Automatic Creation of Arabic Named Entity Annotated Corpus Using Wikipedia

Maha Althobaiti, Udo Kruschwitz, and Massimo Poesio
School of Computer Science and Electronic Engineering
University of Essex
Colchester, UK
{mjaltha, udo, poesio}@essex.ac.uk

Abstract

In this paper we propose a new methodology to exploit Wikipedia features and structure to automatically develop an Arabic NE annotated corpus. Each Wikipedia link is transformed into an NE type of the target article in order to produce the NE annotation. Other Wikipedia features - namely redirects, anchor texts, and inter-language links - are used to tag additional NEs, which appear without links in Wikipedia texts. Furthermore, we have developed a filtering algorithm to eliminate ambiguity when tagging candidate NEs. Herein we also introduce a mechanism based on the high coverage of Wikipedia in order to address two challenges particular to tagging NEs in Arabic text: rich morphology and the absence of capitalisation. The corpus created with our new method (*WDC*) has been used to train an NE tagger which has been tested on different domains. Judging by the results, an NE tagger trained on *WDC* can compete with those trained on manually annotated corpora.

1 Introduction

Supervised learning techniques are well known for their effectiveness to develop Named Entity Recognition (NER) taggers (Bikel et al., 1997; Sekine and others, 1998; McCallum and Li, 2003; Benajiba et al., 2008). The main disadvantage of supervised learning is that it requires a large annotated corpus. Although a substantial amount of annotated data is available for some languages, for other languages, including Arabic, more work is needed to enrich their linguistic resources. In fact, changing the domain or just expanding the set of classes always requires domain-specific experts and new annotated data, both of which cost time and effort. Therefore, current research focuses on approaches that require minimal human intervention to facilitate the process of moving the NE classifiers to new domains and to expand NE classes.

Semi-supervised and unsupervised learning approaches, along with the automatic creation of tagged corpora, are alternatives that avoid manually annotated data (Richman and Schone, 2008; Althobaiti et al., 2013). The high coverage and rich informational structure of online encyclopedias can be exploited for the automatic creation of datasets. For example, many researchers have investigated the use of Wikipedia's structure to classify Wikipedia articles and to transform links into NE annotations according to the link target type (Nothman et al., 2008; Ringland et al., 2009).

In this paper we present our approach to automatically derive a large NE annotated corpora from Arabic Wikipedia. The key to our method lies in the exploitation of Wikipedia's concepts, specifically anchor texts[1] and redirects, to handle the rich morphology in Arabic, and thereby eliminate the need to perform any deep morphological analysis. In addition, a capitalisation probability measure has been introduced and incorporated into the approach in order to replace the capitalisation feature that does not exist in the Arabic script. This capitalisation measure has been utilised in order to filter ambiguous Arabic NE phrases during annotation process.

The remainder of this paper is structured as follows: Section 2 illustrates structural information about Wikipedia. Section 3 includes background information on NER, including recent work. Section 4 summarises the proposed methodology. Sections 5, 6, and 7 describe the proposed algorithm in detail. The experimental setup and the evaluation results are reported and discussed in Section 8. Finally, the conclusion features comments regarding our future work.

[1]The terms 'anchor texts' and 'link labels' are used interchangeably in this paper.

Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics, pages 106–115,
Gothenburg, Sweden, April 26-30 2014.

2 The Structure of Wikipedia

Wikipedia is a free online encyclopedia project written collaboratively by thousands of volunteers, using MediaWiki[2]. Each article in Wikipedia is uniquely identified by its title. The title is usually the most common name for the entity explained in the article.

2.1 Types of Wikipedia Pages

2.1.1 Content Pages

Content pages (aka Wikipedia articles) contain the majority of Wikipedia's informative content. Each content page describes a single topic and has a unique title. In addition to the text describing the topic of the article, content pages may contain tables, images, links and templates.

2.1.2 Redirect Pages

A redirect page is used if there are two or more alternative names that can refer to one entity in Wikipedia. Thus, each alternative name is changed into a title whose article contains a redirect link to the actual article for that entity. For example, 'UK' is an alternative name for the 'United Kingdom', and consequently, the article with the title 'UK' is just a pointer to the article with the title 'United Kingdom'.

2.1.3 List_of Pages

Wikipedia offers several ways to group articles. One method is to group articles by lists. The items on these lists include links to articles in a particular subject area, and may include additional information about the listed items. For example, 'list of scientists' contains links to articles of scientists and also links to more specific lists of scientists.

2.2 The Structure of Wikipedia Articles

2.2.1 Categories

Every article in the Wikipedia collection should have at least one category. Categories should be on vital topics that are useful to the reader. For example, the Wikipedia article about the United Kingdom in Wikipedia is associated with a set of categories that includes 'Countries bordering the Atlantic Ocean', and 'Countries in Europe'.

[2]An open source wiki package written in PHP

2.2.2 Infobox

An infobox is a fixed-format table added to the top right-hand or left-hand corner of articles to provide a summary of some unifying parameters shared by the articles. For instance, every scientist has a name, date of birth, birthplace, nationality, and field of study.

2.3 Links

A link is a method used by Wikipedia to link pages within wiki environments. Links are enclosed in doubled square brackets. A vertical bar, the 'pipe' symbol, is used to create a link while labelling it with a different name on the current page. Look at the following two examples,

1 - [[a]] is labelled 'a' on the current page and links to taget page 'a'.

2 - [[$a|b$]] is labelled 'b' on the current page, but links to target page 'a'.

In the second example, the *anchor text* (aka *link label*) is 'a', while 'b', a *link target*, refers to the title of the target article. In the first example, the anchor text shown on the page and the title of the target article are the same.

3 Related Work

Current NE research seeks out adequate alternatives to traditional techniques such that they require minimal human intervention and solve deficiencies of traditional methods. Specific deficiencies include the limited number of NE classes resulting from the high cost of setting up corpora, and the difficulty of adapting the system to new domains.

One of these trends is distant learning, which depends on the recruitment of external knowledge to increase the performance of the classifier, or to automatically create new resources used in the learning stage.

Kazama and Torisawa (2007) exploited Wikipedia-based features to improve their NE machine learning recogniser's F-score by three percent. Their method retrieved the corresponding Wikipedia entry for each candidate word sequence in the CoNLL 2003 dataset and extracted a category label from the first sentence of the entry.

The automatic creation of training data has also been investigated using external knowledge. An et al. (2003) extracted sentences containing listed entities from the web, and produced a 1.8 million Korean word dataset. Their corpus

performed as well as manually annotated training data. Nothman et al. (2008) exploited Wikipedia to create a massive corpus of named entity annotated text. They transformed Wikipedia's links into named entity annotations by classifying the target articles into standard entity types[3]. Compared to MUC, CoNLL, and BBN corpora, their Wikipedia-derived corpora tend to perform better than other cross-corpus train/test pairs.

Nothman et al. (2013) automatically created massive, multilingual training annotations for named entity recognition by exploiting the text and internal structure of Wikipedia. They first categorised each Wikipedia article into named entity types, training and evaluating on 7,200 manually-labelled Wikipedia articles across nine languages: English, German, French, Italian, Polish, Spanish, Dutch, Portuguese, and Russian. Their cross-lingual approach achieved up to 95% accuracy. They transformed Wikipedia's links into named entity annotations by classifying the target articles into standard entity types. This technique produced reasonable annotations, but was not immediately able to compete with existing gold-standard data. They better aligned their automatic annotations to the gold standard corpus by deducing additional links and heuristically tweaking the Wikipedia corpora. Following this approach, millions of words in nine languages were annotated. Wikipedia-trained models were evaluated against CONLL shared task data and other gold-standard corpora. Their method outperformed Richman and Schone (2008) and Mika et al. (2008), and achieved scores 10% higher than models trained on newswire when tested on manually annotated Wikipedia text.

Alotaibi and Lee (2013) automatically developed two NE-annotated sets from Arabic Wikipedia. The corpora were built using the mechanism that transforms links into NE annotations, by classifying the target articles into named entity types. They used POS-tagging, morphological analysis, and linked NE phrases to detect other mentions of NEs that appear without links in text. By contrast, our method does not require POS-tagging or morphological analysis and just identifies unlinked NEs by matching phrases from an automatically constructed and filtered alternative names with identical terms in the articles texts, see Section 6. The first dataset created by Alotaibi and Lee (2013) is called *WikiFANE(whole)* and contains all sentences retrieved from the articles. The second set, which is called *WikiFANE(selective)*, is constructed by selecting only the sentences that have at least one named entity phrase.

[3]The terms 'type', 'class' and 'category' are used interchangeably in this paper.

4 Summary of the Approach

All of our experiments were conducted on the 26 March 2013 Arabic version of the Wikipedia dump[4]. A parser was created to handle the mediawiki markup and to extract structural information from the Wikipedia dump such as a list of redirect pages along with their target articles, a list of pairs containing link labels and their target articles in the form '*anchor text, target article*', and essential information for each article (e.g., title, body text, categories, and templates).

Many of Wikipedia's concepts such as links, anchor texts, redirects, and inter-language links have been exploited to transform Wikipedia into a NE annotated corpus. More details can be found in the next sections. Generally, the following steps are necessary to develop the dataset:

1. Classify Wikipedia articles into a specific set of NE types.
2. Identify matching text in the title and the first sentence of each article and label the matching phrases according to the article type.
3. Label linked phrases in the text according to the NE type of the target article.
4. Compile a list of alternative titles for articles and filter out ambiguous ones.
5. Identify matching phrases in the list and the Wikipedia text.
6. Filter sentences to prevent noisy sentences being included in the corpus.

We explain each step in turn in the following sections.

5 Classifying Wikipedia Articles into NE Categories

Categorising Wikipedia articles is the initial step in producing NE training data. Therefore, all Wikipedia articles need to be classified into a specific set of named entity types.

[4]http://dumps.wikimedia.org/arwiki/

5.1 The Dataset and Annotation

In order to develop a Wikipedia document classifier, we used a set of 4,000 manually classified Wikipedia articles that are available free online[5]. The set was manually classified using the ACE (2008) taxonomy and a new class (*Product*). Therefore, there were eight coarse-grained categories in total: *Facility*, *Geo-Political*, *Location*, *Organisation*, *Person*, *Vehicle*, *Weapon*, and *Product*. As our work adheres to the CoNLL definition, we mapped these classified Wikipedia articles into CoNLL NE types – namely person, location, organisation, miscellaneous, or other – based on the CoNLL 2003 annotation guidelines (Chinchor et al., 1999).

5.2 The Classification of Wikipedia Articles

Many researchers have already addressed the task of classifying Wikipedia articles into named entity types (Dakka and Cucerzan, 2008; Tardif et al., 2009). Alotaibi and Lee (2012) is the only study that has experimented with classifying the Arabic version of Wikipedia into NE classes. They have explored the use of Naive Bayes, Multinomial Naive Bayes, and SVM for classifying Wikipedia articles, and achieved a F-score ranging from 78% and 90% using different language-dependent and independent features.

We conducted three experiments that used a simple bag-of-words features extracted from different portions of the Wikipedia document and metadata. We summarise the portions of the document included in each experiment below:

Exp1: Experiment 1 involved tokens from the article title and the entire article body.

Exp2: Rich metadata in Wikipedia proved effective for the classification of articles (Tardif et al., 2009; Alotaibi and Lee, 2012). Therefore, in Experiment 2 we included tokens from categories, templates – specifically 'Infobox' – as well as tokens from the article title and first sentence of the document.

Exp3: Experiment 3 involved the same set of tokens as experiment 2 except that categories and infobox features were marked with suffixes to differentiate them from tokens extracted from the article body text. This step of distinguishing tokens based on their location in the document improved the accuracy of document's classification (Tardif et al., 2009; Alotaibi and Lee, 2012).

[5]www.cs.bham.ac.uk/∼fsa081/

In order to optimise features, we implemented a filtered version of the bag-of-words article representation (e.g., removing punctuation marks and symbols) to classify the Arabic Wikipedia documents instead of using a raw dataset (Alotaibi and Lee, 2012). In addition, the same study shows the high impact of applying tokenisation[6] as opposed to the neutral effect of using stemming. We used the filtered features proposed in the study of Alotaibi and Lee (2012), which included removing punctuation marks, symbols, filtering stop words, and normalising digits. We extended the features, however, by utilising the tokenisation scheme that involves separating conjunctions, prepositions, and pronouns from each word.

The feature set has been represented using Term Frequency-Inverse Document Frequency ($TF - IDF$). This representation method is a numerical statistic that reflects how important a token is to a document.

5.3 The Results of Classifying the Wikipedia Articles

As for the learning process, our Wikipedia documents classifier was trained using Liblinear[7]. 80% of the 4,000 hand-classified Wikipedia articles were dedicated to the training stage, while 20% were specified to test the classifier. Table 1 is a comparison of the precision, recall, and F-measure of the classifiers that resulted from the three experiments. The Exp3 classifier performed better than the other classifiers. Therefore, it was selected to classify all of the Wikipedia articles. At the end of this stage, we obtained a list of pairs containing each Wikipedia article and its NE Type. We stored this list in a database in preparation for the next stage: developing the NE-tagged training corpus.

	Exp1			Exp2			Exp3		
	P	R	F	P	R	F	P	R	F
PER	73	60	66	92	86	89	94	95	94
LOC	67	69	68	82	90	86	87	92	89
ORG	60	62	61	89	90	89	89	91	90
MISC	58	53	55	86	89	87	88	91	89
NON	65	55	60	83	88	85	86	88	87
Overall	65	60	62	86	89	87	89	91	90

Table 1: The results of the three Wikipedia document classifiers.

[6]It is also called decliticization or segmentation.

[7]www.csie.ntu.edu.tw/∼cjlin/liblinear/

6 The Annotation Process

6.1 Utilising the Titles of Articles and Link Targets

Identifying corresponding words in the article title and the entire body of text and then tagging the matching phrases with the NE-type can be a risky process, especially for terms with more than one meaning. For example, the title of the article describing the city (كان, 'Cannes')[8] can also, in Arabic, refer to the past verb (كان, 'was'). The portion of the Wikipedia article unlikely to produce errors during the matching process is the first sentence, which usually contains the definition of the term the Wikipedia article is written about (Zesch et al., 2007).

When identifying matching terms in the article title and the first sentence, we found that article titles often contain abbreviations, while the first sentence spells out entire words. This pattern makes it difficult to identify matching terms in the title and first sentence, and frequently appears in biographical Wikipedia articles. For example, one article is entitled (ابو بكر الرازي, 'Abu Bakr Al-Razi'), but the first sentence states the full name of the person: (ابو بكر محمد بن يحيى بن زكريا الرازي, 'Abu Bakr Mohammad Bin Yahia Bin Zakaria Al-Razi'). Therefore, we decided to address the problem with partial matching. In this case, the system should first identify all corresponding words in the title and the first sentence. Second, the system should annotate them and all words that fall between, provided that:

- the sequence of the words in the article title and the text are the same in order to avoid errors in tagging. For example, if the title of the article is (نهر التايمز, 'The River Thames'), but the first sentence reads (التايمز هو نهر يقع في ..., 'The Thames is a river flowing through southern England....'), then the text will not be properly tagged.
- the number of tokens located between matched tokens is less than or equal to five[9].

Figure 1 shows one example of partial matching.

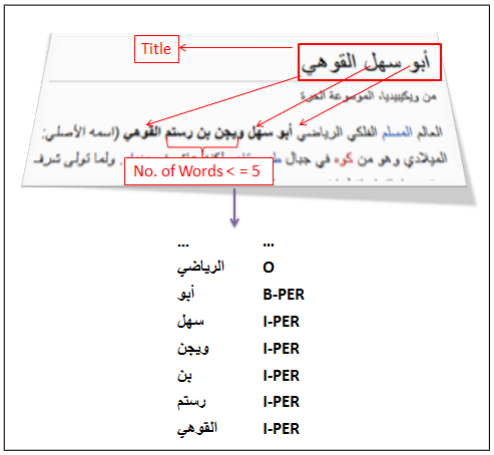

Figure 1: Example of Partial Matching

The next step is to transform the links between Wikipedia articles into NE annotations according to the link target type. Therefore, the link ([[اوباما|باراك اوباما]]/[[Barack Obama|Obama]]) would be changed to (اوباما *PER*) (Obama *PER*), since the link target (Barack Obama) is the title of an article about person. By the end of this stage, all NE anchor texts (anchor texts referring to NE articles) on Wikipedia should be annotated based on the NE-type of the target article.

6.2 Dictionaries of Alternative Names

Depending only on NE anchor texts in order to derive and annotate data from Wikipedia results in a low-quality dataset, as Wikipedia contains a fair amount of NEs mentioned without links. This can be attributed to the fact that each term on Wikipedia is more likely to be linked only on its first appearance in the article (Nothman et al., 2008). These unlinked NE phrases can be found simply by identifying the matching terms in the list of linked NE phrases[10] and the text. The process is not as straightforward as it seems, however, because identifying corresponding terms may prove ineffective, especially in the case of morphologically rich language in which unlinked NE phrases are sometimes found agglutinated to prefixes and conjunctions. In order to detect unlinked and inflected forms of NEs in Wikipedia text, we extended the list of articles titles that were used in the previous step to find and match the possible NEs in the text by including NE anchor texts. Adding NE anchor texts to the list assists in finding possible morphologically inflected NEs in the text while eliminating the need for any morpho-

[8]Throughout the entire paper, Arabic words are represented as follows: (Arabic word,'English translation').

[9]An informal experiment showed that the longest proper Arabic names are 5 to 7 tokens in length.

[10]The list of anchor texts that refer to NE articles

logical analysis. Table 2 shows examples from the dictionary of NE anchor texts.

Anchor Texts	English Gloss
والمغرب	and Morocco
بالمغرب	in Morocco
كالمغرب	such as Morocco
للمغرب	to Morocco
وكالمغرب	and such as Morocco

Table 2: Examples from the dictionary of NE Anchor Texts.

Spelling variations resulting from varied transliteration of foreign named entities in some cases prevent the accurate matching and identification of some unlinked NEs, if only the list of NE anchor texts is used. For example, (انجلترا, 'England') has been written five different ways: (انجلتره, انغلترا, انغلتره, انكلترا, انكلتره). Therefore, we compiled a list of the titles of redirected pages that send the reader to articles describing NEs. We refer to these titles in this paper as *NE redirects*. We consider to the lists of NE redirects and anchor texts a list of alternative names, since they can be used as alternative names for article titles.

The list of alternative names is used to find unlinked NEs in the text by matching phrases from the list with identical terms in the articles texts. This list is essential for managing spelling and morphological variations of unlinked NEs, as well as misspelling. Consequently, the process increases the coverage of NE tags augmented within the plain texts of Wikipedia articles.

6.2.1 Filtering the Dictionaries of Alternative Names

One-word alternative names: Identifying matching phrases in the list of alternative names and the text inevitably results in a lower quality corpus due to noisy names. The noisy alternative names usually occur with meaningful named entities. For example, the article on the person (ابو عبدالله الامين, 'Abu Abdullah Alamyn') has an alternative name consisting only of his last name (الامين, 'Alameen'), which means 'custodian'. Therefore, annotating every occurrence of 'Alamyn' as *PER* would lead to incorrect tagging and ambiguity. The same applies to the city with the name (الجديده, 'Aljadydah'), which literally means 'new'. Thus, the list of alternative names should be filtered to omit one-word NE phrases that usually have a meaning and are ambiguous when taken out of context.

In order to solve this problem, we introduced a capitalisation probability measure for Arabic words, which are never capitalised. This involved finding the English gloss for each one-word alternative name and then computing its probability of being capitalised using the English Wikipedia. To find the English gloss for Arabic words, we exploited Wikipedia Arabic-to-English cross-lingual links that provided us with a reasonable number of Arabic and corresponding English terms. If the English gloss for the Arabic word could not be found using inter-language links, we resorted to an online translator. Before translating the Arabic word, a light stemmer was used to remove prefixes and conjunctions in order to get the translation of the word itself without its associated affixes. Otherwise, the Arabic word (للبلاد) would be translated as (in the country). The capitalisation probability was computed as follows:

$$Pr[EN] = \frac{f(EN)_{isCapitalised}}{f(EN)_{isCapitalised}+f(EN)_{notCapitalised}}$$

where: EN is the English gloss of the alternative name; $f(EN)_{isCapitalised}$ is the number of times the English gloss EN is capitalised in English Wikipedia; and $f(EN)_{notCapitalised}$ is the number of times the English gloss EN is not capitalised in English Wikipedia.

This way, we managed to build a list of Arabic words and their probabilities of being capitalised. It is evident that the meaningful one-word NEs usually achieve a low probability. By specifying a capitalisation threshold constraint, we prevented such words from being included in the list of alternative names. After a set of experiments, we decided to use the capitalisation threshold equal to 0.75.

Multi-word alternative names: Multi-word alternative names (e.g., مصطفى محمود /'MusTafae Mahmud'), احمد عادل /'Ahmad Adel') rarely cause errors in the automatic annotation process. Wikipedians, however, at times append personal and job titles to the person's name contained in the anchor text, which refers to the article about that person. Examples of such anchor texts are (حاكم دبي محمد بن راشد, 'Ruler of Dubai Muhammad bin Rashid') and (رئيس مجلس الوزراء محمد بن راشد, 'President of the Council of Ministers Muhammad bin

Rashid'). As a result, the system will mistakenly annotate words like *Dubai, Council, Ministers* as *PER*. Our solution to this problem is to omit the multi-word alternative name, if any of its words belong to the list of apposition words, which usually appear adjacent to NEs such as (رئيس, 'President'), (وزير, 'Minister'), and (حاكم, 'Ruler'). The filtering algorithm managed to exclude 22.95% of the alternative names from the original list. Algorithm 1 shows pseudo code of the filtering algorithm.

Algorithm 1: Filtering Alternative Names

Input: A set $L = \{l_1, l_2, \ldots, l_n\}$ of all alternative names of Wikipedia articles
Output: A set $RL = \{rl_1, rl_2, \ldots, rl_n\}$ of reliable alternative names

```
for i ← 1 to n do
    T ← split l_i into tokens
    if (T.size() >= 2) then
        /* All tokens of T do not
           belong to apposition list
           */
        if (! containAppositiveWord(T)) then
            add l_i to the set RL
    else
        light_stem ← findLightStem(l_i)
        english_gloss ← translate(light_stem)
        /* Compute Capitalisation
           Probability for English
           gloss                    */
        cap_prob ← compCapProb(english_gloss)
        if (cap_prob > 0.75) then
            add l_i to the set RL
```

The dictionaries derived from Wikipedia by exploiting Wikipedia's structure and adopting the filtering algorithm is shown in Table 3.

Dictionary	Number of entries
Redirects	182,808
• List of NE *Redirects*	94,606
• Filtered list of NE *Redirects*	74,073
Anchor Texts	689,171
• List of NE *Anchor Texts*	130,692
• Filtered list of NE *Anchor Texts*	99,512

Table 3: Dictionaries derived from Wikipedia.

6.3 Post-processing

The goal of Post-processing was to address some issues that arose during the annotation process as a result of different domains, genres, and conventions of entity types. For example, nationalities and other adjectival forms of nations, religions, and ethnic groups are considered *MISC* in the CoNLL NER task in the English corpus, while the Spanish corpus consider them *NOT* named entities (Nothman et al., 2013). As far as we know, almost all Arabic NER datasets that followed the CoNLL style and guidelines in the annotation process consider nationalities *NOT* named entities. On Wikipedia all nationalities are linked to articles about the corresponding countries, which makes the annotation tool tag them as *LOC*. We decided to consider them *NOT* named entities in accordance with the CoNLL-style Arabic datasets. Therefore, in order to resolve this issue, we compiled a list of nationalities, and other adjectival forms of religion and ethnic groups, so that any anchor text matching an entry in the list was re-tagged as a *NOT* named entity.

The list of nationalities and apposition words used in section 6.2.1 were compiled by exploiting the 'List of' articles in Wikipedia such as *list of people by nationality*, *list of ethnic groups*, *list of adjectival forms of place names*, and *list of titles*. Some English versions of these 'List of' pages have been translated into Arabic, either because they are more comprehensive than the Arabic version, or because there is no corresponding page in Arabic.

7 Building the Corpus

After the annotation process, the last step was to incorporate sentences into the corpus. This resulted in obtaining an annotated dataset with around ten million tokens. However, in order to obtain a corpus with a large number of tags without affecting its quality, we created a dataset called Wikipedia-derived corpus (*WDC*), which included only sentences with at least three annotated named entity tokens. The *WDC* dataset contains 165,119 sentences consisting of around 6 million tokens. The annotation style of the *WDC* dataset followed the CoNLL format, where each token and its tag are placed together in the same file in the form $< token > \backslash s < tag >$. The NE boundary is specified using the *BIO* representation scheme, where *B-* indicates the beginning of the NE, *I-* refers to the continuation (Inside) of the NE, and O indicates that the word is not a NE. The *WDC* dataset is available online to the community of researchers[11]

[11] https://www.dropbox.com/sh/27afkiqvlpwyfq0/1hwWGqAcTL

8 Experimental Evaluation

To evaluate the quality of the methodology, we used *WDC* as training data to build an NER model. Then we tested the resulting classifier on datasets from different domains.

8.1 Datasets

For the evaluation purposes, we used three datasets: ANERcorp, NEWS, and TWEETS.

ANERcorp is a news-wire domain dataset built and tagged especially for the NER task by Benajiba et al. (2007). It contains around 150k tokens and is available for free. We tested our methodology on the ANERcorp test corpus because it is widely used in the literature for comparing with existing systems. The NEWS dataset is also a news-wire domain dataset collected by Darwish (2013) from the RSS feed of the Arabic version of news.google.com from October 2012. The RSS consists of the headline and the first 50 to 100 words in the news articles. This set contains approximately 15k tokens. The third test set was extracted randomly from Twitter and contains a set of 1,423 tweets authored in November 2011. It has approximately 26k tokens (Darwish, 2013).

8.2 Our Supervised Classifier

All experiments to train and build a probabilistic classifier were conducted using Conditional Random Fields (CRF)[12]. Regarding the features used in all our experiments, we selected the most successful features from Arabic NER work (Benajiba et al., 2008; Abdul-Hamid and Darwish, 2010; Darwish, 2013). These features include:

- The words immediately before and after the current word in their raw and stemmed forms.
- The first 1, 2, 3, 4 characters in a word.
- The last 1, 2, 3, 4 characters in a word.
- The appearance of the word in the gazetteer.
- The stemmed form of the word.

The gazetteer used contains around 5,000 entries and was developed by Benajiba et al. (2008). A light stemmer was used to determine the stem form of the word by using simple rules to remove conjunctions, prepositions, and definite articles (Larkey et al., 2002).

[12]http://www.chokkan.org/software/crfsuite/

8.3 Training the Supervised Classifier on Manually-annotated Data

The supervised classifier in Section 8.2 was trained on the ANERcorp training set. We refer to the resulting model as the *ANERcorp-Model*. Table 4 shows the results of the *ANERcorp-Model* on the ANERcorp test set. The table also shows the results of the state-of-the-art supervised classifier '*ANERcorp-Model(SoA)*' developed by Darwish (2013) when trained and tested on the same datasets used for *ANERcorp-Model*.

	ANERcorp-Model			ANERcorp-Model(SoA)		
	P	R	F	P	R	F
PER	88.2	69.7	77.87	87	77.7	82.09
LOC	94.07	80.9	86.99	92.3	87.8	89.99
ORG	84.2	58.7	69.17	81.4	66	72.90
Overall	88.82	69.77	78.15	86.9	77.17	81.74

Table 4: The results of Supervised Classifiers.

8.4 Results

We compared a system trained on *WDC* with the systems trained by Alotaibi and Lee (2013) on two datasets, *WikiFANE(whole)* and *WikiFANE(selective)*, which are also automatically collected from Arabic Wikipedia. The evaluation process was conducted by testing them on the ANERcorp set. The results shown in Table 5 prove that the methodology we proposed in this paper produces a dataset that outperforms the two other datasets in terms of recall and F-measure.

Classifier	P	R	F
WikiFAME(whole)	81.53	43.1	56.39
WikiFANE(selective)	88.1	37.52	52.63
WDC	76.44	56.42	64.92

Table 5: Comparison of the system trained on *WDC* dataset with the systems trained on *WikiFANE* datasets.

Table 6 compares the results of the ANERcorp-Model and the WDC-Model when testing them on datasets from different domains. Firstly, We decided to test the ANERcorp-Model and the WDC-Model on Wikipedia. Thus, a subset, containing around 14k tokens, of WDC set was allocated for testing purpose. The results in Table 6 shows that WDC classifier outperforms the F-score of the news-based classifier by around 48%.The obvious difference in the performance of the two classifiers can be attributed to the difference in annotation convention for different domains. For example, many key words in Arabic Wikipedia,

which appear in the text along with NEs (e.g., جامعة/university, مدينة/ city, شركة/company), are usually considered part of NE names. So, the phrase 'Shizuoka Prefecture' that is mentioned in some Arabic Wikipedia articles is considered an entity and linked to an article that talks about Shizuoka, making the system annotate all words in the phrase as NEs as follows: (محافظة *B-LOC* شيزوكا *I-LOC*/ Shizuoka *B-LOC* Prefecture *I-LOC*). On the other hand, in ANERcorp corpus, only the the word after the keyword (ولاية, 'Prefecture') is considered NE. In addition, although sport facilities (e.g., stadiums) are categorized in Wikipedia as *location*, some of them are not even considered entities in ANERcorp test corpus.

Secondly, the ANERcorp-Model and the WDC-Model were tested on the ANERcorp test data. The point of this comparison is to show how well the *WDC* dataset works on a news-wire domain, which is more specific than Wikipedia's open domain. The table shows that the ANERcorp-model outperforms the F-score of the WDC-Model by around 13 points. However, in addition to the fact that training and test datasets for the ANERcorp-Model are drawn from the same domain, 69% of NEs in the test data were seen in the training set (Darwish, 2013).

Thirdly, the ANERcorp-Model and the WDC-Model were tested on NEWS corpus, which is also a news-wire based dataset. The results from Table 6 reveal the quality of the *WDC* dataset on the NEWS corpus. The WDC-Model achieves relatively similar results to the ANERcorp-Model, although the latter has the advantage of being trained on a manually annotated corpus extracted from the similar domain of the NEWS test set.

Finally, testing the ANERcorp-Model and the WDC-Model on data extracted from a social networks like Twitter proves that models trained on open-domain datasets like Wikipedia perform better on social network text than classifiers trained on domain-specific datasets, as shown in Table 6.

In order to show the effect of combining our corpus (*WDC*) with a manually annotated dataset from a different domain, we merged *WDC* with the *ANERcorp* dataset. Table 7 shows the results of a system trained on the combined corpus when testing it on three test sets. The system trained on the combined corpus achieves results that fall between the results of the systems trained on each corpus separately when testing them on the ANERcorp test set and NEWS test set. On the other hand, the results of the system trained on the combined corpus when tested on the third test set (TWEETS) show no significant improvement.

Test set	NE-types	ANERcorp Classifier	WDC Classifier
Wikipedia set	PER	41.57	86.40
	LOC	43.06	79.36
	ORG	20.58	86.46
	Overall	35.40	**84.09**
ANERcorp set	PER	77.87	57.69
	LOC	86.99	70.95
	ORG	69.17	64.45
	Overall	**78.15**	64.92
NEWS set	PER	57.80	56.26
	LOC	65.17	60.78
	ORG	35.23	31.01
	Overall	**53.74**	50.12
TWEETS set	PER	34.57	41.43
	LOC	40.47	39.67
	ORG	15.10	24.36
	Overall	30.99	**35.78**

Table 6: The F-scores of ANERcorp-Model and WDC-Model on ANERcorp, NEWS, & TWEETS datasets.

Test Set	ANERcorp + WDC		
	P	R	F
ANERcorp	86.06	62.33	72.30
NEWS	79.67	39.01	52.37
TWEETS	58.33	26.00	35.97

Table 7: The results of combining *WDC* with *ANERcorp* dataset.

9 Conclusion and Future Work

We have presented a methodology that requires minimal time and human intervention to generate an NE-annotated corpus from Wikipedia. The evaluation results showed the high quality of the developed corpus *WDC*, which contains around 6 million tokens representing different genres, as Wikipedia is considered an open domain. Furthermore, WDC outperforms other NE corpora generated automatically from Arabic Wikipedia by 8 to 12 points in terms of F-measure. Our methodology can easily be adapted to extend to new classes. Therefore, in future we intend to experiment with finer-grained NE hierarchies. In addition, we plan to carry out some domain adaptation experiments to handle the difference in annotation convention for different domains.

References

Ahmed Abdul-Hamid and Kareem Darwish. 2010. Simplified feature set for Arabic named entity recog-

nition. In *Proceedings of the 2010 Named Entities Workshop*, pages 110–115. Association for Computational Linguistics.

Fahd Alotaibi and Mark Lee. 2012. Mapping Arabic Wikipedia into the Named Entities Taxonomy. In *COLING (Posters)*, pages 43–52.

Fahd Alotaibi and Mark Lee. 2013. Automatically Developing a Fine-grained Arabic Named Entity Corpus and Gazetteer by utilizing Wikipedia. In *IJCNLP*.

Maha Althobaiti, Udo Kruschwitz, and Massimo Poesio. 2013. A Semi-supervised Learning Approach to Arabic Named Entity Recognition. In *RANLP*, pages 32–40.

Joohui An, Seungwoo Lee, and Gary Geunbae Lee. 2003. Automatic acquisition of named entity tagged corpus from world wide web. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 2*, pages 165–168. Association for Computational Linguistics.

Yassine Benajiba, Paolo Rosso, and José Miguel Benedíruiz. 2007. Anersys: An Arabic Named Entity Recognition System based on Maximum Entropy. In *Computational Linguistics and Intelligent Text Processing*, pages 143–153. Springer.

Yassine Benajiba, Mona Diab, and Paolo Rosso. 2008. Arabic Named Entity Recognition using optimized feature sets. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 284–293. Association for Computational Linguistics.

Daniel M Bikel, Scott Miller, Richard Schwartz, and Ralph Weischedel. 1997. Nymble: a high-performance learning name-finder. In *Proceedings of the fifth conference on Applied natural language processing*, pages 194–201. Association for Computational Linguistics.

Nancy Chinchor, Erica Brown, Lisa Ferro, and Patty Robinson. 1999. 1999 Named Entity Recognition Task Definition. *MITRE and SAIC*.

Wisam Dakka and Silviu Cucerzan. 2008. Augmenting Wikipedia with Named Entity Tags. In *IJCNLP*, pages 545–552.

Kareem Darwish. 2013. Named Entity Recognition using Cross-lingual Resources: Arabic as an Example. In *ACL*.

Junichi Kazama and Kentaro Torisawa. 2007. Exploiting Wikipedia as external knowledge for named entity recognition. In *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 698–707.

L.S. Larkey, L. Ballesteros, and M.E. Connell. 2002. Improving stemming for Arabic information retrieval: light stemming and co-occurrence analysis. In *Annual ACM Conference on Research and Development in Information Retrieval: Proceedings of the 25 th annual international ACM SIGIR conference on Research and development in information retrieval*, volume 11, pages 275–282.

Andrew McCallum and Wei Li. 2003. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 188–191. Association for Computational Linguistics.

Peter Mika, Massimiliano Ciaramita, Hugo Zaragoza, and Jordi Atserias. 2008. Learning to Tag and Tagging to Learn: A Case Study on Wikipedia. volume 23, pages 26–33.

Joel Nothman, James R Curran, and Tara Murphy. 2008. Transforming Wikipedia into Named Entity training data. In *Proceedings of the Australian Language Technology Workshop*, pages 124–132.

Joel Nothman, Nicky Ringland, Will Radford, Tara Murphy, and James R Curran. 2013. Learning multilingual Named Entity Recognition from Wikipedia. *Artificial Intelligence*, 194:151–175.

Alexander E Richman and Patrick Schone. 2008. Mining Wiki Resources for Multilingual Named Entity Recognition. In *ACL*, pages 1–9.

Nicky Ringland, Joel Nothman, Tara Murphy, and James R Curran. 2009. Classifying articles in English and German Wikipedia. In *Australasian Language Technology Association Workshop 2009*, page 20.

Satoshi Sekine et al. 1998. NYU: Description of the Japanese NE system used for MET-2. In *Proc. of the Seventh Message Understanding Conference (MUC-7)*, volume 17.

Sam Tardif, James R. Curran, and Tara Murphy. 2009. Improved Text Categorisation for Wikipedia Named Entities. In *Proceedings of the Australasian Language Technology Association Workshop*, pages 104–108.

Torsten Zesch, Iryna Gurevych, and Max Mühlhäuser. 2007. Analyzing and accessing Wikipedia as a lexical semantic resource. pages 197–205. Tuebingen, Germany: Gunter Narr, Tübingen.