Morphological Analysis and Generation of Arabic Nouns: A Morphemic Functional Approach

Mohamed Altantawy,* Nizar Habash,* Owen Rambow,* Ibrahim Saleh†

*Center for Computational Learning Systems, Columbia University, New York, USA
{mtantawy,habash,rambow}@ccls.columbia.edu
†Arabic and Translation Studies Division, The American University in Cairo, Egypt
imazied@aucegypt.edu

Abstract

MAGEAD is a morphological analyzer and generator for Modern Standard Arabic (MSA) and its dialects. We introduced MAGEAD in previous work with an implementation of MSA and Levantine Arabic verbs. In this paper, we port that system to MSA nominals (nouns and adjectives), which are far more complex to model than verbs. Our system is a *functional morphological analyzer and generator*, i.e., it analyzes to and generates from a representation consisting of a lexeme and linguistic feature-value pairs, where the features are syntactically (and perhaps semantically) meaningful, rather than just morphologically. A detailed evaluation of the current implementation comparing it to a commonly used morphological analyzer shows that it has good morphological coverage with precision and recall scores in the 90s. An error analysis reveals that the majority of recall and precision errors are problems in the gold standard or a result of the discrepancy between different models of form-based/functional morphology.

1. Goal of This Paper

In previous work, we have presented MAGEAD, a morphological analyzer and generator for Modern Standard Arabic (MSA) verbs, and we have extended that work to cover Levantine Arabic as well (Habash et al., 2005; Habash and Rambow, 2006). In this paper, we port that system to MSA nominals (nouns and adjectives). Our system is a *functional morphological analyzer and generator*, i.e., it analyzes to and generates from a representation consisting of a lexeme and linguistic feature-value pairs, where the features are syntactically (and perhaps semantically) meaningful, rather than just morphologically. In this perspective, nouns turn out to be far more complex than verbs. This is because all variants (MSA and dialects) of Arabic have many "broken plurals" (irregular plurals), which are very common, and irregular feminine forms, which are less common. Furthermore, the same surface morpheme can have different morphological functions depending on context. For example, the morpheme Ta-Marbuta (ة+ +*ħ*),[1] usually associated with the feminine singular (as in شجرة *šjrħ* 'tree'), can appear on the plural form of certain masculine nouns (as in أنظمة *ÂnĎmħ* 'systems'). This discrepancy between the surface form-based morphology and the functional morphology has only recently been addressed in depth in a computational system – see Smrž (2007)'s transformation of the form-based Buckwalter morphological analyzer (Buckwalter, 2004). This paper differs from (Smrž, 2007) in that 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.

Because of the prevalence of irregular inflectional forms among Arabic nominals, the lexicon plays a very important role in a functional morphological analyzer or generator for Arabic: we need to be able to relate irregular forms to their lexemes, and this can only be done with a lexicon. In this paper, we do not present work on a lexicon, and concentrate on the computation of morphology instead, including the interface to the lexicon. 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.

This paper is structured as follows. We present the relevant linguistic facts in more detail in Section 2. We compare our work to related work in Section 3. The computational machinery is presented in Section 4. We present the morphological behavior class hierarchy (the interface to the lexicon) in Section 5. Morphophonemic rules are presented in Section 6. We give an evaluation of MAGEAD in Section 7.

2. Overview of Arabic Nominal Morphology

Arabic is a morphologically rich and complex language. For nominals, the inflectional variants are as follows:

- Number: singular, dual, plural. Some lexemes only have a singular form, such as نمل *nml* 'ants as a collective'.
- Gender: masculine, feminine. Note that only some lexemes (those nouns denoting types of humans, such as كاتب *kAtib* 'writers', and all adjectives) show inflection for gender; however, if the lexeme does not inflect for gender (for example أذن *Âuðun* 'ear'), it has an inherent gender (in this case, feminine).
- Case: nominative, accusative, genitive.
- State: definite, indefinite, construct. We follow Fischer (2001) in his analysis of the morphological determination system for MSA. State is expressed as a suffix and should not be confused with the presence of the +ال *Al+* definite determiner; however, there is an interaction: state cannot be indefinite in the presence

[1] All Arabic transliterations are provided in the Habash-Soudi-Buckwalter transliteration scheme (Habash et al., 2007).

of +ال *Al+*. We see definite state with no +ال *Al+* in the vocative يا رجل! *yA rjl!* 'Man!'.

In addition, nominals take several classes of clitics, which we also handle in MAGEAD in the same way as inflectional affixes:

- Article: presence or absence of the +ال *Al+* proclitic 'the'
- Preposition: presence of a particular preposition (+ك *k+* 'as', +ب *b+* 'by/with', or +ل *l+* 'for')
- Conjunction: presence of a particular conjunction (+و *w+* 'and', +ف *f+* 'then')
- Possessive pronoun: presence of a particular possessive pronoun, e.g, +هم *+hm* 'their'

Not only does Arabic have a large number of inflectional features, they are expressed morphologically using both templatic (i.e., root,[2] pattern, vocalism), and concatenative (prefix and suffix) morphemes. These morphemes do not all have a one-to-one correspondence to linguistic features. For example, the various values of state are often confused because they realize in different ambiguous ways in combination with other features. Compare the singular masculine forms كتابٌ *kitAb+ũ* 'book [indef]' and كتابُ *kitAb+u* 'book [def/construct]' with the dual masculine forms كتابان *kitAb+Ani* 'two books [indef/def]' and كتابا *kitAb+A* 'two books [construct]'. The combination (interdigitation and affixation) of the various morphemes is further complicated as it may involve various phonological and orthographic adjustments. Additionally, there are optional diacritical marks used for short vowels and consonantal duplication. For example, the word لِلمَوَازِينِ *lilmawAziyni* 'for the scales' (typically written as للموازين *llmwAzyn*) can be analyzed into the following morphemes: *li+Al+[wzn]+[mV1V2V3]+[aAI]+i* 'for+the+[scales]+genitive'; the symbols inside the square brackets are templatic morphemes: the root *wzn* 'weight-related' and the pattern and vocalism often used with plurals of instruments/tools. In this example, the number feature is expressed as a pattern choice – compare the plural موازين *mawAziyn [wzn]+[mV1V2V3]+[aAI]* to the singular form ميزان *miyzAn [wzn]+[mV12V3]+[iA]*. This phenomenon is called *broken plurals* and it accounts for almost half of all plurals in Arabic nominals, the rest being sound plurals which use suffixation, e.g., موظفات *mwĎf+At* 'employee+fem.plural'. Furthermore, although the morpheme +ات *+At* is primarily used to indicate the feminine and plural features, it is also used with some masculine nouns to indicate plurality, e.g., امتحانات *AmtHAn+At* 'test+plural'.

Arabic nouns have peculiar agreement rules that depend on lexical features such as humanness/rationality and collectiveness. Specifically, adjectives modifying Arabic nouns agree in gender and number except when the head is an irrational plural noun, in which case the adjective is feminine singular. Compare موظفة جديدة *mwĎfħ jdydħ* 'employee+fem.sg new+fem.sg' and موظفات جديدات *mwĎfAt jdydAt* 'employee+fem.pl new+fem.pl' with ميزان جديد *myzAn jdyd* 'scales+masc.sg new+masc.sg' and موازين جديدة *mwAzyn jdydħ* 'scales+masc.pl new+fem.singular'. The discrepancy between surface and actual gender/number only makes these cases more complex: the word موازين *mwAzyn* in terms of its surface morphology is masculine singular (since it lacks any plural or gender suffix), but functionally it is masculine plural; and being irrational, it takes a feminine singular adjective. Because of these agreement rules, we also record for each lexeme whether it is rational or irrational.

Within the computational approach we use here, a central concept is the *lexeme*, the set of all related inflectional variants. The lemma is a chosen representative of a lexeme; for Arabic nominals, it is the singular nominative definite without any clitics.

We consider all other morphological variation derivational, which means that two words that differ in derivational morphology are from different lexemes. We are aware of the fuzzy cases that border inflectional and derivational morphology, such as the collective plural (compare نملة *nmlħ* 'ant' and its plural نملات *nmlAt* 'ants', versus نمل *nml* 'ants as a collective') or the archaic plurals of paucity/plenty. These words are from different lexemes. See the even more subtle case of كاتب *kAtib* 'writer' in Section 5. In Arabic, all lexemes linked by derivational morphology share a common root: the root is to derivational morphology what the lexeme is to inflectional morphology; the root is a more abstract morphological notion than the lexeme.

3. Related Work

Much work has been done on Arabic morphological analysis and generation in a variety of approaches and at different degrees of linguistic depths (Al-Sughaiyer and Al-Kharashi, 2004). The focus on the lexeme as a central morphological concept is comparable to efforts by (Soudi et al., 2001; Habash, 2004; Smrž, 2007; Dichy and Farghaly, 2007). The implementation using finite state morphology (FSM) is comparable to early work on Arabic morphology, most notably (Kiraz, 1994; Beesley, 1996) and more recent efforts following the multi-tier approach by the authors (Habash et al., 2005; Habash and Rambow, 2006). FSM implementations naturally handle analysis and generation.

We compare MAGEAD to the commonly used Buckwalter Arabic Morphological Analyzer (BAMA) (Buckwalter, 2004) and to the Elixir-FM functional morphology system for Arabic (Smrž, 2007). The development and evaluation of our system were heavily influenced by these two systems.

Elixir-FM and MAGEAD are both functional morphology systems compared to BAMA, which models form-based morphology. These systems also differ in how they model their form-based components. BAMA does not explicitly model templatic morphology or morphological interactions, instead it uses a simple break up of words into prefixes, stems and suffixes that already collapse all templatic,

[2]The root is the traditional notion from Arabic grammar: a sequence typically of three or (rarely) four consonants that abstracts over derivational morphology.

morphophonemic and orthographic decisions. In contrast, Elixir-FM models templatic morphology and orthographic rules; however, some of these are handled by spelling out the allomorphs of a morpheme For example, the pattern for ميزان *myzAn* 'scales', mi12A3, is represented as its allomorph mI2A3,[3] which consolidates a weak radical rule with the pattern. MAGEAD uses a morphemic representation for all morphemes and explicitly defines morphophonemic and orthographic rules to derive the allomorphs. Despite these differences, the BAMA lexicon was heavily used in the creation of Elixir-FM, which extended it to handle functional morphology. Our lexicon is developed by extending Elixir-FM's lexicon.

4. The MAGEAD System: Implementation

MAGEAD is a morphological analyzer and generator for the Arabic language family, by which we mean both MSA and dialects. Building an instance of MAGEAD for a member of the Arabic language family and a class of words, such as MSA nouns or Egyptian verbs, goes through three main phases. Figure 1 shows the overall architecture.

The first phase (shown in Figure 1 in the doted L-shape diagram) involves creating by hand the linguistic resources that are used by a specific MAGEAD instance. Theses resources are: the Morphological Behavior Class Hierarchy MBCH, the context free grammar (CFG) that orders morphemes, the morphophonemic/orthographic rules, and the lexicon. We describe the MBCH and the rules in detail in the following sections. While these resources are unique to a specific instance of MAGEAD, the processes used in the subsequent phases are the same across all instances.

The second phase (shown in Figure 1 in the dashed rectangle) is the compilation of the instance's linguistic resources to produce two finite state transducers (FSTs), one for generation and the other, its inverse, for analysis. MAGEAD is implemented as a multi-tape finite state automata layer on top of the AT&T two-tape finite state transducers (Mohri et al., 2000). We extend the analysis of Kiraz (2000) by introducing a fifth tier. The five tiers are used as follows:

- Tier 1: pattern and affixational morphemes;
- Tier 2: root;
- Tier 3: vocalism;
- Tier 4: phonological representation;
- Tier 5: orthographic representation.

In the generation direction, tiers 1 through 3 are always input tiers. Tier 4 is first an output tier, and subsequently an input tier. Tier 5 is always an output tier. All tiers are read or written at the same time, so that the rules of the multi-tier automaton are rules that scan the input tiers and, depending on the state, write to the output tier.

The compilation phase runs through three consecutive steps. Each step's output is the input to the following step. As a first step, the rules, the MBCH and the CFG as inputs are coded in the Morphtools format, a specification language that we defined for the multi-tape machine used in MAGEAD. Then the Morphtools format is compiled to the Lextools format, an NLP-oriented extension of the AT&T toolkit for finite-state machines (Sproat, 1995). The Lextools specification is compiled by Lextools to the specification language of the AT&T toolkit, which is finally compiled into the desired binary FSTs using the AT&T toolkit. Once we have generated the FSTs there is no need to repeat this phase unless the linguistic resources are modified. For details, see (Habash et al., 2005; Habash and Rambow, 2006).

The third phase is simply the use of MAGEAD for morphological generation or analysis (shown in Figure 1 in the dash-doted rectangle). In this phase, MAGEAD relates (bidirectionally) a lexeme and a set of linguistic features to a surface word form using the FSTs generated by the compilation phase. Conceptually, in a generation perspective, the features are translated to form-based morpheme features that are then ordered, and expressed as concrete morphemes. The concrete templatic morphemes are interdigitated and affixes added, and finally morphological rewrite rules are applied.

5. A Lexical Hierarchy

This section describes the lexical hierarchy which we use as interface between the morphological engine and the lexicon. This hierarchy lets us relate morphological processes to those lexical items which can undergo them.

Within MAGEAD, our operational definition of a lexeme is as a triple of root, morphological behavior class (MBC) and meaning index. The MBC defines exactly how morphemes (both templatic and affixival morphemes) and features are paired for a particular lexeme. Additionally, the MBC specifies relevant morphosyntactic features such as rationality, which determines the agreement rules that are used. The name of the MBC encodes the behavior mnemonically; for example, in MBC mbc:noun-I-M-mi12A3-ma1A2iy3, the I in the MBC class name tells us this is an irrational noun, M that it is has inherent masculine gender, and the two vocalized patterns mi12A3 and ma1A2iy3 are used for the singular and plural, respectively. The meaning index disambiguates between lexemes that have identical word forms, but different meaning; in our current work, we do not use the meaning index, as it pertains only to lexicography and not to morphology. We illustrate our approach with some examples.

Regular inflection معلم *muEal∼im* 'teacher' is regular in both number and gender: the feminine singular form is معلمة *muEal∼imaħ*, the masculine plural form is معلمون *muEal∼imuwn*, and the feminine plural is معلمات *muEal∼imaAt*. Put differently, all inflectional feature combinations map to the same pattern and vocalism morpheme *[mV1V22V3]+[uai]*; however, for the same case and state features, say [CAS:NOM STT:INDEF], [GEN:MAS NUM:SG] gets mapped to the masculine singular morpheme ـٌ+ +ũ, [GEN:MAS NUM:PL] to the masculine plural morpheme ـون+ +*uwna*, [GEN:FEM NUM:SG] to the feminine singular morpheme ـةٌ+ +*aħ*+*ũ*, and [GEN:FEM NUM:PL] to the feminine plural morpheme ـاتٌ+ +*At*+*ũ*. *muEal∼im* is a rational lexeme. Its MBC is thus R-A-mu1a22i3-*, where the R indicates it is rational, A that it can be either masculine or feminine, and mu1a22i3-* says

[3] Smrž (2007) uses a slightly different notation (in this case, MICAL) that maps bijectively to the notation we use here.

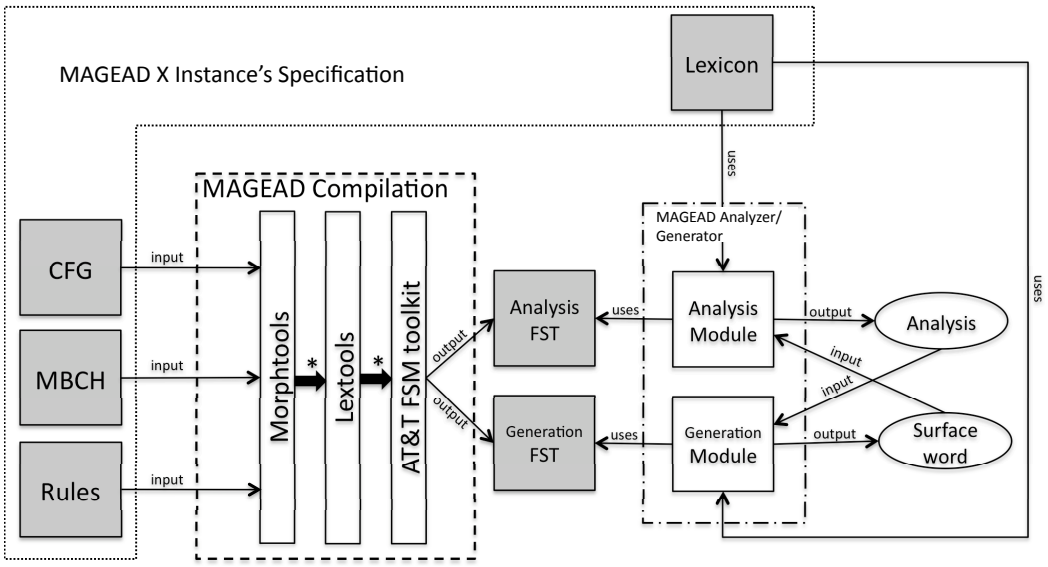

Figure 1: Overview of MAGEAD's Architecture. The asterisk * means that compiled output is input to the following step.

that the vocalized pattern is the same for all forms, and regular inflectional morphology is used to derive them.

Simple broken plural مفتاح *miftAH* 'key' is a masculine irrational noun which has the plural مفاتيح *mafAtiyH* 'keys'. Its MBC is thus mbc:noun-I-M-mi12A3-ma1A2iy3. This MBC maps [GEN:MAS NUM:SG] to the pattern and vocalism morphemes *[mV12V3]+[iA]* and the masculine singular morpheme, and [GEN:MAS NUM:PL] to the pattern and vocalism morphemes *[mV1V2V3]+[aAI]* and (also) the masculine singular morpheme. The fact the مفاتيح *mafAtiyH* is a diptote is handled in with pattern-aware rules, not in the MBC.

Multiple plurals Some words have multiple broken plurals, often with slight meaning variations. Consider the singular form كاتب *kAtib* 'writer'. There are three masculine plurals forms, two of which are broken and one of which is regular: كتّاب *kut∼Ab* 'authors, novelists', كتبة *katabaħ* 'scribes', and the regular كاتبون *kAtibuwn* 'persons writing'. The feminine forms for all three meanings of *kAtib* are the same: كاتبة *kAtibaħ* (singular) and كاتبات *kAtibAt* (plural). We represent this situation by saying that there are in fact three distinct lexemes (all rational), which is necessary as we can only specify one plural form per lexeme per gender, but which is additionally supported by the meaning differences in the plural forms. The three MBCs we have for *kAtib* are thus: mbc:noun-R-A-1A2i3-1u22A3-*, mbc:noun-R-A-1A2i3-1a2a3+ap-*, and mbc:noun-R-A-1A2i3-*. Here, 1A2i3-1a2a3+ap-* (to take one example) means that the masculine singular pattern and vocalism morphemes are *[1V2V3]+[Ai]* with a masculine singular affix, the masculine plural pattern and vocalism morphemes are *[1V2V3]+[aa]* with the feminine singular affix, and the feminine forms use the masculine singular pattern and vocalism morphemes and the regular feminine affixes.

We define MBCs using a special language, in which we can define a hierarchical representation with non-monotonic inheritance. The hierarchy allows us to specify only once those feature-to-morpheme mappings for all MBCs which share them. For example, the root node of our MBC hierarchy is a word, and all Arabic words share certain mappings, such as that from the linguistic feature conj:w to the clitic *w+*. This means that all Arabic words can take a cliticized conjunction. Similarly, the possessive pronominal clitics are the same for all nouns, no matter what their templatic pattern is, and no matter whether the plural is sound or broken. Our current MBC hierarchy specification for MSA nouns comprises 962 classes that can be instantiated in our lexicon, which comprises 32,110 nouns.

We built our lexicon using the lexicon of the Elixir-FM system (Smrž, 2007) with major extensions to convert its allomorphic templatic patterns to something consistent with MAGEAD's representation. The extension was done semi-automatically. We plan to discuss lexicon development for MAGEAD in a future publication.

6. Morphological, Phonological, and Orthographic Rules

This section provides a list of morphological rules for Arabic MSA nominals. Crucially, the rules we present for MSA are different from previous approaches in the explicit separation between orthography and phonology.

We have two basic types of rules. First, morphophonemic/phonological rules map from the morphemic representation to the phonological and orthographic representations. Our nominal system has 79 multi-tier rules of this type. Second, orthographic rules rewrite only the orthographic representation. We use 77 two-tier rules of this type.

A large number of these rules is similar to those we previously presented for verbs (Habash and Rambow, 2007). There are some important differences that stem from particulars of verb and noun morphology differences. For example, nouns have the Ta-Marbuta and definite article morphemes but verbs don't. Similarly, verbs have particular morphemes that require their own rules such as the Waw-of-Plurality واو المضارعة. Some of the shared rules

never apply for some cases because of the different contexts. For example a verb can have no vowels following its last radical in some of its forms (e.g., imperative; perfective with consonant-initial subject suffixes), but nouns always require a case vowels (in MSA). This interacts with Geminate rules. So, whereas the verb مدّ *mad∼+a* 'he extended' has the form مددت *madad+tu* 'I extended'; the noun محتلّ *muHtall+u* 'occupier' never appears as **muHtalil*.

We organize this section by the phenomena we handle rather than the types of rules. As such, each phenomenon will be handled with a combination of different types of rules. We cluster the different phenomena discussed at a coarse level as follows: general default cases, templatic rules, affixational rules, and finally general orthographic rules. We only discuss a portion of the rules for space limitations. We do not discuss the not-so-interesting issue of rule-ordering, which is relevant to making sure the whole system functions smoothly.

6.1. Default Rules

The default cases are sufficiently handled with simple default morphophonemic and orthographic rules that map symbols from tier to appropriate tier. These cases all require the general cleaning of morphemic boundary markers, such as '+', stem initial/final, word initial/final markers, e.g., *[Al+1V2V3+u]+[ktb]+[uu]* ⇒ /Al+kutub+u/ ⇒ *Alkutubu* 'the books'.

6.2. Templatic Rules

Form VIII Rules Nominals derived from verb form VIII (i1ta2a3 افتعل) experience a change in the pattern consonant *t* based on the form of the first root radical. The pattern consonant ت *t* changes to د *d* when the first root radical is ز *z*, د *d* or ذ *ð*. Similarly, the same pattern consonant changes to ط *T* when the first root radical is an emphatic consonant (ص *S*, ض *D*, ط *T* or ظ *Ď*). For example, compare the following verbs all of which are in Form VIII deverbal (مصدر Masdar): استلام *AistilAm* 'reception' (root *slm*, default case), ازدهار *AizdihAr* 'prosperity' (root **z***hr*) and اصطبار *AiSTibAr* 'longanimity/forbearance' (root **S***br*).

Weak Radical Rules The root radicals *w* and *y* are called weak because they often change form due to the application of various morphophonemic rules. There are numerous vocalic and consonantal conditions that cause these changes to take place. We only illustrate some examples of these changes without further discussion:

- *['V12V3]+[wDH]+[iA]* ⇒ *'iwDAH* ⇒ *'IDAH* ⇒ *ĂiyDAH* إيضاح 'clarification'
- *[mV12V3]+[θwr]+[ui]* ⇒ *muθwir* ⇒ *muθIr* ⇒ *muθiyr* مثير 'exciting'
- *[1V2V3]+[jlw]+[aA]* ⇒ *jalAw* ⇒ *jalA'* جلاء 'withdrawal'

Geminate Rules Geminate radicals rules are applied when second and third root radicals have the same consonantal value, e.g., *Hll* or *mdd*. A common geminate rule deletes a vocalism short vowel when preceded by a vocalism vowel and a geminate radical and followed by a geminate radical and a vowel suffix, e.g., *[mV1tV2V3+u]+[Hll]+[uai]* ⇒ *muHtali**l**+u* ⇒ *muHtall+u* ⇒ محتلّ 'occupier'.

6.3. Affixational Rules

Ta-Marbuta When followed by a pronominal enclitic, word-final Ta-Marbuta is rewritten as Ta: مكتبة+نا *mktbħ+nA* becomes مكتبتنا *mktbtnA* 'our library'.

Alif-Maqsura When followed by a pronominal enclitic, Alif-Maqsura typically becomes Alif in nominals, e.g., مستشفى+هم *mstšfý+hm* becomes مستشفاهم *mstšfAhm* 'their hospital'.

Pronominal Clitic Form The *u* vowel in the *+hu-* pronominal enclitics, ه+ *+hu*, هما+ *+humA*, هم+ *+hum*, and هن+ *+hun∼a*, undergoes phonological assimilation to *i* when following a word that ends with *i* as in the nominal genitive case. For example, كتابه 'his book' can be diacritized as *kitAbu+hu*, *kitAba+hu* or *kitAbi+hi*. Similarly, the 1st person singular pronoun clitic ي+ *+iy* has an allomorph *+ya* with words ending with the letters Alif, Ya or Alif-Maqsura, e.g., عيناي *ςaynAya* 'my eyes [nominative]' (عينا+ي *ςaynA+iy*), مولاي *mawlAya* 'my lord' (مولى+ي *mawlaý+iy*). The same pronominal clitic overrides word-final case markers effectively normalizing case for such words, e.g., كتابي *kitAbiy* 'my book' can be underlyingly *kitAb+u+iy*, *kitAb+a+iy* or *kitAb+i+iy* (nominative, accusative or genitive, respectively).

Definite Article The Lam of the definite article +ال *Al+* phonologically assimilates if followed by a so-called Sun letter.[4] Assimilation is indicated by doubling the first letter of the word (with a Shadda diacritic) and counterintuitively not deleting the assimilating letter in the definite article (to preserve the word's morphemic spelling). For example, شمس+ال *Al+šamsu* 'the sun' is written as الشّمس *Alš∼amsu*; however, قمر+ال *Al+qamaru* 'the moon' is written as القمر *Alqamaru*. The Alif of the definite article +ال *Al+* is deleted when preceded by the prepositional proclitic +ل *li+*: ل+الكتاب *li+AlkitAbi* 'for the-book' becomes للكتاب *lilkitAbi*. A similar case of phonological elision occurs with the prepositional proclitic +ب *bi+*, but without the spelling change: ب+الكتاب *bi+AlkitAbi* 'by the-book' is pronounced /bilkitAbi/ but written بالكتاب *biAlkitAbi*.

6.4. General Orthographic Non-Lexical Rules

These rules are purely orthographic rules that do not interact with any other tiers of information. They are the last step in generation mode and the first step in analysis mode. They are presented below in their order of application in generation mode.

Long Vowel Spelling Rule The long vowels are spelled using a short vowel and a glide: ī is written as *iy* and ū as *uw*.

[4]The Sun letters are ت *t*, ث *θ*, د *d*, ذ *ð*, ر *r*, ز *z*, س *s*, ش *š*, ص *S*, ض *D*, ط *T*, ظ *Ď*, ل *l*, and ن *n*.

Sukun Rule A Sukun (no vowel diacritic) is added between any two adjacent consonants or after a consonant at the end of a word.

Shadda Rule The second of two repeated consonants (separated by a Sukun) is replaced with a Shadda diacritic. The Sukun is deleted. For example, *kuttAb* becomes كتّاب *kut∼Ab*.

Hamza Rules The phoneme Hamza (glottal stop) is written using six orthographic symbols (ء ', آ Ā, أ Â, ؤ ŵ, إ Ă and ئ ŷ). The correct symbol is dependent on the context of the Hamza. Buckley (2004) describes over 18 rules for Hamza writing, which we implemented here. Rare exceptions and allowed (less common) alternatives are not handled. The following are some of the rules we used. First a word-initial Hamza followed by a short vowel diacritic is written with Alif Hamza Below (إ Ă) when followed by i and with Alif Hamza Above (أ Â) otherwise. The Hamza is written on a Ya (ئ ŷ) when either followed or preceded by *i* or when following a consonant letter Ya. Alternatively, it is written on a Waw when either followed or preceded by u. Otherwise Hamza is written on Alif. Also, Alif Hamza Above followed by a *long a* (ā) is rewritten as Alif Madda (آ Ā). At the end of base word, the default is to have the Hamza written on the line. The exceptions depend on the preceding short vowel. Some Hamza rules are more complex and involve letter shape. For example, a Hamza at the end of a word directly following a connective letter changes its shape to Ya-Hamza before the suffix اً +Aã, but not if the letter is disconnective, e.g. دفء/دفئًا *df'/dfŷAã* 'warmth' but جزء/جزءاً *jz'/jz'Aã* 'part'.

Vowel Initial Spelling Rule Since initial vowel diacritics cannot appear on their own, we add an additional Alif at the beginning of words with initial vowels. For example, *izdihAru* is modified to its final form ازدهار *AizdihAru*.

7. Evaluation

In this section, we evaluate MAGEAD's nominal implementation in terms of coverage and correctness.

7.1. Testing Corpora

We use the Penn Arabic Treebank (PATB) (Maamouri et al., 2004), specifically Part 3 v 3.1 (Maamouri et al., 2009), as our evaluation corpus. This release of the PATB only provides the analysis choice made by the annotators per word and not all the possible analyses (unlike previous releases). We use the files in the *pos/after-treebank* directory. The entries in these files are tokenized words with links to their surface segments (*INPUT STRING*). We recreate the surface untokenized forms by concatenating the segments and use a concatenation of the POS of each segment as the gold choice for the whole word. The *LEMMA* associated with the base word token (as opposed to clitics) is taken to be the lexeme of the whole word.

To get all possible analyses for each word, we use the Standard Arabic Morphological Analyzer (SAMA) (Graff et al., 2009) – which is version 3.1 of the BAMA analyzer (Buckwalter, 2004).

The test set we report on in this section is based on the nominals[5] appearing among the first 10,000 words in the PATB. These add up to a total of 4,870 nominal word tokens (non-unique). We exclude three kinds of analyses for specific reasons. First, since we are not evaluating our lexicon coverage, we exclude all analyses not in our lexicon. Second, we also exclude all analyses involving non-trilateral roots and non-templatic word stems since we do not even attempt to handle them in the current version of our rules. Finally, SAMA uses orthographic backoff to allow finding matches for words with common spelling errors, e.g., Hamzated Alif form confusion (ا آ أ إ *Â/Ă/Ā/A*), Ta-Marbuta/Ha confusion (ة/ه *ħ/h*) and Alif-Maqsura/Ya confusion (ى/ي *ý/y*). We only model the first of these three, the Hamzated Alif confusions, which represent over 97% of all cases of spelling confusion.[6] Analyses involving the other cases are excluded. The total number of exclusions is 682 word tokens (14% of all nominal word tokens). The remaining 4,188 word tokens (2,405 unique word types) constitute the test set we report on here. The same exclusion criteria are applied to the SAMA analyses.

We convert all analyses to a morphological feature-value representation that specifies for each word its proclitics, enclitic, gender, number, case and state, in addition to the diacritized word form and the lexeme. We do not distinguish among different types of nominals and do not consider non-nominal analyses. As such, the POS value is not included in this evaluation.

7.2. Testing Metrics

We consider five evaluation metrics that fall into three categories:

- Average Recall (**ATyR** for type recall, **AToR** for token recall): on average, what proportion of the analyses in the gold standard does MAGEAD get for each type/token respectively?
- Average Precision (**ATyP** for type precision, **AToP** for token precision): on average, what proportion of the analyses that MAGEAD gets are also in the gold standard for each type/token respectively?
- Context Token Recall (**CToR**): how often does MAGEAD get the contextually correct analysis for all token?

We do not give context precision figures, as MAGEAD does not determine the contextually correct analysis – this is a tagging problem (Habash and Rambow, 2005). Rather, we interpret the context token recall figures as a measure of how often MAGEAD gets the most important of the analyses (i.e., the correct one) for each token.

Words that return no analysis in MAGEAD are reported separately and not included in the metrics since precision in such cases is not defined.

For each of these metrics, we consider the following subsets of morphological aspects to measure against. These aspects

[5] nouns, adjectives, adverbs, comparative adjectives, noun numbers, adjective numbers and quantitative nouns

[6] Statistic computed over the Penn Arabic Treebank choices and their difference from input word.

are considered in isolation and in selected combinations for evaluation in the next section.

- DIAC is the fully diacritized word form. Due to inconsistencies between our gold standard and MAGEAD in how the Sukun diacritic (or no vowel diacritic) is used, we drop it and consider the absence of a diacritic in a fully diacritized word to be a sufficient representative of the Sukun.
- DIACNE is the same as DIAC except that the last (typically case marking) diacritic is dropped.
- LEXEME is the lemma representation of the lexeme.
- FEATCL is the set of clitic features: the definite article, the preposition clitics, the conjunction clitics and the pronominal clitics.[7]

The inflectional features (case, state, gender and number) are ignored in our evaluation because MAGEAD uses functional morphology while SAMA uses form-based morphology. We discuss this issue in more detail below in Section 7.4.

7.3. Quantitative Analysis

Of the 4,188 word tokens (2,405 unique word types) in our test set, MAGEAD fails to produce an analysis in 243 word tokens (5.8% of all word tokens) corresponding to 153 word types (6.4% of all word types). For 124 word types, MAGEAD produces analyses that do not have a matching MBC in the lexicon. For the remaining 29 word types, MAGEAD does not produce a single analysis. The results of our performance on the remaining 3,945 word tokens (2,252 word types) in terms of the metrics discussed above are presented in Table 1. We consider six sets of morphological aspects to compare.

Table 1: Results of MAGEAD's performance on analyzing Arabic nominals. Values are percentages.

	ATyP	ATyR	AToP	AToR	CToR
FEATCL	99.7	97.7	99.7	98.1	99.7
LEX	94.6	95.1	93.8	95.5	97.6
LEX+FEATCL	94.5	94.7	93.8	95.3	97.5
DIACNE	95.9	93.0	95.5	94.0	97.5
DIAC	94.5	72.3	94.5	73.2	94.7
DIACNE+LEX+FEATCL	92.8	91.3	92.2	92.4	95.9

With one exception, DIAC recall, all of our scores are in the 90s. All of the CToR scores are above 94%, which suggests that the errors MAGEAD is making have less effect on the correct in-context choices. Precision is higher than recall except for LEX and LEX+FEATCL cases. This is likely to be the result of lexicon errors or mismatches – see Section 7.4. The best performance we have is on FEATCL; this is significant as it suggests that this implementation of MAGEAD can be used for certain NLP applications such as tokenization and simple stemming. The sharp increase in ATyR and AToR when ignoring the last diacritic (in DIACNE compared to DIAC) together with the small increase in precision suggests that MAGEAD is failing to produce some diacritizations but is not over-generating. In fact, we see in the next section that the largest recall error type is related to an extraneous diacritization that SAMA produces and MAGEAD doesn't. The DIACNE+LEX+FEATCL combination still shows good performance, although there is a lot of potential for improvement still.

7.4. Qualitative Analysis

To understand in more detail the types of errors in our systems, we took a random sample of 100 word types, of which six cases produced no analysis (three complete failures and three lexicon misses). The failures seem to be mostly a result of missing/incorrect rules and in two cases incorrect lexical entries. For the remaining 94 cases, we classify all of the recall and precision errors when comparing all morphological features including gender-number-case-state. Under this harsh comparison condition, the ATyP is 47.5%, ATyR is 53.7%.

Recall Errors As for recall errors, only 22.2% (relative) are valid errors that are the result of missing lexical items, incorrect or missing morphophonemic rules or uncommon particles that MAGEAD does not model. All of these errors are recoverable and will be addressed in the future.

The rest of the errors are actually SAMA issues that should not reflect negatively on MAGEAD: the largest contributor (34.1% relative) is an additional analysis that SAMA adds which contains no case choice, when case otherwise would not be indicated with a written letter, e.g., producing the analysis *kitAb* 'book' in addition to *kitAb{u,a,i,ũ,ĩ}*. For some of these words, SAMA does not produce a case marker since the surface form of the case marker is *nil*, صحارى *SaHAraý* 'deserts [nom/acc/gen]'.

The second largest class (22.9% relative) are cases of mismatches in form and function of the gender and number features, e.g. دول *duwalu* 'states is masculine-singular in SAMA but it is feminine-plural in MAGEAD (plural of دولة *dawlaħ* 'state'). Finally, the last class of errors (20.8% relative) are cases where SAMA has an incorrect morphological prefix or lexeme. The large majority of these cases are the result of a very rare preposition تاء القسم 'Ta of Oath', which is reserved for a few number of words, being applied to any word. For example, the word تأليف *tÂlyf* commonly analyzable as 'authoring' is also analyzed as ت+أليف *t+Âlyf* 'by+domesticated', a very odd analysis.

Precision Errors As for precision errors, only 18% are real problems in MAGEAD. Half of these roughly are rule and morpheme-specification errors and the rest are lexicon errors. An example of a rule errors is a missing exception to the geminate rules that leads to extra vowel deletion, e.g. analyzing محقق 'investigator' as *muHaq∼qu* instead of *muHaq∼iqu*. An example of a wrong lexical entry is using the lemma غربا *γarbAã* 'lit. west [indef.acc]' but translated

[7] We ignore the different clitic POS values available in SAMA, e.g., the various forms of the proclitic +ف*fa*+ (coordinating/ subordinating conjunction or connective/ response-conditional particle) are all represented as *prt:f*.

commonly as 'westward' instead of the correct غرب *γarb* 'west'.

The rest of the precision errors (82%) are false errors that are the result of functional morphology modeling. Around 60% are due solely to the over-generation of definite-construct variants, while the rest are mostly due to mismatches of functional/form-based gender and number cases.

This analysis shows that the majority of errors are in fact either problems in the gold standard or a result of the discrepancy between different models of form-based/functional morphology. Interestingly, these results are comparable to our previously published error analysis of verbs (Habash and Rambow, 2006). None of the real errors are theoretically challenging to our framework and approach. They can and will be addressed in future work on MAGEAD.

8. Conclusion and Future Work

In this paper, we presented the details of an implementation of MSA nominals in MAGEAD, a morphological analyzer and generator for Modern Standard Arabic (MSA) and its dialects. A detailed evaluation of the current implementation comparing it to a commonly used morphological analyzer shows that it has good coverage and usability with high precision and recall.

In the immediate future, we plan to continue improving the morphological rules used in MAGEAD. We also plan to work on creating an improved lexicon and extending MAGEAD's nominal implementation to Arabic dialects.

Acknowledgments

The work reported in this paper was supported by NSF Award 0329163 and the DARPA GALE program, contracts HR0011-06-C-0023 and HR0011-08-C-0110. We would like to thank Tim Buckwalter, Otakar Smrž, Richard Sproat and Ryan Roth for helpful discussions.

9. References

Imad Al-Sughaiyer and Ibrahim Al-Kharashi. 2004. Arabic Morphological Analysis Techniques: A Comprehensive Survey. *Journal of the American Society for Information Science and Technology*, 55(3):189–213.

Kenneth Beesley. 1996. Arabic Finite-State Morphological Analysis and Generation. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING-96)*, pages 89–94, Copenhagen, Denmark.

Ron Buckley. 2004. *Modern Literary Arabic: A Reference Grammar*. Librairie du Liban.

T. Buckwalter. 2004. Buckwalter Arabic Morphological Analyzer Version 2.0. Linguistic Data Consortium, University of Pennsylvania, 2002. LDC Cat alog No.: LDC2004L02, ISBN 1-58563-324-0.

Joseph Dichy and Ali Farghaly. 2007. Grammar-Lexis Relations in the Computational Morphology of Arabic. In A. van den Bosch and A. Soudi, editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.

Wolfdietrich Fischer. 2001. *A Grammar of Classical Arabic*. Yale Language Series. Yale University Press, third revised edition. Translated by Jonathan Rodgers.

David Graff, Mohamed Maamouri, Basma Bouziri, Sondos Krouna, Seth Kulick, and Tim Buckwalter. 2009. Standard arabic morphological analyzer (sama) version 3.1. Linguistic Data Consortium LDC2009E73.

Nizar Habash and Owen Rambow. 2005. Arabic Tokenization, Part-of-Speech Tagging and Morphological Disambiguation in One Fell Swoop. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 573–580, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Nizar Habash and Owen Rambow. 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. Association for Computational Linguistics.

Nizar Habash and Owen Rambow. 2007. Morphophonemic and orthographic rules in a multi- dialectal morphological analyzer and generator for arabic verbs. In *International Symposium on Computer and Arabic Language (ISCAL)*, Riyadh, Saudi Arabia.

Nizar Habash, Owen Rambow, and George Kiraz. 2005. Morphological Analysis and Generation for Arabic Dialects. In *Proceedings of the Workshop on Computational Approaches to Semitic Languages at 43rd Meeting of the Association for Computational Linguistics (ACL'05)*, pages 17–24, Ann Arbor, Michigan.

Nizar Habash, Abdelhadi Soudi, and Tim Buckwalter. 2007. On Arabic Transliteration. In A. van den Bosch and A. Soudi, editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.

Nizar Habash. 2004. Large Scale Lexeme Based Arabic Morphological Generation. In *Proceedings of Traitement Automatique des Langues Naturelles (TALN-04)*, pages 271–276. Fez, Morocco.

George Kiraz. 1994. Multi-tape Two-level Morphology: A Case study in Semitic Non-Linear Morphology. In *Proceedings of Fifteenth International Conference on Computational Linguistics (COLING-94)*, pages 180–186, Kyoto, Japan.

George Anton Kiraz. 2000. Multi-tiered nonlinear morphology using multi-tape finite automata: A case study on Syriac and Arabic. *Computational Linguistics*, 26(1):77–105.

Mohamed Maamouri, Ann Bies, Tim Buckwalter, and Wigdan Mekki. 2004. The penn arabic treebank : Building a large-scale annotated arabic corpus.

Mohamed Maamouri, Ann Bies, Seth Kulick, Fatma Gaddeche, Wigdan Mekki, Sondos Krouna, and Basma Bouziri. 2009. The penn arabic treebank part 3 version 3.1. Linguistic Data Consortium LDC2008E22.

Mehryar Mohri, Fernando C. N. Pereira, and Michael Riley. 2000. The design principles of a weighted finite-state transducer library. *Theoretical Computer Science*, 231:17–32, January.

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

Abdelhadi Soudi, Violetta Cavalli-Sforza, and Abderrahim Jamari. 2001. A Computational Lexeme-Based Treatment of Arabic Morphology. In *Proceedings of the Arabic Natural Language Processing Workshop, Conference of the Association for Computational Linguistics (ACL 2001)*, pages 50–57, Toulouse, France.

Richard Sproat. 1995. Lextools: Tools for finite-state linguistic analysis. Technical Report 11522-951108-10TM, Bell Laboratories.