the BP of the derived noun *miEowal* ‘mattock’ is analysed with the root of its derivational base, here *Ewl*. An inflectional phenomenon is thus analysed with a derivational concept. By imposing one of the pieces of the jigsaw (the root), this practice constrains all others, and happens to blur regularities in the system of inflectional patterns.

For the PRIM model, the objective of consistency with derivational analyses is only secondary to the simplicity of the taxonomy. By relaxing this constraint, we can capture more of the regularity of the inflectional system.

4.3.1. Nouns with *m-* prefixes

Many nouns have a *ma-*, *mu-* or *mi-* prefix before a triliteral root. Traditional morphology excludes these prefixes from the root, and consequently includes them in the pattern, on the basis of the derivational history of these words:

49	mattock	miEowal	maEaAowil	**mi**FoEaL-**ma**FaaEiL	FvEvLvB-FaEaaLiB-1234	معول معاول

The prefix is common to the singular and BP of the derived noun. If we analyse the initial *m-* as a part of a quadriliteral root, most of these nouns enter in independently existing inflectional classes. ‘Initial *m(i)-*, although originally a prefix, is annexed to the root and treated as a C1 as far as BP formation is concerned’ (Kihm, 2006:83). For PRIM, the 9 prefixed nouns below inflect exactly like (A) or (B):

A	**dagger**	**xanojar**	**xanaAojir**	**FaEoLaB-FaEaaLiB**	**FvEvLvB-FaEaaLiB-1234**	خنجر **خناجر**
50	theater	masoraH	masaAoriH	**ma**FoEaL-**ma**FaaEiL	FvEvLvB-FaEaaLiB-1234	مسرح مسارح
51	house	manozil	manaAozil	**ma**FoEiL-**ma**FaaEiL	FvEvLvB-FaEaaLiB-1234	منزل منازل
52	museum	mutoHaf	mataAoHif	**mu**FoEaL-**ma**FaaEiL	FvEvLvB-FaEaaLiB-1234	متحف متاحف
53	sieve	munoxul	manaAoxil	**mu**FoEuL-**ma**FaaEiL	FvEvLvB-FaEaaLiB-1234	منخل مناخل
54	pulpit	minobar	manaAobir	**mi**FoEaL-**ma**FaaEiL	FvEvLvB-FaEaaLiB-1234	منبر منابر
B	**cluster**	**Eunoquwod**	**EanaAoqiyod**	**FuEoLuuB-FaEaaLiiB**	**FvEvLvvB-FaEaaLiiB-1234**	عنقود عناقيد
55	letter	makotuwob	makaAotiyob	maFoEuuL-maFaaEiiL	FvEvLvvB-FaEaaLiiB-1234	مكتوب مكاتيب
56	gutter	mizoraAob	mazaAoriyob	miFoEaaL-maFaaEiiL	FvEvLvvB-FaEaaLiiB-1234	مزراب مزاريب
57	poor	misokiyon	masaAokiyon	miFoEiiL-maFaaEiiL	FvEvLvvB-FaEaaLiiB-1234	مسكين مساكين
58	napkin	minodiyol	manaAodiyol	miFoEiiL-maFaaEiiL	FvEvLvvB-FaEaaLiiB-1234	منديل مناديل

The only reasons to discriminate them are alien to inflectional morphology. In the traditional analysis, both the singular and BP patterns explicitly contain the prefix, which makes them specific to this set of nouns. Even if we strip the prefix off the patterns, we do not always obtain triliteral patterns observable in other BP nouns. Therefore, the traditional analysis increases the number of patterns. By implementing the alternative analysis, PRIM conflates simultaneously the singular pattern and the BP pattern with those of (A) or (B), which simplifies the taxonomy.

The following examples are less regular, but also follow independently observed patterns:

59	building	mabonaY	mabaAoniy	maFoEaL-maFaaEiL	FvEvLvB-FaEaaLiB-123y	مبنى مباني
60	school	madorasap	madaAoris	maFoEaLap-maFaaEiL	FvEvLvB-FaEaaLiB-1234	مدرسة مدارسة
61	tragedy	maOosaAop	maCaAosiy	maFoEaLap-maFaaEiL	FvEvLvB-FaEaaLiB-1h3y	مأساة مآسي
62	foreigner	OajonabiyG	OajaAonib	FaEoLaBiyy-FaEaaLiB	FvEvLvB-FaEaaLiB-1234	أجنبي أجانب
63	appointment	mawoEid	mawaAoEiyod	maFoEiL-maFaaEiiL	FvEvLvB-FaEaaLiiB-1234	مَوْعِد مَواعِيد
64	starling	zurozur	zaraAoziyor	FuEoLuB-FaEaaLiiB	FvEvLvB-FaEaaLiiB-1234	زرزور زرازير

In example (61), the character *C* (آ) is an obligatory substitute for the sequence *OaAo*. The morphology of nouns with *ma-*, *mu-* or *mi-* prefix relates them with verb participles. Their derivational patterns are traditionally labelled with semantic features of patient, e.g. *[ktb & maFoEuuL]= makotuwob* مكتوب 'letter', derived from the triliteral root *ktb* 'write', or of instrument, e.g. *[zrb & miFoEaaL] = mizoraAob* مزراب 'gutter', derived from *zrb* 'flow'. Some of these nouns denote places, e.g. *[nzl & maFoEiL] = manozil* 'house' منزل, from *nzl* 'go down'.

4.3.2. Other cases of diachronically motivated morphological segmentation

In a similar way, some nouns with 4 consonants are traditionally analysed as triliteral, by assigning one of the consonants to the pattern, usually because of a diachronical relation of the noun with a triliteral root, or for some other etymological reason. These nouns can usually be traced back to roots through derivational patterns for participles, deverbal nouns, instrumental nouns… The consonants thus discarded from the root are often *s, n, t, h, m, w, y* or the glottal stop [ʔ], noted by the allographs *c, O, e, W* and *I* (ء, أ, ئـ, ؤ , إ) depending on context. Some of these consonants are more likely to be discarded if they occur in some position in relation to the root. We list below 8 examples of such nouns. If analysed as quadriliteral, all enter in the independently existing inflectional class of *TarobuwoM* 'tarboosh' (C), just as if they were synchronically reanalysed as quadriliteral nouns for inflectional purposes:

	gloss	singular	plural	TM patterns	PRIM codes	Arabic
C	tarboosh	TarobuwoM	TaraAobiyoM	FaEoLuuB-FaEaaLiiB	FvEvLvvB-FaEaaLiiB-1234	طربوش طرابيش
65	expression	taEobiyor	taEaAobiyor	taFoEiiL-taFaaEiiL	FvEvLvvB-FaEaaLiiB-1234	تعبير تعابير
66	week	OusobuwoE	OasaAobiyoE	OuFoEuuL-OaFaaEiiL	FvEvLvvB-FaEaaLiiB-1234	أسبوع أسابيع
67	pumpkin	yaqoTiyon	yakaAoTiyon	yaFoEiiL-yaFaaEiiL	FvEvLvvB-FaEaaLiiB-1234	يقطين يقاطين
68	nostril	xayoMuwom	xayaAoMiyom	FayoEuuL-FayaaEiiL	FvEvLvvB-FaEaaLiiB-1234	خيشوم خياشيم
69	pig	xanoziyor	xanaAoziyor	FanoEiiL-FanaaEiiL	FvEvLvvB-FaEaaLiiB-1234	خنزير خنازير
70	address	EunowaAon	EanaAowiyon	FuEowaaL-FaEaawiiL	FvEvLvvB-FaEaaLiiB-1234	عنوان عناوين
71	coffin	taAobuwot	tawaAobiyot	FaEoLuut-FaEaaLiit	FvEvLvvB-FaEaaLiiB-1234	تابوت توابيت
72	plant	rayoHaAon	rayaAoHiyon	FaEoLaan-FaEaaLiin	FvEvLvvB-FaEaaLiiB-1234	ريحان رياحين

(65) is a deverbal noun with the derivational pattern *taFoEiiL* related to the verbal pattern *FaEEaLa*.

(66) *OusobuwoE* أسبوع 'week' is related to *saboE* سبع 'seven'.

In (67) and (68), *y* is considered exterior to the root, probably for some etymological reason.

In (69) *xanoziyor* خنزير 'pig', there is no agreement in traditional dictionaries such as Ibn Manzur (1290) and Al-Fairuzabadi (v. 1400): dictionaries consider the *n* in this word as a root consonant or not, because an *n* after the 1st root letter may have a special value.

In (70), *w* after the 2nd root letter may have a special value, and *EunawaAon* 'address' may be related to the triliteral root *EnY* عنى 'signify'.

(71) ends with *-uwot*, a suffix of Aramaic origin, so the final *t* is not considered a root consonant. However, Tarabay (2003) classifies it in both *FaEoLuut-FaEaaLiit* and *FaEoLuuB-FaEaaLiiB*.

In (72), *-aAon* is a suffix, so the final *n* is not considered a root consonant.

The assignment of a consonant to the patterns by traditional morphology makes the patterns of examples (68-70) distant from typical inflectional patterns for nouns, in which phonetic consonants sometimes occur before the 1st root letter, as in *OaFoEaaL* (cf. (11-12), Section 4.1.1), or after the last, as in *FaEoLap* (cf. (36), Section 4.2), but not between root letters, be it in the singular or in the BP. In the PRIM taxonomy, we analyse (65)-(72) as quadriliteral as far as inflection is concerned.

5. Root alternations

The root letters of most BPs have the same surface form as those of the singular, as in *Euqodap* 'knot' vs. *Euqad* 'knots'. Other BPs show root alternations, i.e. changes in the surface value of root letters, as in *qabow* 'cave' vs. *Oaqobiyap* 'caves', or in the number of root letters, as in *TaAobiE* 'stamp' طابع vs. *TawaAobiE* 'stamps' طوابع . In the PRIM model, root alternations are represented by a mapping between surface roots from the singular to the BP. This mapping is specified in a straightforward way by **root codes**, a new device.

5.1. Bypassing deep roots and rules

In traditional morphology, most root alternations are obtained by applying rules to deep stems. This model has two major drawbacks. First, rules are not very adequate for a phenomenon with such lexical dependency as BP; the few authors that formalized the rules of traditional morphology (Beesley, 1996; Habash, Rambow, 2006; Smrž, 2007) did not publish them in a readable, updatable way. Second, deep roots are not directly observable, which complicates decisions about what their exact value should be. We abandoned this model for root codes, a new device that simplifies the encoding of lexical items, as the following examples show.

5.1.1. Morphophonological alternations of the 2nd root letter

Some nouns with BP are analysed with their 2nd root letter realised as *A* in the singular, and as *w* or *y* in the plural:

	gloss	sing.	plural	root and patterns(TM)	PRIM codes	in Arabic
73	door	baAob	OabowaAob	bwb FaEoL-OaFoEaaL	FvEvL-OaFoEaaL-1w3	بـاب أبـواب
74	tooth	naAob	OanoyaAob	nyb FaEoL-OaFoEaaL	FvEvL-OaFoEaaL-1y3	نـاب أنـيـاب

Traditional morphology describes this with the aid of a deep root, displayed in the examples above just before the TM patterns: *bwb*, *nyb*. In the deep root, the 2nd root letter is the consonant observed in the plural and in derived words. Morphophonological rules change this letter to *A* in the singular, and leave it unchanged in the plural.

In the PRIM model, we specify the presence of *w* or *y* as the 2nd letter of the surface form of the BP root, through the root codes displayed in the examples above at the end of the PRIM codes: *1w3*, *1y3*. The surrounding slots are represented in the root code, as usual, by a digit corresponding to their rank. We stick to directly observable facts. The transducer associated to the inflectional code generates *w* or *y* at the position of the 2nd letter root in the BP. The root code specifies the value of BP root letters when they differ from the corresponding singular root letters. As a simplification, the value of the 2nd letter in the plural is encoded in the root code whenever it is *y*, *w*, a glottal stop [ʔ], or *A*. This is not strictly necessary for the generation of the plural of *suwor* 'wall', which is *OasowaAor*, since root code *123* would yield the same result as *1w3*, but it simplifies the manual encoding of entries.

The following example illustrates the converse situation. The 2nd root letter *y* is replaced by *A* in the plural:

	gloss	singular	plural	TM root and patterns	PRIM codes	in Arabic
75	politician	siyaAosiyG	saAosap	sys FiEaaLiyy-FaAoLap	FvEvvL-FaEolap-1A3	سيـاسيَ ساسة

When the 2nd root letter of a triliteral noun is realised in the singular as [ʔ], the corresponding letter in the plural may be, unpredictably, [ʔ], *y*, *w* or *A*:

	gloss	singular	plural	TM root and patterns	PRIM codes	in Arabic
76	sad	baAoeis	baOasap	bcs FaaEiL-FaEaLap	FvvEvL-FaEaLap-1h3	بائس بأسة
77	betrayer	xaAoein	xawanap	xwn FaaEiL-FaEaLap	FvvEvL-FaEaLap-1w3	خائن خونة
78	undecided	HaAoeir	Hayarap	Hyr FaaEiL-FaEaLap	FvvEvL-FaEaLap-1y3	حائر حيرة
79	seller	baAoeiE	baAoEap	byE FaaEiL-FaEoLap	FvvEvL-FaEoLap-1A3	بائع بائعة

The letters *c* and *O* note allographs of the glottal stop [ʔ]. Traditional morphology postulates deep roots. In (79), the underlying *y* of the deep root occurs neither in the singular nor in the plural; rules change it to *e* in the singular and to *A* in the BP.

We encode the 2nd root letter of the plural in the root code: *1h3, 1w3, 1y3, 1A3*. In root codes, the symbol *h* stands for [ʔ]. There are much less distinct root codes in the PRIM model than roots in TM: all the deep roots of triliteral nouns with alteration of the 2nd root letter conflate to the 4 code roots cited above.

5.1.2. Morphophonological alternations of the 3rd root letter

The situation is the same for nouns which alter their 3rd root letter. In the BP, this letter is realised as *y* or *c*, or as the long vowel [a:], noted *A* or *Y*:

	gloss	sing.	plural	TM root and patterns	PRIM codes	in Arabic
80	organ	EuDow	OaEoDaAoc	ED- FuEow-OaFoEaaL	FvEvL-OaFoEaaL-12h	عُضْو أعْضاء
81	cloth	zayG	OazoyaAoc	zy- FaEE-OaFoEaaL	FvEvL-OaFoEaaL-12h	زِيّ أزْياء
82	climate	jawG	OajowaAoc	jw- FaEE-OaFoEaaL	FvEvL-OaFoEaaL-12h	جوّ أجواء
83	enemy	EaduwG	OaEodaAoc	Ed- FaEuuw-OaFoEaaL	FvEvvL-OaFoEaaL-12h	عدوّ أعداء
84	cave	qabow	Oaqobiyap	qb- FaEow-OaFoEiLap	FvEvL-OaFoEiLap-12y	قبو أقبية
85	pot	wiEaAoc	OawoEiyap	wE- FiEaac-OaFoEiLap	FvEvvL-OaFoEiLap-12y	وعاء أوعية
86	boy	fataY	futoyaAon	ft- FaEaY-FuEoLaan	FvEvL-FuEoLaan-12y	فتى فتيان
87	boy	fataY	fitoyap	ft- FaEaY-FiEoLap	FvEvL-FiEoLap-12y	فتى فتية
88	judge	qaAoDiy	quDaAop	qD- FaaEiy-FuEaap	FvvEvL-FuEoLap-12A	قـَاضِي قـُضَاة
89	jewel	Hiloyap	HilaY	Hl- FiEoyap-FaEaY	FvEvL-FiEaL-12Y	حلية حلًى
90	step	xuTowap	xuTaY	xT- FuEowap-FuEaY	FvEvL-FuEaL-12Y	خطوة خطى

Since scholars may disagree on the value of the 3rd letter of the traditional deep root, we omit it above. In the PRIM model, the surface value of the 3rd root letter in the plural is encoded in the root code whenever it is *y*, [ʔ], *A* or *Y*:

91	valley	waAodiy	Oawodiyap	wd- FaaEiL-OaFoEiLap	FvvEvL-OaFoEiLap-12y	وادي أودية
92	pastor	raAoEiy	ruEoyaAon	rE- FaaEiL-FuEoLaan	FvvEvL-FuEoLaan-12y	راعي رعيان

5.1.3. Orthographic alternations of glottal stop in roots

Roots with the glottal stop [ʔ] undergo purely orthographic alternations. The glottal stop [ʔ] has 6 allographs in the Arabic alphabet: *c*, *e*, *W*, *O*, *I* and *C* (ء, أ, ئـ, ؤ , إ, آ) . In general, the choice of the allograph depends on orthographic context, and in particular on the preceding and following vowels.[17] For example, an initial [ʔ] is written *O* (أ) when it is followed by *a* or *u*, and *I* (إ) when followed by *i*. The character *C* (آ) is an obligatory substitute for the sequences *OaAo* and *OaOo*. The allographs can be different between the singular and the plural, because they are inserted in different patterns:

93	kettle	Iiboriyoq	OabaAoriyoq	IiFoEiiL-OaFaaEiiL	FvEvLvvB-FaEaaLiiB-h234	إبريق أباريق
94	African	IiforiyoqiyG	OafaAoriqap	IiFoEiiLiyG-OaFaaEiLap	FvEvLvvB-FaEaaLiBap-h234	إفريقي أفارقة

Because of these spelling changes, we systematically register in root codes the presence of [ʔ]. In root codes, the symbol *h* stands for [ʔ]. Then, the plural pattern is sufficient to determine the allograph in the BP:

[17] In some configurations, no standard is actually applied to determine the allograph, and practice depends on regions and authors. In Arabic dialects, initial [ʔ] admits phonetic variants, and some of them may have an influence on spelling in Modern Standard Arabic.

95 trouble	**ma**O̲ozaq	ma**C̲**ziq	**ma**FoEaL-**ma**FaaEiL	FvEvLvB-Fa**E̲**aaLiB-1h̲34	مأزق مآزق
96 twin	ta**w**o**O̲**am	tawaAo**e̲**im	Fa**w**oEaL-Fa**w**aaEiL	FvEvLvB-FaEaa**L̲**iB-12h̲4	توءم توائم
97 congrat.	tahonieap	tahaAonie	taFoEiLap-taFaaEiL	FvEvLvB-FaEaaLiB-123h	تهنئة تهانئ
98 principle	**ma**bodaO̲	mabaAodi**e̲**	**ma**FoEaL-**ma**FaaEiL	FvEvLvB-FaEaaLi**B̲**-123h̲	مبدأ مبادئ
99 pearl	luW̲oluW̲	la**C̲**li**e̲**	FuEoFuE-FaEaaFiE	FvEvLvB-Fa**E̲**aaLi**B̲**-1**h̲**3**h̲**	لؤلؤ لآلئ

The correct allograph of [ʔ] is inserted by the transducer associated to the inflectional code. It is not necessary to specify it in the root code, since it depends on the context, which is encoded in the BP pattern.[18]

Even when the allograph is the same in the singular and in the plural, we encode the presence of the glottal stop in the root code (100, 101). This is not strictly necessary for the generation of the plural, since in such case root code *1234* would yield the same result as *h234*, but it simplifies the manual encoding of entries:

100 warehouse	O̲anobaAor	O̲anaAobir	O̲aFoEaaL-OaFaaEiL	FvEvLvvB-FaEaaLiB-h234	أنبار أنابير
101 teacher	O̲usotaAoJ	O̲asaAotiJap	O̲uFoEaaL-OaFaaEiLap	FvEvLvvB-FaEaaLiBap-h234	أستاذ أساتذة

The allography of [ʔ] poses problems in stem-final position. The allograph may depend on graphically agglutinated pronouns:

ruWasaAoc̲i	'presidents'	رؤساء
ruWasaAoe̲ihaA	'its presidents'	رؤسائها

In these examples, the final *i* is an inflectional suffix and *-haA* is a clitic pronoun in the genitive. This problem is dealt with in Section 8.

Nouns with initial [ʔ] and BP pattern *OaFoEaaL* pose another problem of allography. In the plural, the combination of the root with the pattern produces an underlying form that begins with the sequence *OaOo*. Due to morphophonological rules, this initial sequence is not pronounced [ʔaʔ] but [ʔa:], and the surface form is not scripted *OaOo* or *OaAo*, but *C* آ :

102 horizon	Oufuq	CfaAoq	FuEuL-OaFoEaaL	FvEvL-OaFoEaaL-h23	أفق آفاق

The PRIM transducers actually produce *C*, but we named the root code *h23* and not *A23*, to remind the underlying [ʔ]: since words in Arabic never begin with a long vowel, it is not natural to Arabic speakers to consider that a root begins with *A*.

5.1.4. Biliteral nouns

There are less than 20 biliteral nouns in Arabic. When they admit a BP, it is always triliteral, often with the addition of a final consonant, generally *c*:

	gloss	sing.	plural	TM root and patterns	PRIM codes	in Arabic
103	blood	dam	dimaAo**c̲**	dmc FaE-FiEaaL	FvE-FiEaaL-12h	دم دماء
104	father	Oab	CbaAo**c̲**	Obw FaE-OaFoEaaL	FvE-OaFoEaaL-h2h	أب آباء
105	brother	Oax	Iixo**w̲**ap	Oxw FaE-FiEoLap	FvE-FiEoLap-h2w	أخ إخوة

Traditional morphology generally describes such nouns with a triliteral deep root in which the 3rd root letter is not realised in the singular. Some scholars disagree on this notion of false biliteral, and analyse these roots as underlyingly biliteral. The PRIM taxonomy uses a biliteral singular-pattern code.

A small series of nouns begin with *Ii* in the singular,[19] and have two other consonants; this initial part is pronounced only if the word is preceded by a pause:

106	son	Iibon	OabonaAoc	bnc FoE-OaFoEaaL	FvEvL-OaFoEaaL-23h	إبن أبناء
107	name	Iisom	OasomaAoc	smc FoE-OaFoEaaL	FvEvL-OaFoEaaL-23h	إسم أسماء

According to traditional morphology, this initial letter does not count as a root letter, so these nouns are biliteral. We encode them as triliteral.

[18] (97) admits an alternative plural, *tahaAoniy*, which is assigned to another lexical entry (cf. Section 3.2).
[19] Recall that *I* is an allograph of [ʔ].

5.2. Shifting information from broken-plural patterns to root codes

In some cases, traditional morphology accounts for consonant insertions through special BP patterns such as *FawaaEiL*, *FaEaaeiL*, *FaEaayiL* (فواعل, فعائل, فعايل). By encoding such insertions in root codes, we reduce the number of BP patterns.

5.2.1. Triliteral lemmas with insertion of *y*, *w* or [ʔ]

The following nouns have 3 phonetic consonants in the singular, excluding suffixes, and 4 in the BP:

	gloss	singular	plural	TM patterns	PRIM codes	in Arabic
108	stamp	TaAobiE	TawaAobiE	FaaEiL-FawaaEiL	FvvEvL-FaEaaLiB-1w23	طابع طوابع
109	order	Oamor	OawaAomir	FaEoL-FawaaEiL	FvEvL-FaEaaLiB-1w23	أمر أوامر
110	brothel	maAoxuwor	mawaAoxir	FaaEuuL-FawaaEiL	FvvEvvL-FaEaaLiB-1w23	ماخُور مَواخِر
111	last	Cxir	OawaAoxir	FaaEiL-FawaaEiL	FvvEvL-FaEaaLiB-hw23	آخر أواخر
112	revenue	EaAoeid	EawaAoeid	FaaEiL-FawaaEiL	FvvEvL-FaEaaLiB-1wh3	عائد عوائد
113	darling	Habiyob	HabaAoyib	FaEiiL-FaEaayiL	FvEvvL-FaEaaLiB-12y3	حبيب حبايب
114	old	Eajuwoz	EajaAoeiz	FaEuuL-FaEaaeiL	FvEvvL-FaEaaLiB-12h3	عَجُوز عَجَائِز
115	first	OawGal	OawaAoeil	FaEEaL-FaEaaeiL	FvEEvL-FaEaaLiB-12h3	أوّل أوائل
116	angel	malaAok	malaAoeikap	FaEaaL-FaEaaeiLap	FvEvvL-FaEaaLiBap-12h3	ملاك ملائكة

Traditional morphology postulates that the deep root is the same for all the forms of a lexical entry. In consequence, the BP of these nouns has to be analysed with triliteral roots; the additional consonant can only be assigned to the pattern. This generates several additional BP patterns which specify the position and value of the additional consonant, as *FawaaEiL*. The fact that the additional consonant occurs between the slots for root letters in these patterns makes them distant from other inflectional patterns for nouns, as *FaEaaLiB*. Recall that in typical inflectional patterns for nouns, be it in the singular or in the BP, phonetic consonants sometimes occur before the 1st slot, as in *OaFoEaaL*, or after the last, as in *FaEoLap*, but not between slots (Section 4.3.2).

In contrast, if we analyse the nine BPs above (108-116) with quadriliteral roots, all their patterns conflate with *FaEaaLiB* and *FaEaaLiBap*, which are independently needed for other BPs. We adopted this solution for the PRIM taxonomy. We use the root code to specify the insertion of the additional consonant in the plural root. This analysis simplifies the BP pattern taxonomy by merging classes. It changes the BP patterns, but it remains straightforward to Arabic speakers, since it reuses familiar BP patterns.

In these nouns, the position of the additional consonant of the BP is often occupied by a long vowel in the singular. For a couple of them, an alternative analysis is possible, in which the singular has a quadriliteral root, and one of the root letters codes the long vowel of the singular, as in (117a):

	gloss	singular	plural	TM root and patterns	PRIM codes	in Arabic
117	missile	S**aAo**ruwox	Sa_waAoriyox	Srx FaaEuuL-FawaaEiiL	FvvEvvL-FaEaaLiiB-1w23	صاروخ صواريخ
117a				Swrx FaEoLuuB-FaEaaLiiB		
118	wheel	d**uwo**laAob	da_waAoliyob	dlb FuuEaaL-FawaaEiiL	FvvEvvL-FaEaaLiiB-1w23	دولاب دوالـيب
118a				dwlb FuEoLaaB-FaEaaLiiB		

The two alternative analyses (117) and (117a) do not correspond to distinct interpretations of the form: they are two formal accounts for a single linguistic object. This situation requires a choice, so that the morphological analysis reports a single analysis. The solution of (117a) has the advantage of being closer to the encoding of lemmas with 2 phonetic consonants, such as *baAob* 'door' (Section 5.1.1). However, we opted for the solution of (117) which is consistent with (108)-(116). The availability of several solutions to describe the same phenomenon is a flaw in a descriptive model. In order to reduce this indeterminacy in the encoding of entries, we adopted the following rule:

> *For nouns with at least 3 phonetic consonants in the singular stem, long vowels occurring between the first 3 consonants are assigned to the pattern.*

For example, as *SaAoruwox* 'missile' has 3 phonetic consonants *S*, *r* and *x*, the long vowel *aA* is assigned to the pattern, which is specified by picking the singular-pattern code *FvvEvvL*. This rule leads to familiar patterns: for example, *FaEaaLiiB*, in (117) and (118), is independently needed for other nouns. The rule does not apply to

baAob ‘door’ since this noun has only 2 phonetic consonants. In this type of nouns, the long vowel between the two consonants is unanimously analysed as a root letter.

Traditional morphology has still another analysis for similar nouns, adopting the root of their derivational base:

	gloss	singular	plural	TMroot patterns	PRIM codes	in Arabic
119	port	miyonaAoc	ma**w**aAonie	? miFoEaaL-maFaaEiL	FvvEvvL-FaEaaLiB-1w2h	ميناء موانئ
120	scale	miyozaAon	ma**w**aAoziyon	wzn miFoEaaL-maFaaEiiL	FvvEvvL-FaEaaLiiB-1w23	ميزان موازين
121	cave	magaAorap	magaAo**w**ir	gwr maFoEiLap-maFaaEiL	FvEvvL-FaEaaLiB-12w3	مَغارة مَغاوِر
122	defect	maEaAobap	maEaAo**y**ib	Eyb maFoEaLap-maFaaEiL	FvEvvL-FaEaaLiB-12y3	مَعابة مَعايب

We opted for the solution of (108-116), for the same reasons as in Section 4.3.1. [20]

The noun *EaAodap* ‘habit’ shows, in addition to the insertion of *w* before the 2nd root letter, the substitution of *e* for *A* as 2nd root letter:

123	habit	E**aAo**dap	Ea**w**aAo**e**id	FaEoLap-FawaaeiL	FvEvL-FaEaaLiB-1wh3	عادة عوائد

We have analysed all the nouns in this section with a triliteral root in the singular, and a quadriliteral root in the plural. In the following sections, we survey other examples of this configuration, where the additional root consonant is obtained by reduplicating one of those of the singular, or by inserting a prefix or a suffix. Then, we discuss the case of nouns with 5 consonants in the singular, and 4 in the BP, obtained by removing one of the 5 consonants.

Most quadriliteral BPs show no root alterations as compared to the singular (cf. (16-20), Section 4.1.1). They have one of the three following patterns: *FaEaaLiB*, *FaEaaLiBap* and *FaEaaLiiB*.

5.2.2. Triliteral lemmas with geminated consonant and quadriliteral BP

A number of lemmas with a geminated consonant have a quadriliteral BP. In general, the geminated consonant appears in the plural as two simple occurrences, with a long vowel between them:

124	ladder	su**lG**am	sa**l**aAo**l**im	FuEEaL-FaEaaEiL	FvEEvL-FaEaaLiB-1223	سُلَّم سَلالِم
125	pillow	Ta**rG**aAoHap	Ta**r**aAo**r**iyoH	FaEEaaLap-FaEaaEiiL	FvEEvvL-FaEaaLiiB-1223	طَرّاحَة طراريح
126	mighty	ja**bG**aAor	ja**b**aAo**b**irap	FaEEaaL-FaEaaEiLap	FvEEvvL-FaEaaLiBap-1223	جبّار جبابرة
127	dragon	tinGiyon	tanaAoniyon	FiEEiiL-FaEaaEiiL	FvEEvvL-FaEaaLiiB-1223	تنّين تنانين
128	ox	fidGaAon	fadaAodiyon	FiEEaaL-FaEaaEiiL	FvEEvvL-FaEaaLiiB-1223	فدّان فدادين
129	needle	dabGuwos	dabaAobiyos	FaEEuuL-FaEaaEiiL	FvEEvvL-FaEaaLiiB-1223	دبّوس دبابيس

The geminated consonant of the singular is analysed as a single letter of a triliteral root, and the gemination is assigned to the singular pattern (cf. Section 3.3). The root code *1223* specifies the repetition of the 2nd root letter. In *OawGal* ‘first’, the geminated consonant of the singular is realised as a simple consonant in the plural, but an additional *e* (ئ) is inserted:

130	first	Oa**wG**al	Oa**w**aAo**e**il	FaEEaL-FaEaaeiL	FvEEvL-FaEaaLiB-12h3	أوّل أوائِل

In *MidGap* ‘trouble’, the geminated consonant corresponds to two letters of a triliteral root, and an additional *e* is inserted between them:

131	trouble	Mi**dGap**	Ma**d**aAo**e**i**d**	FiEoLap-FaEaaeiL	FvEvL-FaEaaLiB-12h2	شدّة شدائد

Some triliteral nouns have a quadriliteral BP with a reduplication of the 2nd root letter and a long vowel between the two occurrences:

132	dinar	diyo**n**aAor	da**naAon**iyor	FiiEaaL-FaEaaEiiL	FvvEvvL-FaEaaLiiB-1223	دينار دنانير
133	lighthouse	fa**n**aAor	fa**naAon**iyor	FaEaaL-FaEaaEiiL	FvEvvL-FaEaaLiiB-1223	فَنار فَنانير
134	mortar	haAo**w**un	ha**waAow**iyon	FaaEuL-FaEaaEiiL	FvvEvL-FaEaaLiiB-1223	هاوُن هَواوِين

These nouns seem to have atypical origins, since they are not related to attested verbal forms.

5.2.3. Triliteral lemmas with BP in *-iy* or *-aY*

Some triliteral lemmas have a quadriliteral BP with *-iy* or *-aY* appended to the root (cf. (37)-(42), Section 4.2):

[20] (119) admits an alternative plural, *mawaAoniy*, which is assigned to another lexical entry (cf. Section 3.2).

135	bottle	qanGiyon**ap**	qanaAon**iy**	FaEEiiLap-FaEaaLiy	FvEEvvL-FaEaaLiB-123y	قنّينة قناني
136	land	OaroD	OaraAoD**iy**	FaEoL-FaEaaLiy	FvEvL-FaEaaLiB-h23y	أرض أراضي
137	night	layolap	layaAoliy	FaEoLap-FaEaaLiy	FvEvL-FaEaaLiB-123y	ليلة ليالي
138	snake	OafoE**aY**	OafaAoE**iy**	FaEoLaY-OaFaaEiy	FvEvL-FaEaaLiB-h23y	أفعى أفاعي
139	virgin	EaJor**aAoc**	EaJaAor**aY**	FaEoLaac-FaEaaLaY	FvEvL-FaEaaLiB-123Y	عذراء عذارى

In most of these examples, the singular has a suffix such as *-ap* or *-aY*, which suggests that the ending *-iy* is also a suffix. However, by analysing these endings as part of the stem, we homogenize the nouns with other quadriliteral BPs with pattern *FaEaaLiB*.

In the following examples, *y* is the 3rd consonant of the singular root, and a *w* is inserted before the 2nd consonant, as in (108)-(112), Section 5.2.1:

140	suburb	DaAoHiy**ap**	Da**w**aAoHiy	FaaEiLap-FawaaEiL	FvvEvL-FaEaaLiB-1w2y	ضاحية ضواحي
141	whore	EaAoriy**ap**	Ea**w**aAoriy	FaaEiLap-FawaaEiL	FvvEvL-FaEaaLiB-1w2y	عارية عواري

5.2.4. Triliteral lemmas with BP in *Oa-* or *ma-*

Some triliteral nouns have a BP with an initial *Oa-*, often in concurrence with another plural.[21] We encode the BP in *Oa-* as quadriliteral if it matches one of the three independently known quadriliteral BP patterns (143), and as triliteral otherwise (142):

	gloss	singular	plural	PRIM codes	in Arabic
142	place	makaAon	**Oa**mokinap	FvEvvL-OaFoEiLap-123	مَكان أمكنة
143	place	makaAon	**Oa**maAokin	FvEvvL-FaEaaLiB-h123	مَكان أماكِن

In TM, the BP in (143) is marked as 'plural of plural' and obtained by re-pluralizing the BP in (142):

	gloss	singular	plural	pl. of pl.	TM patterns	in Arabic
144	place	makaAon	**Oa**mokinap	**Oa**maAokin	FaEaaL-OaFoEiLap-OaFaaEiL	مَكان أمكنة أماكِن

Recall that we do not formalize the 'plural of plural' mark in our model (cf. Section 3). Here is a similar example, but both BPs have quadriliteral patterns:

	gloss	singular	plural	PRIM codes	in Arabic
145	pregnant	HabolaY	HabaAolaY	FvEvL-FaEaaLiB-123Y	حبلى حبالى
146	pregnant	HabolaY	**Oa**HaAobiyol	FvEvL-FaEaaLiiB-h123	حبلى أحابيل

	gloss	sing.	plural	pl. of pl.	TM patterns	in Arabic
147	pregnant	HabolaY	HabaAolaY	**Oa**HaAobiyol	FaEoLaY-FaEaaLaY-OaFaaEiiL	حبلى حبالى أحابيل

The noun *Hadiyov* 'talk' has only one BP in *Oa-*:

	gloss	singular	plural	TM patterns	PRIM codes	in Arabic
148	talk	Hadiyov	**Oa**HaAdiyov	FaEiiL-OaFaaEiiL	FvEvvL-FaEaaLiiB-h123	حَديث أحاديث

Finally, some triliteral nouns have a quadriliteral BP with an initial *ma-*:

149	feeling	MuEuwor	maMaAoEir	FuEuuL-maFaaEiL	FvEvvL-FaEaaLiB-m123	شعور مشاعر
150	danger	xaTar	maxaAoTir	FaEaL-maFaaEiL	FvEvL-FaEaaLiB-m123	خطر مخاطر
151	drawback	sayGicap	masaAowie	FaEEiLap-maFaaEiL	FvEEvL-FaEaaLiB-m1wh	سَيِّئَة مَساوِئ

Dictionaries describe this type of plural, but grammarians have paid little attention to them. Tarabay (2003) does not mention them. These nouns usually denote abstract entities and are derived from verbs or adjectives. The *ma-* insertion can be compared with *Oa-* and with derivational prefixes in *m-* occurring in past participles and deverbal nouns. Diachronically, the singular and the plural of such pairs may have come from distinct lexical items. However, synchronically, their association within a single item is confirmed by comparing sentences such as:

[21] As a rule, we generate at most one plural of a given lexical entry. When several plurals are observed, they are assigned to distinct entries, no matter whether they are equivalent or not (cf. Section 3.2).

جلس الشيخ في قاعة الإجتماعات يراجع حساباته الانتخابية
jalasa Al-Mayoxu fiy qaAEapi Al-IijtimaAEaAti yuraAjiEu HisaAbaAti-hi Al-IntixaAbiyap
sat the-sheikh in the-room-meeting review calculation-his electoral
"The sheikh sat in the meeting room reviewing his electoral calculation"

جلست المشايخ في قاعة الإجتماعات تراجع حساباتها الانتخابية
jalasat Al-maMaAyixu fiy qaAEapi Al-IijtimaAEaAti turaAjiEu HisaAbaAti-hA Al-IntixaAbiyap
sat the-sheikhs in the-room-meeting review calculation-her electoral
"The sheikhs sat in the meeting room reviewing their electoral calculations"

The only semantic difference between these two sentences is about the number of the subject. Such differential semantic evaluation (Gross, 1975) is a particularly reliable and reproducible type of introspective evidence about semantic facts.

5.2.5. Lemmas with 5 or 6 consonants

From a 5-consonant singular, the formation of a quadriliteral BP requires the omission of one of the 5 consonants. The first consonant is never omitted. The consonants *y*, *w* or an *n* are often omitted:

```
152 philosopher
   fayolasuwof   falaAosifap FayoEaLuuB-FaEaaLiBap  FvEvLvBvvD-FaEaaLiBap-1345  فيلسوف فلاسفة
153 program
   baronaAomaj   baraAomij   FaEonaaLaB-FaEaaLiB    FvEvLvvBvD-FaEaaLiB-1245  برنامج برامج
154 elephant (female)
   EaqaroTal     EaqaAoril   FaEaLoBaD-FaEaaLiD     FvEvLvBvD-FaEaaLiB-1235      عقرطل عقارل
155 cylinder
   OusoTuwaAonap OasaAoTiyon FuEoLuwaaBap-FaEaaLiiB FvEvLvBvvD-FaEaaLiiB-h235 أسطوانة اساطين
```

Note that in the singular, for TM, the consonant omitted in the BP is assigned to the pattern in (152, 153, 154), but to the root in (155).

The 5th consonant is often omitted:

```
156 quince
    safarojal     safaAorij   FaEaLoBaD-FaEaaLiB   FvEvLvBvD-FaEaaLiB-1234    سفرجل سفارج
157 octopus
    OaxoTabuwoT   OaxaAoTib   FaEoLaBuuD-FaEaaLiB  FvEvLvBvvD-FaEaaLiB-h234  أخطبوط أخاطب
```

Here is a similar example with 6 consonants:

```
158 emperor  IimobaraAoTuwor  OabaAoTirap      FvEvLvBvvDvvJ-FaEaaLiBap-h356  إمبراطور أباطرة
```

A few 5-consonant nouns deviate from the standard quadriliteral BP patterns in that all 5 root consonants are retained in the BP, with the 3rd and 4th ones jointly in the 3rd slot of the BP pattern:

```
159 crab     siloToEaAon salaAoToEiyon    FvEvLvBvvD-FaEaaLiiB-12345   سلطعان سلاطعين
160 pot      miroTobaAon maraAoTobiyon    FvEvLvBvvD-FaEaaLiiB-12345    مرطبان مراطبين
161 thimble  kiMotobaAon kaMaAotobiyon    FvEvLvBvvD-FaEaaLiiB-12345    كشتبان كشاتبين
```

The surface pattern actually handled by the PRIM transducers of these BPs is *FoEaaLoBiiD*. However, we analyse this pattern as a variant of quadriliteral *FaEaaLiiB*, and we use the label of this pattern in the inflectional codes. These nouns deviate from general rules in several ways. First, all other BP roots have at most 4 consonants. Second, these BPs are pronounced in three syllables as Cv-CvvC-CvvC with unusual CvvC second syllables: [sala:tˤʕi:n mara:tˤbi:n kaʃa:tbi:n ʔatˤa:rmi:zˤ], as if the attraction to a quadriliteral BP pattern were stronger than phonotactic constraints. We are not aware of any prior mention of these exceptional nouns in literature about Arabic.

Unlike standard Arabic, we report, in the Lebanese dialect, the existence of initial consonant clusters for examples (159-161) as *solaAoToEiyon*, pronounced in two syllables as CCvvC-CvvC [sla:tˤʕi:n mra:tˤbi:n kʃa:tbi:n]. (163) is a similar example with an initial consonant cluster, but in a triliteral BP pattern; (162) is the

BP of this word in standard Arabic. A probable template for (163) in standard modern Arabic is the inflectional class of (164), with a standard BP pattern *FiEaL*:

162 strip	MariyoTap-MaraAoeiT	FvEvvL-FaEaaLiB-12e4	شريطة شرائط
163 strip	MoriyoTap-MoriyaT	F_1F_2vEvL-F_1F_2iEaL-1y3	شريطة شرِيَط
164 uprising	fitonap-fitan	FvEvL-FiEaL-123	فتنة فتن

Two other plurals of the same noun are observed in the Lebanese dialect: a suffixal plural *MoriyoT-aAot* شريطات and a variant of (162), *MaraAoyiT*.

6. Quantitative data about the taxonomy

Our BP lexicon is composed of 3 198 noun entries, among which 1 662 admit a triliteral BP, and 1 536 a quadriliteral BP. We have 985 BPs with the *FaEaaLiB* pattern. Table 1 shows how entries with this BP pattern are distributed according to the singular-pattern taxonomy.

Singular-Pattern Code		Example			Entries	In Arabic script
		Gloss	Plural	Singular		
FvEvLvB	FvEvLvB	dirham	daraAhim	diroham	556	درهم دراهم
	FvEvLvB-ap	tornado	zawaABiE	zawobaEap		زَوْبَعَة زَوابَع
	FvEvLvB-iyy	foreigner	OajaAnib	OajonabiyG		أجنبي أجانب
	FvEvLvB-iyyap	rifle	banaAdiq	bunduqiyGap		بُنْدُقِيَّة بَنادِق
	FvEvLvB-p	turtle	salaAHif	suloHaFaAp		سُلَحْفاة سَلاحِف
FvEvvLvB		sample	namaAzij	namuwozaj	1	نموذج نماذج
FvEvLvvB		bat	waTaAwiT	wuTowaAT	19	وَطْواط وَطاوِط
FvEvLLvB		buildings	majaAmiE	mujamGaE	4	مجمّع مجامع
FvvEvL		stamp	tawaAbiE	TaAobiE	165	طابع طوابع
FvEEvvL		bottle	**qanaAniy**	qanGiynap	1	قنّينة قناني
FvvEvvL		port	**mawaAnie**	miyonaAoc	6	ميناء موانئ
FvEvvL		cave	magaAwir	magaAorap	197	مَغارة مَغاوِر
FvEEvL		ladder	salaAlim	sulGam	5	سُلّم سَلالِم
FvEvL		order	OawaAmir	Oamor	25	أمر أوامر
FvEvLvBvD		quince	safaArij	safarojal	4	سفرجل سفارج
FvEvLvvBvD		program	baraAmij	baronaAomaj	1	برنامج برامج
FvEvLvBvvD		octopus	OaxaATib	OaxoTabuwoT	1	أخطبوط أخاطب
		TOTAL			**985**	

Table 1. Distribution of lexical items with the *FaEaaLiB* BP pattern according to the singular-pattern taxonomy.

The 3 198 entries with BP are inflected by means of finite-state transducers in number, definiteness and case (3×3×3). An entry which does not inflect in gender produces 27 surface forms. An entry which inflects also in gender produces 2×3×3×2 forms for the singular and the dual, which inflect in gender, and 1×3×3×1 for the BP, which does not inflect in gender (cf. Section 7); this totals to 45. The size of the full-form dictionary is 97 002 surface forms. It occupies 4.9 Megabytes in Unicode little Endian in plain text. It is compressed and minimized into 430 Kilobytes, and loaded to memory for fast retrieval. The generation, compression and minimization of the full-form lexicon lasts a few seconds on a Windows laptop.

The number of inflectional graphs is 300 : 25 BP patterns, 75 singular pattern/BP pattern pairs, 160 singular pattern/BP patterns/root code triples, and 300 when we take into account the generation of gender and inflectional suffixes in the singular. In addition, the main graphs invoke approximately 20 sub-graphs.

This number of inflectional graphs (300) is to be compared with the nearly 390 inflectional graphs for nouns for Brazilian Portuguese constructed also for Unitex (Muniz *et al.*, 2005) which deals with gender, number and degree (base, diminutive and augmentative), as in *casa(s)* 'house(s)', *casinha(s)* 'small house(s)', *casarão/casarões* 'large house(s)'. Another 245 inflectional graphs for adjectives deal with gender, number and degree: *lindo(s)/linda(s)* 'beautiful' (base), *lindinho(s)/lindinha(s)* (diminutive), *lindão/lindões/lindona(s)* (augmentative) and *lindíssimo(s)/lindíssima(s)* (superlative). With suffixal plurals, which will require at most 20 additional graphs, the number of inflectional graph for Arabic nouns does not reach the number of graphs for the Unitex Portuguese (Brazil) dictionary.

7. Rules of agreement with broken plural nouns

The difference between BP and suffixal plural in Arabic is obviously a matter of inflectional morphology, but not only. Grammatical agreement of plural nouns with adjectives, participles or verbs is slightly different depending on whether the plural noun is a BP or a suffixal plural. The difference is observed both with human and non-human nouns, but agreement follows distinct rules.

7.1. Human nouns

A human noun in the plural can agree with adjectives and participles in the broken or suffixal plural, or with both, if the adjective has both plurals. This rule applies independently of whether the plural noun is a BP, as *EulamaAocu* 'scientists', or a suffixal plural, as *muraAoqibuwona* 'observers'. In the following examples, the *:q* code marks BPs, and *:p* marks suffixal plurals:

....والعلماء (العاملون + النشطاء) في حقل الكيمياء ...

wa-Al-EulamaAcu Al-(nuMaTaAc + EaAmiluwna) fiy Haqoli Al-kiymoyaAc
and-the-scientists:q the-(active:q + working:p) in area the-chemistry
'and the scientists (active + working) in the area of chemistry'

....و المراقبون الدوّليون (العاملون + النشطاء) في سوريا ...

wa-Al-muraAqibuwna Al-duwGaliyGuna Al-(nuMaTaAc + EaAmiluwna) fiy suwriyGaA
and-the-observers:p the-international the-(active:q + working:p) in Syria
'and the international observers (active + working) in Syria'

However, if the human noun is in the BP, it can also agree with an adjective or participle in the feminine singular (*:fs* code below), no matter the gender of the noun or the sex of its referent:[22]

....والعلماء العاملة في حقل الكيمياء ...

wa-Al-EulamaAcu Al-EaAmilapu fiy Haqli Al-kiymoyaAc
and-the-scientists:mq the-working:fs in area the-chemistry
'and the scientists working in the area of chemistry'

This additional possibility of agreement is not observed with suffixal plurals of human nouns (the '*' symbol signals unacceptability here):

... والمراقبون الدوّليون العاملة في سوريا ...

**wa-Al-muraAqibuwna Al-duwGaliyGuna Al-EaAmilapu fiy suwriyGaA*
*and-the-observers:p the-international the-working:fs in Syria
'and the international observers working in Syria'

Agreement of adjectives in the feminine singular with BP human nouns may surprise non-Arabic speakers. It is less frequent than agreement of adjectives in the plural, but handbooks definitely consider it as grammatical, and it occurs in literary works:

[22] The adjective or participle could be analysed and labeled as an alternative plural, with the same form as a feminine singular (Smrž, 2007:27).

الرجال شحيحة في مصر الآن...
Al-rijaAlu MaHiHapun fiy misra AaloCn
the-<N:mq> <A:fs> in Cairo presently
'Men are rare in Cairo presently'
(Rim Basyuwniy, *Smell of The Sea*, http://arabicorpus.byu.edu/)

The rules of grammatical agreement between subject noun and verb, when the verb occurs after the subject, are similar to the rules above. A BP human noun subject can agree with the verb in the feminine singular, whereas a suffixal plural human noun subject cannot:

القضاة (غادرت + غادروا + غادرن) ظهراً
Al-quDaApu (gaAdarat + gaAdaruwA + gaAdarona) ZuhoraAF
The-judges:q (left:fs + left:mp + left:fp) at-mid-day
'The judges left at-mid-day'

المراقبون (*غادرت + غادروا + *غادرن) ظهراً
*Al-muragibuwna (*gaAdarat + gaAdaruwA + *gaAdarona) ZuhoraAF*
The-observers:mp (*left:fs + left:mp + *left:fp) at-mid-day
'The observers left at mid-day'

7.2. Non-human nouns

With non-human nouns, agreement rules are slightly different, but they still discriminate between BPs and suffixal plurals. Both types of plural can agree with an adjective or participle in the feminine singular, but only suffixal plurals can agree with an adjective or participle in the plural (*:fp* code below):

إستعملت (*المعاول + الحلقات) الصالحات
*IistaEomaltu Al-(*maEaAwilu + HalaqaAtu) SaAliHaAtun*
I used the-(*mattocks:q + rings:fp) good:fp
'I used the good (mattocks + rings)'

A dozen non-human nouns with BP, often denoting female animals, are exceptions to this rule and can agree with an adjective or participle in the plural.

7.3. Codification

The formalization of agreement rules in parsers and generators requires discrimination between the BP and suffixal plural of Arabic nouns. We opted for the straightforward solution of distinguishing two values for number, *q* and *p*. Taking into account the singular and the dual, our morpho-syntactic model of Arabic totals 4 values for number of nouns and adjectives. The MAGEAD system (Altantawy *et al*., 2011) has 3 values for number: singular, dual and plural. The Smrz (2007) parser has 3 values also.

We lack bases to define the gender of a BP. Broken plural shows no morphological difference in gender, even when the singular does: *qaAoDiy* 'male judge' and *qaAoDiyap* 'female judge' have the same BP *quDaAop* 'male or female judges or both'. Rules of agreement of a human BP with adjectives in the suffixal plural: <*A:mp*>, <*A:fp*>, or with verbs in the plural, depends on the sex of the referent. In the case of a non-human BP, an agreeing adjective is obligatorily in the feminine singular. Thus, our model represent BPs without any gender, tagging them as <*N:q*>.

8. Clitic-related spelling variants

In Arabic, a token can be analysed as a sequence of segments. Each segment in a token is a morpheme. A nominal token may contain a single morpheme <*N*>, or the concatenation of up to 5 morphemes as in:

<*CONJC*> <*PREP*> <*DET*><*N*> <*PRO+Gen*>

where <*CONJC*> is a coordinating conjunction, <PREP> a preposition, <*DET*> the determiner *Al-*, and <*PRO+Gen*> a pronoun in the genitive. The combination of morphemes obeys a number of constraints. A

<*PREP*> constrains the noun to be in the genitive case.[23] The presence of a clitic, graphically agglutinated <*PRO+Gen*> constrains another inflectional feature of the noun, definiteness, to have the construct-state value, while two other values, definite and indefinite, are possible otherwise. By checking such constraints, wrong segmentations can be discarded.

8.1. Segmentation

With the Unitex system, we represent nouns with four inflectional features: gender (masculine, feminine), number (singular, dual, suffixal plural, BP), definiteness (definite, indefinite, construct-state) and case (nominative, accusative, genitive). The segmentation into morphemes is performed with the aid of graphs. The output of this process is saved in the text automaton as in Fig. 1.

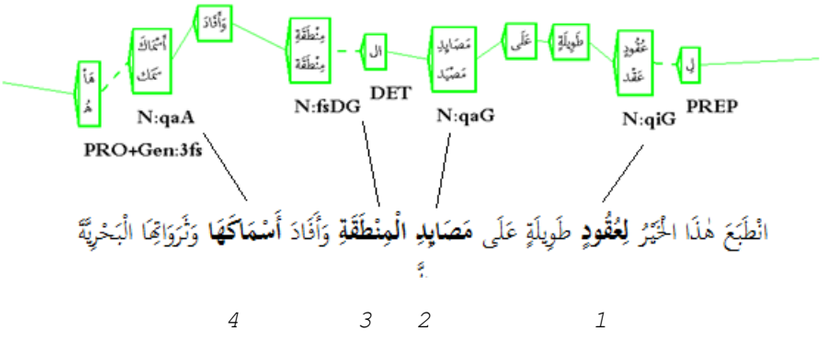


Fig. 1. Nouns tagged in text. Text automaton resulting from the application of graphs of morphological segmentation. Dashed lines connect segments inside the same token.

The sequence displayed in Fig. 1 contains 4 nouns, among which 3 BPs:

No.		Token	Lexical item
1	BP	*li_Euquwd-K*	*Eaqod,FvEvL-FuEuuL-123*
2	BP	*maSaAyid*	*maSoyad,FvEvLvB-FaEaaLiB-1234*
3	sing.	*Al_minoTaqap-i*	*minoTaqap,FvEvLvB-FaEaaLiB-1234*
		(This singular noun is labelled by the analyser since it admits a BP)	
4	BP	*OasmaAk-i_haA*	*samak,FvEvL-OaFoEaaL-123*

Dashed lines connect segments inside the same token. Abbreviations read as follows: PREP (preposition), DET (determiner), PRO (pronoun), Gen (genitive). Genders: **m**asculine, **f**eminine. Numbers: **s**ingular, **d**ual, suffixal **p**lural, **q** for broken plural. Definitenesses: **D**efinite, **i**ndefinite, and **a** for construct-state. Cases: **N**ominative, **A**ccusative, **G**enitive.

8.2. Orthographic adjustments

Most inflected noun forms are insensitive to graphically agglutinated pronouns, but some forms undergo an orthographic adjustment, e.g. forms with the suffix *-ap* or ending with a glottal stop. The suffix *-ap* is realised as its allograph *-at-*. In the full-form dictionary, those morphological variants that combine with the pronoun are marked as <*N+pro*>. Segmentation graphs select the <*N+pro*> variants from the dictionary. Fig. 2 shows the text automaton resulting from the morphological analysis of *OanoMiTatihaA* 'its activities':

No.	Token	Lexical item
1 BP	*OanoMiTa<u>t</u>-i-haA*	*naMaAT,FvEvvL-OaFoEiLap-123*

[23] <*CONJC*> combines freely with any inflected noun.

The segmentation graph checks that the agglutinated variant is marked as <*N+pro*> in the dictionary. Dashed lines connect segments inside the same token.

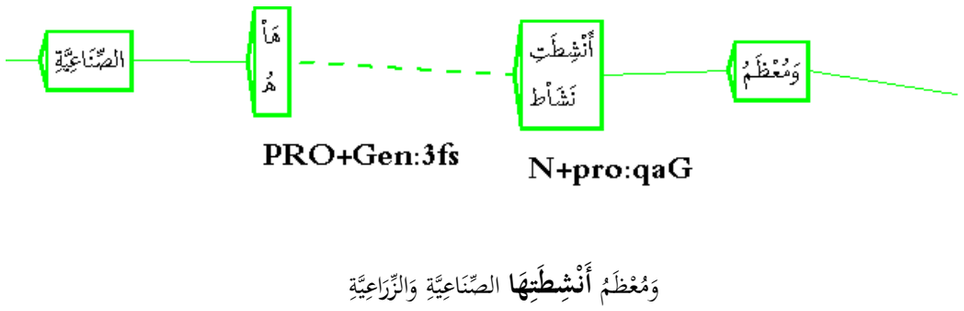


Fig. 2. Text automaton resulting from morphological segmentation.

The generation of the orthographically adjusted variants of an inflected noun is performed directly during the compilation of the dictionary of word forms. This process applies rules of orthographical variation, but makes use of lexical information encoded in entries. During analysis, the segmentation graph links each morphological variant to the correct context: again, this process implements rules, but takes advantage of formalized lexical information. The variants are generated during the compilation of the resources, not at analysis time as in rule-based systems in which a rule should compute each morphological variant at run time, then link each variant to the correct context. Our method simplifies and speeds up the process of annotation.

The system generates the inflected forms with the aid of an inflectional transducer (Fig. 3), as in Silberztein (1998). This transducer invokes sub-graphs; one of them, displayed in Fig. 4, specifies the generation of the orthographically adjusted construct-state variants (with the form *-at-* of the suffix) of an inflected form. The generation is performed during the compilation of the dictionary.

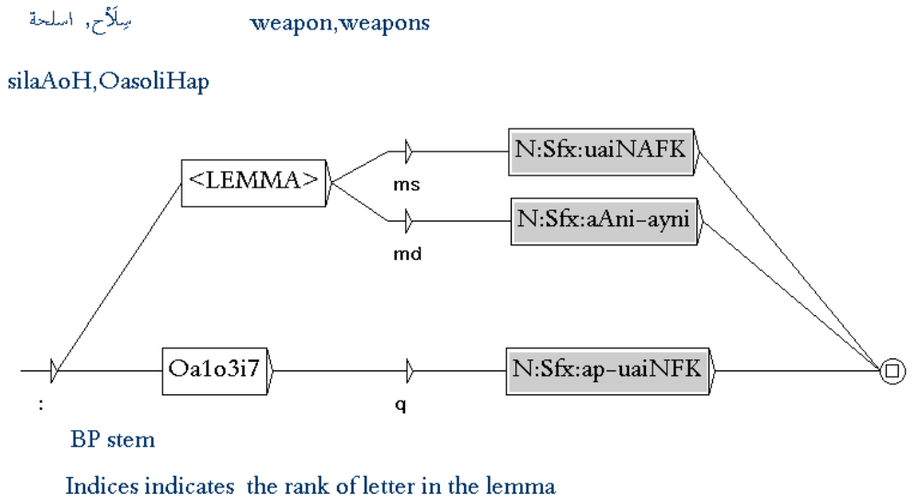


Fig 3. Inflectional transducer N300-m-FvEvvL-OaFoEiLap-123. Each path contains a stem pattern and a call to a subgraph of suffixes for definiteness and case variations (3×3).

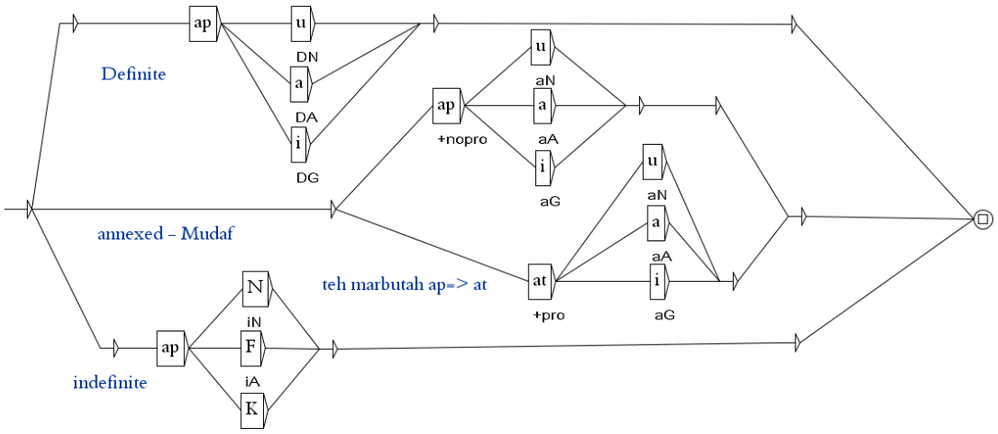


Fig. 4. Subgraph ap-uaiNFK represents definiteness/case suffix variations for nouns ending with the suffix *-ap*.

9. Evaluation

Since our BP lexicon is partial, we have chosen to measure its lexical coverage, and the feasibility of the extension of lexical coverage.

9.1. Corpus

We used a small sample of the NEMLAR Arabic Written Corpus (Attia *et al.*, 2005). This corpus was produced and annotated by RDI, Egypt, for the Nemlar Consortium.[24] During the construction of our lexicon of BPs, we did not use any part of the corpus: our sources of information were handbooks, reference dictionaries and native speaker competence. Thus, the evaluation tool is independent from the evaluated resource.

We selected three documents totalling 3 550 tokens (about 10 pages) and containing scientific popularization about three topics: pollution and fishing in Egypt, earthquakes in the world, and quality of water. We used the documents in the fully diacritized version.[25]

9.2. Coverage

We have extracted manually 388 occurrences of plural nouns and adjectives: 267 BPs and 121 suffixal plurals, among which 8 in the masculine and 113 in the feminine. Our lexicon (3 198 entries with BP) covered 195 occurrences out of the 267, i.e. 73% of occurrences. The sample did not contain any adjective in the BP.

The 195 covered occurrences of BPs are forms of 84 different lemmas of nouns, while the 72 remaining occurrences are forms of 25 lemmas of nouns: the lexicon covered 77% of the lemmas in the sample.

The 267 occurrences of BPs belong to 33 different inflectional classes, which had all been encoded in the system before evaluation. During the evaluation experiments, 5 descriptions of classes were found to contain errors affecting the recognition or tagging of forms. Therefore, the system covered 100% of the inflectional classes relevant for the sample, and 85% of them without errors.

	Sample	Covered	Coverage
Occurrences	267	195	73%
Lemmas	109	84	77%
Inflectional classes	33	33	100%

[24] It consists of about 500 thousand words of Arabic text from 13 different genres. Each text is provided in 4 different versions: raw text, fully diacritized text, text with Arabic lexical analysis, and text with Arabic POS-tags.
[25] The annotated corpus (10 pages) will be freely available in a file named *Fishing-Earthquakes-Water.txt* in the Unitex/Arabic/Corpus folder.

BP occurrences make up 7.5% of all tokens of the sample, but 69% of all occurrences of plural nouns and adjectives, a surprisingly high proportion. In order to check this point, we made another study with another document from the Nemlar corpus, belonging to another genre: a 2 510-token biographical text (4 pages) by Tawfiq Hakim, an Egyptian playwright. We counted 158 BP occurrences, which make up 6.3% of all tokens, and 73% of the 216 plural nouns and adjectives.

Thus, in spite of the fact that BPs are irregular, their presence in Arabic text is predominant over suffixal plurals. To our knowledge, this quantitative predominance had not been discovered before.

Among the 267 BP occurrences, 170 occurrences (64%) are graphically agglutinated with other segments and 97 are not. This means that graphical agglutination affects nouns in a massive way.

9.3. Feasibility of the extension of lexical coverage

The 72 occurrences of BP missing in the lexicon were analysed as forms of 25 distinct lemmas, for which 25 new entries were inserted. All new entries were assigned to already encoded inflectional classes. The new entries were tested by compiling the lexicon and tagging the evaluation corpus. The description of one of the classes had to be corrected because of a filename error. The analysis, encoding, testing and correction required 4 hours' work.

This experiment validates the feasibility of a comprehensive BP lexicon on the basis of the PRIM model.

The following list is a part of a concordance of the 267 occurrences of BPs in the evaluation corpus. It has been produced after lexicon update, by submitting the <*N:q*> lexical mask to Unitex:

اَلنَّيْتُرُوجِينِيَّةِ وَالْفِسْفُورِيَّةِ وَالْمَوَادِّ الْعُضْوِيَّةِ الْأُخْرَي الَّتِي
عُضْوِيَّةِ الْأُخْرَي الَّتِي تُعْتَبَرُ غِذَاءً لِلْأَسْمَاكِ وَالَّتِي تَفِدُ إِلَي الْبَحْرِ
الْبَحْرِ وَالْبُحَيْرَاتِ مَعَ مِيَاهِ النِّيلِ وَالْمَصَارِفِ وَالْقَنَوَاتِ الزِّرَاعِيَّةِ
الْآدَمِيَّةِ الَّتِي تَذْهَبُ لِلْبَحْرِ أَمَامَ سَوَاحِلِ الدَّلْتَا وَمَصَايِدِهَا مُنْذُ بِدَايِ
تَذْهَبُ لِلْبَحْرِ أَمَامَ سَوَاحِلِ الدَّلْتَا وَمَصَايِدِهَا مُنْذُ بِدَايَةِ الثَّمَانِينِيَّات
دُ بِدَايَةِ الثَّمَانِينِيَّاتِ .وَهٰذِهِ الْأَسْبَابُ هِيَ :أَوَّلًا :زِيَادَةُ النَّع
حُوظُ فِي حَجْمِ وَمِسَاحَةِ شَبَكَاتِ الْمِيَاهِ وَالْمَجَارِي وَمَحَطَّاتِ الصَّرْفِ الْآدَمِيِّ
وَمَحَطَّاتِ الصَّرْفِ الْآدَمِيِّ بِخَاصَّةٍ فِي مُدُنِ الْقَاهِرَةِ وَالْإِسْكَنْدَرِيَّةِ خِلَالَ
حَقِيقَةٌ أَمْ وَهْمٌ؟ !لَمْ يَتَّفِقْ بَعْضُ الْعُلَمَاءِ مَعَ رَأْيِ نِكْسُونْ وَمِنْهُمْ
لْمِصْرِيُّ الدُّكْتُورُ يُوسُفُ حَلِيمِ بِقِسْمِ عُلُومِ الْبِحَارِ بِجَامِعَةِ الْإِسْكَنْدَرِيَّة
يُّ الدُّكْتُورُ يُوسُفُ حَلِيمِ بِقِسْمِ عُلُومِ الْبِحَارِ بِجَامِعَةِ الْإِسْكَنْدَرِيَّةِ، حَيْث
فِي الْمَقَامِ الْأَوَّلِ إِلَى زِيَادَةِ عَدَدِ سُفُنِ وَمَرَاكِبِ الصَّيْدِ وَتَحَسُّنِ كَفَاءَةِ
مَقَامِ الْأَوَّلِ إِلَى زِيَادَةِ عَدَدِ سُفُنِ وَمَرَاكِبِ الصَّيْدِ وَتَحَسُّنِ كَفَاءَةِ الْمُعَد
الصَّيْدِ وَتَحَسُّنِ كَفَاءَةِ الْمُعَدَّاتِ وَالْأَجْهِزَةِ الْمُسْتَخْدَمَةِ فِي عَمَلِيَّاتِ
زِيَادَةِ نِسْبَةِ الْهَائِمَاتِ النَّبَاتِيَّةِ وَالْمَوَادِّ الْعُضْوِيَّةِ .كَمَا يُؤَخَذُ
مْلَةِ الْمُلَوِّثَاتِ الَّتِي ذَكَرَتْهَا بَيْنَ الْمَوَادِّ الْعُضْوِيَّةِ الْمَوْجُودَةِ فِي الْأ
الْمَوَادِّ الْعُضْوِيَّةِ الْمَوْجُودَةِ فِي الْأَسْمِدَةِ وَالْمُخَلَّفَاتِ الْآدَمِيَّةِ وَب

In order to investigate the feasibility of the extension of lexical coverage beyond BPs and verbs (Neme, 2011), we inserted in the lexicon 750 items for all the words occurring in the evaluation corpus and not found in the lexicon. We encoded 52 inflectional classes for suffixal plural nouns, suffixal plural adjectives, grammatical words and for 2 classes of verbs missing in Neme (2011). The encoding and the testing/correction loop required 60 hours' work. After this extension, the evaluation corpus was entirely covered.

This experiment validated our intuition that, besides verb conjugation and BPs, Arabic morpho-syntactic tagging does not pose any serious challenges to resource-based language processing.

Conclusion

By keeping inflection apart from derivational morphology and dealing with morphophonological alternations in a factual way, the PRIM model simplifies the encoding of BP. Its strong points can be summed up as follows:

1. It complies with the conventions in traditional morphology that we found useful to noun inflection, in particular with most of the traditional patterns in the sense of Semitic morphology. Thus, the PRIM language resources can be easily updated by Arabic-speaking linguists in order to extend lexical coverage and control the evolution of the accuracy of systems that use them. We have dropped conventions related to semantic description.

2. The updatable lexicon is structured in lexical entries, as traditional dictionaries, and not in stem entries, as in the multi-stem approach.

3. Inflected forms are generated from their observable surface lemma, and not from a deep root.

4. The pattern of a singular noun is abstracted from the stem without gender or number suffixes, and without definiteness and case markers. The pattern of a BP is abstracted from the stem without definiteness or case markers.

5. The taxonomy of singular patterns specifies vowel quantity, noted as *v* or *vv*, but ignores vowel quality and derivational history.

6. Patterns are not used to represent morpho-syntactic features in lexical tags. Lexical tags are accurate and informative and consist of a lemma and a set of feature-value pairs, generally gender, number, definiteness and case.

7. Root alternations are encoded independently from patterns. They are explicitly represented as separate pieces of lexical information, instead of being obtained through the interaction of a deep level with general rules. They are encoded as mappings from the surface root of the singular to the surface root of the plural. Orthographical variations of the glottal stop are encoded in the same way.

8. Root letter substitutions and insertions are restricted to *w*, *y*, *A*, to allographs of the glottal stop, and to copies of root letters available in the lemma.

9. The PRIM taxonomy for noun inflection is simple, orderly and detailed. The number of classes, including suffixal plural and BP, is smaller than for Brazilian Portuguese.

10. A transducer corresponds to each inflectional class of nouns, and generates all the inflected forms of any lemma in the class. Transducers are edited in graphical form with the Unitex system, and handle roots in Semitic languages straightforwardly. They can be quickly corrected when an error is detected.

11. Morphological analysis of Arabic text is performed directly with a dictionary of words and without morphological rules, which simplifies and speeds up the process.

12. Agglutinated clitics are analysed without generation of artificial ambiguity. Clitic agglutination is described independently from inflection, in separate graphs.

13. The PRIM model is compatible with solutions to the other challenges to Arabic processing: verb conjugations, including alternations of *w*, *y*, *A* and the glottal stop (Neme, 2011); recognition of partially diacritized text with fully diacritized resources, excluding incompatible analyses.

Our distinctive approach consists in considering language resources as the key point of the problem. We integrate all complex operations among resource management operations.

Bibliography

Abdel-Nour, Jabbour (2006). *Dictionnaire Abdel-Nour al-Mufassal Arabe-Français*. Dar El-Ilm Lil-Malayin. 10th edition. 2034 pages, 3 columns.

Altantawy, Mohamed; Habash, Nizar; Rambow, Owen (2011). Fast Yet Rich Morphological Analysis. In *Proceedings of the 9th International Workshop on Finite State Methods and Natural Language Processing* (FSMNLP), pages 116-124.

Altantawy, Mohamed; Habash, Nizar; Rambow, Owen; Saleh, Ibrahim (2010). Morphological Analysis and Generation of Arabic Nouns: A Morphemic Functional Approach. In *Proceedings of the Language Resource and Evaluation Conference* (LREC), Malta, pages 851-858.

Attia., M., Yaseen., M., Choukri., K. (2005). *Specifications of the Arabic Written Corpus produced within the NEMLAR project*, www.NEMLAR.org.

Beesley, Kenneth R. (1996). Arabic finite state morphological analysis and generation. In *Proceedings of the International Conference on Computational Linguistics* (COLING), Copenhagen, Center for Sprogteknologi, volume 1, pages 89-94.

Beesley, Kenneth R. (2001). Finite-State Morphological Analysis and Generation of Arabic at Xerox Research: Status and Plans in 2001. In *Proceedings of the ACL/EACL Workshop 'Arabic Language Processing: Status and Prospects'*, pages 1-8.

Boudlal, Abderrahim; Lakhouaja, Abdelhak; Mazroui, Azzeddine; Meziane, Abdelouafi (2010). Alkhalil Morpho SYS1: A Morphosyntactic Analysis System for Arabic Texts. *International Arab Conference on Information Technology* (ACIT).

Brame, M. (1970). *Arabic Phonology: Implications for Phonological Theory and Historical Semitic*, unpublished Ph.D. dissertation, MIT.

Buckwalter, Timothy (1990). Lexicographic notation of Arabic noun pattern morphemes and their inflectional features. In *Proceedings of the Second Cambridge Conference on Bilingual Computing of Arabic and English*. 7 pages.

Buckwalter, Timothy. *Arabic Morphological Analyzer* Version 1.0. (2002). LDC Catalog No.: LDC2002349.

Buckwalter, Timothy (2007). Issues in Arabic Morphological Analysis. In Antal van den Bosch and Abdelhadi Soudi (eds.), *Arabic Computational Morphology. Knowledge-based and Empirical Methods*. Text, Speech and Language Technology, volume 38, Berlin: Springer, pages 23-41.

Courtois, Blandine (1990). Un système de dictionnaires électroniques pour les mots simples du français, *Langue Française* 87, Paris: Larousse, p.11-22.

El-Dahdah Antoine (1992), *A dictionary of universal Arabic Grammar*. Librairie du Liban Publishers. Bilingual, 250 p. in Arabic/250 p. in English.

Daille, Béatrice; Fabre, Cécile; Sébillot, Pascale (2002). Applications of Computational Morphology. In Boucher, Paul (ed.), *Many Morphologies*, Somerville: Cascadilla Press, p. 210-234.

Al-Fairuzabadi (v. 1400), *Al-Qamus al-Muhit* (Comprehensive Dictionary). Ed. 2007, Beirut: Dar Al Kotob Al Ilmiyah, 1440 pages.

Ferrando, Ignacio (2006). The plural of paucity in Arabic and its actual scope. On two claims by Siibawayhi and al-Farraa'. In: Boudelaa, Sami (ed.), *Perspectives on Arabic Linguistics*, XVI, Current Issues in Linguistic Theory, 266, Amsterdam/Philadelphia: Benjamins, p. 39-61.

Gaber, Gaber Meftah (2012). *An Optimality Theory Account of the Non-concatenative Morphology of the Nominal System of Libyan Arabic, with Special Reference to the Broken Plural*, PhD Dissertation, Durham University, http://etheses.dur.ac.uk/3511/

Al-Ghalāyini, Mustafa (2007). *Jāmi3 al-durūs al-'arabiyah* (A university grammar textbook). 1st edition 1912. Dar El Fikr Printers-Publishers, Beirut. 570 pages. In Arabic.

Gross, Maurice (1975). *Méthodes en syntaxe. Régime des constructions complétives*. Paris: Hermann.

Habash, Nizar; Rambow, Owen (2006). MAGEAD: A Morphological Analyzer and Generator for the Arabic Dialects. *In Proceedings of the International Conference on Computational Linguistics and Annual Meeting of the Association for Computational Linguistics* (COLING-ACL), Sydney, Australia, pages 681–688.

Habash, Nizar (2010). *Introduction to Arabic Natural Language Processing*. Morgan & Claypoll Publishers.

Al-Hadithy, Khadija (2003). *Morphological forms in the Sibawayh's Kitāb. A dictionary and a study*. Republication of the dissertation in the Master of Arts, Faculty of Literature in Cairo, first published in 1961. 370 pages. In Arabic.

Huh, Hyun-Gue; Laporte, Éric (2005). A resource-based Korean morphological annotation system. In *Proceedings of the International Joint Conference on Natural Language Processing* (IJCNLP), Jeju, Korea.

Ibn Manzur (1290). *Lisān al-ʿArab* (The Arabic Language). Ed. 1955-1956, Beirut: Dar Sadir, 15 volumes.

Kihm, Alain (2006). Nonsegmental concatenation : a study of Classical Arabic broken plurals and verbal nouns, *Morphology* 16, 69-105.

Kiraz, George Anton (1994). Multi-tape Two-level Morphology: A Case study in Semitic Non-Linear Morphology. In *Proceedings of the International Conference on Computational Linguistics* (COLING), Kyoto, Japan, pages 180–186.

Kiraz, George Anton (2001). *Computational Nonlinear Morphology, with Emphasis on Semitic Languages* Cambridge, U.K.: Cambridge University Press. Studies in Natural Language Processing, 171 pages.

Levy, M. M. (1971). *The plural of the nouns in Modern Standard Arabic*, PhD dissertation, University of Michigan.

McCarthy, J. J. (1981). A prosodic theory of nonconcatenative morphology. *Linguistic Inquiry* 12, 373-418.

McCarthy, J. J. & Prince, A. S. (1990). Foot and word in prosodic morphology: the Arabic broken plural. *Natural Language and Linguistic Theory* 8(2), 209–283.

Muniz, Marcelo C.M.; Maria das Graças V. Nunes; Éric Laporte (2005). UNITEX-PB, a set of flexible language resources for Brazilian Portuguese. In *Proceedings of the TIL Workshop*. pages 2059–2068.

Neme, Alexis (2011) A lexicon of Arabic verbs constructed on the basis of Semitic taxonomy and using finite-state transducers. In *Proceedings of the International Workshop on Lexical Resources* (WoLeR) at ESSLLI.

Paumier, Sébastien. (2011). *Unitex - manuel d'utilisation 2.1* , Université Paris-Est Marne-la-Vallée.

Ratcliffe, Robert R. (1998), *The 'broken' plural problem in Arabic and comparative Semitic: allomorphy and analogy in non-concatenative morphology*, John Benjamins, Foreign Language Study, 261 pages.

Ratcliffe, Robert R. (2001), Analogy in Semitic morphology: where do new roots and new patterns come from? In Zaborsky, Andrzej (ed.), *New Data and New Methods in Afroasiatic Linguistics. Robert Hetzron in memoriam*, Wiesbaden, Harrassowitz, p. 153-162.

Ryding C. Karin (2005), *A Reference Grammar Of Modern Standard Arabic*, Cambridge University Press, 708 pages.

Sibawayh (arround 800 CE), *Kitaabu Siibawayhi 'Abii Bišrin `Amri bni `Utmaana bni Qunbur*, ed. S. M. Haaruun (1977, 2ª), Cairo, 5 vols.

Silberztein, Max (1998). INTEX: An integrated FST toolbox. In Derick Wood, Sheng Yu (eds.), *Automata Implementation*, p. 185-197, Lecture Notes in Computer Science, vol. 1436. Second International Workshop on Implementing Automata, Berlin/Heidelberg: Springer.

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

Soudi, Abdelhadi; Cavalli-Sforza, Violetta; Jamari, Abderrahim (2002), The Arabic Noun System Generation.

Tarabay, Adma (2003). *A dictionary of Arabic plurals*. Librairie du Liban Publishers. 590 pages. In Arabic.

Zbib, Rabih; Soudi, Abdelhadi (2012). Introduction. Challenges for Arabic machine translation.In Soudi, Abdelhadi; Farghaly, Ali; Neumann, Günter; Zbib, Rabih (eds.), *Challenges for Arabic Machine Translation*, Natural Language Processing, 9, Amsterdam: Benjamins, p. 1-13.

A lexicon of Arabic verbs constructed on the basis of Semitic taxonomy and using finite-state transducers

Alexis Amid Neme

A lexicon of Arabic verbs constructed on the basis of Semitic taxonomy and using finite-state transducers

Alexis Amid Neme

Laboratoire d'informatique Gaspard-Monge – LIGM
Université Paris-Est, 77454 Marne-la-Vallée Cedex 2, France.

http://infolingu.univ-mlv.fr

E-mail: alexis.neme@gmail.com

Abstract

We describe a lexicon of Arabic verbs constructed on the basis of Semitic patterns and used in a resource-based method of morphological annotation of written Arabic text. The annotated output is a graph of morphemes with accurate linguistic information. An enhanced FST implementation for Semitic languages was created. This system is adapted also for generating inflected forms. The language resources can be easily updated. The lexicon is constituted of 15 400 verbal entries.

We propose an inflectional taxonomy that increases the lexicon readability and maintainability for Arabic speakers and linguists. Traditional grammar defines inflectional verbal classes by using verbal pattern-classes and root-classes, related to the nature of each of the triliteral root-consonants. Verbal pattern-classes are clearly defined but root-classes are complex. In our taxonomy, traditional pattern-classes are reused and root-classes are simply redefined.

Our taxonomy provides a straightforward encoding scheme for inflectional variations and orthographic adjustments due to assimilation and agglutination. We have tested and evaluated our resource against 10 000 diacriticized verb occurrences in the Nemlar corpus and compared it to Buckwalter resources. The lexical coverage is 99.9 % and a laptop needs two minutes in order to generate and compress the inflected lexicon of 2.5 million forms into 4 Megabytes.

1. Introduction

Arabic morphology can be described by many formal representations. However, Semitic morphology or *root-and-pattern* morphology (Kiraz, 2004) is a natural representation for Arabic [1]. The *root* represents a morphemic abstraction, usually for a verb a sequence of three consonants, like *ktb*. A *pattern is* a template of characters surrounding the root consonants, and in which the slots for the root consonants are shown by indices. The combination of a root with a pattern produces a surface form. For example, *kataba* and *yakotubu* are represented by the root *ktb* and the patterns *1a2a3a* or *ya1o2u3u*.

Root-and-pattern morphology is standard in Arabic and is learned in grammar text books. Arabic linguists use *root-and-pattern* representation in order to list verbal entries and related inflected forms. On the other hand, FSTs have shown their simplicity and efficiency in inflectional morphology for western languages. Computer scientists appoint FSTs as standard devices for inflection. Various formal representations for Arabic morphology have been created by computer scientists to avoid root-and-pattern representation. The point that motivated this trend is that FSTs formalism would not be fitted for Semitic morphology since FSTs are concatenative whereas Semitic morphology is not. In concatenative representation, the root-and-pattern representation is replaced by a stem- or lexeme-based representation. For these formalisms, a stem is a basic morpheme that undergoes affixations with other morphemes in order to form larger morphological or syntactic units. For root-and pattern morphology, a stem derives from a root and a particular pattern and subsequently undergoes affixations.

At the operational level, the lexical representation of the concatenative model is entirely concatenative in order to compel with the *[prefix][stem][suffix]* representation. However, these representations imply a manual stem precompilation based on a root-and-pattern representation. The concatenative models are generally composed of three components: lexicon, rewrite rules, and morphotactics. The lexicon consists of multiple sublexica, generally *prefix*, *stem*, and *suffix*. The rewrite rules map the multiple lexical representations to a surface representation. The morphotactics component aims with a subjacent representation to generate or to parse the surface form *[prefix][stem][suffix]* and performs alternation rules at morpheme boundaries such as deletion, epenthesis, and assimilation.

Any formal representation that is not adapted to Semitic morphology will be rejected by the majority of Arabic-speaking linguists. When linguists work in a newly created formalism, they continue to work with *root-and-pattern* representation on paper and subsequently, they unfold their descriptions for a specific formalism. Their contribution for updating and correcting lexical resources is complex and time-consuming, and therefore error-prone.

Our approach resorts to classical techniques of lexicon compression and lookup in an inflected full-form dictionary that includes orthographic variations related to morpheme agglutination. The formalization of all possible verbal tokens requires complex and interdependent rules. For these issues, we define a taxonomy for Arabic verbs composed of 460 inflectional

1 We would like to thank Eric Laporte and Sébastien Paumier for helpful discussions, contributions and for the adaptation of Unitex to Arabic. Unitex is an open source multilingual corpus processor. More than 12 European languages, Korean and Thai with their linguistic resources are operational in Unitex. http://www-igm.univ-mlv.fr/~unitex

classes. We demonstrate that FSTs are compatible with root-and-pattern representation. Our taxonomy encodes simultaneously in the lexical representation three variations at the surface level:
- inflectional classes of a lemma;
- inflectional subclasses related to morphophonemic assimilation;
- orthographic adjustments related to the agglutination of a pronoun.

In our orthographic representation, we use a fully diacriticized lexicon and we take advantage of the clear boundary, already defined in traditional grammar, between verbal inflection and verbal agglutination to describe these two levels independently. In order to satisfy both computer scientists and Arabic linguists, we have created in Unitex an enhanced version of FSTs adapted to root-and-pattern representation.

In Section 2, we outline the state-of-the-art approaches to Arabic morphological annotation. Section 3 describes the methodology and particularly the inflectional verbal taxonomy. Section 4 describes agglutination as morpheme combinatorics. Section 5 reports the construction of the lexicon. Section 6 reports the evaluation of the lexicon. A conclusion and perspectives are presented in Section 7.

2. State of the Art

Several morphological annotators of Arabic are available. The Buckwalter Arabic Morphological Analyzer (BAMA) is one of the best Arabic morphological analysers and is available as open source. The BAMA uses a concatenative lexicon-driven approach where morphotactics and orthographic adjustment rules are partially applied into the lexicon itself instead of being specified in terms of general rules that interact to realize the output (Buckwalter, 2002).

The BAMA has three components: the lexicon subdivided in A, B, C sublexica, the compatibility tables (AB, BC, AC) and the analysis engine. An Arabic word is viewed as a concatenation of three regions, a prefix region (A), a stem region (B) and a suffix region (C). The prefix and suffix regions can be null. An entry in A may be the concatenation of proclitics and an inflectional prefix. An entry in C may be the concatenation of an inflectional suffix and an enclitic. The A and C lexica are composed of 561 and 989 entries which represent all possible combinations of inflectional and agglutinative morphemes for nouns and verbs. For each stem in B, a morphological compatibility category, an English gloss and part-of-speech (POS) data are specified. A list of stems is assigned to a lemma, and the lemma is not used in the analysis process. The B lexicon is composed of 82 000 stems which represent nearly 40 000 lemmas. Verbal stems are 33393[2] and represent 8709 verbal lemmas. A full ABC form must be allowed by the three compatibility tables AB, BC, AC.

2 Verbal stems are for perfect active (17008) stems, imperfect active (13241), perfect passive (403), imperfect passive (2611), and for imperative 130 stems. BAMA resource does not include all imperfect active stems, for instance.

qr>	qara>	PV->	qara>/VERB_PERFECT
qr|	qara|	PV-|	qara|/VERB_PERFECT
qr&	qara&	PV_w	qara&/VERB_PERFECT
qr>	qora>	IV	qora>/VERB_IMPERFECT
qr>	qora>	IV_wn	qora>/VERB_IMPERFECT
qr|	qora|	IV-|	qora|/VERB_IMPERFECT
qr&	qora&	IV_wn	qora&/VERB_IMPERFECT
qr}	qora}	IV_yn	qora}/VERB_IMPERFECT
qr>	qora>	IV_Pass	yuqora>/VERB_IMPERFECT

Table 1. BAMA stem lexicon using Buckwalter transliteration. A list of stems related to the lemma-identifier qara>-a_1 "to read". The 9 stems are related to the orthographic variants of the 3rd root consonant, here glottal stop (*hamza*), depending on the next inflectional suffix and the existence of an agglutinated pronoun.

The Buckwalter representation for the Arabic lexicon is not fitted for generation but only for text analysis. In ElixirFM (http://elixir-fm.sourceforge.net/), Smrz (2007) adapted the Buckwalter resources for generation and the project is implemented in Haskell, a functional programming language. In the ALMORGEANA project, Habash (2004) proposed also a version of Buckwalter resources adapted to generation and analysis. Below an example *lilkutubi* "books" :

lilkutubi⇔ [kitAb-1 POS: N l+ Al+ +PL +GEN]
li_l_kutub-i⇔
[lemma-ID NOUN PREP+DET+ (plustem) + Genitive]

Although the lexicon is an open linguistic resource, the procedure for updating it is complex. For instance, adding a new verb is an intricate operation. First, the A and C lexica are composed of 561 and 989 entries. Although the two disjoint sets of inflectional and agglutination suffix morphemes are clearly defined in Arabic, the *[prefixes] [stem][suffixes]* representation does not allow two suffix subsets to be defined. Second, the stem lexicon entries corresponding to a lemma are numerous and need to be subcategorized. In other words, a lemma is unfolded into many stems, and one uses a cumbersome subcategorization which mixes up inflectional and agglutinative features of verb stems in order to match with 3 compatibility tables, composed respectively of 2050, 1660, 1200 entries. Such composite data are complex and not transparent for Arabic linguists.
Mesfar (2008) adopts a "lemma-based lexicon" and FSTs for inflection. The project claims 10 000 verb lemmas. The framework is similar to ours since it resorts to classical techniques of lexicon compression and lookup in a full list of inflected -forms. The project does not use root-and-pattern representation. As far as we know, no figures on testing and evaluating the systems are available. The lemma lexicon is wordy such as the extract of the lexicon from Mesfar (2008):

```
ضَرَبَ,V+Tr+FLX=Vdaraba1+DRV=N_daraba1:FlxDRV+DRV=A_daraba1:FlxDRV
# le verbe "ذَكَرَ" et "كَتَبَ" se conjuguent et se dérivent selon les même modèles
ذَكَرَ,V+Tr+FLX=Vdakara2+DRV=N_dakara2:FlxDRV+DRV=A_dakara2:FlxDRV
كَتَبَ,V+Tr+FLX=Vdakara2+DRV=N_dakara2:FlxDRV+DRV=A_dakara2:FlxDRV.
```

FST are difficult to read and maintain (Mesfar, 2006, page 3):

> " أَلَّمَ ", V+Tr+FLX [8] = V_kallama (kallama – *to speak with someone*)
> Among the 122 inflectional transformations which are described in the flexional paradigm "V_kallama", here is one: (<LW> يُ <R4><S> <R><S> /ُ A+P+3+m+s). This NooJ transformation means: position the cursor (|) at the beginning of the form(<LW>) (|kallama), insert "يُ" (yu) into the head of the form (yu|kallama), skip four letters (<R4>) (yukall|ama), erase a letter (<S>) (yukall|ma), insert the vowel "ِ " (i) (yukalli|ma), skip a letter (<R>) (yukallim|a), delete of the following letter (<S>) (yukallim|)and finally insert the final vowel "ُ " (u) (yukallimu|)."

For their morpho-phonological system and in addition to concatenative rules, Carnegie Melon Univ. uses transformational rules to describe alternation of root letters (Cavalli-Sforza, et al., 2005). As far as we know, no figures on lexical coverage or evaluation are available.

The SARF project (Al-Bawab et al., 1994, http://sourceforge.net/projects/sarf/) is based on root-and-pattern representation. Starting from three- and four-consonant roots, it can generate Arabic verbs, derivative nouns, and gerunds, and inflect them. It has over 20 000 verb lemmas. The project uses conventional programming techniques with the Java language and roots encoded in XML files. It uses transformational rules in order to handle alternation of root letters in the Java programs. The patterns are hard-coded in the form of Java code. This work has the advantage of being clearly built on a strong linguistic basis that is the standard morphology in Arabic. However, it neither includes the use of a test collection nor reports a success rate; in addition, updating and correcting a language resource included in source code is complex since it involves two expertises: an Arabic linguist and a programmer; updating data and updating source code obey to different professional practices.

At Université de Lyon 2, the DIINAR project (Dichy & Ferghali, 2004) was developed for terminological and translation purposes. DIINAR.1 includes a total number of 119,693 lemmas, fully vowelled, among which 19,457 verb lemmas. A conventional programming framework and databases are used for generation and analysis with a lemma-based lexicon encoded according to this framework. As far as we know, no figures on testing and evaluating the system for morphological annotation are available.

For a complete survey of morphological parsers, readers should consider Al-Sughaiyer & Al-Kharashi (2004) and Habash (2010).

3. Method of description

3.1 A taxonomy for verb inflection

Our method is based on a precompiled diacriticized full-form dictionary with all possible inflected forms and their orthographic variations due to morphophonemic alternations. We exclude from this inflectional representation agglutinated prefixes and suffixes such as conjunctions and pronouns. We associate morphosyntactic feature values to each entry in the generated list of 2.43 million surface forms. In order to obtain this list, we provide a list of lemmas manually associated to codes defined by a taxonomy, each code representing a transducer. The full-form list is produced after inflecting each lemma by applying the encoded transducer (Silberztein, 1998).

Arabic and other Semitic languages have long been described in terms of a *root* interwoven with a *pattern*. The root is a sequence of consonants. Each Arabic verb contains 3 or 4 consonants that remain generally unchanged in all conjugated forms and make up the consonantal root; all the remaining information on a conjugated form is called 'pattern'. For example, *yakotubuwna = [ktb & ya1o2u3uwna]* is obtained through the interdigitation of the root *ktb* with the pattern of active-Perfect-3person-masculine-plural-indicative *ya1o2u3uwna*. Below some precisions:

- Some root consonants change. They are the glottal stop, noted *h* in the taxonomy, and glides, noted *w*, *y*; those that never change are written in patterns in the form of their position 1, 2, 3 or 4.
- At the surface level, the orthographic representation of glottal stop and glides can change. The glottal stop is represented by six allographs depending on the context. At phonological level, the glides become short vowels */ i, u/* or long vowels */a:*, *i:*, *u:/* or are omitted and transcripted as *zero-vowel*, *o*[3] (see also footnote 4).
- A pattern indicates the position of its letters relative to the root consonants. Generally, these letters are vowels and/or affixes related to derived verb form such as *IisotakotabuwA = [ktb & Iisota1o2a3uwA]*. The surface form may also be subdivided in *[prefix] [stem] [sufix]*. The *stem pattern* formalizes all infixation operations such as *kotub = [ktb & 1o2u3]*. Inflectional prefixes and suffixes can be concatenated subsequently to the stem form y*akotubuwna = [ya] [ktb & 1o2u3] [uwna]*.
- The third root consonant can be identical to the second one. In the root, it is represented by a gemination mark *G*, and in the pattern, by *2*, such as *madadota = [mdG & 1a2a2ota]*.
- By convention, the perfect-3rd person-masculine -singular is the form used as lemma. The corresponding pattern is called the canonical pattern. All patterns are defined in function of the canonical pattern.

Verbal pattern classes are clearly defined in Arabic grammar but root-classes are intricate and involve a complex terminology. Root-classes are defined according to the nature of some of the root consonants: regular, weak, geminated, with glottal stop, and to their position 1, 2, 3 or 4. In this terminology, *qaAla/yaquwlu* قال "say" is a *hollow verb of w kind*, with a weak consonant *w* at the second position; whereas *baAEa/yabiyEu* باع "sell" is a *hollow verb of y kind*. Moreover, two or three special values of the root consonants can appear at the same time. A verb like *OataY/yaOotiy* أتى "arrive" has a glottal stop at the first position and a weak consonant *y* at the third position. A classification with nature/position criteria and each with 4 sub-criteria yields to an intricate terminology and is not

3 The zero-vowel marks the absence of vowel between two consonants.

consensual in Arabic grammar.
Our classification is bi-dimensional like the traditional one and based on the traditional pattern-classes which are reused and root-classes which are redefined more simply. Traditional grammar defines an inflectional verbal class by a pattern-class and a root-class. Triliteral verbs are compatible with 16 possible canonical patterns and quadrilateral verbs with 4 canonical patterns. Our classification defines 31 root-classes. The root classes are defined according to the nature of the root consonants. The special values for the consonants are *w, y* and the glottal stop (h). An irregular root is a root with at least one special value in its consonants. The inflected forms of a verb are easily predictable on the basis of the features of the root. We revisited and simplified, with no loss of information, the root-based traditional classification by using three consonantic slots, noted *123*, except for special values: glottal stop (*h*), *w, y,* for each slot; and when the 3rd root consonant is identical to the 2nd, the slots are noted *122*. Thereby, the lemma *ktb* will be encoded *$V3au-123* where:

$ is the Semitic mode for FST which means the root consonants interdigitate into the pattern: *[ktb & ya1o2u3u] = yakotubu;*
V is the verbal POS;
3au is the class of triliteral verbs used with the patterns *1a2a3/ya1o2u3* for perfect/ imperfect;
123 is the class of roots in which no slot is occupied by a special value.

Each root/canonical-pattern pair corresponds to a lemma. This representation seems well-founded and also well-established in Arabic morphology. Above all, it is ubiquitous in the Arabic-speaking world. Below, some examples from the lexicon:

```
/Lemma,encoding/ canonical-patt. Special values
-----------------------------------------------
/ simple forms
نقش,$V3au-123 / 1a2a3a/ya1o2u3u no special values
جرّ,$V3au-122 /    third root identical to second
عاد,$V3au-1w3 /      with waw as a second root
غفا,$V3au-12w /      with waw as a third root
فتح,$V3aa-123/ 1a2a3a/ya1o2a3u
لمز,$V3ai-123 / 1a2a3a/ya1o2i1u
حاك,$V3ai-1y3 /   with yeh as a second root
سرى,$V3ai-12y /   with yeh as a third root
أوى,$V3ai-hwy /   with hamza, waw and yeh
علم,$V3ia-123 / 1a2i3a/ya1o2a3u
وطئ,$V3ia-w2h /   waw and hamza as 1rst and 3rd
كرُم,$V3uu-123 / 1a2u3a/ya1o2u3u
حسب,$V3ii-123 / 1a2i3a/ya1o2i3u
/ Derived forms
أقبل,$V61-123  / Aa1o2a3a
دشّن,$V62-123   / 1a2Ga3a
داهم,$V63-123  / 1aA2a3a
إنشغل,$V64-123 / Iino1a2a3a
إنطلى,$V64-12y /   with yeh as a third root
إختنق,$V65-123 / Ii1ota2a3a
إزهرّ,$V66-123  / Ii1o2a3Ga
تهاجن,$V67-123 / ta1aA2a3a
تآكل,$V67-h23  /   with  hamza as a first root
تحدّد,$V68-122  / ta1a2Ga2a with identical 3rd root
تلكّأ,$V68-12h  /   with hamza as a third root
إستبسل,$V69-123 / Iisota12a3a
اعشوشب,$V70-123 / Ii1o2aw2a3a
/ Quadriliteral roots
بعثر,$V40-1234  / 1a2o3a4a a quadriliteral root
طمأن,$V40-12h4  /   with hamza as a third root
دمدم,$V40-1212  /   a geminated quadriliteral root
تبعثر,$V41-1234 / ta1a2o3a4a
تلألأ,$V41-1h1h  /    a geminated root with 2 hamzas
```

Below, some of the 31 possible combinations of root-classes related to class-pattern V3ia. Some root-classes are empty which means that there is no verb with such root-classes for class-pattern V3ia:

```
/Lemma,encoding/  /lemma-transliteration
علم ,$V3ia-123   /Elm
ظل  ,$V3ia-122   /ZlG
أمَ  ,$V3ia-h22   /OmG
ألف ,$V3ia-h23   /Olf
رئف ,$V3ia-1h3   /ref
ظمئ ,$V3ia-12h   /Zme
//First weak root consonant
ودَ  ,$V3ia-w22   /wdG
    ,$V3ia-wh3
وطىء,$V3ia-w2h   /wTe
وجع ,$V3ia-w23   /wjE
    ,$V3ia-y22
يئس ,$V3ia-yh3   /yes
يقظ ,$V3ia-y23   /yqZ
```

The format of the lexicon is a list of lemma entries. In our format, the string before comma transcribes plain letters and the gemination mark but no short vowel diacritics. The pattern includes the encoding of short vowels (*a, i, u).* This transcript choice is consistent with usual practice in traditional paper dictionaries.
Our full-form lexicon is produced by FSTs. The FST output format is *surface-form,lemma.V:feature-values* such as *:*
كتب,تَـكْـتُـبُ.V:aI3fsN
/active-Imperfect-3rdpers-fem-sing-iNdicative

The *feature values* are :
- Voice: active (a), passive (b);
- Tense: Perfect, Imperfect, Imperative (Y);
- Person: 1, 2, 3;
- Gender: masculine, feminine;
- Number: singular, dual, plural;
- Mode: indicative (N), Subjunctive, Jussive, Energetic.

In the following two sub-sections, we present first inflectional transducers and then inflection-related orthographic adjustments.

3.2 The inflection transducers

An inflection transducer specifies the inflectional variations of a word. It is shared by the class of words that inflect in the same way. The input parts of the transducer encode the modifications that have to be applied to the canonical forms. The corresponding output parts contain the codes for the inflectional features. A transducer is represented by a graph and can include subgraphs. The transducers are displayed in Unitex style, i.e. input parts are displayed in the nodes, and output parts below the nodes.

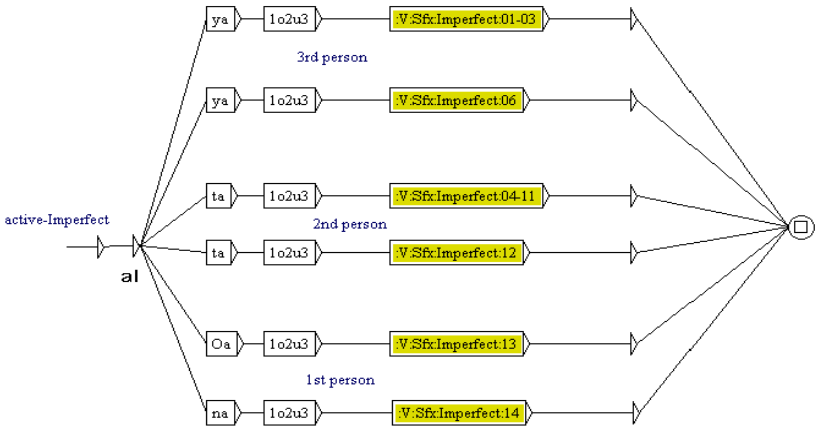


Fig 1. The active imperfect (aI) subgraph. Each path contains a prefix, a stem-pattern and a subgraph of suffixes. The Person-Gender-Number variations are numbered from 01 to 14.

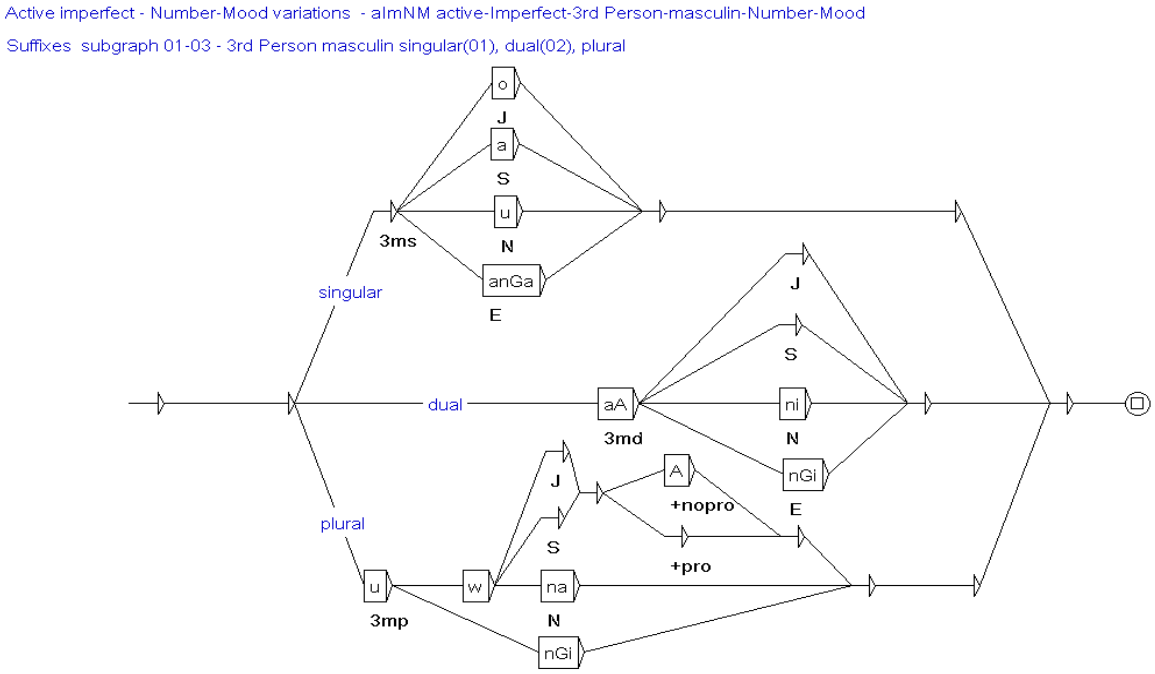


Fig 2. The 01-03 subgraph represents Number-Mode suffix variations for active Imperfect 3rd Person masculine, related to Person-Gender-Number-Mode variations.

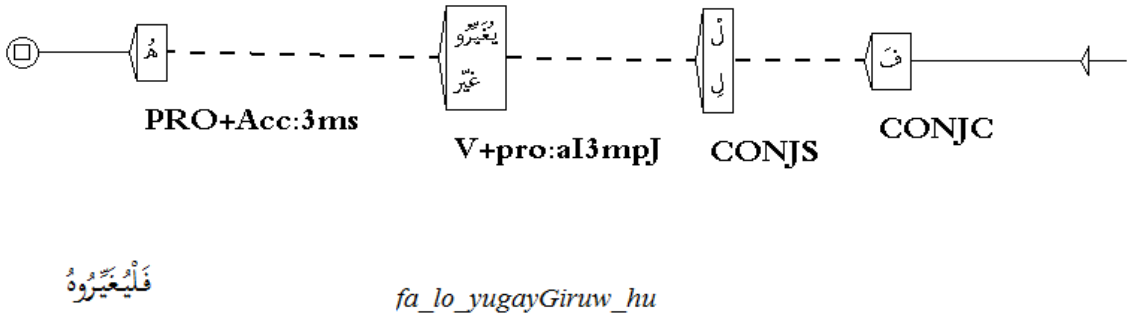


Fig 3. Text automaton as output of the application of a graph dictionary. Here a morphological analysis of *faloyugayGirohu (and_to_change-they_it)*. The morphological dictionary graph restricts the selection to V+pro agglutinated variant only. Dashed lines connect segments in the same token.

A Buckwalter transliteration is used as a standard to map Arabic characters into Latin ones. An XML version of this transliteration was created in order to handle this format. We create a modified version of the XML version where all special characters such as (', |, * , $, ~) are respectively replaced by (c, C, J, M, G) [4]. Many systems use special characters in a special way.
In order to generate the full-form dictionary, the following steps are accomplished.
- The lemma lexicon is transliterated.
- The FSTs are applied to the list and produces a transliterated full-form dictionary output.
- The output is transliterated into Arabic script.
So, both the lemma lexicon and the full-form dictionary are in Arabic script which is handier to read for Arabic linguists.

For example, the lexical entry `ktb,$V3au-123` is processed by the transducer named *V3au-123* in order to get all inflected forms. The main graph contains five subgraphs referring to the five voice-tense variations. In turn, each subgraph (Fig. 1) contains suffixes of Person, Gender, Number for the perfect and Person, Gender, Number, Mode for the Imperfect (Fig. 2).

3.3 Inflection-related adjustments

The inflectional taxonomy takes into account variations due to orthographic adjustment and morphophonemic assimilations. The phoneme involved in the variation is replaced by a gemination mark or by another phoneme. At morpheme boundaries between a stem and a suffix, the first letter *n* and *t* of the perfect suffix is changed to gemination mark like in *daxGan+naA => daxGanGaA* , "smoked-we"; *Oavobat + tu => OavobatGu* "demonstrated-I". Our taxonomy includes the inflectional classes Vpp-12n, Vpp-12t in order to take into account such phenomena. In our resource, we have counted 614 entries in Vpp-12n and 154 in Vpp-12t root-classes.

Due to morphophonemic variations, the *t* in the canonical pattern V65 or *Ii1ota2a3a* (اِفْتَعَلَ) has an orthographic variation depending on the value of the first root consonant. It is replaced by emphatic *T*, or *d*, or by gemination mark *G*. The subclasses V65T, V65d, V65G encode the *t* variation, we have counted: 46 entries with V65T-rrr such as `ISTfY,$V65T-12y` إصطفى; 31 entries with V65d-rrr such as `Izdwj,$V65d-1w3` إزدوج; and 114 entries with V65G-rrr such as `ItGbE,$V65G-123` إتّبع or `ItGSl,$V65G-w23` إتّصل.

4. Agglutination and omission of diacritic

4.1 Orthographic adjustments and agglutination

In Arabic, a token delimited by spaces or punctuation symbols is composed of a sequence of segments. Each segment in a token is a morpheme. In Unitex, this segmentation is formalized via a morphological dictionary graph. Such graphs introduce morphological analyses in the text automaton (Fig 3) where dashed lines connect segments.

The combination of a sequence of morphemes obeys a number of constraints. Checking these constraints is necessary to discard wrong segmentations. In Arabic, a verbal token is composed by one morpheme <*V*> or the concatenation of up to 4 morphemes such as:
<*CONJC*> <*CONJS*> <*V*> <*PRO+accusative*>
where <CONJC> is a coordinating conjunction, <CONJS> is a subordinating conjunction and <*PRO+accusative*> an agglutinated object pronoun.

<*CONJC*> combines freely with any inflected verb. The <*CONJS*> constraints the verb to the Imperfect Subjunctive or Jussive. Finally, an inflected verb form is often insensitive to the agglutinated pronoun but some forms are sensitive like forms with a glottal stop as the third root consonant.

The subgraph selects only V+pro variants from the full-form dictionary (cf. Fig 3). When followed by a pronoun, a verbal segment may have an orthographic adjustment. This is often the case when the verbal segment ends with a long */a:/ A*, its allograph *Y*, or a glottal stop which has 6 allographs depending on its position and the surrounding vowels. For verbs, the roots with a glottal stop as the third consonant change their graphemic representation. A suffix subgraph related to classes Vpp-rrh represents the orthographic variations of an ending glottal stop due to pronoun agglutination.

The generation of the agglutinable variants of an inflected verb is performed directly with a lexicon of words, which is another way to implement a rule. In fact, the dictionary graph links each morphological variant to the correct context, which also expresses a rule. The variants are generated during the compilation of the resources, not at analysis time as in rule-based systems in which a rule should compute each morphological variant at run time, then link each variant to the correct context. The advantage of our method is that it simplifies and speeds up the process of annotation.

4.2 Diacritics

Diacritics are often omitted in Arabic written text. According to our corpus study of 6930 tokens from Annahar newspaper, 209 tokens (3%) include at least a diacritic. 140 tokens (2 %) are with the *F* diacritic *(–an)* and 57 (1 %) are with gemination mark *G*, in which nearly 0.8 % is related to a verbal form. 9 are with the short vowel *u*. For the *u* diacritic, 7/9 involve a passive verbal form. For the gemination diacritic, 49/57 involve a verbal form and are the following.
- 41 to V62 refer to *1a2**G**a3a* derived form (فَعَّلَ).
- 5 to V68 refer to *ta1a2**G**a3a* derived form (تَفَعَّلَ).
- 2 to V65G refer to *Ii1**G**a2a3a* derived form (اِفْتَعَلَ).
- 1 to V3au refers to *ya1o2**u**lu* a triliteral simple form (فَعَل يَفعُل).

4 The Transliteration in Unitex Arabic <=> Latin: ء, c; ا, C; أ, O; ؤ, W; إ, I; ئ, e; ا, A; ب, B; ة, P; ت, T; ث, V; ج, J; ح, H; خ, X; د, d; ذ, J; ر, r; ز, z; س, s; ش, M; ص, S; ض, D; ط, T; ظ, Z; ع, E; غ, g; ف, f; ق, q; ك, k; ل, l; م, m; ن, n; ه, h; و, w; ى, Y; ي, y; ً, F; ّ , N; ْ , K; َ, a; ُ, u; ِ, i; ً, G; ْ, o;

Editors generally display diacritics for unusual forms such as passive verb forms. When some are displayed, they can avoid misinterpretations to the reader. For verbs, diacritics are the short vowels *(a, i, u)* or the gemination mark followed by a short vowel. Arabic verbs can include a sequence of two diacritics: the gemination mark followed by a short vowel. In the case of two diacritics, diacritics omission is not totally free. One can omit the two diacritics or the last diacritic but never the gemination mark alone.

Consequently, processing written Arabic text should take into account undiacriticized and partially diacriticized text. A lookup procedure in Unitex[5] has been adjusted to deal with omission of diacritics in Arabic. This procedure finds in the diacriticized full-form dictionary all possible diacriticized candidate forms compatible with a given undiacriticized or partially diacriticized form. When a diacritic is present in a surface form, the lookup procedure excludes the candidates in the lexicon which do not have that diacritic at the same position.

5. Some figures

Our lexicon is composed of 15 400 entries. Each entry is inflected into 144 surface forms and in average 158 forms if we include orthographic variations due to agglutination. The size of the full-form dictionary is 2.43 million surface forms. The size of the full-form dictionary in plain text is 132 Megabytes in Unicode little Endian and is compressed and minimized into 4 Megabytes which is loaded to memory for fast retrieval. The generation, compression and minimization of the full-form lexicon lasts two minutes[6] on a Windows laptop.
The number of main inflectional graphs is 460. Each main graph is composed of 5 subgraphs for voice-tense features variations, that is 2300 subgraphs. These subgraphs use also 540 suffix subgraphs related to person-gender-number-mode features. In all, the number of graphs and subgraphs is 3300 (460+2300+540), to be compared with nearly 100 graphs and subgraphs dedicated to the verbal inflection system for Brazilian Portuguese constructed also for Unitex (Muniz et al. 2005). A sample will be freely available from the time of the workshop.

We have noticed that many simple triliteral verbs may have orthographical variants related to the variation of the vowel after the second root consonant. However, these variations may correspond to meaning differences; therefore we should have different entries. In order to facilitate the encoding scheme, all orthographic variants of verbs are encoded in separate entries. In our lexicon, a verb may have several inflectional codes. These codes can correspond to different lexical items or to orthographic variants of the same item. In the future, we plan to encode different lemmas if the different inflectional behaviours are correlated to differences at other levels, e.g. semantic, which is the case of `Hsb,$V3au-123` "count", and `Hsb, $V3ii-123` "think". One should also encode a single lemma if the inflectional behaviours are a free variation, such as for `kfl,$V3au-123` and `kfl,$V3ai-123` "grant". Out of a total 4135 simple triliteral root in the lexicon, 1278 triliteral root have several inflection al codes.

Some inflectional classes are redundant such as V62-122, which is identical to V62-123, whereas V65-122 is different from V65-123. In order to make the encoding scheme easier to handle for Arabic linguists, we have duplicated the inflectional graph V62-122. The 122 root-class delimits two classes in nearly all other cases. We estimate such redundancy at 15%. We offer a simple encoding scheme with duplicated inflectional classes in order to make it unnecessary for Arabic linguists to memorize in which cases some features have to be marked.

6. Evaluation

We have chosen the NEMLAR Arabic Written Corpus (Attia et al., 2005), first to improve our lexicon of verbs, and then to constitute our test collection. The Nemlar data consists of about 500 thousand words of Arabic text from 13 different genres. The text is provided in 4 different versions: raw text, fully diacriticized text, text with Arabic lexical analysis, and text with Arabic POS-tags. The database was produced and annotated by RDI, Egypt, for the Nemlar Consortium.
The extraction of occurrences of verbs from "text with Arabic POS-tags" provided 50 000 occurrences of verbs. These occurrences were split in two disjoint parts: nearly 40 000 token occurrences (11050 token types) for correcting the resource and a test collection of 10 000 token occurrences (5222 token types) for testing it after the correction stage.
The test collection shows that 10 verbs lemmas were missing in our lexicon[7]. Hence, the fault rate of the resource is 0.1% in this corpus. Let us assume that a page is composed of 50 lines/page, 10 tokens/line, 1 verb/10 tokens. In other words, in 20 pages of real corpus, our resource fails to recognize 1 verb.
In order to compare our lexicon with the Buckwalter resource, we ran BAMA on the first 550 occurrences of verbs of the same test collection. 14 occurrences of verbs were unrecognized, which represents a 2.5 % error rate, i.e. 25 times the error rate of our resource. The unrecognized tokens involve: 10 missing passive stems, 2 imperative stems and 2 missing verb lemmas.
Morphosyntactic tagging is generally part of a pipeline of written text processing. In a common undiacriticized Arabic corpus, most verbs have two possible analyses, one as active and one as passive. The lack of passive stems in the Buckwalter resource leads to assign only the active tag to verbs, which can jeopardize a subsequent deep syntactic parsing of a sentence.
A fallback procedure in order to assign morphosyntactic

5 The lookup procedure was adjusted by Sébastien Paumier.

6 At Columbia University, MAGEAD Project constructs an Arabic resource according to Buckwalter's Prefixes-Stem-Suffixes representation. They describe an Arabic lexicon based on root-and-pattern representation and rules dedicated to orthographic variations due morphophonemic alternations; and other rules dedicated to orthographic adjustment due to agglutinations (Habash & Rambow, 2006). The program needs more than 15 hours to generate such resource (Owen Rambow, personal communication).

7 jzm,$V32-123; qrGZ,$V62-123; thrGb,$V68-123; rDb,$V33-123; kfl,$V34-123; tnAqM,$V67-123; sAb,$V32-1y3; zEq,$V33-123; DnG,$V32-1nn; tAh,$V32-1y3

features to unrecognized tokens is often included in a language processing pipeline. Since our fault rate is 0.1 %, it might be useless to construct a fallback procedure for unrecognized verbs when this resource is used.

7. A conclusion and perspectives

We elaborated a model for Arabic verbs with the following features. A detailed and simple taxonomy is based on Semitic morphology. Lemma-based verbs are used as entries in the lexicon. FSTs are used to produce inflected forms. Agglutination is described independently from inflection.Our experimentation shows that the method outperforms state-of-the-art systems of Arabic morphological annotation.
We made language resources the central point of the problem. All complex operations were integrated among resource management operations. The output of our system is accurate and informative; the language resources used by the system can be easily updated by an expert of Arabic independently from computational linguistics experts, which allows users to control the evolution of the accuracy of the system. Morphological annotation of Arabic text is performed directly with a lexicon of words and without morphological rules, which simplifies and speeds up the process. The undiacriticized, partially and fully diacriticized Arabic text can be annotated excluding incompatible analyses.

We reuse traditional Semitic patterns and we provide a clear scheme for root-class encoding by avoiding intricate terms. Root-and-pattern representation facilitates our task in encoding the lexicon since it is a standard but also it helps to debug our transducers quickly which is not the case of a rule-based system.
This work opens several perspectives. The resources can be extended by running the annotator and analysing output. Another perspective is to extend this methodology to inflection of noun and adjective, mainly to encode singular and the plural under the same lemma entry using Semitic patterns فَـعِـيل فُـعَـلاء. For example, the pair *raeiys*, *ruWasaAc* (رَئِيس رُؤَسَاء) "president" will be represented by one entry:

```
raeiys,$N3_1a2iy3-1u2a3Ac-1h3
nabiyl,$N3_1a2iy3-1u2a3Ac-123
```

where number *3* denotes a triliteral root; *1a2iy3-1u2a3aAc* is a pattern pair that represents singular-plural variations; and *1h3* (vs *123*) encode the glottal stop variations of the 2nd consonant root *(e =>W)*.

8. References

Al-Bawab, M., Mrayati, M., Alam, Y.M., Al-Tayyan, M.H. (1994). A computerized morpho-syntactic system of Arabic. In *The Arabian Journal of Science and Engineering*, 19, 461-480. Published by KFUPM, Saudi Arabia.

Attia., M., Yaseen., M., Choukri., K. (2005). Specifications of the Arabic Written Corpus produced within the NEMLAR project, www.NEMLAR.org.

Beesley, Kenneth R. (1996). Arabic finite state morphological analysis and generation. *In COLING'96, volume 1*, pages 89– 94, Copenhagen, August 5-9. Center for Sprogteknologi. The 16th International Conference on Computational Linguistics, 1996.

Buckwalter, T. (2004). Issues in Arabic Orthography and Morphology Analysis. *In Proceedings of the COLING 2004*. Workshop on Computational Approaches to Arabic Script-based Languages, pages 31–34.

Buckwalter Arabic Morphological Analyzer Version 1.0. (2002). LDC Catalog No.: LDC2002349.

Cavalli-Sforza, Soudi, Mitamura (2000). Arabic Morphology Generation Using a Concatenative Strategy. In *Proceedings of the 6th Applied Natural Language ProcessingConference (ANLP 2000)*, pages 86–93, Seattle, Washington, USA.

Dichy, J., Farghaly, A. (2003). Roots & Patterns vs. Stems plus Grammar-Lexis Specifications: there what basis should is multilingual lexical database centred on Arabic be built? *In Workshop on Machine Translation for Semitic Languages*, New Orleans, USA.

Habash, N., Rambow, O. (2005). Arabic Tokenization, Morphological Analysis, and Part-of-Speech Tagging in One Fell Swoop. *In Proceedings of the Conference of American Association for Computational Linguistics* (ACL05).

Habash, N., Rambow, O. (2006). MAGEAD: A Morphological Analyzer and Generator for the Arabic Dialects. *In Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, pages 681–688, Sydney, Australia, July.*

Habash, N. (2010). *Introduction to Arabic Natural Language Processing*. Morgan & Claypoll Publishers.

Huh, H.-G. Laporte É. (2005). A resource-based Korean morphological annotation system. *In Proc. Int. Joint Conf. on Natural Language Processing*, Jeju, Korea, 2005.

Kiraz, A. (2004): http://www.scribd.com/doc/46443095/Computational-Nonlinear-Morphology-With-Emphasis-on-Semitic-Languages-Studies-in-Natural-Language-Processing-9780521631969-41686

Mesfar, S. (2008). Analyse morpho-syntaxique automatique et reconnaissance des entités nommées en arabe standard. Thèse, novembre 2008, Université de Franche-Comté.

Mesfar, Slim. (2006). Standard Arabic formalization and linguistic platform for its analysis in *Proceedings of Arabic NLP/MT conference*, London, 2006

Marcelo C.M. Muniz, Maria das Graças V. Nunes, and Éric Laporte (2005). UNITEX-PB, a set of flexible language resources for Brazilian Portuguese. *Workshop TIL'05*. pp. 2059–2068.

Paumier, Sébastien. (2011). *Unitex - manuel d'utilisation 2.1* , University of Marne-la-Vallée.

Silberztein, Max. (1998). INTEX: An integrated FST toolbox, in Derick WOOD, Sheng YU (éd.), *Automata Implementation*, p. 185-197, Lecture Notes in Computer Science, vol. 1436. Second International Workshop on Implementing Automata, Berlin/Heidelberg: Springer.

Smrz, Otakar. (2007). ElixirFM — Implementation of Functional Arabic Morphology. *In Computational Approaches to Semitic Languages*, ACL 2007, Prague.

Al-Sughaiyer, Imad A., Al-Kharashi, Ibrahim A. (2004). Arabic Morphological Analysis Techniques: A Comprehensive Survey. *In Journal of the American Society for Information Science and Technology*, 55(3):189–213.