A Morphological Analyzer for Egyptian Arabic

Nizar Habash and **Ramy Eskander** and **Abdelati Hawwari**
Center for Computational Learning Systems
Columbia University
New York, NY, USA
{habash,reskander,ahawwari}@ccls.columbia.edu

Abstract

Most tools and resources developed for natural language processing of Arabic are designed for Modern Standard Arabic (MSA) and perform terribly on Arabic dialects, such as Egyptian Arabic. Egyptian Arabic differs from MSA phonologically, morphologically and lexically and has no standardized orthography. We present a linguistically accurate, large-scale morphological analyzer for Egyptian Arabic. The analyzer extends an existing resource, the Egyptian Colloquial Arabic Lexicon, and follows the part-of-speech guidelines used by the Linguistic Data Consortium for Egyptian Arabic. It accepts multiple orthographic variants and normalizes them to a conventional orthography.

1 Introduction

Dialectal Arabic (DA) refers to the day-to-day native vernaculars spoken in the Arab World. DA is used side by side with Modern Standard Arabic (MSA), the official language of the media and education (Holes, 2004). Although DAs are historically related to MSA, there are many phonological, morphological and lexical differences between them. Unlike MSA, DAs have no standard orthographies or language academies. Furthermore, different DAs, such as Egyptian Arabic (henceforth, EGY), Levantine Arabic or Moroccan Arabic have important differences among them similar to those seen among Romance languages (Erwin, 1963; Cowell, 1964; Abdel-Massih et al., 1979; Holes, 2004). Most tools and resources developed for natural language processing (NLP) of Arabic are designed for MSA. Such resources are quite limited when it comes to processing DA, e.g., a state-of-the-art MSA morphological analyzer has been reported to only have 60% coverage of Levantine Arabic verb forms (Habash and Rambow, 2006). Most efforts to address this gap have been lacking. Some have taken a quick-and-dirty approach to model shallow morphology in DA by extending MSA tools, resulting in linguistically inaccurate models (Abo Bakr et al., 2008; Salloum and Habash, 2011). Others have attempted to build linguistically accurate models that are lacking in coverage (at the lexical or inflectional levels) or focusing on representations that are not readily usable for NLP text processing, e.g., phonological lexicons (Kilany et al., 2002).

In this paper we present the Columbia Arabic Language and dIalect Morphological Analyzer (CALIMA) for EGY.[1] We built this tool by extending an existing resource for EGY, the Egyptian Colloquial Arabic Lexicon (ECAL) (Kilany et al., 2002). CALIMA is a linguistically accurate, large-scale morphological analyzer. It follows the part-of-speech (POS) guidelines used by the Linguistic Data Consortium for EGY (Maamouri et al., 2012b). It accepts multiple orthographic variants and normalizes them to CODA, a conventional orthography for DA (Habash et al., 2012).

The rest of the paper is structured as follows: Section 2 presents relevant motivating linguistic facts. Section 3 discusses related work. Section 4 details the steps taken to create CALIMA starting with ECAL. Section 5 presents a preliminary evaluation and statistics about the coverage of CALIMA. Finally, Section 6 outlines future plans and directions.

[1] Although we focus on Egyptian Arabic in this paper, the CALIMA name will be used in the future to cover a variety of dialects.

Proceedings of the Twelfth Meeting of the Special Interest Group on Computational Morphology and Phonology (SIGMORPHON2012), pages 1–9,
Montréal, Canada, June 7, 2012.

2 Motivating Linguistic Facts

We present some general Arabic (MSA/DA) NLP challenges. Then we discuss differences between MSA and DA – specifically EGY.

2.1 General Arabic Linguistic Challenges

Arabic, as MSA or DA, poses many challenges for NLP. Arabic is a morphologically complex language which includes rich inflectional morphology, expressed both templatically and affixationally, and several classes of attachable clitics. For example, the MSA word وسيكتبونها *wa+sa+ya-ktub-uwna+hA*[2] 'and they will write it' has two proclitics (+و *wa+* 'and' and +س *sa+* 'will'), one prefix -ي *ya-* '3rd person', one suffix -ون *-uwna* 'masculine plural' and one pronominal enclitic +ها *+hA* 'it/her'. The stem *ktub* can be further analyzed into the root *ktb* and pattern *12u3*.

Additionally, Arabic is written with optional diacritics that primarily specify short vowels and consonantal doubling, e.g., the example above will most certainly be written as *wsyktbwnhA*. The absence of these diacritics together with the language's complex morphology lead to a high degree of ambiguity, e.g., the Standard Arabic Morphological Analyzer (SAMA) (Graff et al., 2009) produces an average of 12 analyses per MSA word.

Moreover, some letters in Arabic are often spelled inconsistently which leads to an increase in both sparsity (multiple forms of the same word) and ambiguity (same form corresponding to multiple words), e.g., variants of the Hamzated Alif, أ Â or إ Ă, are often written without their Hamza (ء *'*): ا *A*. and the Alif-Maqsura (or dot-less Ya) ى ý and the regular dotted Ya ي *y* are often used interchangeably in the word-final position (Buckwalter, 2007).

Arabic complex morphology and ambiguity are handled using tools for disambiguation and tokenization (Habash and Rambow, 2005; Diab et al., 2007).

[2]Arabic orthographic transliteration is presented in the HSB scheme (Habash et al., 2007): (in alphabetical order)
ا ب ت ث ج ح خ د ذ ر ز س ش ص ض ط ظ ع غ ف ق ك ل م ن ه و ي
A b t θ j H x d ð r z s š S D T Ď ς γ f q k l m n h w y
and the additional letters: ' ء, Â أ, Ă إ, Ā آ, ŵ ؤ, ŷ ئ, ى ý, ħ ة.

We distinguish between *morphological analysis*, whose target is to produce all possible morphological/POS readings of a word out of context, and *morphological disambiguation*, which attempts to tag the word in context (Habash and Rambow, 2005). The work presented in this paper is only about *morphological analysis*.

2.2 Differences between MSA and DA

Contemporary Arabic is in fact a collection of varieties: MSA, which has a standard orthography and is used in formal settings, and DAs, which are commonly used informally and with increasing presence on the web, but which do not have standard orthographies. DAs mostly differ from MSA phonologically, morphologically, and lexically (Gadalla, 2000; Holes, 2004). These difference are not modeled as part of MSA NLP tools, leaving a gap in coverage when using them to process DAs. All examples below are in Egyptian Arabic (EGY).

Phonologically, the profile of EGY is quite similar to MSA, except for some important differences. For example, the MSA consonants ق/ذ/ث *q/ð/θ* are generally pronounced in EGY (Cairene) as *'/z/s* (Holes, 2004). Some of these consonants shift in different ways in different words: e.g., MSA ذنب *ðanb* 'fault' and كذب *kiðb* 'lying' are pronounced *zanb* and *kidb*. EGY has five long vowels compared with MSA's three long vowels. Unlike MSA, long vowels in EGY predictably shorten under certain conditions, often as a result of cliticization. For example, compare the following forms of the same verb: شاف *šAf* /š**ā**f/ 'he saw' and شافها *šAf+hA* /š**a**fha/ 'he saw her' (Habash et al., 2012).

Morphologically, the most important difference is in the use of clitics and affixes that do not exist in MSA. For instance, the EGY equivalent of the MSA example above is وحيكتبوها *wi+Ha+yi-ktib-uw+hA* 'and they will write it'. The optionality of vocalic diacritics helps hide some of the differences resulting from vowel changes; compare the undiacritized forms: EGY *wHyktbwhA* and MSA *wsyktbwnhA*. In this example, the forms of the clitics and affixes are different in EGY although they have the same meaning; however, EGY has clitics that are not part of MSA morphology, e.g., the indirect pronominal object clitic (*+l+uh* 'for him')

وحيكتبوهاله *wi+Ha+yi-ktib-uw+hA+l+uh* 'and they will write it for him'. Another important example is the circumfix negation ش+ +ما *mA+* +š which surrounds some verb forms: ماكتبش *mA+katab+š* 'he did not write' (the MSA equivalent is two words: لم يكتب *lam yaktub*). Another important morphological difference from MSA is that DAs in general and not just EGY drop the case and mood features almost completely.

Lexically, the number of differences is very large. Examples include بس *bas* 'only', طربيزة *tarabayzaħ* 'table', مرات *mirAt* 'wife [of]' and دول *dawl* 'these', which correspond to MSA فقط *faqaT*, طاولة *TAwilaħ*, زوجة *zawjaħ* and هؤلاء *hawˆlA'*, respectively.

An important challenge for NLP work on DAs in general is the lack of an orthographic standard. EGY writers are often inconsistent even in their own writing. The differences in phonology between MSA and EGY are often responsible: words can be spelled as pronounced or etymologically in their related MSA form, e.g., كدب *kidb* or كذب *kiðb*. Some clitics have multiple common forms, e.g., the future particle ح *Ha* appears as a separate word or as a proclitic +ح/+ه *Ha+*/*ha+*, reflecting different pronunciations. The different spellings may add some confusion, e.g., كتبو *ktbw* may be كتبوا *katabuwA* 'they wrote' or كتبه *katabuh* 'he wrote it'. Finally, shortened long vowels can be spelled long or short, e.g., شفها/شافها *šAf+hA*/*šf+hA* 'he saw her'.

3 Related Work

3.1 Approaches to Arabic Morphology

There has been a considerable amount of work on Arabic morphological analysis (Al-Sughaiyer and Al-Kharashi, 2004; Habash, 2010). Altantawy et al. (2011) characterize the various approaches explored for Arabic and Semitic computational morphology as being on a continuum with two poles: on one end, very abstract and linguistically rich representations and morphological rules are used to derive surface forms; while on the other end, simple and shallow techniques focus on efficient search in a space of precompiled (tabulated) solutions. The first type is typically implemented using finite-state technology and can be at many different degrees of sophistication and detail (Beesley et al., 1989; Kiraz, 2000; Habash and Rambow, 2006). The second type is typically implemented using hash-tables with a simple search algorithm. Examples include the Buckwalter Arabic Morphological Analyzer (BAMA) (Buckwalter, 2004), its Standard Arabic Morphological Analyzer (SAMA) (Graff et al., 2009) incarnation, and their generation-oriented extension, ALMOR (Habash, 2007). These systems do not represent the morphemic, phonological and orthographic rules directly, and instead compile their effect into the lexicon itself, which consists of three tables for prefixes, stems and suffixes and their compatibilities. A prefix or suffix in this approach is a string consisting of all the word's prefixes and suffixes, respectively, as a single unit (including null affix sequences). During analysis, all possible splits of a word into compatible prefix-stem-suffix combination are explored. More details are discussed in Section 4.5. Numerous intermediate points exist between these two extremes (e.g., ElixirFM (Smrž, 2007)). Altantawy et al. (2011) describe a method for converting a linguistically complex and abstract implementation of Arabic verbs in finite-state machinery into a simple precompiled tabular representation.

The approach we follow in this paper is closer to the second type. We start with a lexicon of inflected forms and derive from it a tabular representation compatible with the SAMA system for MSA. However, as we do this, we design the tables and extend them in ways that capture generalizations and extend orthographic coverage.

3.2 Arabic Dialect Morphology

The majority of the work discussed above has focused on MSA, while only a few efforts have targeted DA morphology (Kilany et al., 2002; Riesa and Yarowsky, 2006; Habash and Rambow, 2006; Abo Bakr et al., 2008; Salloum and Habash, 2011; Mohamed et al., 2012). These efforts generally fall in two camps. First are solutions that focus on extending MSA tools to cover DA phenomena. For example, both Abo Bakr et al. (2008) and Salloum and Habash (2011) extended the BAMA/SAMA databases (Buckwalter, 2004; Graff et al., 2009) to accept DA prefixes and suffixes. Both of these efforts were interested in mapping DA text to some MSA-like form; as such they did not model DA lin-

guistic phenomena, e.g., the ADAM system (Salloum and Habash, 2011) outputs only MSA diacritics that are discarded in later processing.

The second camp is interested in modeling DA directly. However, the attempts at doing so are lacking in coverage in one dimension or another. The earliest effort on EGY that we know of is the Egyptian Colloquial Arabic Lexicon (ECAL) (Kilany et al., 2002). It was developed as part of the CALLHOME Egyptian Arabic (CHE) corpus (Gadalla et al., 1997) which contains 140 telephone conversations and their transcripts. The lexicon lists all of the words appearing in the CHE corpus and provides phonological, orthographic and morphological information for them. This is an important resource; however, it is lacking in many ways: the orthographic forms are undiacritized, no morpheme segmentations are provided, and the lexicon has only some 66K fully inflected forms and as such lacks general morphological coverage. Another effort is the work by Habash and Rambow (2006) which focuses on modeling DAs together with MSA using a common multi-tier finite-state-machine framework. Although this approach has a lot of potential, in practice, it is closer to the first camp in its results since they used MSA lexicons as a base. Finally, two previous efforts focused on modeling shallow dialectal segmentation using supervised methods (Riesa and Yarowsky, 2006; Mohamed et al., 2012). Riesa and Yarowsky (2006) presented a supervised algorithm for online morpheme segmentation for Iraqi Arabic that cut the out-of-vocabulary rates by half in the context of machine translation into English. Mohamed et al. (2012) annotated a collection of EGY for morpheme boundaries and used this data to develop an EGY tokenizer. Although these efforts model DA directly, they remain at a shallow level of representation (undiacritized surface morph segmentation).

We use the ECAL lexicon as a base for CALIMA and extend it further. Some of the expansion techniques we used are inspired by previous solutions (Abo Bakr et al., 2008; Salloum and Habash, 2011). For the morphological representation, we follow the Linguistic Data Consortium guidelines which extend the MSA POS guidelines to multiple dialects (Maamouri et al., 2006; Maamouri et al., 2012b). To address the problem of orthographic variations, we follow the proposal by Habash et al. (2012) who designed a conventional orthography for DA (or CODA) for NLP applications in the CALIMA databases. However, to handle input in a variety of spellings, we extend our analyzer to accept non-CODA-compliant word forms but map them only to CODA-compliant forms as part of the analysis.

4 Approach

We describe next the various steps for creating CALIMA starting with ECAL. The details of the approach are to some degree dependent on this unique resource; however, some aspects of the approach may be generalizable to other resources, and languages or dialects.

4.1 The Egyptian Colloquial Arabic Lexicon

ECAL has about 66K entries: 27K verbs, 36K nouns and adjectives, 1.5K proper nouns and 1K closed classes. For each entry, the lexicon provides a phonological form, an undiacritized Arabic script orthography, a lemma (in phonological form), and morphological features, among other information. There are 36K unique lemmas and 1,464 unique morphological feature combinations. The following is an example ECAL entry for the word مبيكلموش *mbyklmwš* 'he did not talk to him'.[3] We only show Arabic orthography, phonology, and lemma+features:

```
mbyklmwš
mabiykallimU$[4]
kallim:verb+pres-3rd-masc-sg+DO-3rd-masc-sg+neg
```

Our goal for CALIMA is to have a much larger coverage, a CODA-compliant diacritized orthography, and a morpheme-based morphological analysis. The next steps allow us to accomplish these goals.

4.2 Diacritic Insertion

First, we built a component to diacritize the ECAL undiacritized Arabic script entries in a way that is consistent with ECAL phonological form. This was implemented using a finite-state transducer (FST) that maps the phonological form to multiple possible

[3]The same orthographic form has another reading 'they did not talk' which of course has different morphological features.

[4]The phonological form as used in ECAL. For transcription details, see (Kilany et al., 2002).

diacritized Arabic script forms. The form that is the same as the undiacritized ECAL orthography (except for diacritics) is used as the diacritized orthography for the rest of the process. The FST consists of about 160 transformations that we created manually. All except for 100 cases are generic mappings, e.g., two repeated *b* consonants are turned into بّ *b*∼,[5] or a short vowel *u* can be orthographically a short vowel (just the diacritic *u*) or a long vowel *uw* which shortened. The exceptional 100 cases were specified by hand in the FST as complete string mappings. These were mostly odd spellings of foreign words or spelling errors. We did not attempt to correct or change the ECAL letter spelling; we only added diacritics.

After diacritization, we modify the Arabic orthography in the example above to: `mabiykal~imuwš`.

4.3 Morphological Tag Mapping

Next, we wrote rules to convert from ECAL diacritized Arabic and morphology to CODA-compliant diacritized Arabic and LDC EGY POS compliant tags. The rules fall into three categories: ignore rules specify which ECAL entries to exclude due to errors; correction rules correct for some ECAL entry errors; and prefix/suffix/stem rules are used to identify specific pairs of prefix/suffix/stem substrings and morphological features to map to appropriate prefix/suffix/stem morphemes, respectively. For stems, the majority of the rules also identify roots and patterns. Since multiple root-pattern combinations may be possible for a particular word, the appropriate root-pattern is chosen by enforcing consistency across all the inflected forms of the lemma of the word and minimizing the overall number of roots in the system. We do not use or report on root-patterns in CALIMA in this paper since this information is not required by the LDC tags; however, we plan on using them in future efforts exploiting templatic morphology.

At the time of writing this paper, the system included 4,632 rules covering all POS. These include 1,248 ignore rules, 1,451 correction rules, 83 prefix rules, and 441 suffixes rules. About 1,409 stem rules are used to map core POS tags and identify templatic roots and patterns. Some rules were semi-automatically created, but all were manually checked. The rules are specified in a simple format that is interpreted and applied by a separate rule processing script. Developing the script and writing the rules took about 3 person-months of effort.

As an example, the following three rules are used to handle the circumfix *ma++š* 'not' and the progressing particle *bi+*.

```
PRE: ma,+neg  => ϕ,+neg  >> mA/NEG_PART#
PRE: bi,+pres => ϕ,+subj >> bi/PROG_PART+
SUF: š,+neg   => ϕ,ϕ     >> +š/NEG_PART
```

The input to the rule processor is a pair of surface form and morphological features. Each rule matches on a surface substring and a combination of morphological features (first two comma-separated tokens in the rule) and rewrites the parts it matched on (second two comma-separated tokens in the rule after =>). The type of the rule, i.e. prefix or suffix rule, determines how the matching is applied. In addition, the rule generates a substring of the target tag (last token in the rule). The first and third rules above handle a circumfix; the *+neg* feature is not deleted in the first rule (which handles the prefix) to allow the third rule (which handles the suffix) to fire. The second rule rewrites the feature *+pres* (present tense) as *+subj* (subjunctive) which is consistent with the form of the verb after removing the progressive particle *bi+*. After applying these rules in addition to a few others, the above example is turned into CODA and EGY POS compliant forms (# means word boundary):[6]

```
mA#bi+yi+kal~im+huw+š
NEG_PART#PROG_PART+IV3MS+IV+IVSUFF_DO:3MS+NEG_PART
```

The stem rules, whose results are not shown here, determine that the root is *klm* and the pattern is *1a22i3*.

We extended the set of mapped ECAL entries systematically. We copied entries and modified them to include additional clitics that are not present with all entries, e.g., the conjunction +ف *fa+* 'then', and the definite article +ال *Al+*.

4.4 Orthographic Lemma Identification

The ECAL lemmas are specified in a phonological form, e.g., in the example above, it is *kallim*. To determine the diacritized Arabic orthography spelling

[5]The ∼ diacritic or *Shadda* indicates the presence of consonantal doubling.

[6]CODA guidelines state that the negative particle ما *mA* is not to be cliticized except in a very small number of words (Habash et al., 2012).

of the lemma, we relied on the existence of the lemma itself as an entry and other ad hoc rules to identify the appropriate form. Using this technique, we successfully identified the orthographic lemma form for 97% of the cases. The remainder were manually corrected. We followed the guidelines for lemma specification in SAMA, e.g., verbs are cited using the third person masculine singular perfective form. For our example, the CALIMA lemma is *kal∼im*.

4.5 Table Construction

We converted the mapped ECAL entries to a SAMA-like representation (Graff et al., 2009). In SAMA, morphological information is stored in six tables. Three tables specify complex prefixes, complex suffixes and stems. A complex prefix/suffix is a set of prefix/suffix morphemes that are treated as a single database entry, e.g., *wi+Ha+yi* is a complex prefix made of three prefix morphemes. Each complex prefix, complex suffix and stem has a class category which abstract away from all similarly behaving complex affixes and stems. The other three tables specify compatibility across the class categories (prefix-stem, prefix-suffix and stem-suffix). We extracted triples of prefix-stem-suffix and used them to build the six SAMA-like tables. The generated tables are usable by the sama-analyze engine provided as part of SAMA3.1 (Graff et al., 2009). We also added back off mode support for NOUN_PROP.

Prefix/stem/suffix class categories are generated automatically. We identified specific features of the word's stem and affixes to generate specific affix classes that allow for correct coverage expansion. For example, in a complex suffix, the first morpheme is the only one interacting with the stem. As such, there is no need to give each complex suffix its own class category, but rather assign the class category based on the first morpheme. This allows us to automatically extend the coverage of the analyzer compared to that of the ECAL lexicon.

We also go further in terms of generalizations. For instance, some of the pronoun clitics in EGY have two forms that depend on whether the stem ends with vowel-consonant or two consonants, e.g., كتابها *kitAb+hA* 'her book' as opposed to ابنها *Aibn+ahA* 'her son'. This information is used to give the suffixes *+hA* and *+ahA* different class categories that are generalizable to other similarly behaving clitics.

At this stage of our system, which we refer to as CALIMA-core in Section 5.2, there are 252 unique complex prefixes and 550 unique complex suffixes, constructed from 43 and 86 unique simple prefixes and suffixes, respectively. The total number of prefix/suffix class categories is only 41 and 78, respectively.

4.6 Various Table Extensions

We extended the CALIMA-core tables in a similar approach to the extension of SAMA tables done by Salloum and Habash (2011). We distinguish two types of extensions.

Additional Clitics and POS Tags We added a number of clitics and POS tags that are not part of ECAL, e.g., the prepositional clitic +ع *Ea+* 'on' and multiple POS tags for the proclitic +ف *fa+* (as `CONJ`, `SUB_CONJ` and `CONNEC_PART`). Here we copied a related entry and modified it but kept its category class. For example, in the case of +ع *Ea+* 'on', we copied a prepositional clitic with similar distribution and behavior: +ب *bi+* 'with'.

Non-CODA **Orthography Support** We extended the generated tables to include common non-CODA orthographic variants. The following are some examples of the expansions. First, we added the variant +و *+w* for two suffixes: +ه *+uh* 'his/him' and +وا *+uwA* 'they/you [plural]'. Second, we added the form *ha+* for the future particle *Ha+*. Third, we introduced non-CODA-compliant Hamza forms as variants for some stems. Finally, some of the extensions target specific stems of frequently used words, such as the adverb برضه *brDh* 'also' which can be written as برده *brdh* and برضو *brDw* among other forms. The non-CODA forms are only used to match on the input word, with the returned analysis being a *corrected* analysis. For example, the word هيكتبو *hyktbw* returns the analysis حيكتبوا *Hyktbwa* `Ha/FUT_PART+yi/IV3P+ktib/IV+uwA/3P` 'they will write' among other analyses. The orthographic variations supported include 16 prefix cases, 41 stem cases, and eight suffix cases.

After all the clitic, POS tag and orthographic extensions, the total number of complex prefix entries

substantially increases from 352 to 2,421, and the number of complex suffix entries increases from 826 to 1,179. The number of stem entries increases from around 60K to 100K. The total number of recognizable word forms increases from 4M to 48M. We will refer to the system with all the extensions as CALIMA in Section 5.

5 Current Status

In this section, we present some statistics on the current status of the CALIMA analyzer. As with all work on morphological analyzers, there are always ways to improve the quality and coverage.

5.1 System Statistics

CALIMA has 100K stems corresponding to 36K lemmas. There are 2,421 complex prefixes and 1,179 complex suffixes (unique diacritized form and POS tag combinations). The total number of analyzable words by CALIMA is 48M words (compared to the 66K entries in ECAL). This is still limited compared to the SAMA3.1 analyzer (Graff et al., 2009) whose coverage of MSA reaches 246M words. See Table 1.

5.2 Coverage Evaluation

We tested CALIMA against a manually annotated EGY corpus of 3,300 words (Maamouri et al., 2012a) which was not used as part of its development, i.e., a completely blind test.[7] This evaluation is a POS recall evaluation. It is not about selecting the correct POS answer in context. We do not consider whether the diacritization or the lemma choice are correct or not. We compare CALIMA coverage with that of ECAL and a state-of-the-art MSA analyzer, SAMA3.1 (Graff et al., 2009). For the purpose of completeness, we also compare CALIMA-core and an extended version of SAMA3.1. The SAMA3.1 extensions include two EGY verbal proclitics (`Ha/FUT_PART` and `bi/PROG_PART`), some alternative suffixes that have no case or mood, and all the orthographic variations used inside CALIMA. We also compare the performance of different merged versions of SAMA3.1 and CALIMA. The results are presented in Table 1.

The second column in Table 1, *Correct Answer* indicates the percentage of the test words whose correct analysis in context appears among the analyses returned by the analyzer. The third column, *No Correct Answer*, presents the percentage of time one or more analyses are returned, but none matching the correct answer. The fourth column, *No Analysis*, indicates the percentage of words returning no analyses. The last column presents the total number of recognizable words in the system.

CALIMA provides among its results a correct answer for POS tags over 84% of the time. This is almost 27% absolute over the original list of words from ECAL and almost 21% absolute over the SAMA3.1 system. The various extensions in CALIMA give it about 10% absolute over CALIMA-core (and increase its size 10-fold). The limited extensions to SAMA3.1 reduce the difference between it and CALIMA-core by 50% relative. The overall performance of CALIMA-core merged with SAMA3.1 is comparable to CALIMA, although CALIMA has three times the number of no-analysis cases. Merging CALIMA and extended SAMA3.1 increases the performance to 92%, an 8% absolute increase over CALIMA alone. The final rate of no-analysis cases is only 1%.

5.3 Error Analysis

We analyzed a sample of 100 cases where no answer was found (*No Correct Answer* + *No Analysis*) for CALIMA+extended SAMA3.1. About a third of the cases (30%) are due to gold tag errors. Irrecoverable typographical errors occur 5% of the time, e.g., فين *fyn* instead of في *fy* 'in'. Only 2% of the cases involve a speech effect, e.g., جمييييل *jmyyyyyl* 'beautiful!!!'. A fifth of the cases (22%) involve a non-CODA orthographic choice which was not extended, e.g., the shortened long vowel in حجات *HjAt* instead of the CODA-compliant حاجات *HAjAt* 'things'. Another fifth of the cases (20%) are due to incomplete paradigms, i.e., the lemma exists but not the specific inflected stem. Finally, 21% of the cases receive a SAMA3.1 analysis that is almost correct, except for the presence of some mood/case mark-

[7] We ignore some specific choices made by the annotators, most importantly the use of ".VN" to mark verbal nominals, which is not even supported in SAMA3.1. We also ignore some annotation choices that are not consistent with the latest LDC guidelines (Maamouri et al., 2012b), such as using gender-marked plurals in some contexts, e.g., 3MP instead of 3P.

	Correct Answer	No Correct Answer	No Analysis	Words
ECAL	57.4%	14.7%	27.9%	66K
SAMA3.1	63.7%	27.1%	9.3%	246M
extended SAMA3.1	68.8%	24.9%	6.3%	511M
CALIMA**-core**	73.9%	10.8%	15.3%	4M
CALIMA	84.1%	8.0%	7.9%	48M
CALIMA**-core +** SAMA3.1	84.4%	12.8%	2.8%	287M
CALIMA **+ extended** SAMA3.1	92.1%	7.0%	1.0%	543M

Table 1: Comparison of seven morphological analysis systems on a manually annotated test set. The second column indicates the percentage of the test words whose correct analysis in context appears among the analyses returned by the analyzer. The third column presents the percentage of time one or more analyses are returned, but none matching the correct answer. The fourth column indicates the percentage of words returning no analyses. The last column presents the total number of recognizable words in the system.

ers that are absent in EGY, and which we did not handle. Overall, these are positive results that suggest the next steps should involve additional orthographic and morphological extensions and paradigm completion.

6 Outlook

We plan to continue improving the coverage of CALIMA using a variety of methods. First, we are investigating techniques to automatically fill in the paradigm gaps using information from multiple entries in ECAL belonging to different lemmas that share similar characteristics, e.g., hollow verbs in Form I. Another direction is to update our tables with less common orthographic variations, perhaps using information from the phonological forms in ECAL. Manual addition of specific entries will also be considered to fill in lexicon gaps. Furthermore, we plan to add additional features which we did not discuss such as the English and MSA glosses for all the entries in CALIMA. We also plan to make this tool public so it can be used by other people working on EGY NLP tasks, from annotating corpora to building morphological disambiguation tools.

Acknowledgments

This paper is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-12-C-0014. Any opinions, findings and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of DARPA. We thank Mohamed Maamouri, Owen Rambow, Seth Kulick, Mona Diab and Mike Ciul, for helpful discussions and feedback.

References

Ernest T. Abdel-Massih, Zaki N. Abdel-Malek, and El-Said M. Badawi. 1979. *A Reference Grammar of Egyptian Arabic*. Georgetown University Press.

Hitham Abo Bakr, Khaled Shaalan, and Ibrahim Ziedan. 2008. A Hybrid Approach for Converting Written Egyptian Colloquial Dialect into Diacritized Arabic. In *The 6th International Conference on Informatics and Systems, INFOS2008*. Cairo University.

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.

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

Kenneth Beesley, Tim Buckwalter, and Stuart Newton. 1989. Two-Level Finite-State Analysis of Arabic Morphology. In *Proceedings of the Seminar on Bilingual Computing in Arabic and English*, page n.p.

Tim Buckwalter. 2004. Buckwalter arabic morphological analyzer version 2.0. LDC catalog number LDC2004L02, ISBN 1-58563-324-0.

Tim Buckwalter. 2007. Issues in Arabic Morphological Analysis. In A. van den Bosch and A. Soudi, editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.

Mark W. Cowell. 1964. *A Reference Grammar of Syrian Arabic*. Georgetown University Press.

Mona Diab, Kadri Hacioglu, and Daniel Jurafsky, 2007. *Arabic Computational Morphology: Knowledge-based and Empirical Methods*, chapter Automated Methods for Processing Arabic Text: From Tokenization to Base Phrase Chunking. Springer.

Wallace Erwin. 1963. *A Short Reference Grammar of Iraqi Arabic*. Georgetown University Press.

Hassan Gadalla, Hanaa Kilany, Howaida Arram, Ashraf Yacoub, Alaa El-Habashi, Amr Shalaby, Krisjanis Karins, Everett Rowson, Robert MacIntyre, Paul Kingsbury, David Graff, and Cynthia McLemore. 1997. CALLHOME Egyptian Arabic Transcripts. In *Linguistic Data Consortium, Philadelphia*.

Hassan Gadalla. 2000. *Comparative Morphology of Standard and Egyptian Arabic*. LINCOM EUROPA.

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.

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.

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, Mona Diab, and Owen Rabmow. 2012. Conventional Orthography for Dialectal Arabic. In *Proceedings of the Language Resources and Evaluation Conference (LREC)*, Istanbul.

Nizar Habash. 2007. Arabic Morphological Representations for Machine Translation. In Antal van den Bosch and Abdelhadi Soudi, editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Kluwer/Springer.

Nizar Habash. 2010. *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers.

Clive Holes. 2004. *Modern Arabic: Structures, Functions, and Varieties*. Georgetown Classics in Arabic Language and Linguistics. Georgetown University Press.

H. Kilany, H. Gadalla, H. Arram, A. Yacoub, A. El-Habashi, and C. McLemore. 2002. Egyptian Colloquial Arabic Lexicon. LDC catalog number LDC99L22.

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, Mona Diab, Nizar Habash, Owen Rambow, and Dalila Tabessi. 2006. Developing and Using a Pilot Dialectal Arabic Treebank. In *Proceedings of the Language Resources and Evaluation Conference (LREC)*, Genoa, Italy.

Mohamed Maamouri, Ann Bies, Seth Kulick, Dalila Tabessi, and Sondos Krouna. 2012a. Egyptian Arabic Treebank Pilot.

Mohamed Maamouri, Sondos Krouna, Dalila Tabessi, Nadia Hamrouni, and Nizar Habash. 2012b. Egyptian Arabic Morphological Annotation Guidelines.

Emad Mohamed, Behrang Mohit, and Kemal Oflazer. 2012. Annotating and Learning Morphological Segmentation of Egyptian Colloquial Arabic. In *Proceedings of the Language Resources and Evaluation Conference (LREC)*, Istanbul.

Jason Riesa and David Yarowsky. 2006. Minimally Supervised Morphological Segmentation with Applications to Machine Translation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas (AMTA06)*, pages 185–192, Cambridge,MA.

Wael Salloum and Nizar Habash. 2011. Dialectal to Standard Arabic Paraphrasing to Improve Arabic-English Statistical Machine Translation. In *Proceedings of the First Workshop on Algorithms and Resources for Modelling of Dialects and Language Varieties*, pages 10–21, Edinburgh, Scotland.

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