Department of
Information Engineering
and Computer Science
DISI

UNIVERSITY
OF TRENTO - Italy

DISI - Via Sommarive 14 - 38123 Povo - Trento (Italy)
http://www.disi.unitn.it

A Single-Model Approach for Arabic Segmentation, POS-Tagging and Named Entity Recognition

Abed Alhakim Freihat1, Gabor Bella,
Hamdy Mubarak, and Fausto Giunchiglia

April 25-26, 2018

Technical Report # DISI-18-008

In proceedings of the second International Conference on Natural Language and Speech Processing ICNLSP 2018 – Algiers, Algier

A Single-Model Approach for Arabic Segmentation, POS-Tagging and Named Entity Recognition

Abed Alhakim Freihat[1], Gabor Bella[1], Hamdy Mubarak[2], and Fausto Giunchiglia[1]

[1] University Of Trento, 38122 Trento, Italy,
{abed.freihat, gabor.bella,fausto.giunchiglia}@unitn.it
[2] Qatar Computer Research Institute, Doha, Qatar
hmubarak@hbku.edu.qa

Abstract. This paper presents an entirely new, one-million-word annotated corpus for a comprehensive, machine-learning-based preprocessing of text in Modern Standard Arabic. Contrarily to the conventional pipeline architecture, we solve the NLP tasks of word segmentation, POS tagging and named entity recognition as a single sequence labeling task. This single-component configuration results in a faster operation and is able to provide state-of-the-art precision and recall according to our evaluations. The fine-grained output tag set output by our annotator greatly simplifies downstream tasks such as lemmatization. Provided as a trained OpenNLP component, the annotator is publicly free for research purposes.

Keywords: NLP; segmentation, POS-tagging, lemmatization, named entity recognition, machine learning

1 Introduction

Common natural language understanding tasks, such as information retrieval, word sense disambiguation, or query answering, are usually built on top of a set of basic NLP preprocessing operations. These operations are supposed to bring text to a more canonical form with dictionary words (lemmas) and named entities clearly identified. The precise solutions applied depend greatly on the language; however, state-of-the-art approaches typically involve a pipeline of components, such as a part-of-speech tagger, a morphological analyzer, a lemmatizer and a named entity recognizer (NER). Compared to English, both lemmatization and NER are harder for Arabic text: for the former because of the inflectional complexity and ambiguity inherent to the language, and for the latter because Arabic does not mark named entities by capitalization among other reasons.

There has been extensive research on solving each of the tasks mentioned above. In the case of Arabic POS tagging, the approaches are typically based

on statistical classifiers such as SVM [1, 2], sometimes combined with rule-based methods [3] or with a morphological analyzer [4–6]. The idea of POS tagging applied to unsegmented words has been investigated in [5] and in [7].

Likewise, for NER several solutions and tools have been reported. They can be classified as rule-based systems such as the approach presented in [8], machine-learning-based ones such as [9, 10], and hybrid systems such as [11]. The correlation between NER and POS tagging is illustrated in [12].

Good-quality Arabic annotated corpora for machine learning are few and far between. The *Penn Arabic Treebank* [13] is a non-free, half-a-million-words annotated corpus, destined for POS tagging and syntactic parsing, on which large number of research results are based. The KALIMAT corpus[3] [14], while freely available, is a silver standard corpus that reports a POS tagging accuracy of 96%.

In terms of NLP pipeline architecture, most existing solutions perform the aforementioned tasks as a cascade of several processing steps. For example, POS-tagging in FARASA [15, 16] and in MADAMIRA [17] supposes that word segmentation has been done as a previous step. Segmentation, in turn, relies on further preprocessing tasks such as morphological analysis in MADAMIRA.

Likewise, NER and lemmatization are often implemented as separate downstream tasks that rely on the results of POS tagging, base phrase chunking and morphological analysis. In several Arabic pipelines in the literature [18], however, upstream tasks such as POS tagging are implemented in a coarse-grained manner, which amounts to delegating the resolution of certain cases of ambiguity to downstream components. For example, by using single VERB and PART tags, the POS tagger in [1] avoids challenging ambiguities in Arabic verbs and particles respectively. Consequently, a further downstream component is needed for morphological disambiguation, e.g., to find out whether تعرف is an imperative (تَعَرَّفْ/*recognize*), past (تَعَرَّفَ/*recognized*), or present tense verb (تَعْرِفُ/*you know or she nows*); whether the noun سحب is singular (in which case it means *withdrawal*) or plural (meaning *clouds*); or whether أن is an accusative (أَنَّ) or a subordinate particle (أَنْ).

In this paper, we present a corpus-based approach that performs word segmentation, POS tagging and named entity recognition as a single processing component and without any other auxiliary or preprocessing tool. Such a single-step solution has several advantages:

- it is faster to execute than a running several machine learning models in series;
- it is easier to reuse as part of a natural language understanding application;

[3] `https://sourceforge.net/projects/kalimat/`

– it does not suffer from the problem of cumulative errors that are inherent to solutions that solve the same tasks in series.

Another design goal of our corpus was to provide a rich output by using fine-grained POS tags in the training corpus. This way, a great deal of ambiguity in Arabic text is resolved by our tool and subsequent operations such as lemmatization are largely simplified.
The tool can be tested online[4] and it is free for research upon request.

The rest of the paper is organized as follows. Section 2 highlights challenging cases of ambiguity in Arabic that are solved by our tool. Section 3 provides a high-level overview and principles of our solution, followed by a detailed presentation of the tag set used. Section 4 presents the corpus building process. In section 5 we demonstrate how the output of the tool serves downstream tasks such as lemmatization and named entity extraction. Sections 6 and 7 present the evaluation and the conclusions respectively.

2 Ambiguity in Arabic

For the purpose of illustrating the challenging cases that low-level NLP tasks such as word segmentation or lemmatization typically need to solve, in the following we list some common examples of ambiguity in Arabic.

Ambiguity in word segmentation. Certain words can be segmented into morphemes in more than one valid way. In such cases the correct segmentation can only be determined in context. In table 1 we list some common examples of ambiguity that occur at the segmentation level.

Ambiguity in POS tagging. While correct segmentation decreases the ambiguity in Arabic text, polysemy and the lack of short vowels result in morphemes having multiple meanings with distinct parts of speech. In table 2 we show some examples of this kind.

Ambiguity in fine-grained POS-tags Even with correct segmentation and POS tagging, challenging cases of ambiguity still remain at the fine-grained POS-tags level, taking into consideration that MSA words are overwhelmingly written without diacritics. An important point here is that our methodology reduces the ambiguity but it is not a word sense disambiguation method. Thus, we do not consider transitive/ditransitive verb ambiguity. For example, verbs such as علم (عَلِمَ/*knew* or عَلَّمَ/*taught*) remain ambiguous according to our current annotation tag set. In the following, we list some ambiguity examples that we deal with at the fine-grained POS-tags level.

– Passive voice vs active voice verbs: Many verbs in Arabic have the same form in the active or passive voice cases. Verbs like سجّل/*reported, or has*

[4] http://www.arabicnlp.pro/alp/

Table 1. Ambiguity examples at the segmentation level

Ambiguity	**Example**
Nouns vs conjunction+pronoun	وهن/*weakness* vs وهن/*and they (feminine)*
Noun vs conjunction+verb	وحل/*mud* vs وحل/*and (he) solved*
Noun vs conjunction+noun	وصفة*receipt* vsوصفة/*and (a) character*
Noun vs singular noun+pronoun	كتابي/*two books (in genitive)* vs كتابي/*my book*
Noun vs preposition+noun	لسعة/*sting* vs لسعة/*for capacity*
Proper noun vs preposition+noun	بعقوبة/*a city in Iraq* vs بعقوبة/*with punishment*
Proper noun vs conjunction+noun	وهران/*a city in Algeria* vs وهران/*and two cats*
Proper noun vs definite article+ noun	الباب/*a city in Syria* vs الباب/*the door*
Noun vs interrogative particle+ negation particle	ألم/*pain* vs ألم/*did I not*
Adjective vs noun+pronoun	جانبي/*lateral* vs جانبي/*my side*
Adjective vs preposition+noun	بحرية/*nautical* vs بحرية/*to freedom*
Verb vs conjunction+pronoun	فهم/*(he) understood* vs فهم/*and they (masculine)*
Verb vs conjunction+verb	وفر/*saved* vs وفر/*and (he) escaped*
Verb vs verb+pronoun	علمنا/*we knew* vs علمنا/*(he) taught us*
Verb vs interrogative particle+verb	أتذكر/*(I) remember* vs أتذكر/*do (you) remember*

Table 2. Ambiguity examples at the POS-tagging level

Ambiguity	**Example**
Verb vs noun	حمل/*carried* vs حمل/*carrying*
Verb vs comparative	أثقل/*overburdened* vs اثقل/*heavier*
Verb vs adjective	سهل/*facilitate* vs سهل/*easy*
Verb/noun vs particle	لم/*gathered* vs لم/*not*
Verb vs number	تسع/*expanded* vs تسع/*nine*
Verb vs proper noun	طلعت/*rised* vs طلعت/*Talat*
Noun vs number	سبع/*lion* vs سبع/*seven*
Noun vs proper noun	إحسان/*philanthropy* vs إحسان/*Ehsan*
Adjective vs noun	نوعية/*qualitative* vs نوعية/*quality*
Adjective vs proper noun	جميل/*nice* vs جميل/*Jamil*
Interrogative particle vs relative pronoun	According to their position in the sentence
Particle ambiguity in أن	أنْ/*subordinating* vs أنَّ/*accusative*
Particle ambiguity in إن	إنْ/*conditional* vs إنَّ/*accusative*
Particle ambiguity in لم	لمْ/*negation* vs لمَ/*interrogative*
Particle ambiguity in ما	ما/*negation* vs ما/*interrogative*

been reported can be only through the context disambiguated.

– Past vs present tense verbs: We have the following verb tense ambiguities:
  - The same verb word form that denotes a verb in first person singular present denotes (another) verb in third person singular masculine past. Consider for example the verb أجمل which can be أُجْمِلُ/*(I) illustrate* can also be أَجْمَلَ/*(he) illustrated*.
  - Third person singular feminine present verb form denotes (another) verb in third person singular masculine past. Consider for example the verb تحمل which can be تَحْمِلِ/*(she) carries*, can be also تَحَمَّلَ/*(he) sustained*.

– Imperative verbs:
  - Imperative verb (second person singular masculine) form can be read as third person singular masculine past tense verb. For example the verb تعرف which may be an imperative verb (تَعَرَّفْ/*recognize*) or a past tense verb (تَعَرَّفَ/*(he) recognized*).
  - Imperative verb (second person plural masculine) form can be read as third person plural masculine past tense verb. It can be also third person plural masculine present tense. For example the verb تعرفوا which may be an imperative verb (تَعَرَّفُوا/*recognize*), a past tense verb (تَعَرَّفُوا/*(they) recognized*), or a present tense verb in cases like (كَيْ تَعْرِفُوا/*so that (you) know*).
  - Imperative verb (second person singular feminine) form can be read as second person singular feminine present tense verb (after some particles). For example the same form تعرفي can be second person singular feminine imperative (تَعَرَّفِي/*recognize*) or second person singular feminine present tense after subordination particles such as in the case (كَيْ تَعْرِفِي/*so that you know*).

– **Noun ambiguities**
  - Singular vs plural nouns: In Arabic, there are several word forms that denote (different) singular and plural nouns. For example the word سحب denotes the singular noun سَحْب/*dragging* and the plural noun سُحُب/*clouds*.
  - Dual vs singular nouns: The ا accusative case ending in Arabic leads to dual singular ambiguity. For example, the word form كتابا may be read as singular noun كِتَابًا/*one book* or dual كِتَابَا/*two books* (in genitive dual cases such as كِتَابَا العُلُوم).

- Dual vs plural nouns: Dual form nouns and masculine plural noun in general are ambiguous. For example the word مؤمنين can be read as مؤمِنَيْن/*dual form* or as مؤمِنِيْن/masculine plural form.
- Feminine vs masculine singular nouns: There are cases in which the same word form denotes singular but with different gender For example the word قدم can be feminine قَدَم/*foot* or masculine قِدَم/*antiquity*.

Ambiguity in named entity recognition. Below we present two examples of ambiguity related to NER, referring to reader to [19] for a more detailed treatise on the matter.

– Inherent ambiguity in named entities: It is possible for a word or a sequence of words to denote named entities that belong to different classes. For example واشنطن denotes both a person and location. Another problem is that it is frequent to name organizations and establishments after the name of persons. For example, جامعة الملك عبد الله للعلوم التقنية/
King Abdullah University of Science and Technology.

– Ellipses: Ellipses (omitting parts of nominal phrases and entity names) contribute to the high ambiguity of natural languages. Considering the lack of orthographic features in Arabic, ellipses increase the ambiguity. For example, a text about البحر الأبيض المتوسط/*The Mediterranean Sea* mentions it explicitly at the beginning of the text. After that, it may omit البحر الأبيض/the White Sea and refers to it by المتوسط/*the Mediterranean*. This word is used mostly as an adjective (which means *the average*) and there is no orthographic triggers that may disambiguate the entity from the adjective token.

3 Annotation Design

The core idea of our approach is to combine three linguistically related tasks; part-of-speech tagging, named entity recognition and word segmentation; into a single task that we solve through supervised sequence labeling. A further design goal is to provide as much information as possible to the downstream NLP tasks such as lemmatization.

We achieve these goals by annotating a single large training corpus with complex and fine-grained tags that encode information with respect to part of speech, word segments and named entities. These three main kinds of tags are composed as follows:

```
<TAG>      ::= <PREFIX> <BASETAG> <POSTFIX>
<BASETAG>  ::= <POSTAG> | <NERTAG>
<PREFIX>   ::= <PREFIX> | <PROCLITIC> "+" | ""
<POSTFIX>  ::= <POSTFIX> | "+" <ENCLITIC> | ""
```

A tag is thus composed of a mandatory base tag and of zero or more (i.e., optional) proclitics and enclitics concatenated with the "+" sign indicating word segments. A base tag, in turn, is either a POS tag or a NER tag, but not both (in other words, we do not annotate named entities by part of speech). For example, the full tag of the word وبكتابه is a noun tag preceded by two proclitic tags (conjunction and preposition) and followed by a pronoun enclitic tag. The choice of our base and clitic tags was inspired by the coarse-grained tags used in MADA 2.32 [2] and 3.0 [17], as well as by the more fine-grained tags used in the Quran Corpus [20]. For compatibility with other NLP tools, mapping our tags to MADA 2.32, MADA 3.0 and Stanford [21] or any other tag sets is straightforward.

3.1 Base POS Annotation

The POS tag set consists of 58 tags classified into five main categories:

```
<POSTAG> ::= <NOUN> | <ADJECTIVE> | <VERB> | <ADVERB> |
<PREPOSITION>|<PARTICLE>
```

Nouns: The noun class has 13 tags as shown in table 3. The first 9 tags are fine-grained annotations of common nouns that we classify according to their number (singular, dual or plural) and gender (masculine, feminine or irregular). We use the feature *irregular* to annotate the irregular plural nouns. As it is the case in MADA, we consider quantifiers, numbers and foreign words as special noun classes. Following the Quran corpus, we consider pronouns, demonstrative pronouns and relative pronouns as special noun classes.

Adjectives: The adjective class has 9 tags as described in table 3. Similar to nouns, the first 7 tags are fine-grained annotations of adjectives that we classify according to their number and gender. As it is the case in MADA, we consider comparative adjectives and numerical adjectives as special adjective classes.

Verbs: The verb class contains 5 tags as described in table 3. The first four tags are fine-grained annotations of verbs that we classify according to their passive marking (active or passive) and tense (past or present). Annotating future tense in Arabic is explained in the particle class. For imperative verbs, we use the tag *IMPV*.

Adverbs: It is not clear in the modern Arabic linguistics community, whether *adverb* belongs to the Arabic part of speech system or not. In this study, we follow FARASA and MADA in considering adverbs as a category of the Arabic part of speech system, where we consider adverbs as *predicate* modifiers that we classify in three classes as shown in table 3.

Particles: This class contains 28 particles tags from the Quran Corpus tagset, that we list in table 4.

Table 3. Noun, adjective, verb, and adverb tags

Tag	Explanation	Example
SMN	Singular masculine noun	رجل , جبل ,كتاب
SFN	Singular feminine noun	بنت , قرية ,جريدة
DMN	Dual masculine noun	كتابي ,كتابا , كتابين ,كتابان
DFN	Dual feminine noun	قريتي ,قريتا , قريتين ,قريتان
PMN	Plural masculine noun	موظفي ,موظفو , موظفين ,موظفون
PFN	Plural feminine noun	كتابات ,حالات , إسهامات ,موظفات
PIN	Plural irregular noun	رجال ,بنات , قرى ,كتب
FWN	Foreign noun	سيناتور ,لابتوب , موبايل ,بنسلين
NQ	Quantifiers	أي ,بعض , كل ,جميع
NM	Numbers	اثنين ,اثنان , ١ ,واحد
PRO	Pronouns	ـها ,هي , ـه ,هو
DM	Demonstrative pronouns	تلك ,هؤلاء , هذان ,هذا
REL	Relative pronouns	اللواتي ,اللذان , التي ,الذي
SMAJ	Singular masculine adjective	سليم , قوي ,جميل
SFAJ	Singular feminine adjective	سليمة , قوية ,جميلة
DMAJ	Dual masculine adjective	جميلي ,جميلا , جميلين ,جميلان
DFAJ	Dual feminine adjective	سليمتان , قويتان ,جميلتين
PMAJ	Plural masculine adjective	جميلي ,جميلو , جميليون ,جميلين
PFAJ	Plural feminine adjective	سليمات , قويات ,جميلات
PIAJ	Plural irregular adjective	حمر , أقوياء ,أصحاء
AJCMP	Comparative adjectives	أسلم , أقوى ,أجمل
AJNM	ordinal adjectives	ثالث , ثاني ,أول
PRSV	Present verb (active voice)	يحمل , يسأل ,يقول
PSTV	Past verb (active voice)	حمل , سأل ,قال
PPRSV	Present verb (passive voice)	يحمل , يسأل ,يقال
PPSTV	Past verb (passive voice)	حمل , سئل ,أقيل
IMPV	Imperative verb	احمل , اسأل ,قل
T	Temporal adverb	بعد , أحيانا ,صباحا
LC	Location adverb	بعد , تحت ,فوق
AV	Adverb	تماما , خاصة ,يوميا

Table 4. Particle tags

Tag	Explanation	Example	Tag	Explanation	Example
D	Definite article	ال	*RES*	Restriction particle	إلا
C	Conjunctions	ف , أو ,و	*CERT*	Certainty particle	قد
P	Prepositions	ل , الى ,من	*SUR*	Surprise particle	إذا ,إذن
Q	Interrogative particles	كيف ,هل ,ماذا	*EMPH*	Emphatic particle	لـ
COND	Conditional particles	إنْ ,إذا ,لو	*PRP*	Purpose particle	لـ
NEG	Negation particles	لن ,لا ,لم	*RET*	Retraction particle	بل
ACC	accusative particles	لعل , لكن ,إن	*REM*	Resumption particles	و ,ف
SUB	subordinate particles	كي ,أنْ	*INTG*	Interogative particle	أ
FUT	Future particles	سـ, سوف	*PREV*	Preventive particle	ما
VOC	Vocative particles	يا ,ـم	*INC*	Inceptive particle	ـم ,ألا
ANS	Answer particles	نعم, كلالا,	*IMPV*	Imperative particle	ل
EXL	Explanation particles	أما ,أي	*PR*	Prohibition particle	لا
EXP	Exceptive particles	عدا ,سوى	*ABB*	Abbreviation	د (دكتور)
EXC	Exclamation particles	يا ,ما	*PX*	Punctuation	., :

3.2 Word Segment Annotation

We represent the morphology of words through complex tags that correspond to their internal structure. As shown above, the structure of a complex tag is

$$[\text{PROCLITIC+}]^* \text{ BASETAG } [\text{+ENCLITIC}]^*$$

where BASETAG is one of the base POS tags, ENCLITIC, when presents, stands for one or two clitic tags at the end of the word, and PROCLITIC, when presents, is the combination of one to three tags out of a set of the proclitic tags at the beginning of the word.

In our corpus, the number of distinct individual tags (including both simple and complex tags) is 358, as shown in table 5.

Table 5. Clitic tags

# clititcs	# tags	Examples
0	78	*SMN* كتاب
1	163	*C+PRSV* ويفعل
2	105	*C+PIN+PRO* وأصدقاؤه
3	12	*C+FUT+PRSV+PRO* وسيكتبه

Table 6. Named entity tags

Tag	Explanation	Example
B-PER, *I-PER*	Persons	نجيب_B-PER محفوظ_I-PER
B-LOC, *I-LOC*	Locations	البحر_B-LOC الأبيض_I-LOC المتوسط_I-LOC
B-ORG, *I-ORG*	Organizations	حزب_B-ORG الحرية_I-ORG والعدالة_I-ORG
B-EVENT, *I-EVENT*	Events	الحرب_B-EVENT العالمية_I-EVENT الثانية_I-EVENT
B-MISC, *I-MISC*	Miscellaneous	درب_B-MISC التبانة_I-MISC

3.3 Named Entity Annotation

```
<NERTAG>   ::= <POSITION> "-" <CLASS>
<POSITION> ::= "B" | "I"
<CLASS>    ::= "PER" | "LOC" | "ORG" | "EVENT" | "MISC"
```

Our approach does not mark named entities with POS tags; rather, we annotate them directly with named entity tags. Following the conventions of CONLL-2003[5], the NER tags provide both the class of the entity and its boundaries through indicating the positions of the tokens composing it. `B-` stands for *beginning*, i.e., the first token of the entity, while `I-` stands for *internal*, marking subsequent tokens of the same entity.

Our corpus currently distinguishes between the most common types of named entities: Persons, Locations, Organizations, Events and Others. We did not yet classify entity classes such as date, time, currency or measurement, nor subclasses of organizations (e.g., we do not differentiate between a football team and a university).

Thus the total number of NER tags is eight as shown in table 6; however, as shown earlier, NER tags can be further combined with clitic tags.

4 Annotation Method

In order to be free from licensing restrictions and modeling choices of existing resources such as the Penn Arabic Treebank [13], we assembled and hand-annotated an entirely new corpus. In the following, we present our corpus annotation method that takes into consideration the challenging ambiguities discussed in section 2.

[5] http://www.cnts.ua.ac.be/conll2003/ner/annotation.txt

4.1 Sources

The corpus was assembled from documents and text segments from a varied set of online resources in Modern Standard Arabic, such as medical consultancy web pages, Wikipedia, news portals, online novels and social media, as shown in table 7. The diversity of sources serves the purpose of increasing the robustness of the model with respect to changes in domain, style, form and orthography. For consistency within the corpus and with the type of texts targeted by our annotator, we removed all short vowels from the input corpora.

The current corpus consists of more than 90k annotated sentences with more than one million Arabic words and 120k punctuation marks.

Table 7. Resources used in the training corpus

Resource	**Proportion**
Aljazeera online	30%
Arabic Wikipedia	20%
Novels	15%
Alquds newspaper	10%
Altibbi medical corpus	10%
IslamWeb	5%
Social networks (Facebook, Twitter)	5%
Other resources	5%

4.2 The Annotation Process

The annotation was performed semi-automatically in two major steps:

1. annotation of a corpus large enough to train an initial machine learning model;
2. iterative extension of the corpus, where new sets of sentences were annotated by the initial trained model, added to the corpus after hand-correction, and the model retrained after each iteration.

Step 1 was an iterative process. It was bootstrapped using a 200-sentence gold-standard *seed corpus* that was fully hand-annotated. The goal of each iteration was to add a new set of 100 new sentences to the seed corpus, until about 15k sentences were tagged. Each iteration consisted of the following steps:

1.a For each word in the untagged corpus that was already tagged in the seed corpus, simply copy the tag from the tagged word (this of course can lead to wrong tagging as the process does not take the context into account; we fix such mistakes later).

1.b Find the 100 sentences with the highest number of tags obtained through replacement in the previous step.
1.c Manually verify, correct and complete the annotation of the sentences extracted in step 1.b.
1.d Add the annotated and verified sentences to the seed corpus and repeat.

At the end of step 1 many rounds of manual verification were performed on the annotated corpus.

In step 2, we extended the corpus in an iterative manner:

2.a Train an initial machine learning model from the annotated corpus.
2.b Use this model to label a new set of 100 sentences.
2.c Verify and correct the new annotations obtained.
2.d Add the newly annotated sentences to the annotated corpus and repeat.

For training the machine learning model we used the POS tagger component of the widely-known OpenNLP tool with default features and parameters. The annotation work was accomplished by two linguists, the annotator and a consultant who was beside the design of the tag set active in corrections and consultations especially in the first phase.

In Figure 1, we provide an example of a complete annotated sentence.

```
وقال_C+PSTV الخبير_D+SMN الفلكي_D+SMAJ بدار_P+B-ORG التقويم_I-ORG القطري_I-ORG بشير_B-PER
مرزوق_I-PER إن_ACC زخة_SFN شهب_PIN الرباعيات_D+PFN من_P الشهب_D+PIN المميزة_D+PIAJ جدا_AV
؛_PX وذلك_C+DM لأن_P+ACC معدل_SMN سقوطها_SMN+PRO في_P السماء_D+SFN عند_T ذروتها_SFN+PRO
يكون_PRSV كبيرا_SMAJ إذ_T يصل_PRSV إلى_P 80_NM شهابا_SMN في_P الساعة_D+SFN بحسب_P+SMN
تقدير_SMN خبراء_PIN الفلك_D+SMN المتخصصين_D+PMAJ في_P رصد_SMN الشهب_D+PIN ._PX
```

Fig. 1. An example of annotated sentence

5 Pipeline Integration

In this section, we show through a few typical examples how our annotator can be integrated into an Arabic NLP pipeline and its output effectively used for downstream language processing tasks.

The input of the annotator is expected to be UTF-8-encoded, whitespace-tokenized, but otherwise unannotated text in Modern Standard Arabic. We are also supposing that sentences have been previously split by the usual sentence-end markers (“.”, “!”, “?”, “...”) and newlines.

5.1 Word Segmentation

Word segmentation is executed based on the clitic tags provided by the annotator. The input of the method is a word and its corresponding tag. The output is a list of tokens that correspond to the PROCLITIC, BASETAG and the ENCLITIC tags. Given that clitics are linguistically determined, segmentation becomes a simple string splitting task. An example of the output of a segmentation tool we implemented is shown in figure 2.

س+يشهد سكان ال+نصف ال+شمالي من ال+كرة ال+أرضية أولى زخات ال+شهب ل+هذا ال+عام مطلع يناير / كانون الثاني ممثلة في " شهب ال+رباعيات " التي تصل ذروة+ها مساء الأربعاء ال+مقبل و+تمتد حتى بزوغ فجر الخميس .
و+قال ال+خبير ال+فلكي ب+دار التقويم القطري بشير مرزوق إن زخة شهب ال+رباعيات من ال+شهب ال+مميزة جدا ؛ و+ذلك ل+أن معدل سقوط+ها في ال+سماء عند ذروة+ها يكون كبيرا إذ يصل إلى 80 شهابا في ال+ساعة ب+حسب تقدير خبراء ال+فلك ال+متخصصين في رصد ال+شهب .

Fig. 2. Segmentation output example

5.2 Lemmatization

For lemmatization, we distinguish regular and irregular cases. Regular cases include singular nouns and adjectives, regular plural nouns and regular verbs. The fine-grained tags output by our annotator encode the morphological features of number and gender for nouns and adjectives, and the tense and voice (active or passive) for verbs. This makes lemmatization a straightforward task of *normalization* that removes inflectional prefixes and suffixes. For example, the verb form يحملوا is normalized into the lemma حمل. In the case of singular and regular plural nouns, it is sufficient to remove the plural suffixes and case endings. For example, the lemma of the dual noun form ولدين is ولد.

Irregular cases including broken plurals and irregular verbs, are more complex and are typically processed using finite-state transducers and/or lemmatization dictionaries, such as *AraComLex* [22] or the *OpenNLP lemmatizer*[6].

5.3 Named Entity Extraction

By *named entity extraction*, we mean the identification of the all entity occurrences present in the text and the extraction of their canonical names. Based on the NER tags output by our annotator, identifying the start and end of a named entity is a trivial task. Then through subsequent word segmentation, the clitics can be removed from the entity and the canonical name obtained. An example result of this process can be seen in figure 3.

<START:LOC> رئيس وزراء <END> <START:LOC> السعودية <END> إلى <START:DAY> السبت <END> يصل اليوم
<END> شهباز شريف <START:PER> للقاء شقيقه <END> نواز شريف <START:PER> السابق <END> باكستان
<START:LOC> كبير وزراء <END> إقليم البنجاب <START:LOC> والذي سبقه إلى هناك قبل أربعة أيام على متن ، طائرة سعودية خاصة .
<END> نواز شريف <START:PER> الباكستانية أن " <END> إكسبريس تريبيون <START:ORG> " وذكرت صحيفة سيجري كذلك اجتماعات على مستوى رفيع مع مسؤولين سعوديين .
. وتثير الزيارة علامات استفهام كثيرة تباينت أجوبة الصحف الباكستانية عنها
نواز <START:PER> إلى مصادر لم تُسمها القول إن زيارة <END> إكسبريس تريبيون <START:ORG> ونسبت
<END> إلى <START:LOC> السعودية <END> ما قد يفسر سبب إرسال " وهو <START:LOC> ذات طابع سياسي ،
<END> السعودية <START:PER> طائرة خاصة لنقل <END> شهباز <START:PER> " .
<END> <START:DAY> الثلاثاء <START:LOC> السعودية <END> وصل <START:PER> شهباز شريف <END> وأضافت أن
<END> بن علي <START:PER> الماضي والتقى عددا من المسؤولين ، من بينهم رئيس الوزراء التركي يلدرم <END> .

Fig. 3. Example text with named entities segmented and their boundaries within the text indicated.

[6] https://opennlp.apache.org/docs/1.8.4/manual/opennlp.html#tools.lemmatizer

6 Evaluation

To evaluate the performance of the proposed solution, we trained a machine learning model on the annotated corpus using the OpenNLP Maximum Entropy POS tagger with default features and *cutoff* = 3. We did not apply any preliminary normalization to the evaluation corpus. The evaluation corpus was taken from two sources: the *Aljazeera* news portal and the *Altibbi* medical consultancy web portal. The text contained 9990 tokens (9075 words and 915 punctuations). Manual validation of the evaluation results was performed. The per-task accuracy figures are shown in table 8.

Table 8. Evaluation results

Error type	Number of errors	Accuracy
Segmentation	25	99.7%
Coarse-grained POS	131	98.7%
Fine-grained POS	206	97.9%

The *Segmentation* error type refers to words that were not segmented correctly. The *Coarse-grained POS* error type refers to words that were correctly segmented but the base POS tag was wrong. This also includes incorrect named entity POS tags. Finally, the *Fine-grained POS* error type means that the word segmentation and the coarse-grained POS tag were correct but the fine-grained information within the tag was incorrect in one of the following ways:

- for nouns and adjectives: number/gender error;
- for verbs: tense error or passive/active voice error.

In some cases, the tag included more than one type of error. For example, the ومضرة_SFN tag (instead of C+SFAJ) includes segmentation and POS-tagging errors and was counted twice.

We also evaluated named entity recognition separately. Our evaluation corpus contains 674 named entity tags that denote 297 named entities (For example روبرت_B-PER واتسون_I-PER is one named entity that contains two named entity tags). The total number of true positives (correctly detected and classified named entities) was 265 (89.2% precision). The number of false negatives (assigning a non-named-entity tag, partial tagging, named entity boundary error, or a wrong named entity class applied) was 32 and the number of the false positives 15 (94.6% recall). F1-Measure = 91.8%. In table 9 we provide some examples of these errors. The evaluation data, and how to replicate the evaluation tests are available online[7].

[7] http://www.arabicnlp.pro/alp/eval.zip

Table 9. Evaluation results

Error type	Example
Non-NER tag	وفلوريدا_C+SMN instead of C+B-LOC
Partially tagged	وفيك_SMN سيجبيسما_I-PER instead of وفيك_C+B-PER سيجبيسما_I-PER
Boundary error	شركة_SFN سيبريان_B-ORG الروسية_I-ORG instead of شركة_SFN سيبريان_B-ORG الروسية_D+SFAJ
Wrong classification	غرين_B-LOC (in ودعا غرين) instead of غرين_B-PER
False positive	للنظم_P+B-ORG الآيكولوجية_I-ORG instead of للنظم_P+D+PIN الآيكولوجية_D+PIAJ

7 Conclusion and Future Work

In this study, we have demonstrated a single-corpus and single-model approach to solving low-level Arabic NLP tasks in Arabic. We showed how our tool manages to resolve a large number of cases of ambiguity in Arabic, facilitating subsequent operations such as lemmatization. A trained model and corresponding tools are available online and are free for research purposes upon request.

We are also planning to release the annotated corpus itself in the near future. The work is still in progress and can be improved in multiple manners.

Fine-tuning: while the tool reached very good results with default OpenNLP features, we believe that they can still be improved by customizing the classifier and the features, or using another machine or deep learning algorithms such as CRF and biLSTM;

Noun classification: in the current tag set, we do not differentiate between gerunds (المصدر) and other noun classes. For example, the noun قلب/*heart* is tagged the same as the gerund قلب/*overthrow*.

Named entity classification: the classification of named entities in our corpus is still incomplete and coarse-grained. For example, العباسيين, الشعب الألماني, or اللغة العربية are not classified as named entities. We plan to introduce new classes such as dates and currencies, as well as a finer-grained classification of organizations.

Other tools and corpora: we plan to use the same corpus and tag set to produce annotations for other NLP tasks such as chunking and parsing.

References

1. K. Darwish, H. Mubarak, A. Abdelali, and M. Eldesouki, "Arabic pos tagging: Don't abandon feature engineering just yet," in *Proceedings of the Third Arabic Natural Language Processing Workshop*, pp. 130–137, 2017.

2. M. Diab, "Second generation amira tools for arabic processing: Fast and robust tokenization, pos tagging, and base phrase chunking," in *2nd International Conference on Arabic Language Resources and Tools*, vol. 110, 2009.
3. S. Khoja, "Apt: Arabic part-of-speech tagger," in *Proceedings of the Student Workshop at NAACL*, pp. 20–25, 2001.
4. H. Aldarmaki and M. Diab, "Robust part-of-speech tagging of arabic text," in *ANLP Workshop 2015*, p. 173, 2015.
5. N. Habash and O. Rambow, "Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop," in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pp. 573–580, Association for Computational Linguistics, 2005.
6. M. Sawalha and E. Atwell, "Fine-grain morphological analyzer and part-of-speech tagger for arabic text," in *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, pp. 1258–1265, European Language Resources Association (ELRA), 2010.
7. E. Mohamed and S. Kübler, "Is arabic part of speech tagging feasible without word segmentation?," in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 705–708, Association for Computational Linguistics, 2010.
8. K. Shaalan and H. Raza, "Arabic named entity recognition from diverse text types," in *Proceedings of the 6th International Conference on Advances in Natural Language Processing*, GoTAL '08, (Berlin, Heidelberg), pp. 440–451, Springer-Verlag, 2008.
9. M. Althobaiti, U. Kruschwitz, and M. Poesio, "A semi-supervised learning approach to arabic named entity recognition," in *Recent Advances in Natural Language Processing, RANLP 2013, 9-11 September, 2013, Hissar, Bulgaria*, pp. 32–40, 2013.
10. K. Darwish, "Named entity recognition using cross-lingual resources: Arabic as an example," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, pp. 1558–1567, 2013.
11. S. Abdallah, K. F. Shaalan, and M. Shoaib, "Integrating rule-based system with classification for arabic named entity recognition," in *Computational Linguistics and Intelligent Text Processing - 13th International Conference, CICLing 2012, New Delhi, India, March 11-17, 2012, Proceedings, Part I*, pp. 311–322, 2012.
12. S. AlGahtani, "Arabic named entity recognition: A corpus-based study, ph.d. thesis." 2011.
13. M. Maamouri, A. Bies, T. Buckwalter, and W. Mekki, "The penn arabic treebank: Building a large-scale annotated arabic corpus," in *NEMLAR conference on Arabic language resources and tools*, vol. 27, pp. 466–467, 2004.
14. M. El-Haj and R. Koulali, "Kalimat a multipurpose arabic corpus," in *Second Workshop on Arabic Corpus Linguistics (WACL-2)*, pp. 22–25, 2013.
15. K. Darwish and H. Mubarak, "Farasa: A new fast and accurate arabic word segmenter.,"
16. A. Abdelali, K. Darwish, N. Durrani, and H. Mubarak, "Farasa: A fast and furious segmenter for arabic," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pp. 11–16, 2016.
17. A. Pasha, M. Al-Badrashiny, M. T. Diab, A. El Kholy, R. Eskander, N. Habash, M. Pooleery, O. Rambow, and R. Roth, "Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic.," in *LREC*, vol. 14, pp. 1094–1101, 2014.

18. M. Diab, "Second generation tools (amira 2.0): Fast and robust tokenization, pos tagging, and base phrase chunking," in *Proceedings of the Second International Conference on Arabic Language Resources and Tools* (K. Choukri and B. Maegaard, eds.), (Cairo, Egypt), The MEDAR Consortium, April 2009.
19. K. Shaalan, "A survey of arabic named entity recognition and classification," *Comput. Linguist.*, vol. 40, pp. 469–510, June 2014.
20. K. Dukes and N. Habash, "Morphological annotation of quranic arabic," in *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)* (N. C. C. Chair), K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, eds.), (Valletta, Malta), European Language Resources Association (ELRA), may 2010.
21. K. Toutanova, D. Klein, C. D. Manning, and Y. Singer, "Feature-rich part-of-speech tagging with a cyclic dependency network," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, (Stroudsburg, PA, USA), pp. 173–180, Association for Computational Linguistics, 2003.
22. M. Attia, P. Pecina, A. Toral, L. Tounsi, and J. van Genabith, "An open-source finite state morphological transducer for modern standard arabic," in *Proceedings of the 9th International Workshop on Finite State Methods and Natural Language Processing*, FSMNLP '11, (Stroudsburg, PA, USA), pp. 125–133, Association for Computational Linguistics, 2011.