RENAR: A Rule-Based Arabic Named Entity Recognition System

WAJDI ZAGHOUANI, University of Pennsylvania

2

Named entity recognition has served many natural language processing tasks such as information retrieval, machine translation, and question answering systems. Many researchers have addressed the name identification issue in a variety of languages and recently some research efforts have started to focus on named entity recognition for the Arabic language. We present a working Arabic information extraction (IE) system that is used to analyze large volumes of news texts every day to extract the named entity (NE) types person, organization, location, date, and number, as well as quotations (direct reported speech) by and about people. The named entity recognition (NER) system was not developed for Arabic, but instead a multilingual NER system was adapted to also cover Arabic. The Semitic language Arabic substantially differs from the Indo-European and Finno-Ugric languages currently covered. This article thus describes what Arabic language-specific resources had to be developed and what changes needed to be made to the rule set in order to be applicable to the Arabic language. The achieved evaluation results are generally satisfactory, but could be improved for certain entity types.

Categories and Subject Descriptors: I.2.7 [**Artificial Intelligence**]: Natural Language Processing

General Terms: Algorithms, Languages, Design

Additional Key Words and Phrases: Named entity recognition, rule-based systems, Arabic natural language processing, information extraction

ACM Reference Format:
Zaghouani, W. 2012. RENAR: A rule-based Arabic named entity recognition system. ACM Trans. Asian Lang. Inform. Process. 11, 1, Article 2 (March 2012), 13 pages.
DOI = 10.1145/2090176.2090178 http://doi.acm.org/10.1145/2090176.2090178

1. INTRODUCTION

The named entity recognition (NER) task was introduced for the first time in 1995 by the Message Understanding Conference (MUC-6) [Grishman and Sundheim 1996]. Three subtasks were defined for MUC-6: ENAMEX for the proper names (person, locations, and organizations), NUMEX for the numeric expressions, and TIMEX for temporal expressions.

NER is used in various areas in the field of natural language processing (NLP) and Information Retrieval (IR) [Sekine 2004]. For instance, news analysis systems such as NewsVine, SiloBreaker and the Europe Media Monitor (EMM) application NewsExplorer [Steinberger et al. 2009] go beyond this IR information setting, by additionally extracting information from the news text and by further linking information found in the news.

Most such systems are monolingual (typically English). NewsExplorer, in contrast, currently covers 19 languages, including Arabic. Such high multilinguality is most

Author's address: W. Zaghouani, University of Pennsylvania, Linguistic Data Consortium, 3600 Market Street, Suite 810, Philadelphia, PA, 19104, USA; email: zaghouani.wajdi@courrier.uqam.ca.

DOI 10.1145/2090176.2090178 http://doi.acm.org/10.1145/2090176.2090178

likely to be achieved only if the effort for each language is limited. The EMM family of applications processes an average of approximately 100,000 articles per day. The Arabic language is relatively well represented within EMM.

Steinberger et al. [2008] propose to use language-independent rules and as few language-specific resources as possible. These should furthermore be simple and easy to produce, and they should be organized in a compositional manner so that any new language can simply be plugged in to the overall system once the language-specific resources are available. In this article, we present the effort of adding the Semitic language Arabic to EMM-NewsExplorer.

Arabic is significantly different from the other, mostly Indo-European EMM languages [Vergyri et al. 2004], and is thus a good test case for the proposed method. In the next section, we discuss the state of the art of Arabic NER. In Section 3, we present some challenges specific to the Arabic language. In Section 4, we discuss the method we used to build our resources. Section 5 illustrates the architecture of the system and its various components. In Section 6, we present the evaluation results achieved with the described method. Section 7 summarizes the results.

2. RELATED WORK

NER by now is a known and common task and has been integrated into several products. For a few widely used languages, a large variety of NER tools exist [Nadeau and Sekine 2009], and three main approaches have been used to implement these tools: linguistic rule based, statistical based, and hybrid.

Rule-based methods are usually based on an existing lexicon of proper names and a local grammar that describes patterns to match NEs using internal evidence (gazetteers) and external evidence provided by the context in which the NEs appear.

Statistical and machine learning approaches generally require a large amount of manually annotated training data. Hybrid methods are a combination of the statistical and the rule-based approaches. A remaining challenge in the field is how to develop such systems quickly with minimal costs. For Arabic, available software is mostly commercial, including ANEE[1] (Coltec), IdentiFinder[2] (BBN), NetOwlExtractor[3] (NetOwl), Siraj[4] (Sakhr), Clear Tags[5] (ClearForest), and InXight-Smart-Discovery-Entity-Extractor[6] (InXight). Little information is available on the inner workings and on formal evaluation results of these systems.

Maloney and Niv [1998] presented TAGARAB, an Arabic name recognizer that combines the pattern-matching module with a morphological analyzer to improve performance. TAGARAB was evaluated on a corpus of 3,214 tokens and the overall performance obtained for the various categories (time, person, location, and number) was a precision of 89.5%, a recall of 80.8% and an F-measure of 85%.

Abuleil [2004] developed a rule-based system that makes use of hand-written rules and trigger words. The system starts by marking the phrases that could include names, then it builds up a graph that represents the words in these phrases and the relationships between them and finally, rules are applied to classify and generate the names before saving them in a database. Abuleil's system has been evaluated on a corpus of 500 news articles from the Alraya newspaper and has obtained a precision of 90.4% on person, 93% on location and 92.3% on organization.

[1]See http://www.coltec.net/Portals/0/COLTEC_PDFs/ANEE_NEW.pdf.
[2]See http://www.bbn.com/technology/speech/identifinder.
[3]See http://www.sra.com/netowl/entity-extraction/.
[4]See online demo version available at http://siraj.sakhr.com/.
[5]See http://www.clearforest.com/solutions.html.
[6]See http://www.inxightfedsys.com/products/sdks/tf/default.asp.

Doaa et al. [2005] used a parallel corpus in Arabic and in Spanish as well as a Spanish Named Entity tagger in order extract NE in the Arabic corpus. A simple mapping technique has been used to transliterate the words in the Arabic text and return those matching with named entities in the Spanish text as named entities in Arabic. The size of the parallel corpus was limited to 1,200 sentence pairs and 300 sentences were randomly selected from the Spanish corpus with their equivalent in Arabic. For each pair, the output of the NE tagger was compared to the manually annotated gold standard set. Moreover, a stop word filter was applied to exclude the stop words from the potential transliterated candidates. The filter improved the precision from 84% to 90% and the recall was very high at 97.5%.

Zitouni et al. [2005] presented an Arabic mention detection system that looks for nominals, pronominals, references to entities, and named entities. A Maximum Entropy Markov model was implemented using lexical and syntactic features. The system was evaluated against the ACE 2004 data set. The overall score obtained was an F-measure of 69%.

Traboulsi [2006] presented NExtract, a rule-based named entity recognition model that uses local grammar and dictionaries. He showed that his approach could lead to good results when tested in a small-scale experiment with a Reuters corpus. Later on, Traboulsi [2009] proposed a local grammar-based approach combined with a finite state automata to detect Arabic named entities. No further details are available at this time about this method.

Benajiba et al. [2008] proposed a system that combines two machine learning approaches: Support Vector Machines (SVMs) and Conditional Random Fields (CRFs). Moreover, the system uses lexical, syntactic, and morphological features and a multi-classifier approach where each classifier is designed to tag a NE class separately. The system obtained an F-measure of 83.5% when tested with the ACE 2003.

Benajiba [2009a] compared in his thesis the results obtained by ANERsys from various machine learning (ML) approaches such as the Maximum Entropy, Support Vector Machines, and Conditional Random Fields. Benajiba concluded that no single ML approach is considered better than the other for the Arabic NER task and that the best results were obtained when he used a multi-classifier approach where each classifier used the best ML technique for the specific named entity class.

In another experiment Benajiba et al. [2009b] explored a combination of lexical, contextual, and morphological features. The impact of the different features has been measured in isolation and combined. The best F-measure of 82.71% was obtained when he combined language independent features with language specific features.

Shaalan and Raza [2009] presented a NER system for Arabic (NERA) using a rule-based approach, a dictionary of names, a local grammar in the form of regular expressions, and a filtering mechanism. The filter serves mainly to revise the system output by using a blacklist to reject the incorrect named entities. NERA obtained an F-measure of 87.7% for person, 85.9% for locations, 83.15% for organizations, and 91.6% for dates.

LingPipe[7] is a tool kit of various information extraction tools. The named entity recognition module is based on a supervised training model combined with a dictionary and a regular expression matching method. The NER module has been trained on ANERCorp, a corpus of 150,000 words created by Benajiba et al. [2007]. The default detection in LingPipe distinguishes between three types of entities (person, place, and organization). LingPipe was evaluated and obtained an F-measure of 67%.

[7]LingPipe is freely available at http://alias-i.com/lingpipe/.

Similar to the other rule-based systems, our own system, RENAR, is also based on hand-written local patterns. However, a big difference is that we use a set of language-independent rules in combination with language-specific parameter files, containing the relevant vocabulary and an optional set of extra language-specific rules. The general mechanism will be explained in Section 4.3. In the same section, we will discuss the changes made to support the peculiarities of the Arabic language.

The choice of a rule-based approach over a machine learning approach was motivated mainly by the fact that the current architecture of the EMM interface is optimized for rule-based systems and by the encouraging results obtained by various similar Arabic rule-based systems such as Maloney and Niv [1998], Abuleil [2004], Shaalan and Raza [2009].

3. CHALLENGES WITH ARABIC NER

In RENAR, we addressed many challenges posed by the peculiarities of the Arabic language [Zaghouani 2009], which is significantly different from the other European languages that are supported by the EMM-NewsExplorer. We review below some issues that need to be taken into consideration when building a NER system for Arabic.

Non-vocalization and ambiguity. The lack of diacritics in the Arabic written text (e.g., short vowels), which is common in the news articles, causes a high degree of ambiguity since many words can be diacritized multiple ways, producing different meanings [Debili and Achour 1998]. In order to alleviate the impact of this issue, contextual information will be used in our system as shown in Section 4.

Absence of capital letters. Unlike Latin script languages, Arabic does not distinguish upper and lowercase letters (uppercase helps to identify the beginning and end of potential NEs in most Latin script languages).

Complex morphology. The Arabic language has a very systematic but complex morphological structure based on root-pattern schemes and is considered a highly inflectional language [Shaalan 2005]. Usually a given lemma in Arabic could have more than one word form which includes a root, prefixes, suffixes, and clitics. This issue should be dealt with in order to detect correctly the NEs in the text. For instance, the attached clitics (e.g., ب Ba, ل Laam, و waw) should go through a morphological preprocessing step in order to obtain the original word form.

Lack of standardization of the Arabic spelling. Arabic text, like many other languages, has many spelling variants when it comes to proper names and especially foreign names, which may lack a standardized spelling or be prone to common typos such as the spelling of Monica Seles مونيكا سيليش (correct) or مونكا سليش (wrong).

4. BUILDING REQUIRED RESOURCES FOR RENAR

In order to build RENAR, we have used freely available corpora since we were not able to license any commercially available corpora. Moreover, we built a stop word list, a modifiers list, and gazetteer for person, location, and organization names.

4.1. Corpus

We used the freely available ArabiCorpus[8] (AC) as the main corpus to build our resources. AC is a collection Arabic text (68,943,447 words) from various sources such as

[8]ArabiCorpus is freely available at http://arabicorpus.byu.edu/.

newspapers, the Quran and Arabic literature. AC includes a user interface that offers advanced searching capabilities, which we used in our preparation.

4.2. Trigger Words

Proper names are usually surrounded by cue words or trigger words such as titles like Dr. and Mr. or verbs such as said, declared. The trigger word list was created manually by using semi-automatic procedures, by looking at the most frequent left and right-hand-side contexts of known Arabic NEs, or at the context of NEs that are found using the rules with initial lists of seed words. A list of 3,400 trigger words was created including 1,100 modifiers.

4.3. Stop Word List

Stop words are frequent words that cannot be part of named entities. We built a list of 1,009 most frequent stop words that occurred in the AC. This list is mainly composed of prepositions, adverbs, and verbs.

4.4. Gazetteers

The name part lists contain a total of 19,600 names. The person names list is composed of a first names list (17,000 entries), including an extensive list of common name parts (not only Arabic first names, but also common international names such as جون John, جان Jean, خوان Juan written in Arabic), a list of name infixes (بن bin, عبد abd, أبو abu, آل Al, etc.), and finally a last names list of 2,600 entries.

The locations are recognized through a simple gazetteer lookup procedure, without any grammatical patterns. The gazetteer consists of currently 2,200 names of countries, cities, towns, villages, states, and political regions found mainly in the multilingual KNAB gazetteer produced by the Institute of the Estonian Language[9] and enriched recently by the Arabic Wikipedia[10]. The organizations lexicon is limited to a list of 4,000 company and organization names that we extracted from the AC.

5. DESCRIPTION OF THE ARABIC NER SYSTEM

We first present the architecture of the proposed multilanguage system, then we describe the morphological pre-processing step and the working of the NER rules.

5.1. The Architecture of the System

The architecture of the system is shown in Figure 1. The system relies on three main processing steps: pre-processing (segmentation rules), lookup of full known names, and recognition of unknown names using local grammars and a set of dictionaries. Names found repeatedly (at least twice) in the course of long-term multilingual news analysis are manually checked then stored in a database.

The multilanguage database currently contains over one million known names and variants for these names. Every day, lists of known entities are exported to a finite-state automaton, which identifies the known names in the news texts before the extraction rules apply. A daily search of known names and their variants on Wikipedia and, if found, are added to the multilingual name variants found there to the list of known entities [Pouliquen et al. 2005].

[9]See http://www.eki.ee/knab/p_mm_en.htm.
[10]See http://ar.wikipedia.org/wiki/.

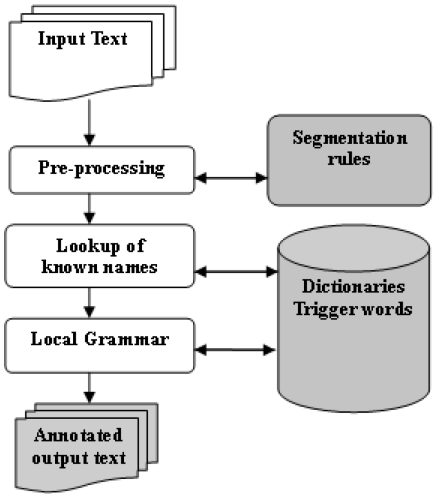

Fig. 1. Architecture of the system.

5.2. Morphological Pre-Processing

Arabic morphology is relatively complex in that it uses prefixes, infixes, and suffixes. This various morphological variation can be dealt with by using hand-crafted rules to strip off possible prefixes and suffixes from the word stem before applying the NER grammars [Shaalan 2008]. For each type of NE, several rules were built, and each rule was applied to all input words. By doing this, particles attached to words are stripped, allowing a better match between the words found in the text and those in the dictionaries. For example, the attached conjunction *wa*, the preposition *li* and the definite article *al* are stripped off all words and names, like in the example below. Transformation one removes the attached conjunction و /wa/ (and), and transformation two removes the attached preposition ل /li/ (for) and the definite article ال /al/ (the):

وللرئيس/walilra'iis /	(and for the president).
للرئيس/lilra'iis/	(for the president).
رئيس/ra'iis/	(president).

5.3. Local Grammars for Person and Organization Name Recognition

We first explain how the multilingual EMM grammars for NER work in general. We then describe the Arabic-specific differences. The rule set is mostly language-independent (meaning that the same rules are applied to all languages), but they refer to language-specific words, multi-word expressions and regular expressions that are stored in language-specific parameter files. Due to this modularity, it is relatively easy to add a new language to the tool set. Whenever a rule needs to be added for a specific language, as is the case for Arabic, this rule will be stored in the language-specific parameter file. The language-specific resources are hand-compiled, but bootstrapping methods are used to build the word lists quickly [Pouliquen et al. 2005].

We refer to the language-specific words and expressions that are parts of the local patterns and that help our system to identify named entities as trigger words. We use this unspecific term because not only titles are included, but also verbal phrases, regular expressions, and more. The rules refer to words and multi-word expressions that

are stored in various dictionary files. The person name recognition tool distinguishes various types of trigger words, lists of modifiers, and stop words.

The trigger word lists include titles (السيدة Mrs., استاذ Prof., دكتور Dr., etc.), professions or positions (مدير director, رئيس president, محام lawyer, الكاهن priest, etc.), country adjectives (التونسي Tunisian, الكندي Canadian, etc.), religious and ethnic groups (الكاثوليكي Catholic, السنية Sunni, بربر Berber) and many other expressions indicating, for example, that in Latin-based languages some uppercase words (X) may be names (e.g., 'X declared', 'X died', [0-9]+-year-old X, etc.).

The word lists for each language are kept in a language-specific parameter file. The language-specific parameter file furthermore contains a word list loosely named modifiers, that is, words that can appear in certain places between the name mention and the trigger words. The modifier list can contain all sorts of modifiers, but also auxiliary verbs and more (e.g., has, yesterday). The rule set thus refers to different subsets of these language-specific word lists. Here are some examples, using the notation: (\w+) represents an unknown word, \b an obligatory word boundary (white space, possibly with punctuation); + indicates one or more elements, * means zero or more elements:

(1) PERSON_TRIGGER+\bUppercaseWord\bUppercaseWord
(2) UppercaseWord\bUppercaseWord(\bMODIFIER)*\bPERSON_TRIGGER+

Rule (1) will recognize any combination of at least two uppercase words as person names if they are found next to trigger words or expressions (e.g., Mr. Ahmed Issa). Note that we require at least two name parts because all names are unambiguously grounded to real-world entities so that the extracted information can be displayed on NewsExplorer[11].

Rule (2) captures apposition constructions such as "Hamid Karzai, the newly elected Afghan president." The words "the", "newly", and "elected" can be found in the English modifier lists, while both Afghan and president are part of the trigger word lists. Our recognition expressions (e.g., for modifier) can be more complex than what is shown here. Details of the approach that are worth highlighting here are (a) determiners, adverbs and other elements may be loosely combined under the heading modifier, as the application would not benefit from a distinction; (b) the order of elements within the groups modifier and person trigger is also not specified.

These are both examples of under-specified rules that make it easier to write the rules and apply them to different languages. While generation grammars would need this information, our recognition grammars do not. These generic rules refer to case information, which is not available in Arabic or Farsi. When a person trigger expression is found in these Arabic script languages, we cannot know whether the preceding or following words are names or not (because we do not have access to generic dictionaries, part-of-speech or syntactic information). Furthermore, we would not know where the name boundaries are. For this reason, we needed to write separate, safer rules, which were then placed in the language-specific parameter file so that they only apply to Arabic. The following are examples of such Arabic-specific rules:

(3) KNOWN_NAME+\b(\w+)\bNAME_INFIX*\bKNOWN_NAME
(4) (\w+)\bNAME_INFIX+\b(\w+)
(5) PERSON_TRIGGER+\b(\w+)\bKNOWN_NAME

[11]See, for example, the page for Iraqi politician Nouri al-Maliki:
http://emm.newsexplorer.eu/NewsExplorer/entities/en/77049.html.

(6) NAME_STOP_WORDS\b(\w+)(\bMODIFIER)*\bPERSON_TRIGGER+
(7) ORGANIZATION_TRIGGER_BEG+\bKNOWN_NAME\bNAME_INFIX*\b KNOWN_NAME*\bTRIGGER_ORG_END

Rule (3) recognizes combinations of known name parts (first names or last names in some Arabic countries, the distinction is mostly irrelevant). The names can optionally be separated by one or more name parts (e.g., بن bin, عبد abd, أبو abu, آل Al), referred to as name infixes, to stay in line with other languages where we can find name infixes such as van der, de la, della, von, etc. This rule would successfully recognize the name محمد علي بن حليمة (Mohammed ali ben Halima), assuming that both Mohammed and Halima are in the list of known names. The known name list used contains thousands of name parts from different parts of the Arab world. Similar lists exist for most EMM languages.

Rule (4) recognizes combinations of unknown (or, optionally, known) words if they are linked by name infixes. Similar rules will also allow for longer combinations of three, four or more names, as long as each element is linked to the others via name infixes.

Rule (5) will recognize an unknown word and a known name part as a name if they follow a trigger expression (e.g., عيسى أحمد السيد –Mr. Issa Ahmed, assuming that Ahmed is a known name and Issa is unknown). Rule (6) is the Arabic equivalent to rule (2), that is, it will capture apposition constructions. However, in order to recognize the left-hand-side boundary of the name, we make use of stop words. Stop words are typically high-frequency words from a long list of words that cannot be name parts.

In the example below, the rule will thus recognize "Hamid Karzai" or any other name as a person, even if the words involved are not known name parts. The words "and said", found in the name stop word list, ensure that the left-hand-side border of the name is correctly identified as Hamid. The uppercase condition applicable to most EMM languages is thus replaced in Arabic by the introduction of the name stop word list:

وقال حامد كرزاي الرئيس الأفغاني المنتخب الجديد
"And said Hamid Karzai, the newly elected Afghani president"

An alternative would have been to use a part-of-speech tagger or full dictionaries with part-of-speech information, but developing these would have required effort far beyond the scope of our project. More importantly, the parallelism between languages would have been lost. While new rules needed to be introduced especially to deal with the Arabic language, the format of the Arabic language-specific resources remains consistent with those of other languages.

Rule (7) is an illustration of an actual organization rule that recognizes a combinations of known organization beginning and ending parts (e.g., شركة the company, و إخوان and brothers). This rule will recognize complex company names that include person names by using a known names list:

شركة محمد أبو الحمداني و إخوان
"The company Mohamed Abu Alhamadani and brothers"

5.4. Difficulties and Challenges

At times, it was difficult to create rules because of the complexity of organization and person names in Arabic. A major challenge was to predict the boundaries of the named entities, especially with long and composed Arabic names. We found cases where the

Table I. Distribution and Ratio of Proper Names in the Evaluation Corpus

Category	Ratio
Person	39.00%
Location	30.40%
Organizations	20.60%
Miscellaneous	10.00%

full name is composed of eight words. Moreover, it was very difficult to cover with our local grammar all regional variants of the names of organizations because labels and standards differ from one country to another and from one culture to another. For instance, companies that originate from the Maghreb region will frequently use French words in their names (such as أوتو محمد داوود بياس –Mohamed Daoud Pièces Auto).

On the other hand, companies in the Gulf region will use English words (such as مدني تايلورز –Madani Tailors). We have therefore limited our coverage to a lookup of major internationally known organizations, as well as those from the Arab world.

6. EVALUATION

6.1. Corpus

The results of our system were evaluated against a standard evaluation corpus: Benajiba's ANerCorp[12], which is freely available online. Evaluation results are available for other systems that were tested against this corpus [Benajiba and Rosso 2007; Benajiba et al. 2007; Lingpipe[13]], so we are able to compare these directly to the performance of RENAR.

Moreover, this allows a fair comparison of the performances of our system with three existing systems which have been evaluated already against the same corpus.

The corpus consists of 316 articles and 150,286 tokens which were collected from various online news sources[14]. Proper names represent 11% of the whole corpus and their distribution ratio per class is shown the Table I.

6.2. Results

We have used the CoNLL 2002 evaluation tool[15] to process the results obtained by RENAR. According to the official CoNLL evaluation guidelines, a named entity is considered correctly detected only if the classification is correct and all the constituent

[12]See http://www.dsic.upv.es/~ybenajiba.

[13]See http://alias-i.com/lingpipe/demos/tutorial/ne/read-me.html.

[14]This is the list of sources used and the articles distribution ratio [Benajiba et al. 2008]:

— http://www.aljazeera.net 34.8%
— Other newspapers and magazines 17.8%
— http://www.raya.com 15.5%
— http://ar.wikipedia.org 6.6%
— http://www.alalam.ma 5.4%
— http://www.ahram.eg.org 5.4%
— http://www.alittihad.ae 3.5%
— http://www.bbc.co.uk/arabic/ 3.5%
— http://arabic.cnn.com 2.8%
— http://www.addustour.com 2.8%
— http://kassioun.org 1.9%

[15]Available at http://bredt.uib.no/download/conlleval.txt.

Table II. Results Obtained by RENAR for the Various Named Entity Types

Category	Precision	Recall	*F*-measure
Person	71.18	54.20	61.54
Organization	58.79	47.00	52.23
Location	90.21	85.20	87.63
Overall	**73.39**	**62.13**	**67.13**

Table III. Precision, Recall and *F*-Measure Obtained by Existing Arabic NER Systems

System	Person			Location			Organization		
	P	R	F	P	R	F	P	R	F
RENAR	**71.18**	54.20	61.54	90.21	**85.20**	**87.63**	58.79	47.00	52.23
Lingpipe[16]	63.40	**65.70**	**64.50**	78.20	78.80	78.50	**60.90**	**52.70**	**56.50**
Benajiba ANERsys 1.0[17]	54.21	41.01	46.69	82.17	78.42	80.25	45.16	31.04	36.79
Benajiba ANERsys 2.0[18]	56.27	48.56	52.13	**91.69**	82.23	86.71	47.95	45.02	46.43

words of the named entity are recognized. Table II summarizes the results obtained by our system. We used the standard evaluation measures (i.e., precision, recall, and *F*-measures), allowing direct comparison with the results from other systems. We have included the precision, recall, and *F*-measure values for each named entity category except for the miscellaneous category, which was not applicable to our system.

Finally, Table III summarizes the recognition accuracy, in terms of precision, recall, and *F*-measure, achieved by RENAR and the other NER systems for Arabic. All systems in Table III: RENAR, Lingpipe, Benajiba's ANERsys 1.0 and ANERsys 2.0, have been evaluated against Benajiba's AnerCorp.

6.3. Discussion of Results and Error Analysis

The results showed clearly that the performance of the various systems varied significantly by entity type, and the accuracy scores of the various systems within each type are relatively close to each other, especially the scores for Renar, Lingpipe and ANERsys 2.0.

With our system, the person names were correctly identified in 71.18% of the cases, which is still better than the three other systems (63.4% for Lingpipe and 54.21% and 56.27% for the Benajiba systems). For the location category, Renar obtained the best *F*-measure (87.63%). This can be explained by the good coverage of Renar's location gazetteer and the safe lookup method. For person and organization, our system ranked second with *F*-measures of 61.54% and 52.23%, respectively. These scores may seem low, but when compared to the other systems, it becomes obvious that all tested systems have some difficulties with these two categories.

Regarding the person category, Renar achieved the best overall precision (71.18), more than 6 points better than Lingpipe. On the other hand, it appears that the

[16]Results obtained from http://alias-i.com/lingpipe/demos/tutorial/ne/read-me.html.
[17]Results obtained by Benajiba et al. [2007].
[18]Results obtained by Benajiba and Rosso [2007].

organization category is the most challenging one, with a low F-measure ranging from 36.79% (ANERsys 2.0), 52.23% (Renar) to 56.50% (Lingpipe). The analysis of detection errors for the organization category showed they are sometimes due to factors such as the inconsistency of transcribing foreign organization names to Arabic, or the extended length of the name and the relatively small size of our organizations-name gazetteer.

Moreover, our tool was not able to identify correctly many entries as shown by the low recall of 47%. A deeper analysis has revealed many cases of incorrect categorization due to the ambiguity of some Arabic words. Another type of error was that names were only partially recognized, especially with organization and person names. In addition, the absence of rigorous standards for keyboarding Arabic text has led to inconsistencies in the spelling of some words and therefore has influenced our results. For example, the use of the letter Hamza ء, an Arabic glottal stop, varies in word-initial position (e.g., a name like Ahmad could be written as احمد or as أحمد).

Some errors were also caused by the spelling variants of translated or transliterated entities that were not present in our gazetteer (e.g., a location name like Los Angeles could be written in four different ways as لوس انجليس , لوس انجيليس , لس انجلوس or لوس انجليس). In addition to that, the ambiguity and length of some named entities prevented our system from detecting all parts of multi-word names (e.g., detecting Bin Ahmed Talel طلال بن أحمد instead of the full name Katib Bin Ahmed Talel طلال كاتِب بن أحمد), the ambiguity here involves the word Katib كاتِب which can be the past participle of the verb "to write" or a person's name. More fine-grained rules will be needed to address such cases.

With this small-scale evaluation, we showed promising results. Our long-term evaluation plan involves a larger evaluation corpus to check the accuracy and comprehensiveness of our system. We are also planning to improve our detection rules and to increase the coverage of the detection rules to cover more named entities combinations, especially for the person and organization categories.

7. CONCLUSION

We have presented our work on adapting a multilingual NER system to the Arabic language. Many of the existing language-independent rules had to be adapted to Arabic, mostly because the lack of orthographic case makes it difficult to know where a name starts and ends. More than for other languages, we needed long lists of potential name parts, and we had to make much more use of stop words. Not having access to full dictionaries and to part-of-speech information made the NER task rather difficult for Arabic, more so than for the EMM languages using the Latin script (and thus orthographic case). In NewsExplorer, we are currently mostly making use of safe NER rules (such as those using lists of known name parts and name infixes). The reason is that we did not have an Arabic speaker in the group for a long time. Not having any control instance, we needed to optimize precision at the cost of lowering recall.

We are now planning to work on improving the recall. Moreover, we showed that there are some important factors that have greatly influenced the above achieved results such as the relative lack of standardized orthography in Arabic, especially in the transcription of foreign names, the high ambiguity found in Arabic text and the relatively small size of our gazetteers. Having said this, we feel that the results are relatively good, considering the simplicity of the approach and having ensured that the approach remains the same across all 20 languages in which we currently recognize named entities.

ACKNOWLEDGMENTS

The author would like to thank Ralf Steinberger and Bruno Pouliquen of the Language Technology Group at the Joint Research Center, for all the help provided during this project. A special thanks to David Graff of the Linguistic Data Consortium for his valuable suggestions and comments.

REFERENCES

ABULEIL, S. 2004. Extracting names from Arabic text for question-answering systems. In *Proceedings of Coupling Approaches, Coupling Media and Coupling Languages for Information Retrieval (RIAO'04)*. 638–647.

BENAJIBA,D Y. 2009a. Named entity recognition. Doctoral dissertation, Universidad Politecnica de Valencia.

BENAJIBA Y. AND ROSSO, P. 2007. ANERsys 2.0: Conquering the NER task for the Arabic language by combining the maximum entropy with POS-tag information. In *Proceedings of the Workshop on Language-Independent Engineering (LIE'07)*.

BENAJIBA Y., ROSSO P., AND BENEDI, J.-M. 2007. Arabic ANERsys: An Arabic named entity recognition system based on maximum entropy. In *Proceedings of the Conference on Computational Linguistics and Intelligent Text Processing (CLITP'07)*. 143–153.

BENAJIBA, Y., DIAB, M., AND ROSSO, P. 2008. Arabic named entity recognition using optimized feature sets. In *Proceedings of the Joint Meeting of the Conference on Empirical Methods in Natural Language Processing (EMNLP'08)*. 284–293.

BENAJIBA, Y., DIAB, M., AND ROSSO, P. 2009b. Using language independent and language specific features to enhance Arabic NER. *Int. Arabic J. Inf. Technol.*, 463–471.

DEBILI, F. AND ACHOUR, H. 1998. Voyellation automatique de l'arabe. In *Proceedings of the Workshop on Computational Approaches to Semitic Languages (CASL'98)*. 42–49.

DOAA, S., MORENO-SANDOVAL, A., AND GUIRAO, J.-M. 2005. A proposal for an Arabic named entity tagger leveraging a parallel corpus (Spanish-Arabic). In *Proceedings of Recent Advances in Natural Language Processing (RANLP'05)*. 459–465.

GRISHMAN, R. AND SUNDHEIM, B. 1996. Message Understanding Conference - 6: A brief history. In *Proceedings of the International Conference on Computer Linguistics (COLING'96)*. 466–471.

MALONEY, J. AND NIV, M. 1998. TAGARAB: A fast, accurate Arabic name recognizer using high precision morphological analysis. In *Proceedings of the Workshop on Computational Approaches to Semitic Languages (CASL'98)*. 8–15.

NADEAU, D. AND SEKINE, S. 2009. A survey of named entity recognition and classification. In *Named Entities – Recognition, Classification and Use*. S. Sekine and E. Ranchhod Eds., Benjamins Current Topics, Vol. 19, John Benjamins Publishing Company, Amsterdam.

POULIQUEN, B., STEINBERGER, R., IGNAT, C., TEMNIKOVA, I., WIDIGER, A., ZAGHOUANI, W., AND ŽIŽKA, J. 2005. Multilingual person name recognition and transliteration. *Corela, Numéros spéciaux, Le traitement lexicographique des noms propres*.

SEKINE, S. 2004. amed entity: History and future. http://cs.nyu.edu/~sekine/papers/NEsurvey200402.pdf.

SHAALAN, K. 2005. Arabic GramCheck: A grammar checker for Arabic. *Softw. Prac. Exp. 35*, 7, 643–665.

SHAALAN, K. AND RAZA, H. 2008. Arabic named entity recognition from diverse text types. In *Proceedings of the 6th International Conference (GoTAL'09)*. 440–451.

SHAALAN, K. AND RAZA, H. 2009. NERA: Named entity recognition for Arabic. *J. Amer. Soc. for Inf. Sci. Technol. 60*, 8, 1652–1663.

STEINBERGER, R., POULIQUEN, B., AND IGNAT, C. 2008. Using language-independent rules to achieve high multilinguality in Text Mining. In *Mining Massive Data Sets for Security*. F.-S. Françoise, D. Perrotta, J. Piskorski, and R. Steinberger Eds., IOS Press, 217–240.

STEINBERGER, R., POULIQUEN, B. AND VAN DER GOOT, E. 2009. An introduction to the Europe media monitor family of applications. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR-CLIR'09)*. F. Gey, N. Kando, and J. Karlgren Eds. 1–8.

TRABOULSI, H. N. 2006. *Named Entity Recognition:* A local grammar-based approach. Doctoral dissertation, Department of Computing, Surrey University, Guildford, U.K.

TRABOULSI, H. N. 2009. Arabic named entity extraction: A local grammar-based approach. In *Proceedings of the International Multiconference on Computer Science and Information Technology (IMCSIT'09)*. 139–143.

VERGYRI, D., KIRCHHOFF, K., DUH, K., AND STOLCKE, A. 2004. Morphology-based language modeling for Arabic speech recognition. In *Proceedings of International Conference on Spoken Language Processing (ICSLP'04)*. 2245–2248.

ZAGHOUANI, W. 2009. Le repérage automatique des entités nommées dans la langue arabe: Vers la création d'un système à base de règles. Master's thesis, University of Montreal.

ZITOUNI, I., SORENSEN, J., LUO, X., AND FLORIAN, R. 2005. The impact of morphological stemming on Arabic mention detection and coreference resolution. In *Proceedings of the Workshop of Computational Approaches to Semitic Languages (ACL'05)*. 79–86.

Received June 2010; revised February 2011, April 2011; accepted May 2011