An automatically built Named Entity lexicon for Arabic

Mohammed Attia† **Antonio Toral**† **Lamia Tounsi**† **Monica Monachini**∗ **Josef van Genabith**†

†NCLT, School of Computing, Dublin City University, Ireland
{mattia,atoral,ltounsi,josef}@computing.dcu.ie
∗Istituto di Linguistica Computazionale, CNR, Pisa, Italy
{firstname.lastname}@ilc.cnr.it

Abstract

We have successfully adapted and extended the automatic Multilingual, Interoperable Named Entity Lexicon approach to Arabic, using Arabic WordNet (AWN) and Arabic Wikipedia (AWK). First, we extract AWN's instantiable nouns and identify the corresponding categories and hyponym subcategories in AWK. Then, we exploit Wikipedia inter-lingual links to locate correspondences between articles in ten different languages in order to identify Named Entities (NEs). We apply keyword search on AWK abstracts to provide for Arabic articles that do not have a correspondence in any of the other languages. In addition, we perform a post-processing step to fetch further NEs from AWK not reachable through AWN. Finally, we investigate diacritization using matching with geonames databases, MADA-TOKAN tools and different heuristics for restoring vowel marks of Arabic NEs. Using this methodology, we have extracted approximately 45,000 Arabic NEs and built, to the best of our knowledge, the largest, most mature and well-structured Arabic NE lexical resource to date. We have stored and organised this lexicon following the Lexical Markup Framework (LMF) ISO standard. We conduct a quantitative and qualitative evaluation of the lexicon against a manually annotated gold standard and achieve precision scores from 95.83% (with 66.13% recall) to 99.31% (with 61.45% recall) according to different values of a threshold.

1. Introduction

MINELex (Multilingual, Interoperable Named Entity Lexicon)[1] (Toral, 2009) contains Named Entities (NEs) for English, Italian and Spanish which are connected to general-domain lexicons (English WordNet, Spanish WordNet and the Italian SIMPLE-CLIPS) and two ontologies (SUMO and SIMPLE), in a format compliant with the ISO Lexical Markup Framework (LMF) (Francopoulo et al., 2009) standard in order to facilitate interoperability with other resources and tools. The NE lexicon was automatically derived by following a methodology that combines three ingredients: Language Resources (LRs), Web 2.0 and representation standards. MINELex contains 974,567 NEs for English, 137,583 for Spanish and 125,806 for Italian. Its knowledge has been applied to validate questions regarding NEs in a state-of-the-art Question Answering system (Ferrández et al., 2007), providing a 28% increment in accuracy. Its methodology is used in PANACEA[2] to create repositories which can store different pieces of information acquired and merged with new or legacy data.

In this paper we apply the MINELex methodology to Arabic, a Semitic language, to empirically prove the generic nature of the approach. The resources used are the Arabic WordNet (AWN) (Rodríguez et al., 2008; Elkateb et al., 2006) and the Arabic Wikipedia (AWK)[3].

AWN was constructed according to the methods and techniques used in the development of Princeton WordNet for English (PWN) (Fellbaum, 1998) and EuroWordNet (Vossen, 1998). It utilizes SUMO as an interlingua to link AWN to previously developed WordNets. This ensured that the overall topology of the wordnets is similar and a high degree of correspondence and compatibility is achieved. It also enables the translation on the lexical level from Arabic to English and other languages included in EuroWordNet. AWN consists of 11,269 synsets containing a total of 23,481 Arabic expressions. This number includes 1,142 NEs which were extracted automatically and checked by the lexicographers.

Wikipedia (WK) is a freely-available online multilingual encyclopedia built by a large number of contributors. Currently WK is published in 269 languages, with each language varying in the number of articles and the average size (number of words) of articles. Wikipedia contains additional information that proved to be helpful for linguistic processing such as a category taxonomy and cross-referencing. Each article in WK is assigned a category and may be also linked to equivalent articles in other languages through what is called ”interwiki links”. It also contains ”disambiguation pages” for resolving the ambiguity related to names that are spelt the same. AWK has about 104,000 articles (as of September 2009[4]) compared to 3.1 million articles in the English Wikipedia. Arabic is ranked 20^{th} among all languages included in the Wikipedia, and it also has a high growth rate. From September 2007 to September 2008 it grew by almost 100% and in September 2009 it grew further by over 30%.

The rest of the paper is organised as follows. Section 2 surveys the related work. Section 3 describes our methodology, which is evaluated in section 4. Finally, we present the conclusions and outline future work.

2. Background

NEs are a crucial factor in the improvement of Information Retrieval, Machine Translation, and Question-Answering systems (Gey, 2000); (Abuleil, 2004), as well as in parallel text processing and alignment of parallel corpora (Samy et al., 2005). One obvious reason for the importance of NEs is

[1] http://www.ilc.cnr.it/ne-repository

[2] http://panacea-lr.eu

[3] http://ar.wikipedia.org

[4] http://stats.wikimedia.org/EN/TablesWikipediaAR.htm

their pervasiveness. (Benajiba and Rosso, 2007) found that NEs constituted 11% of their corpus, and (Gey, 2000) suggested that 30% of content words are proper names. Statistics from the Penn Arabic Treebank (ATB) confirm the high frequency of NEs in texts. The ATB consists of 23,611 sentences, 553,363 words, and 428,761 content words (nouns, verbs, adjectives and adverbs). The number of NEs in the ATB reaches 54,398 which is 10% of the overall words and 13% of the content-bearing words.

NEs in Arabic are particularly challenging as Arabic is a morphologically-rich and case-insensitive language. NE Recognition in many other languages relies heavily on capital letters as an important feature of proper names. In Arabic there is no such distinction. In Arabic cliticized conjunctions and prepositions can be attached to the base form, further masking the NE, as shown by the following examples:

- Person: وكوفي أنان *wa-kuwfy ʾanān*, “and Kofi Annan”
- Location: بالبحر الأحمر *bi-’l-baḥr al-ʾaḥmar*, “in The Red Sea”
- Organization: وللأمم المتّحدة *wa-lil-ʾumam al-muttaḥidati*, “and to The United Nations”

Most of the literature on Arabic NEs concentrates on NE recognition (Maloney and Niv, 1998); (Abuleil, 2004); (Mesfar, 2007); (Farber et al., 2008); (Benajiba and Rosso, 2007); (Benajiba et al., 2008); (Shaalan and Raza, 2009); (Elsebai et al., 2009). NE Extraction is viewed largely as a subset of the task of NE Recognition. Most of the previous work uses data from bilingual dictionaries, lexicons or just simple lists of proper names. (Benajiba and Rosso, 2007) developed an annotated corpus (ANERcorp) collected from various news websites and the AWK. They also manually compiled gazetteers (ANERgazet) for location, person and organization names that contained about 4,500 NEs.

(Shaalan and Raza, 2009) compiled gazetteers of NEs collected from annotated corpora such as the ACE and ATB, from a database provided by government organizations and from Internet resources. The size if the database is presumably huge, yet due to the extremely heterogeneous nature of the sources and the lack of a detailed taxonomy, it cannot be considered as a standard language resource. Similarly (Benajiba et al., 2008) tried to make up for the lack of Arabic NE lexical resources by including hand-crafted gazetteers for person, location and organization names, and then semi-automatically enriched the location gazetteer using the AWK, taking the page labelled ”Countries of the world” as a starting point to crawl AWK and retrieve location names. The resulting list went through manual validation to ensure quality.

(Alkhalifa and Rodríguez, 2009) presented an approach to automatically attaching 3,854 Arabic NEs to English NEs using AWN, PWN, AWK and EWP as knowledge sources. Their approach is quite different as they start with an English NE collected from the PWN and EWP and try to obtain the Arabic counterpart from the AWK. Therefore they cannot capture Arabic NEs that have been originally compiled in Arabic and have no English equivalent. The AWK grows constantly and translation does not always keep pace. Our approach is more intuitive and linguistically motivated as we conduct the NE extraction cycle using Arabic resources. Using our methodology we have already extracted 974,567, 137,583 and 125,806 NEs for English, Spanish and Italian respectively (Toral, 2009). Judging by the size of the AWK we expect to be able to extract 35,000 Arabic NEs, which will be the largest mature and well-structured Arabic NE lexical resource to date. This lexical resource will be conducive to research on NE identification in unrestricted texts. Moreover, as the method is fully automated, the number of NEs will grow with the growth of AWK.

The applicability of LMF to Arabic has been the object of recent studies, such as the representation of HPSG-based syntactic lexicons (Loukil et al., 2007) and inflectional paradigms of verbs (Khemakhem et al., 2007). This paper will contribute to the application of LMF to Arabic by studying the formalisation of NEs.

3. Methodology

In this section we describe the different phases of our methodology and for each of them explain the challenges posed by Arabic and the decisions taken to tackle them. We begin by identifying the nouns of AWN that can instantiate NEs, these are mapped to the corresponding AWK categories. Then we identify which of the articles of these categories are NEs, these are extracted, connected to AWN and inserted in the NE repository. In a subsequent post-processing step further NEs are acquired by exploiting inter-lingual links. Finally the NEs acquired are diacritised. A schema depicting the overall process is presented in figure 1.

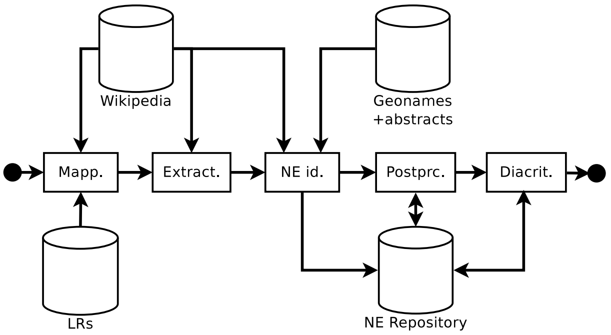

Figure 1: Method diagram

The following subsections cover in detail each of the phases.

3.1. Mapping

The first step consists of identifying the senses of AWN that can be extended with NEs. In other words, we are interested in instantiable nouns. Neither AWN nor PWN contain explicit information regarding the instantiability of their senses but both contain instance relations. Thus, we can obtain the set of instantiated nouns[5]. This set is built as a union of the instantiated synsets of both resources. In order to obtain the synsets from AWN that correspond to the instantiated synsets in PWN, we use the connections of

[5] If a synset is instantiated, it is instantiable.

AWN that link to the synsets of the former. Finally, we recursively add the hyponyms of the instantiable synsets to the set[6].

Following this, we obtain 384 instantiated synsets for AWN and 1,475 for PWN. The union of both sets contains 1,572 synsets, corresponding to 1,187 nouns (866 monosemous and 321 polysemous).

This set of nouns is then mapped to the categories of AWK. These mappings are obtained by comparing (string matching) the lemmas of the nouns to those of the categories. In order to do this, the categories of AWK are lemmatised with MADA+TOKAN (Habash and Rambow, 2005).

We obtain a mapping for 309 of the 866 monosemous nouns (35.68%) and for 173 of the 321 polysemous ones (53.89%), i.e. 40.6% of the whole set are mapped.

3.2. Extraction and Identification

Once the mapping has been established, we extract the articles from the mapped categories and their hyponym subcategories. In order to identify which of the subcategories are hyponyms we define a set of regular-expression-like patterns which can also hold Part-of-Speech tags. In the case of Arabic we have found out that just a very simple pattern (the name of the category followed by space and any string) is enough:

- `ˆcategory_`, e.g. recognises the subcategories سياسيّون حسب الجنسيّة "politicians by nationality" and سياسيّون بريطانيّون "British politicians" as hyponyms of the category سياسيّون "politicians".

Administrative categories (categories with the string "ويكيبيديا *wiykiybiydyaā*" (Wikipedia)) are discarded as they pertain to meta content and administrative purposes rather than real content.

Subsequently, we extract the articles that belong to the mapped categories (and subcategories) and identify which of them are NEs. For English, Italian and Spanish we relied on capitalisation norms that apply to these languages, i.e. that proper nouns (NEs) begin with capital letters while common nouns do not. As Arabic does not follow these rules we propose to take advantage of the inter-lingual links of WK (links that connect equivalent entries in different languages) in order to circumvent the issue; for each extracted article from AWK, we obtain the corresponding articles in a set of ten languages[7] that follow these norms. However, this covers only 62.5% of AWK's articles.

3.2.1. Abstract Keyword Searching

Relying on the capitalization of the interlingual-links is an effective method of validation, but when only 62.5% of the Arabic articles have links to other languages, this means that 37.5% of the entries have no chance at all of being detected or included in the NE repository. Therefore, in order to improve recall, we consider two other heuristics for validation: keyword searching in the AWK article abstract and looking up a database of geographical names (geonames). The geonames list is described in Section 3.4.1, and here we describe the process of keyword searching.

The AWK dump site[8] provides an abstract file. This is an XML file that includes all the titles of the entries followed by a short description of the entry. This file was found to be very useful. For the entries for which there are no interlingual links we use, as a back-off method, keyword searching in the abstract file. We developed a regular expression that looks into the abstract for hints (or keywords) that have a high likelihood that the entry is talking about a person or a location. We limited the method only to these two types of NEs because they have high frequency in the data and we believe that most of the other types of NEs will be captured by the main system. For locations we collected all names where the definition starts with دولة *dawlat* country, مدينة *madiynat* city, محافظة *muḥaāfaẓat* governorate, مقاطعة *muqaāṭaʿat* district, قرية *qariyat* village, جبل *ǧabal* mountain, بحر *baḥr* sea, , etc. We made a list of 16 location keywords and were able to collect 4,587 location names. For persons we looked for definitions where we can find phrases such as ولد في *wulida fiy* born in, مات في *maāta fiy* died in, شغل منصب *šaġala manṣib* worked as, عاش في *ʿaāša fiy* lived in, درس في *darasa fiy* studied in, حصل على *ḥaṣala ʿalaā* obtained a degree, etc. We compiled a list of 60 keywords. With persons the collected data was noisy as it included disambiguation pages, names of films, series, prizes, materials, locations, etc. So we had to use an exclusion list (of 160 keywords) to filter out the noise, and we collected a total of 16,038 NEs for persons.

3.3. Postprocessing

We perform a post-processing step of cross-fertilisation. Further Arabic NEs can be obtained by exploiting, on the one hand, the links between the English, Spanish and Italian NEs and their corresponding LRs and, on the other hand, the interconnections holding among these LRs. E.g. if we have an NE for Spanish that has an equivalent in AWK but has not been extracted, we can treat it as a candidate Arabic NE and following the connection to PWN present both in the Spanish WN and AWN, the new NE can be added to MINELex and duly connected to AWN. In turn, the NEs extracted for Arabic can provide further NEs for the other languages of MINELEx.

3.4. Diacritization

As most Arabic texts that appear in the media (whether in printed documents or digitalized format) are undiacritized, restoring diacritics is a necessary step for various NLP tasks that require disambiguation or involve speech processing. All the entries in the Arabic WordNet are fully diacritized, including NEs, and it is desirable, if not required, for compatible additions to be diacritized as well. A diacritic in Arabic is a small mark placed either above or under a letter to indicate what short vowel will follow that letter. Long vowels are usually indicated by one of three designated letters.

There are several publications on the automatic diacritization of Arabic, using statistical and Machine Learning algorithms, or a hybrid of rule-based and statistical methods.

[6] If a synset is instantiable, its hyponyms are instantiable.

[7] Catalan, Dutch, English, French, Italian, Norwegian, Portuguese, Romanian, Spanish, Swedish

[8] `http://download.wikimedia.org/arwiki/`

(Elshafei and Al-Ghamdi, 2006) used a hidden Markov Model (HMM). (Nelken and Shieber, 2005) presented an algorithm for restoring diacritics to undiacritized MSA texts using a cascade of probabilistic finite-state transducers trained on the Arabic treebank, integrating a word-based and a letter-based language model, and reported accuracy of over 90%. (Habash et al., 2009) use a morphological analyser and Support Vector Machines (SVM) classifiers. (Rashwan et al., 2009) developed a hybrid system that combines morphological information with probability estimation algorithms. (Zitouni and Sarikaya, 2009) use a Maximum Entropy approach for diacritics restoration.

(Alkhalifa and Rodríguez, 2009) used an approach similar to ours for extracting 3850 NEs from AWK and applied four different heuristics for diacritization. Firstly, transliterations of foreign names were left unvowelized when long vowels are used and no short vowels are needed. Secondly, in case transliterated names have some short vowels that need to be restored, translations into English, French, Italian and Spanish were checked to recover the missing vowels. Thirdly, if the component words of the compound Arabic NEs were found in AWN they are assigned the same vowelization as in AWN. And lastly if none of these worked, the NE was left for manual vowelization. However, their pipeline is not clear as they did not describe how they decided how words belonged to each of the categories, and they did not report numerical results as to how much NEs were diacritized automatically and how much were done manually.

We developed a diacritization pipeline for restoring vowel marks for Arabic NEs extracted from AWK. The pipeline uses different methods ranging from matching with lists of diacritized names, running words through a state-of-the-art SVM diacritizer, and linguistically-motivated rule-based methods as described in Figure 2 below.

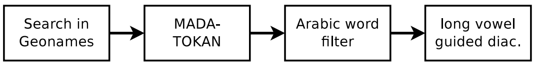

Figure 2: Arabic Named Entity Diacritization Pipeline

3.4.1. Searching in Geonames

In this processing stage we extract geonames from two sources, www.geonames.org and www.geonames.de, and the names extracted are compared to the NEs that we extract from AWK, and when they are matched, the correct diacritization is selected.

www.geonames.org

This is a geographical database that covers 7,226,537 locations around the word with geographical information on longitude/latitude, population size, administrative divisions, etc. The web site provides dumps for downloading the data. One interesting linguistic feature of the database is that it provides translations into various languages. There are 35,125 locations that have equivalences in Arabic. The transliteration system used is UNGEGN (United Nations Conference on the Standardization of Geographical Names). The Buckwalter transliteration system (used in AWN) is an ASCII-only, strictly orthographical transliteration scheme, representing Arabic orthography on a basis of one-to-one letter transformation. This is different from the former transliteration scheme that take into account phonological information not expressed in the Arabic script. The process of transforming UNGEGN into Buckwalter involves three steps: mapping, scaling and normalizing.

1. Mapping. Some Arabic names are provided with transliterations while others are provided with translations. In this step we decide whether the Latin script is a direct transliteration of Arabic or not. This involves resolving ambiguities related to *hamza*s, *taa' marboutah*, assimilated definite article, doubled letters, etc. The result we obtained for the mapping is that 29181 out of 35125 names were mapped correctly (83%).

2. Scaling. In this step we use the vowel marks in the transliteration and scale it up to give the diacritics for the words in the native Arabic script. Almost all the NEs that were correctly mapped were scaled to have full diacritic marks, that is 29148 names out of 29181 or 99.9%.

3. Normalizing. We use the combined information from Arabic script and UNGEGN transliteration to resolve ambiguities and correct common misspellings. This process involves two steps. First, resolving the ambiguity in the UNGEGN transliteration related to:

- '*h*' in the final position which could be *haa'* ه or *taa' marboutah* ة.
- *Hamza* '*'*' could stand for one of five letters in Arabic أ, إ, ؤ, ئ and ء.

Second, correcting common misspellings in the Arabic script words related to:

- *taa' marboutah* ة in the final position could be misspelled as ه or *haa'*. This misspelling is pointed out when the transliteration has a final 't'.
- *Hamza* on or under *alif* could be dropped. We make use of the general morphological rule that "a *hamza* must be expressed in proper names" and we rely on hints from the pronunciation indicated by the transliterations to restore the correct *Hamza*.

For example, Muḩāfaz̧at al Iskandarīyah – محافظة الإسكندرية – مُحَافَظَةُ الإِسْكَنْدَرِيَّةُ, "Governorate of Alexandria".

There is a disadvantage of the geonames.org database, however, in that it is hugely over-representative of Iraqi locations. We found that the data contains 26,756 names of locations from Iraq alone. This is 92% of the Arabic dataset, which leaves us with only 2,392 geonames for the rest of the world.

www.geonames.de

This is another source of geographical names that provides information on world countries and their administrative divisions, with translation into various languages. A dump is not provided and we had to write wrappers to crawl the web site and extract information from web pages and align the results. geonames.de uses a different transliteration system called DIN 31635 . This is a DIN (Deutsches Institut

für Normung, or the German Institute for Standardization) standard for the transliteration of the Arabic alphabet. This is the system used for the transliteration of Arabic through this paper, except where indicated otherwise. The total size of the Arabic dataset is 5,947, including names that have spelling variants (the number of unique headwords is 5,272). We do not use mapping here because all Arabic words are provided with transliterations, but we still need to use scaling and normalizing as mentioned above. One common mistake in this dataset that we needed to nromalize was using *alif maqsoura* ى into *yaa*. 5,826 genames were scaled successfully for this resource. For example, al-Im̄ar̄aṭu lʿArab̄ıyâtu l-Muttaḥidâ – الإمارات العربيّة
الإِمَارَاتُ العَرَبِيَّةُ المُتَّحِدَةُ – المتّحدة, "United Arab Emirates".

The unique combined list from both geonames.org and geonames.de is 30,838 location names. When the list is matched with the AWK NEs, 10% of the names were recognized. In more details, out of 36,567 Arabic NEs extracted from the AWK, 943 were found in geonames.org and 2,656 were found in geonames.de. This means that although geonames.de is far smaller in size than geonames.org, yet it is more efficient because, as mentioned above, geonames.org is highly over-representative of Iraqi locations.

3.4.2. MADA+TOKAN Diacritization

In the second step of our diacritization pipeline, we use MADA+TOKAN (Habash et al., 2009), a state-of-the-art, freely-available toolkit. MADA operates by examining a list of all possible analyses for each word generated by the Buckwalter Morphological Analyser (BAMA), and then selecting the analysis that matches the current context best by means of support vector machine models classifying for 19 distinct, weighted morphological features to provide complete diacritic, lexemic, glossary and morphological information. One limitation of MADA+TOKAN is that if no analysis is given by BAMA, no lemmatization or diacritization is undertaken. BAMA is already limited in its coverage of proper nouns; out of 40,222 lemmas in BAMA there are only 2034 lemmas classified as proper nouns. Another limitation is that MADA+TOKAN is trained and tested on the Penn Arabic Treebank (ATB) and therefore its coverage and quality with other text types is not guaranteed.

We took the list of NEs that were not matched with the geonames list and analysed it using MADA+TOKAN. The result is that out of a total of 26894 unique words (after breaking the list of NEs into single types and removing repetitions), 10083 words received an analysis by MADA+TOKAN, which means a coverage of 37%. Here are two examples of the output: اَلكَعبَةُ, AlkaEobap, "Kaaba" and جَامِعَةُ الخَرطُوم, jAmiEap AlxaroTuwm, "University of Khartoum".

However, an analysis of the results from MADA+TOKAN shows that not all words are analysed as proper nouns. Only 2955 out of the 10083 (29%) were detected as proper nouns, and an incorrect POS analysis usually signals a potentially wrong diacritization choice.

3.4.3. Arabic Word Filter

Next we need to consider words for which no analysis by MADA+TOKAN was found, or out-of-vocabulary (OOV) words. We build our post-processing cycles around the following assumptions:

- Most unknown words are foreign names transliterated into Arabic.
- Transliteration of foreign names usually employs long vowels instead of short vowels to facilitate correct pronunciation without resorting to short vowels which are usually ignored in writing.
- Arabic names do not follow this assumption and need to be excluded.
- The phonetic properties of many Arabic letters (or sounds) are not found in European languages.

Based on these assumptions we create a filter to exclude potential Arabic words from the subsequent long-vowel-guided diacritization step. The Arabic alphabet consists of 28 letters that can be extended to 35 when adding *taa' marboutah*, *alif maqsourah* and the five different shapes of *hamza*. Out of those 35 characters, there are 12 characters (roughly one-third) ض ط ظ ق ع ح ة ء ؤ ئ ى
ص that are restricted to native Arabic words and seldom, if ever, used in transliteration. This is due to the phonological fact that they denote sounds specific, to a great extent, to Arabic and some Semitic languages. By checking these letters on the International Phonetic Alphabet (IPA) for Arabic[9], we find that most of them have no equivalent in English. When we use this filter, 45% of the total NE database is identified as Arabic names and the remaining 55% as transliterated. But when we apply the filter to the words tagged as unknown by MADA+TOKAN, only 8% of the words were detected as Arabic native words. These were found to be either transliterations from Hebrew, unconventional transliterations or misspellings.

3.4.4. Long-Vowel-Guided Diacritization

Having filtered out possible Arabic words and assuming that the list we have consists of foreign names transliterated into Arabic, we apply our long-vowel-guided diacritization algorithm. The algorithm is a simple set of rules based on the fact that there are three long vowels in Arabic and these are represented by letters not diacritics. These are ا *alif*, و *waw* and ي *yaa'*. *Alif* is the only unambiguous vowel, *waw* and *yaa* can stand for glides besides long vowels. If any letter is followed by an *alif*, it must have the short vowel *fathah*, then, if it is followed by a *waw* or *yaa* (that are not themselves followed by *alif*), it will have the short vowel *dammah* or *kasrah*, respectively. Then, if a letter is found preceded by long vowels (*alif*, *waw* or *yaa*) and followed by a diacritized letter, it will have *sukoun*. The result of this step is that 59% of the unknown words are fully diacritized, as shown by the following examples.

Victor Hugo – فِيكْتُوْرْ هُوجُو – فيكتور هوجو
Barack Obama – بَارَاك أُوْبَامَا – باراك أوباما

[9] http://en.wikipedia.org/wiki/Wikipedia:IPA_for_Arabic

Nicolas Sarkozy – نيكولا ساركوزي – نِيكُولاَ سَاْرْكُوزِي

The result of the combined processes of the diacritization pipeline is that 73% of the NEs in Wikipedia are fully diacritized. We consider this as a satisfactory result although it is not possible to compare it to other work. Previous diacritization systems report on the results of diacritization for normal texts that, in addition to NEs, contain common nouns, verbs, adjectives and function words. The quality reported is usually over 90%, but no separate statistics are given for NEs. Out of vocabulary (OOV) words are generally responsible for the drop in accuracy in these systems and most of these OOV words are NEs.

3.5. LMF

Finally, in order to make the procedures independent from specific LRs we provide an output format compliant to standards. The elements that are part of this output are mainly NEs, orthographic variants of these NEs and classes to which these NEs belong (by means of “instance of” relations). Due to the fact that this data could be naturally represented by means of a LR and because the final aim is to extend LRs with this information we have decided to follow LMF, an ISO standard for the representation of lexicons, in order to encode the output.

We have developed a NE repository as a database whose structure is compliant with LMF. Compared to the initial design of the MINELex database, we have added the representation of diacritics and meta content (confidence, number of occurrences) of the acquired NEs. The reason to add this meta content is to allow the use of the repository with different levels of granularity for different tasks, i.e. one could use only the NEs above a certain confidence threshold or use only the most widely known NEs (those that occur more than a given number of times).

The appendix shows an example of the LMF representation for an Arabic NE, both in the XML and database formats.

4. Results and Discussion

The data used for the evaluation comprises AWN, PWN 2.1, the automatic mappings between PWN 2.1 and PWN 2.0 (Daudé et al., 2003) and a dump of AWK obtained in February 2010, which contains 234,109 articles and 32,746 categories. In order to evaluate the NE identification, we have manually annotated a randomly selected set of 1,000 articles that belong to the categories and subcategories covered by the mapping classifying them in two categories: NEs (93.9%) and non NEs (6.1%).

Tables 1 and 2 present the results for NE identification over the annotated set using (i) only the translations in the 10 aforementioned languages and (ii) also the lists described in 3.2.1., respectively. The tables show the precision, recall, $F_{\beta=1}$ and $F_{\beta=0.5}$ for different values of a threshold (the minimum percentage of occurrences beginning with capital letters to be considered a NE).

As expected, exploiting the NE list obtained from analysing the abstracts has a notable impact on recall (around 15% absolute improvement), while precision increases very slightly.

Table 1: NE identification using Wikipedia for ten languages

Threshold	**Precision**	**Recall**	$\mathbf{F}_{\beta=1}$	$\mathbf{F}_{\beta=0.5}$
0.01	94.70	51.33	66.57	81.01
0.11	95.82	51.22	66.76	81.61
0.21	96.39	51.22	66.9	81.94
0.31	97.74	50.69	66.76	82.44
0.41	98.33	50.16	66.43	82.49
0.51	98.52	49.73	66.1	82.36
0.61	98.5	48.99	65.43	81.94
0.71	98.88	46.96	63.68	80.98
0.81	99.31	45.69	62.58	80.43
0.91	99.25	42.39	59.4	78.25

Table 2: NE identification using Wikipedia for ten languages and keyword search

Threshold	**Precision**	**Recall**	$\mathbf{F}_{\beta=1}$	$\mathbf{F}_{\beta=0.5}$
0.01	95.83	66.13	78.26	87.94
0.11	96.72	66.03	78.48	88.5
0.21	97.18	66.03	78.63	88.8
0.31	98.09	65.5	78.54	89.2
0.41	98.55	65.07	78.38	89.35
0.51	98.7	64.64	78.12	89.29
0.61	98.69	64	77.65	89.04
0.71	98.99	62.62	76.71	88.69
0.81	99.31	61.45	75.92	88.42
0.91	99.28	58.68	73.76	87.21

Table 3 shows the amount of NEs, instance relations and written forms acquired for the different intervals of the threshold both without and with NE lists.

The postprocessing phase (see 3.3.) adds additional NEs by exploiting the English, Spanish and Italian NEs of the repository. 11,784 English, 6,869 Italian and 6,937 Spanish NEs have an equivalent in Arabic. Discarding duplicates in these three sets and NEs that have been already extracted for Arabic, there remain 6,586 NEs, which are added to the Arabic repository. Finally, the repository contains 44,315 Arabic NEs.

5. Conclusions

We have adapted and extended a generic methodology to automatically create a NE lexicon by exploiting AWN and AWK. The different steps regarding the construction of this resource including mapping, NE identification, postprocessing and diacritisation have been discussed and evaluated. We use the LMF standard for representation in order to provide a classification of entities in the nodes of taxonomy. The resulting resource contains approximately 45,000 Arabic NEs and can be used with different levels of granularity for NE recognition. We believe that the resource created is very useful for real world applications, such as parsing, Machine Translation and Question Answering systems. In the future, we intend to develop more heuristics to improve the recall and thus capture more NEs. The resulting NE repository and the manually annotated

Table 3: Extracted NEs

Lists	Threshold	NEs	Relations	Written forms
no	≥ 0.91	23,910	27,422	24,887
	≥ 0.81	25,620	29,398	26,717
	≥ 0.71	26,480	30,421	27,649
	≥ 0.61	27,077	31,138	28,323
	≥ 0.51	27,469	31,619	28,778
	≥ 0.41	28,048	32,287	29,451
	≥ 0.31	28,562	32,890	30,034
	≥ 0.21	29,261	33,671	30,866
	≥ 0.11	30,079	34,593	31,875
	≥ 0.01	30,354	34,901	32,205
yes	≥ 0.91	31,284	36,271	32,386
	≥ 0.81	32,995	38,247	34,212
	≥ 0.71	33,855	39,270	35,143
	≥ 0.61	34,452	39,987	35,815
	≥ 0.51	34,844	40,468	36,268
	≥ 0.41	35,423	41,136	36,940
	≥ 0.31	35,937	41,739	37,522
	≥ 0.21	36,636	42,520	38,354
	≥ 0.11	37,454	43,442	39,363
	≥ 0.01	37,729	43,750	39,693
	-	7,375	8,849	7,573

NE set can be found at `http://www.ilc.cnr.it/ne-repository`.

Acknowledgements

This research has been partially funded by the PANACEA consortium, which is a 7^{th} Framework Research Programme of the European Union (EU), contract number 7FP-ITC-248064, and by funding received from the IRCSET (Irish Research Council for Science, Engineering and Technology) and from Enterprise Ireland.

6. References

Saleem Abuleil. 2004. Extracting names from arabic text for question-answering systems. In *RIAO'04*, pages 638–647, University of Avignon (Vaucluse), France.

Musa Alkhalifa and Horacio Rodríguez. 2009. Automatically extending ne coverage of arabic wordnet using wikipedia. In *CITALA'09*, Rabat, Morocco.

Y. Benajiba and P. Rosso. 2007. Anersys 2.0 : Conquering the ner task for the arabic language by combining the maximum entropy with pos-tag information. In *IICAI-2007*, Pune, India.

Y. Benajiba, M. Diab, and P. Rosso. 2008. Arabic named entity recognition using optimized feature sets. In *EMNLP-2008*, Honolulu, Hawaii.

Jordi Daudé, Lluís Padró, and German Rigau. 2003. Making wordnet mappings robust. In *Proceedings of the 19th Congreso de la Sociedad Española para el Procesamiento del Lenguage Natural, SEPLN*, Universidad Universidad de Alcalá de Henares. Madrid, Spain.

S. Elkateb, W. Black, H. Rodríguez, M. Alkhalifa, P. Vossen, A. Pease, and C. Fellbaum. 2006. Building a wordnet for arabic. In *LREC'06*, Genoa, Italy.

A. Elsebai, F. Meziane, and F.Z. Belkredim. 2009. A rule based persons names arabic extraction system. In *The IBIMA*, Cairo, Egypt.

Husni Al-Muhtaseb Mustafa Elshafei and Mansour Al-Ghamdi. 2006. Statistical methods for automatic diacritization of arabic text. In *The Saudi 18th National Computer Conference (NCC18)*, Riyadh, Saudi Arabia.

Benjamin Farber, Dayne Freitag, Nizar Habash, and Owen Rambow. 2008. Improving ner in arabic using a morphological tagger. In *LREC'08*, Marrakech, Morocc.

C. Fellbaum. 1998. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press, May.

Sergio Ferrández, Antonio Toral, Óscar Ferrández, Antonio Ferrández, and Rafael Muñoz. 2007. Applying wikipedia's multilingual knowledge to cross-lingual question answering. In Zoubida Kedad, Nadira Lammari, Elisabeth Métais, Farid Meziane, and Yacine Rezgui, editors, *NLDB*, volume 4592 of *Lecture Notes in Computer Science*. Springer.

G. Francopoulo, N. Bel, M. George, N. Calzolari, M. Monachini, M. Pet, and C. Soria. 2009. Multilingual resources for nlp in the lexical markup framework. In *Language Resources and Evaluation Journal (forthcoming)*.

F. Gey. 2000. Research to improve cross-language retrieval - position paper for clef. In *Lecture Notes in Computer Science 2069*, pages 83–88, Berlin: Springer.

Nizar Habash and Owen Rambow. 2005. Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop. In *Proceedings of ACL05*, pages 573–580, Ann Arbor, Michigan. ACL.

Nizar Habash, Owen Rambow, and Ryan Roth. 2009. Mada+tokan: A toolkit for arabic tokenization, diacritization, morphological disambiguation, pos tagging, stemming and lemmatization. In *The 2nd International Conference on Arabic Language Resources and Tools (MEDAR)*, Cairo, Egypt.

A. Khemakhem, B. Gargouri, A. Abdelwahed, and G. Francopoulo. 2007. Modélisation des paradigmes de flexion des verbes arabes selon la norme lmf - iso 24613. In *Proceedings of TALN*.

N. Loukil, K. Haddar, and A. Ben Hamadou. 2007. Normalisation de la représentation des lexiques syntaxiques arabes pour les formalismes d'unification. In *Proceedings of Colloque Lexique et Grammaire*.

J. Maloney and M. Niv. 1998. Tagarab: A fast, accurate arabic name recogniser using high precision morphological analysis. In *The Workshop on Computational Approaches to Semitic Languages*, pages 8–15, Montreal, Canada.

S. Mesfar. 2007. Named entity recognition for arabic using syntactic grammars. In *The 12th International Conference on Application of Natural Language to Information Systems*, pages 305–316, Paris, France.

Rani Nelken and Stuart M. Shieber. 2005. Arabic diacritization using weighted finite-state transducers. In *The ACL 2005 Workshop On Computational Approaches To Semitic Languages the ACL 2005 Workshop On Compu-*

tational Approaches To Semitic Languages, Ann Arbor, Michigan.

M. Rashwan, M. Al-Badrashiny, M. Attia, and S. Abdou. 2009. A hybrid system for automatic arabic diacritization. In *The 2nd International Conference on Arabic Language Resources and Tools*, Cairo, Egypt.

R. Rodríguez, D. Farwell, J. Farreres, M. Bertran, M. Alkhalifa, M.A. Mart., W. Black, S. Elkateb, J. Kirk, A. Pease, P. Vossen, and C. Fellbaum. 2008. Arabic wordnet: Current state and future extensions. In *The Fourth Global WordNet Conference*, Szeged, Hungary.

Doaa Samy, Antonio Moreno, and Jose M. Guirao. 2005. A proposal for an arabic named entity tagger leveraging a parallel corpus. In *RANLP*, Borovets, Bulgaria.

K. Shaalan and H. Raza. 2009. Nera: Named entity recognition for arabic. In *JASIST, John Wiley and Sons*, pages 1652–1663, NJ, USA.

Antonio Toral. 2009. *Enrichment of Language Resources by Exploiting New Text and the Resources Themselves. A case study on the acquisition of a NE lexicon*. Ph.D. thesis, Universitat d'Alacant.

P. Vossen. 1998. *EuroWordNet A Multilingual Database with Lexical Semantic Networks*. Kluwer Academic publishers.

Imed Zitouni and Ruhi Sarikaya. 2009. Arabic diacritic restoration approach based on maximum entropy models. In *Computer Speech Language*, volume Volume 23, Issue 3, pages 257–276.

Appendix. LMF output

Figure 3 shows the LMF compliant XML code for the NE الأمم المتّحدة "United Nations". Tables 4, 5, 6, 7, 8, 9, 10 and 11 show the equivalent content in MINELex database format.

Table 4: NE Repository. LexicalEntry table

LE id	LE pos
ar_le_الأمم المتّحدة	PN
ar_le_mnZm	N
en_le_United_Nations	PN

Table 5: NE Repository. FormRepresentation table

LE id	written form	v. type	script	orthog. n.
ar_le_الأمم المتّحدة	ar_الأمم المتّحدة	full	Arab	arabicUnpointed
ar_le_الأمم المتّحدة	ar_اَلأُمَم المُتَّحِدَة	full	Arab	arabicPointed
ar_le_mnZm	ar_mnZm	full	Latin	
en_le_United_Nations	en_United_Nations	full	Latin	

Table 6: NE Repository. Sense table

S id	LE id	res.	res. id
ar_s_الأمم المتّحدة	ar_le_الأمم المتّحدة	ar_WK	2270
ar_s_109710501	ar_le_mnZm	ar_WN	109710501
en_s_United_Nations	en_le_United_Nations	en_WK	31769

```
<LexicalResource dtdVersion="16">
 <GlobalInformation>
  <feat att="label" val="MINELex"/>
 </GlobalInformation>
 <Lexicon>
  <feat att="Language" val="ar"/>
  <LexicalEntry id="ar_le_الأمم_المتحدة">
   <Lemma>
    <feat att="partOfSpeech" val="proper noun"/>
    <FormRepresentation>
     <feat att="writtenForm" val="الأمم_المتحدة"/>
     <feat att="variantType" val="full"/>
     <feat att="script" val="Arab"/>
     <feat att="orthographyName" val="arabicUnpointed"/>
    </FormRepresentation>
    <FormRepresentation>
     <feat att="writtenForm" val="الأُمَم_المُتَّحِدَة"/>
     <feat att="variantType" val="full"/>
     <feat att="script" val="Arab"/>
     <feat att="orthographyName" val="arabicPointed"/>
    </FormRepresentation>
   </Lemma>
   <Sense id="ar_s_الأمم_المتحدة">
    <SenseRelation targets="ar_s_109710501">
     <feat att="semanticrelation" val="instance_of"/>
    </SenseRelation>
    <MonolingualExternalRef>
     <feat att="external_system" val="ar_Wikipedia"/>
     <feat att="external_reference" val="2270"/>
    </MonolingualExternalRef>
    <Confidence mode="wiki10" occurrences="250" confidence="0.996"/>
   </Sense>
  </LexicalEntry>
  <LexicalEntry id="ar_le_mnZm">
   <Lemma>[...]</Lemma>
   <Sense id="ar_s_mnZm">
    <MonolingualExternalRef>
     <feat att="external_system" val="ar_WN"/>
     <feat att="external_reference" val="109710501"/>
    </MonolingualExternalRef>
   </Sense>
  </LexicalEntry>
 </Lexicon>
 <Lexicon>
  <feat att="Language" val="en"/>
  <LexicalEntry id="en_le_United_Nations">[...]</LexicalEntry>
 </Lexicon>
 <SenseAxis id="sa_001" senses="en_s_United_Nations ar_s_الأمم_المتحدة">
  <feat att="type" val="eq_syn"/>
  <InterlingualExternalRef>
   <feat att="external_system" val="SUMO"/>
   <feat att="external_reference" val="PoliticalOrganization"/>
   <feat att="external_reltype" val="at"/>
  </InterlingualExternalRef>
 </SenseAxis>
</LexicalResource>
```

Figure 3: Method diagram

Table 7: NE Repository. SenseRelation table

source id	target id	relation
ar_s_الأمم المتّحدة	ar_s_109710501	instanceOf

Table 8: NE Repository. SenseAxis table

SA id	type
1	eq synonym

Table 9: NE Repository. SenseAxisElements table

SA id	element
1	ar_s_الأمم المتّحدة
1	en_s_United_Nations

Table 10: NE Repository. SenseAxisExternalRef table

SA id	resource	resource id	relation
1	SUMO	PoliticalOrganization	at

Table 11: NE Repository. Confidence table

S id	mode	occurrences	confidence
ar_s_الأمم المتّحدة	wiki10	250	0.996