Semeval 2007 Task 18: Arabic Semantic Labeling

Mona Diab
Columbia University
mdiab@cs.columbia.edu

Musa Alkhalifa
University of Barcelona
musa@thera-clic.com

Sabri Elkateb
University of Manchester
Sabri.Elkateb@manchester.ac.uk

Christiane Fellbaum
Princeton University
fellbaum@clarity.princeton.edu

Aous Mansouri
University of Colorado, Boulder
aous.mansouri@colorado.edu

Martha Palmer
University of Colorado, Boulder
martha.palmer@colorado.edu

Abstract

In this paper, we present the details of the Arabic Semantic Labeling task. We describe some of the features of Arabic that are relevant for the task. The task comprises two subtasks: Arabic word sense disambiguation and Arabic semantic role labeling. The task focuses on modern standard Arabic.

1 Introduction

Recent years have witnessed a surge in available resources for the Arabic language.[1] The computational linguistics community is just about starting to exploit these resources toward several interesting scientific and engineering goals. The Arabic language is interesting from a computational linguistic perspective. It is significantly different from English hence creating a challenge for existing technology to be easily portable to Arabic. The Arabic language is inherently complex due to its rich morphology and relative free word order. Moreover, with the existence of several interesting varieties, the spoken vernaculars, we are witnessing the emergence of written dialectal Arabic everyday on the web, however there are no set standards for these varieties.

We have seen many successful strides towards functional systems for Arabic enabling technologies, but we are yet to read about large Arabic NLP applications such as Machine Translation and Information Extraction that are on par with performance on the English language. The problem is not the existence of data, but rather the existence of data annotated with the relevant level of information that is useful for NLP. This task attempts a step towards the goal of creating resources that could be useful for such applications.

In this task, we presented practitioners in the field with challenge of labeling Arabic text with semantic labels. The labels constitute two levels of granularity: sense labels and semantic role labels. We specifically chose data that overlapped such that we would have the same data annotated for different types of semantics, lexical and structural. The overall task of Arabic Semantic Labeling was subdivided into 4 sub-tasks: Arabic word sense disambiguation (AWSD), English to Arabic WSD task (EAWSD), argument detection within the context of semantic role labeling, and argument semantic role classification.

Such a set of tasks would not have been feasible without the existence of several crucial resources: the `Arabic Treebank` (ATB) (Maamouri et al., 2004), the `Arabic WordNet` (AWN) (Elkateb et al., 2006), and the `Pilot Arabic Propbank` (APB).[2]

This paper is laid out as follows: Section 2 will describe some facts about the Arabic language; Section 3 will present the overall description of the tasks; Section 4 describes the word sense disambiguation task; Section 5 describes the semantic role labeling task.

2 The Arabic Language

In the context of our tasks, we only deal with MSA.[3]

Arabic is a Semitic language. It is known for its templatic morphology where words are made up of

[1] Author 1 is supported by DARPA contract Contract No. HR0011-06-C-0023. Authors 2, 3 and 4 are supported by the US Central Intelligence Service.

[2] Funded by DARPA subcontract to BBN Inc. to University of Colorado, LDC-UPenn and Columbia University.

[3] In this paper we use MSA and Arabic interchangeably.

Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), pages 93–98,
Prague, June 2007. ©2007 Association for Computational Linguistics

roots and affixes. Clitics agglutinate to words. For instance, the surface word و بحسناتهم *wbHsnAthm*[4] 'and by their virtues[fem.]', can be split into the conjunction *w* 'and', preposition *b* 'by', the stem *HsnAt* 'virtues [fem.]', and possessive pronoun *hm* 'their'. Arabic is different from English from both the morphological and syntactic perspectives which make it a challenging language to the existing NLP technology that is too tailored to the English language.

From the morphological standpoint, Arabic exhibits rich morphology. Similar to English, Arabic verbs are marked explicitly for tense, voice and person, however in addition, Arabic marks verbs with mood (subjunctive, indicative and jussive) information. For nominals (nouns, adjectives, proper names), Arabic marks case (accusative, genitive and nominative), number, gender and definiteness features. Depending on the genre of the text at hand, not all of those features are explicitly marked on naturally occurring text.

Arabic writing is known for being underspecified for short vowels. Some of the case, mood and voice features are marked only using short vowels. Hence, if the genre of the text were religious such as the Quran or the Bible, or pedagogical such as children's books in Arabic, it would be fully specified for all the short vowels to enhance readability and disambiguation.

From the syntactic standpoint, Arabic, different from English, is considered a pro-drop language, where the subject of a verb may be implicitly encoded in the verb morphology. Hence, we observe sentences such as اكل البرتقال *Akl AlbrtqAl* 'ate-[he] the-oranges', where the verb *Akl* encodes that the subject is a 3rd person masculine singular. This sentence is exactly equivalent to هو اكل البرتقال *hw Akl AlbrtqAl* 'he ate the-oranges'. In the Arabic Treebank (ATB), we observe that 30% of all sentences are pro-dropped for subject.

Also Arabic is different from English in that it exhibits a larger degree of free word order. For example, Arabic allows for subject-verb-object (SVO) and verb-subject-object (VSO) argument orders, as well as, OSV and OVS. In the ATB, we observe an equal distribution of both VSO and SVO orders each equally 35% of the time. An example of an SVO sentence is الرجال اكلوا البرتقال *AlrjAl AklwA AlbrtqAl* 'the-men ate-them the-oranges', this is contrasted with اكل الرجال البرتقال*Akl AlrjAl AlbrtqAl* 'ate the-men the-oranges'.

Arabic exhibits more complex noun phrases than English mainly to express possession. These constructions are known as *idafa* constructions. In these complex structures an indefinite noun is followed by a definite noun. For example, رجل البيت *rjl Albyt* 'man the-house' meaning 'man of the house'. Accordingly, MSA does not have a special prepositional use to express possession in a manner similar to English.

3 Overall Tasks Description

Given the differences between English and Arabic, we anticipate that the process of automatically tagging text with semantic information might take more than just applying an English semantic labeler to Arabic. With this in mind, we decided to design a set of tasks that target different types of semantic annotations. We designed an all-words style word sense disambiguation (WSD) task for all the nouns and verbs in Arabic running text. Moreover, we designed another task where the participants are asked to detect and classify semantic role labels (SRL) for a large portion of newswire text. The WSD texts are chosen from the same set used for SRL. All the data is from the Arabic Treebank III ver. 2 (ATB). The ATB consists of MSA newswire data from Annhar newspaper, from the months of July through November of 2002. The ATB is fully annotated with morphological information as well syntactic structural information. The released data for the subtasks is unvowelized and romanized using the Buckwalter transliteration scheme. The part of speech (POS) tag set used in the released data for both the WSD and the SRL sub-tasks is the reduced tag set that is officially released with the ATB.

4 Task: WSD

In the context of this task, word sense disambiguation is the process by which words in context are tagged with their specific meaning definitions from a predefined lexical resource such as a dictionary or taxonomy. The NLP field has gone through a very

[4] We use the Buckwalter transliteration scheme to show romanized Arabic (Buckwalter, 2002).

long tradition of algorithms designed for solving this problem (Ide and Veronis, 1998). Most of the systems however target English since it is the language with most resources. In fact a big push forward dawned on English WSD with the wide release of significant resources such as WordNet.

Arabic poses some interesting challenges for WSD since it has an inherent complexity in its writing system. As mentioned earlier, written MSA is underspecified for short vowels and diacritics. These short vowels and diacritics convey both lexical and inflectional information. For example, كلية *klyp* could mean three different things, ‘all’, ‘kidney’ and ‘college’. Due to the undiacritized, unvowelized writing system, the three meanings are conflated. If diacritics are explicitly present, we would observe a better distinction made between كليّة *kly∼p* ‘all’ or ‘college’, and كلية *klyp* ‘kidney’. Hence, full diacritization may be viewed as a level of WSD. But crucially, naturally occurring Arabic text conflates more words due to the writing system.

To date, very little work has been published on Arabic WSD. This is mainly attributed to the lack in lexical resources for the Arabic language. But this picture is about to change with the new release of an Arabic WordNet (AWN).

Arabic WordNet Arabic WordNet (AWN) is a lexical resource for modern standard Arabic. AWN is based on the design and contents of Princeton WordNet (PWN)(Fellbaum, 1998) and can be mapped onto PWN as well as a number of other wordnets, enabling translation on the lexical level to and from dozens of other languages.

AWN focuses on the the Common Base Concepts (Tufis, 2004), as well as extensions specific to Arabic and Named Entities. The Base Concepts are translated manually by authors 2 and 3 into Arabic. Encoding is bi-directional: Arabic concepts for all senses are determined in PWN and encoded in AWN; when a new Arabic verb is added, extensions are made from verbal entries, including verbal derivations, nominalizations, verbal nouns, etc.

To date, the database comprises over 8,000 synsets with over 15,000 words; about 1,400 synsets refer to Named Entities.

Task design With the release of the AWN, we set out to design a sub-task on Arabic WSD. The task had only trial and test data released in an XML compliant format marking instance, sentence and document boundaries. The relevant words are marked with their gross part of speech and underlying lemma and English gloss information.

The participants are required to annotate the chosen instances with the synset information from AWN. Many of the entries in AWN are directly mapped to PWN 2.0 via the byte offset for the synsets.

The two subtasks data comprised 1176 verb and noun instances: 256 verbs and 920 nouns. The annotators were only able to annotate 888 instances for both English and Arabic due to gaps in the AWN. Hence, the final data set comprised 677 nouns and 211 verbs. The gold standard data is annotated authors 2 and 3 of Arabic (the annotators who created the AWN). There was always an overlap in the data of around 300 instances. In the English Arabic WSD task, participants are provided with a specific English word in translation to an Arabic instance. They are also given the full English translation of the Arabic document. Unfortunately, there were no participants in the task.

5 Task: Semantic Role Labeling (SRL)

Shallow approaches to text processing have been garnering a lot of attention recently. Specifically, shallow approaches to semantic processing are making large strides in the direction of efficiently and effectively deriving tacit semantic information from text. Semantic Role Labeling (SRL) is one such approach. With the advent of faster and powerful computers, more effective machine learning algorithms, and importantly, large data resources annotated with relevant levels of semantic information `FrameNet` (Baker et al., 1998) and `ProbBank` corpora (Palmer et al., 2005), we are seeing a surge in efficient approaches to SRL (Carreras and Màrquez, 2005).

SRL is the process by which predicates and their arguments are identified and their roles defined in a sentence.

To date, most of the reported SRL systems are for English. We do see some headway for other languages such as German and Chinese. The systems for the other languages follow the successful models devised for English, (Gildea and Jurafsky, 2002;

Xue and Palmer, 2004; Pradhan et al., 2003). However, no SRL systems exist for Arabic.

Challenges of Arabic for SRL Given the deep difference between such languages, this method may not be straightforward.

To clarify this point, let us consider Figure 1.

It illustrates a sample Arabic syntactic tree with the relevant part of speech tags and arguments defined. The sentence is مشروع الامم المتحدة فرض مهلة نهائية ل إتاحة الفرصة امام قبرص *m$rwE AlAmm AlmtHdp frD mhlp nhAyp l AtAHp AlfrSp AmAm qbrS*. meaning 'The United Nations' project imposed a final grace period as an opportunity for Cyprus'. As we see in the figure, the predicate is *frD* 'imposed' and it has two numbered arguments: ARG0 is the subject of the sentence which is *m$rwE AlAmm AlmtHdp* 'United Nations project'; ARG1, in the object position, namely, *mhlp nhAyp* 'final grace period'. The predicate has an ARGM-PRP (purpose argument) in *l AtAHp AlfrSp AmAm qbrS* 'as an opportunity for Cyprus'.

As exemplified earlier in Section 2, there are several crucial structural differences between English and Arabic. These differences can make the SRL task much harder to resolve than it is for English.

Pro-drop could cause a problem for Arabic SRL systems that do not annotate traces.

Passivization is marked with a short vowel that hardly ever appears on unvocalized text.

The structural word order could create problems. For instance for a sentence such as بلغ الرجل الولد 'the man reached—told the boy', *Alrjl* 'the man' could be an ARG0 for the VSO, or ARG1 for an VOS. Or for the following structure الولد بلغ الرجل *Alwld blg Alrjl* 'the boy reached the man', *Alwld* 'the boy' could be an ARG0 if it were a SVO sentence, or could be an ARG1 if it were an OVS sentence.

Idafa constructions may cause problems for argument boundary detection systems unless the underlying parser is sensitive to these constructions. For example, in the sentence illustrated in Figure 1, the NP *m$rwE AlAmm AlmtHdp* 'the United Nations' project' is an *idafa* construction, so the scope of the NP has to cover all three words and then assign the ARG boundary to the correct NP.

Arabic Propbank Taking into consideration the possible challenges, an `Arabic Propbank` (APB) was created. APB comprises 200K words from ATB 3 version 2 annotating the proposition for each verb. The chosen verbs occur at least 12 times in the corpus covering 80% of the data. It provides semantic role annotations for 454 verbal predicates. The predicates are fully specified for diacritization hence no two lexically variant verbs are conflated. APB defines an overall 26 argument types. We have excluded here 4 of these argument types, three of which were absent from the training data and ARGM-TER which marks ATB errors. Once the verbs are chosen, the framers come up with frames based on a combination of syntactic and semantic behaviors expressed by the verb and its core arguments. The framers use their native intuition, look at a sample occurrence in the data, and use external sources to aid them in the frame-creating process. If the verb has more than one sense, it is divided into more than one frame depending on how it relates to its arguments. The arguments themselves are chosen based not only on what is deemed semantically necessary, but on frequency of usage, as well. Figure 1 shows an example predicate and its arguments annotated with semantic role labels.

Task Design The Arabic SRL task is split into an argument boundary detection task and an argument classification task. We released data for the 95 most frequent verbs. An important characteristic of the data-set is the use of unvowelized Arabic in the Buckwalter transliteration scheme. We released the gold standard parses in the ATB as a source for syntactic parses for the data. The data is annotated with the reduced Bies POS tag set (in the LDC ATB distribution). The data comprises a development set of 886 sentences, a test set of 902 sentences, and a training set of 8,402 sentences. The development set comprises 1710 argument instances, the test data comprises 1657 argument instances, and training data comprises 21,194 argument instances. For evaluation we use the official CoNLL evaluator (Carreras and Màrquez, 2005). The evaluation software produces accuracy, precision, recall and $F_{\beta=1}$ metrics.

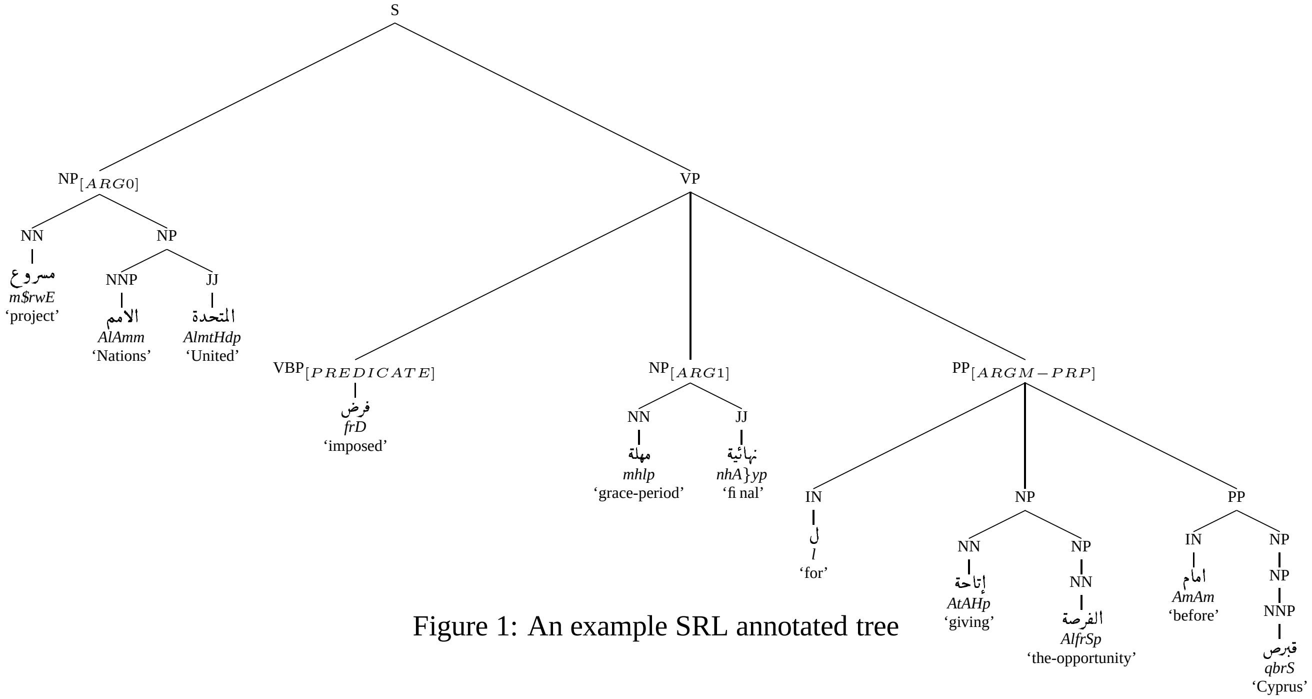

Figure 1: An example SRL annotated tree

5.1 Subtask : Argument Boundary Detection

In this task, the participating systems are expected to detect the boundaries of arguments associated with designated predicates. The systems are expected to identify the arguments with the correct level of scoping. For instance, in our running example sentence, the argument boundaries for the verb فرض *frD* 'imposed' are illustrated as follows: *[m$rwE AlAmm AlmtHdp]*$_{ARG}$ *[frD]*$_{Lemma:faroD}$ *[mhlp nhA}yp]*$_{ARG}$ *[l AtAHp AlfrSp AmAm qbrS]*$_{ARG}$. The three relevant arguments are *m$rwE AlAmm AlmtHdp* 'the United Nations Project', *mhlp nhA}yp* 'final grace-period', and *l AtAHp AlfrSp AmAm qbrS* 'as an opportunity for Cyprus'.

Only one system (CUNIT) participated in the subtask. CUNIT is an SVM based discriminative classification system based on different degrees polynomial kernels. The best CUNIT system (with degree 2 kernel) achieves an F$_{\beta=1}$ argument boundary detection score of 93.68% on the development data and 94.06% on the test data. We note that the results on the test data are higher than on the development data indicating that the test data is relatively easier.

5.2 Subtask: Argument Classification

In this task, the participating systems are expected to identify the class of the arguments detected in the previous step of argument boundary detection. In this sub task we have 22 argument types. Table 1 illustrates the different argument types and their distributions between the dev, train and test sets.

The most frequent arguments are ARG0, ARG1, ARG2 and ARGM-TMP. This is similar to what we see in the English Propbank. We note the additional ARG types with the extension STR. These are for stranded arguments. The tag STR is used when one constituent cannot be selected and an argument has two or more concatenated constituents. An example of this type of ARG is استقر في نيو يورك في بروكلين *{stqr fy nyw ywrk fy brwklyn* 'he settled in New York, in Brooklyn'. In this case, *fy nyw ywrk* 'in New York' is labeled ARG1 and *fy brwklyn* 'in Brooklyn' is labeled ARG1-STR.

Only one system (CUNIT) participated in the SRL subtask. CUNIT is an SVM based discriminative classification system based on different degrees polynomial kernels. The best CUNIT system (with degree 2 kernel) achieves an overall F$_{\beta=1}$ score for all arguments classification of 77.84% on the development data and 81.43% on the test data. It is worth noting that these results are run with the automatic argument boundary detection as an initial step. In both the test and the development results, the precision is significantly higher than the recall. For the development set precision is 81.31% and the recall

	#train	#dev	#test
ARG0	6,328	227	256
ARG0-STR	70	8	5
ARG1	7,858	702	699
ARG1-PRD	38	2	3
ARG1-STR	172	23	13
ARG2	1,843	191	180
ARG2-STR	32	5	4
ARG3	164	13	12
ARG4	15	0	4
ARGM	79	6	1
ARGM-ADV	994	103	115
ARGM-BNF	53	5	7
ARGM-CAU	89	12	11
ARGM-CND	38	6	3
ARGM-DIR	25	3	1
ARGM-DIS	56	8	5
ARGM-EXT	21	0	1
ARGM-LOC	711	82	61
ARGM-MNR	623	85	55
ARGM-NEG	529	76	39
ARGM-PRD	77	14	12
ARGM-PRP	343	42	27
ARGM-TMP	1,347	96	107
Total	21,194	1,710	1,657

Table 1: Distribution of training, development and test instances on the different role types.

is 74.67%. For the test set, the precision is 84.71% and the recall is 78.39%. We note that, similar to the boundary detection sub-task, the results on the test data are significantly higher than on the development data which suggests that the test data is relatively easier.

6 Conclusion

In this paper, we presented a description of Task 18 on Arabic Semantic labeling. Our goal was to rally interest in Arabic Semantic labeling. On the word sense disambiguation front, we have successfully created an all-words sense annotated set of Arabic nouns and verbs in running text. The set is annotated with both Arabic WordNet synset labels and their corresponding English WordNet 2.0 synset labels. Unfortunately, no systems participated in the WSD sub-tasks, however, we have prepared the data for future endeavors and hopefully this will motivate researchers in NLP to start experimenting with Arabic WSD.

On the task of Semantic Role Labeling, we have created a test, training and development set that has been successfully validated through being employed for building the first Arabic SRL system. Hopefully, this data will help propel research in Arabic SRL. It is also worth noting that we currently have effectively created a data set that is annotated for word senses, lexical information such as full morphological specifications, syntactic and semantic parses as well as English glosses and translations.

References

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The berkeley FrameNet project. In *COLING-ACL '98: Proceedings of the Conference, held at the University of Montréal*, pages 86–90.

Tim Buckwalter. 2002. Buckwalter Arabic Morphological Analyzer Version 1.0. Linguistic Data Consortium, University of Pennsylvania, 2002. LDC Catalog No.: LDC2002L49.

Xavier Carreras and Lluiś M`arquez. 2005. Introduction to the CoNLL-2005 shared task: Semantic role labeling. In *Proceedings of CoNLL-2005*, Ann Arbor, Michigan.

S. Elkateb, H. Rodriguez, M. Alkhalifa, P. Vossen, A. Pease, M. Bertran, W. Black, and C. Fellbaum. 2006. The arabic wordnet project. In *Proceedings of the Conference on Lexical Resources in the European Community*, Genoa, Italy, May.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press. http://www.cogsci.princeton.edu/˜wn [2000, September 7].

Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288.

Nancy Ide and Jean Veronis. 1998. Word sense disambiguation: State of the art. In *Computational Linguistics*, number 24, pages 1–40.

Mohamed Maamouri, Ann Bies, Tim Buckwalter, and Wig dan Mekki. 2004. The penn arabic treebank : Building a large-scale annota ted arabic corpus.

Martha Palmer, Dan Gildea, and Paul Kingsbury. 2005. The proposition bank: A corpus anotated with semantic roles. In *Computational Linguistics Journal*, number 31:1.

Sameer Pradhan, Kadri Hacioglu, Wayne Ward, James H. Martin, and Daniel Jurafsky. 2003. Semantic role parsing: Adding semantic structure to unstructured text. In *Proceedings of ICDM-2003*, Melbourne, USA.

Dan Tufi s. 2004. The balkanet project. In *Special Issue of The Romanian Journal of Information Science and Technology*, number 7, pages 1–248.

Nianwen Xue and Martha Palmer. 2004. Calibrating features for semantic role labeling. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 88–94, Barcelona, Spain, July. Association for Computational Linguistics.