# A New Approach for Arabic Named Entity Recognition

Wahiba Karaa and Thabet Slimani
College of Computers and Information Technology, Taif University, KSA

**Abstract**: *A Named Entity Recognition (NER) plays a noteworthy role in Natural Language Processing (NLP) research, since it makes available the detection of proper nouns in unstructured texts. NER makes easier searching, retrieving, and extracting information seeing as the significant information in texts is usually sited around proper names. This paper suggests an efficient approach that can identify Named Entities (NE) in Arabic texts without the need for morphological or syntactic analysis or gazetteers. The goal of our approach is to provide a general framework for Arabic NE recognition. Within this framework; the system learns the recognition of NE automatically and induces NE systematically, starting from sample NE instances as seeds. This method takes advantage from the web, the approach learns from a web corpus. The seeds are used to identify the contexts in the web denoting NE and then the contexts identify new NE. Thorough experimental evaluation of our approach, the performances measured by recall, precision and f-measure conducted to recognize NE are promising. We obtained an overall rate of F-measure equal to 83%.*

**Keywords**: *Arabic NE, machine learning, web document, information retrieval, information extraction.*

*Received October 4, 2014; accepted March 15, 2015*

## 1. Introduction

The Named Entity (NE) recognition is an indispensable sub-task in various systems dealing with automatic Natural Language Processing (NLP), such as information extraction [12], automatic speech recognition, document indexing, document annotation [14], document classification [15], translation, question-answering [10], etc.

The Named Entity Recognition (NER) concept was presented in 1990 at the Message Understanding Conferences (MUC), which were financed by the Defense Advanced Research Projects Agency (DARPA) to promote information extraction new methods. Quality of the NER system impacts directly on the quality of the whole NLP system.

Currently, for only some heavily used languages, a large variety of NER tools exist [17]. Arabic NER stays a little researched area compared to other languages such as English. Arabic NER has not yet been investigated significantly [20].

Arabic is a Semitic language that presents complex morphological and orthographic behaviours that may complicate NER [25]. Some complexities of this language include:

1. Lack Capitalization of Proper Nouns: this is a major obstacle since capital letters are very significant in NE recognition in Latin languages;
2. Absence of Vowels: a non-vowelized word has many ambiguities in meaning or syntactic function. For example, the word كتب/kataba/ can be a verb (write) /kataba/ or a name (book) /kutubon/ depending on the vowelization.
3. Complex Morphology: arabic is a highly inflected language. Coordinating conjunctions, prepositions, possessive pronouns, and determiners are usually linked to words as prefixes or suffixes. All words in Arabic are derived from roots using patterns or templates.

In this paper we suggest a system that uses the advantages of learning technologies, combined with statistical models to extract contexts from Web, and identify the most significant contexts for the recognition of NE by using different measures for ranking contexts [5].

The remaining of the paper is organized as follows: in section 2 introduces related work. Section 3 describes the methodology, and section 4 gives results of our approach. Section 5 is devoted to the work’s conclusion and future works.

## 2. Related Works

Several works are particularly interested in the recognition of named entities in Arabic. For instance, John Maloney and Niv [16] described TAGARAB system for recognizing named entities related to dates, times, and numeric in Arabic-language text. This system is based on a combination of morphological analysis and a pattern matching engine. However, Abuleil [2] presented a technique to extract names from texts in question-answering systems based on graphs representing the words in phrases including names and the relationships (weight) between them. The names then are generated according to some given rules. Samy *et al.* [19] proposed an Arabic NE tagger using an Arabic-Spanish parallel corpus. The idea is that given two sentences where each one is the translation of the other; and given that in one sentence the NE were tagged, then the translated sentence should contain the same NE.

Benajiba *et al.* [9], proposed a NE recognition system where, based on maximum entropy approach, they model the problem as a two steps classification approach, separating the NE boundary detection from the NE classification. Then they pursued with additional work in Benajiba *et al.* [6] where they described an approach based on Support Vector Machines (SVM) and Conditional Random Fields (CRF). They investigated lexical, contextual and morphological features on eight standardized data-sets. Later, Benajiba *et al.* [7] described a novel NER system using SVMs and a combination of both language independent and language dependent features for Arabic NER. The result showed a significantly performance. And then in 2010 they improved accurate Arabic NER system using noisy features and an Arabic-English parallel corpus [8]. Yet, Shaalan and Raza [22] presented their system NERA to recognize ten types of NE. To develop NERA, they used rule-based approach using linguistic grammar-based techniques and a list of dictionaries. Furthermore, Zribi *et al.* [25] proposed a hybrid approach for Arabic named entities recognition based on learning method. A set of rules are extracted manually to correct and improve the result of the learning system. Later, Al-Jumaily *et al.* [3] proposed an approach trying to minimize the impact of Arabic prefixes and suffixes on the quality of the pattern recognition model applied to identify named entities. These patterns are built up by processing and using different existing gazetteers.

Zaghouani [24] presented a rule-based Arabic NER system (RENAR) to recognize person, location and organization named entities. The system is based on three steps: Morphological preprocessing step; searching known NEs; and recognizing unknown named entities using local grammar.

Aboaoga and Aziz [1] introduced a rule-based Arabic NER approach. This approach is composed by three steps: Preprocessing step; automatic NE tagging step using lists of person names and keywords; and extraction of person names applying four main rules.

Recently, in their paper, Shaalan and Oudah [21] propose a hybrid NE recognition approach for Arabic that takes the advantages of rule-based and machine learning-based approaches. The finding is an improvement of the system performance and surmounts many problems such as restricted resources for languages that require deep language processing.

Althobaiti *et al.* [4] in their paper propose a new approach using Wikipedia to automatically build up an Arabic NE annotated corpus. Each Wikipedia link is transformed into an NE type to generate the NE annotation. Other Wikipedia features (redirects, anchor texts, etc.,) are used to tag additional NEs, which appear without links in Wikipedia texts.

In this paper we are interested in recognizing Arabic named entities without using lexical, syntactic or morphological features, without using dictionaries or grammatical rules.

In this paper we consider that the NE which can be: Location, person, company, date, time, etc., is characterized by a given context. The context concerns a set of words preceding or following the NE: The word "شركة" /charika/ (company) in a text, for example, indicates that this word (or context) may be followed by the name of a company. Likewise, a word preceded by the context "ممثل"/mumethil/ (actor), involves that this word is the name of an actor. The NE can be then recognized regarding the context. The context identifying a NE is the one that is most frequent. This frequency is calculated based on the documents in the largest corpus that is the web.

The World Wide Web provides enormous quantities of texts, for a multiplicity of languages on a vast array of topics. Thesauruses and lexicons could be developed directly from the Web. Large numbers of linguistic processes that have been applied to small corpora could be applied to the Web. Further anatomizing of Web text types and domains, thesauruses and lexicons could be developed directly from the Web and all for a multiplicity of languages [13].

## 3. The Proposed Approach

Our approach performs as follows: We introduce a small set of seeds. Then we find all occurrences of those seeds on the Web. From these occurrences we identify contexts (words surrounding the seeds). We call context of NE, the set of n words that follow and precede the NE.

We develop strategies for estimating the consistency of the extracted contexts. After the contexts 'selection, we search the Web for these contexts and find new NE. Iteratively, we can then take these new NE and discover all their occurrences and from these latter we generate more contexts. In each iteration we will get an extended list of NE and significant contexts for discovering them. We describe the approach more formally in the following steps:

We consider Cw a large document corpus namely the World Wide Web.

- *Step 1.* Seeds←Examples. NE examples are provided, by the user. These examples can be limited. In our study, we used generally a list of five examples.

We give a set of seeds, which are instances of named entities of a given category. For example, for the NE "صحيفة" (newspaper) we must provide as instances, newspaper' names such as "الاهرام"/ahram/, "الصباح"/assabah/, "الصريح"/assarih/, etc. For the NE

“ملك” (king) must be provided as instances, king’ names such as “الحسن الثاني” (Hassen II), “عيسى بنحمد” (Issa Benahmed), “عبد الله الثاني” (Abdullah II), “ عبد الله بن عبد العزيز” (Abdullah Bin Abdellaaziz).

- *Step 2*. List-URL← Find-Seed (Seeds, Cw). Find all occurrences of the seeds in the Web Corpus (Cw). The function returns a collection of Uniform Resource Locators (list-URL) related to every seed occurrence in the Cw. For all the seeds examples, a query is sent to a Web search engine (Google, yahoo, etc.,) by using a particular API related to each engine, searching for seeds. This function, also, checks and rejects the same URLs.
- *Step 3*. Doc← Find-Doc(List-URL). For each URL, a new query is send in order to load and save all Documents (Doc) from the List of URLs (List-URL) containing a seed occurrence. The documents (Doc) will be used as a training corpus in the next step.
- *Step 4*. Cont←Find-Context (Seeds, Doc). Find all the contexts which are words surrounding the seed examples (words before, or/and after the example). The length of the context is an input parameter. In our study the context can be one word or two words on the left and on the right of a seed.
- *Step 5*. RankC←Rank-Context (Cont, Doc, Seeds, Cw). The contexts need to be relevant. For this goal, we compute a ranking function to guaranty the quality of the contexts. The relevance of a context is dependent on its occurrences in the documents ‘corpus. Usually relevant contexts are those having higher frequencies. However, some contexts can be irrelevant even if they have high frequencies in the documents ‘corpus. We propose a new method for calculating the relevance of a context. We assume that the pertinence of a context is not only based on its frequency, but also by the seed frequency and the document frequency. In this step, we compute a weight for each context according to a function calculated by a combination of the context frequency, the inverse context frequency, the seed example frequency, the seed example inverse frequency, the document frequency, and the inverse document frequency. To compute the context weight, we use context frequency representations based on Term Frequency-Inverse Document Frequency (tf-idf) version. *tf-idf* is one of the most classics and most common weighting methods used to describe documents in the vector space model [18]. The *tf* considers the term frequency in the document: the more a word occurs in a document, the more it is expected to be significant in this document. In addition, *idf* inverse document frequency measures the term frequency in the corpus: The more a word appears in a corpus, the more it is estimated irrelevant for the document. The term frequency (*tf*) of a term *ti* in a document *dj* is calculated as follows [18]:

$$tfij = \frac{frequencyi\,j}{\sum kfrenquencykj} \qquad (1)$$

Where *frequencyij* is the occurrence number of the term *ti* in the document dj. The denominator is the total occurrences of all terms in the document dj.

The *idf*, Inverse Document Frequency component is computed as follows [18]:

$$idfi = \log \frac{N}{ni} \qquad (2)$$

Where *N* is the total number of documents in the corpus, and *ni* is the number of documents in which the term *ti* emerges.

*tf-idf* weighting is computed as follows [18]:

$$wij = tfi * dfi \qquad (3)$$

A variety of versions have been suggested to the basic *tf-idf* formula, where the *tf* or *idf* part is modified using functions related to selection characteristics. In this paper, *tf-idf* is the staple benchmark, that we use to propose an adaptive *tf-idf* to calculate other frequencies: The Context Frequency (*cfi*), the Inverse Context Frequency (*icfi*), the Seed Example Frequency (*sefi*), the Inverse Seed Example Frequency (*itefi*), the Document Frequency (*dfi)* , and the Inverse Document Frequency (*idfi*). The frequencies are computed using the web as training corpus, based on *tf/idf* [18] frequencies. These frequencies are described in the following definitions:

- *Definition 1*: Context frequency. The *cfi* of a context *ci* in a document corpus is calculated as follows:

$$cfi = \frac{nci}{NC} \qquad (4)$$

We consider the variable *nci*, the occurrence number of a context *ci*, accompanied by a seed example within the training corpus (Doc). *NC* is the occurrence number of all contexts seen with some seeds in the training corpus (Doc).

- *Definition 2*: Inverse context frequency. We considered the hypothesis that a context is significant for a NE, if it does not often appears with other phrases in the corpus. *icfe*, measures the *cfi* associated to all the phrases escorted by the context: the more a context appears in a corpus accompanied with other phrases, the more it is estimated irrelevant for the NE.

*icfi* is calculated using *nci*, the occurrence number of a context within a corpus accompanied by one training example, and the variable $\overline{NC}$ represents the occurrence number of the contexts accompanied with other phrases in the corpus.

$$icfi = \frac{nci}{\overline{NC}} \qquad (5)$$

- *Definition 3*: Document frequency. We used the variable *ndi* to assign the occurrence number of documents in the corpus containing a context *ci*. *ND* is the occurrence number of documents in the training corpus.

The document frequency *dfi* is calculated as follows:

$$dfi = \frac{ndi}{ND} \tag{6}$$

- *Definition 4*: Inverse document frequency. The inverse document frequency i*dfi* is calculated as follows:

$$idfi = \frac{ndi}{\overline{ND}} \tag{7}$$

We used the variable *ndi* to designate the occurrence number of documents in the corpus containing a context *ci*. $\overline{ND}$ is the occurrence number of documents not containing the context *ci* in the training corpus.

- *Definition 5*: Seed example frequency. This frequency is related to one seed example frequency regarding a context:

$$sefi = \frac{nsi}{NCs} \tag{8}$$

We designated *nsi*, the occurrence number of a seed located with a context *ci*, in the corpus, and *NCs* is the occurrence number of all contexts identified in corpus with the seed example.

- *Definition 6*: Inverse seed example frequency:

$$itefi = \frac{nsi}{NCs} \tag{9}$$

We designated *nsi*, the occurrence number of a seed example found with a context *ci*, in the corpus, and $\overline{NCs}$ represents the occurrence number of the contexts accompanied with other phrases than the seed example in the training corpus.

- *Definition 7*: Context weight. The product of the obtained frequencies provides the weight *wi* needed for the detection of the context pertinence:

$$wi = cfi * icfi * dfi * idfi * \sum sefi * \sum itefi \tag{10}$$

In our paper, we introduce a strategy for evaluating the quality of the contexts that are generated in every iteration. Only those contexts that are regarded as being sufficiently pertinent will be kept by the system for the following iterations of the system. We determine a *wi* threshold as the minimum confidence that a context must have to be included in the context set to start the next iteration.

This strategy for generation and filtering of contexts enhances the quality of the extracted patterns significantly in the following step.

- *Step 6*. NE-pattren←Genrate-Pattern(Cont, Seeds). As we observed the step 5 is a crucial step in the context extraction and selection. This current step is the generation of patterns, based on the retained contexts, that can be used to retrieve NE in Arabic documents. For example the following NE pattern can be used to extract NE related to location:

العاصمة<LOCATION>

Where العاصمة is translated: Capital and <LOCATION> is the category of the NE. The generic forms of a pattern can be:

context <NE category>
*<NE category>* context
context1 *<NE category>* context2

- *Step 7*. Seeds← Find-Seed (Cont, Cw). Search the web for phrases matching any of the retained contexts. This function finds segments of text in the document corpus where the context occurs. This segment is considered as a new seed. The set of seeds is updated and then the system has larger seed examples.

- *Step 8*. Return to step 2. The system repeats the different steps iteratively. In each iteration the system learns, generates news contexts, new seeds, and therefore new NE patterns.

## 4. Experimental Results

In the NE context detection may be viewed as a classification problem: The contexts are divided into two categories: Positive (retained) and negative (pruned). Since decision tree algorithms are widely used for data mining [11], classification, etc., we used the algorithm C4.5 [23], which learns a decision tree that combines the different parameters optimally.

In the data provided to the learning algorithm, each context entity is represented by a set of parameters including of course the different *tf-idf* frequencies, but also some features such as the length (number of characters and words) of contexts.

For the system implementation, the Java programming language is used, given its portability and reusability, with mainly Java Development Kit (JDK). JDK has as its primary components a collection of programming tools (javac, javadoc, etc.,) that can work in the form of command lines. The development environment is java eclipse platform.

In order to evaluate the effectiveness of our system, four test runs have been evaluated, with 5 instances as seeds each time in order to recognize the names of organization, capitals, presidents, and sportsmen.

For example for the NE “capital”, we furnished the five following seeds examples: “باريس” (Paris), “تونس” (Tunis), “القاهرة”(Cairo), “الدوحة” (Doha), and “الرياض” (Riyadh). We obtained more than 2 600 000 URLs obtained by queries on the search engine Google.

Therefore, we extracted a corpus of documents from 700 retained URLs. We have noticed that the context number becomes stable from a significant size corpus (700 documents).

After 15 iterations, we get: 19250 contexts composed of two different words on the right, 19782 contexts composed of one word on the (extract table 1), 19788 contexts composed of one word on the left, and 20235 contexts composed of two words on the left (extract table 2). The seed examples are occurred 4264 times in the corpus with the contexts. Only 200 patterns are retained.

Table 1. The right context examples (an extract).

| One word right | Patterns | Two Words Right | Patterns |
|---|---|---|---|
| (Capital)<br>العاصمة | العاصمة<br><CAPITAL> | (Embassy in)<br>السفارة في | السفارة في<br><CAPITAL> |
| (Airport)<br>مطار | مطار<br><CAPITAL> | (Hotels in)<br>فنادق في | فنادق في<br><CAPITAL> |
| (Hotels)<br>فنادق | فنادق<br><CAPITAL> | (Hotel in)<br>فندق في | فندق في<br><CAPITAL> |
| (City)<br>مدينة | مدينة<br><CAPITAL> | (Tourism in)<br>السياحة في | السياحة في<br><CAPITAL> |
| (Markets)<br>أسواق | أسواق<br><CAPITAL> | (A trip to)<br>رحلة إلى | رحلة إلى<br><CAPITAL> |
| (Province)<br>محافظة | محافظة<br><CAPITAL> | (City Market)<br>سوق مدينة | سوق مدينة<br><CAPITAL> |

Table 2. The left context examples (an extract).

| One word Left | Patterns | Two Words Left | Patterns |
|---|---|---|---|
| (The capital)<br>العاصمة | <CAPITAL><br>العاصمة | (The beautiful capital)<br>العاصمة الجميلة | <CAPITAL><br>العاصمة الجميلة |
| (Capital)<br>عاصمة | <CAPITAL><br>عاصمة | (Beautiful capital)<br>عاصمة جميلة | <CAPITAL><br>عاصمة جميلة |
| (The city)<br>المدينة | <CAPITAL><br>المدينة | (The beautiful city)<br>المدينة الجميلة | <CAPITAL><br>المدينة الجميلة |
| (City)<br>مدينة | <CAPITAL><br>مدينة | (Beautiful city)<br>مدينة جميلة | <CAPITAL><br>مدينة جميلة |

For the evaluation, we used ANERcorp[1], widely used in the literature for comparing with existing systems. It is dataset built and tagged especially for the NER task by Benajiba *et al.* [9]. It contains around 150k tokens and is available for free.

Recall, precision, and F-measure are usually used to measure the system performance in this field. We compute *Recall*, *Precision* and *F-measure* as follows:

$$Recall = \frac{number\ of\ \ NE\ recognized\ by\ the\ system}{number\ of\ \ correct\ NE\ in\ the\ corpus} \tag{11}$$

$$Precision = \frac{number\ of\ \ correct\ NE\ recognized\ by\ the\ system}{number\ of\ \ NE\ given\ by\ the\ system} \tag{12}$$

$$F-measure = \frac{2*(recall * precision)}{(recall + precision)} \tag{13}$$

The performances measured by *Recall* and *Precision* and *F-measure* conducted to recognize NE are promising as shown in Table 3:

Table 3. Performance measurement.

| | Precision | Recall | F-measure |
|---|---|---|---|
| **President** | 92,32% | 85,21% | 89% |
| **Sportman** | 90,23% | 78,11% | 84% |
| **Capital** | 93,14% | 87,41% | 90% |
| **Organization** | 80,32% | 60,21% | 69% |
| **Global Performance** | 89,00% | 77,74% | 83% |

## 5. Conclusions

The Web provides a large amount of documents, usually in unstructured forms. Using these documents to extract relations and integrate them into a structured form would create a surprising source of information. We can learn from the Web.

In this paper, we used a learning approach for NE recognition. The goal is to exploit documents obtainable from the web and investigate contexts that appear frequently neighboring named entities. First the user gives to the system a set of seed examples, and then contexts that occur with the training examples were extracted from the Web. Different feature weighting measures were computed in order to conclude contexts' selection. The retained contexts are employed to generate NE patterns. In an iterative manner, the contexts are used to check the web searching for fragments of text that exist with the contexts and thus discovering new NE that are considered as new seed examples and exploited to identify new potential contexts.

Our approach proved to be a suitable solution for the NER in Arabic without using any Preprocessing task, without morphological or syntactic analysis. It can be also considered as a refinement of NE: Person, location, organization, etc. This refinement can define for one category of NE for example: "person", subcategories of NE: "president", "sportsman", "actor", etc. This approach provides adaptability features and it can be easily customized to work with different languages.

However, in many cases, the same context can introduce different NE. For example, the context "السيد" (Mr.) can precede both a sportsman name and a president name: "السيد شوماخر" (Mr. Schumacher) and " السيد جاك شيراك" (Mr. Jacques Chirac). An interesting future work that we believe is to detect and to measure similarity between generated patterns, in order to enhance the performance of our NER system.

## References

[1] Aboaoga M. and Ab-Aziz M., "Arabic Person Names Recognition by Using a Rule Based Approach," *Journal of Computer Science*, vol. 9, no 7, pp. 922-927, 2013.

[2] Abuleil S., "Extracting Names From Arabic Text for Question-Answering Systems," *in Proceeding of the 7th International Conference on Coupling Approaches, Coupling Media, and Coupling Languages for Information Retrieval*, Vaucluse, pp. 638-647, 2004.

[3] Al-Jumaily H., Martínez P., Martínez-Fernández J., and Goot E., "A Real Time Named Entity Recognition System for Arabic Text Mining," *Journal of Language Resources and Evaluation*, vol. 46, no. 4, pp. 543-563, 2012.

[1] http://www1.ccls.columbia.edu/~ybenajiba/downloads.html

[4] Althobaiti M., Kruschwitz, U., and Poesio M., "Automatic Creation of Arabic Named Entity Annotated Corpus Using Wikipedia," *in Proceeding of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, Sweden, pp. 106-115, 2014.

[5] Karaa W. "Named entity Recognition using Web Document Corpus," *International Journal of Managing Information Technology*, vol. 3, no. 1, pp. 46-56, 2011.

[6] Benajiba Y., Diab M., and Rosso P., "Arabic Named Entity Recognition Using Optimized Feature Sets," *in Proceeding of the Conference on Empirical Methods in Natural Language Processing*, Stroudsburg, pp. 284-293, 2008.

[7] Benajiba Y., Diab M., and Rosso P., "Using Language Independent and Language Specific Features to Enhance Arabic Named Entity Recognition," *The International Arab Journal of Information Technology*, vol. 6, no. 5, pp. 464-472, 2009.

[8] Benajiba Y., Zitouni I., Diab M., and Rosso P., "Arabic Named Entity Recognition: Using Features Extracted from Noisy Data," *in Proceeding of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL, pp. 281-285, 2010.

[9] Benajiba Y., Rosso P., and Ruiz J., "ANERsys: An Arabic Named Entity Recognition system based on Maximum Entropy," *in Proceeding of 8th International Conference on Computational Linguistics and Intelligent Text Processing*, Mexico, pp. 143-153, 2007.

[10] Daille B. and Morin E., "Reconnaissance Automatique Des Noms Propres De La langue Écrite: Les Récentes Realizations," *Traitement Automatique des Langues*, vol. 41, no. 3, pp. 601-621, 2000.

[11] Denis F., Gilleron R., and Letouzey F., "Learning from Positive and Unlabeled Examples," *Journal of Theoretical Computer Science*, vol. 348, no. 1, pp. 70-83, 2005.

[12] Fourour N. and Morin E., "Apport du Web Dans la Reconnaissance des Entités Nommées," *Revue Québécoise De Linguistique*, vol. 32, no. 1, pp. 41-60, 2003.

[13] Kilgarriff A. and Grefenstette G., "Introduction to the Special Issue on the Web as Corpus," *Journal of Computational Linguistics*, vol. 29, no. 3, pp. 333-347, 2003.

[14] Kiryakov A., Popov B., Terziev I., Manov D., and Ognyanoff D., "Semantic Annotation, Indexing, and Retrieval," *The Journal of Web Semantics, Elsevier*, vol. 2, no. 1, pp. 49-79, 2004.

[15] Kumaran G. and Allan J., "Text Classification and Named Entities for New Event Detection," *in Proceeding of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New York, pp. 297-304, 2004.

[16] Maloney J. and Niv M., "TAGARAB: A Fast, Accurate Arabic Name Recognizer using High Precision Morphological Analysis," *in Proceeding of the Workshop on Computational Approaches to Semitic Languages*, Montreal, pp. 8-15, 1998.

[17] Nadeau D. and Sekine S., "A Survey of Named Entity Recognition and Classification," *Lingvisticae Investigationes*, vol. 30, no. 1, pp. 3-26, 2007.

[18] Salton G., Wong A., and Yang C., "A Vector Space Model for Information Retrieval," *Journal of the American Society for Information Science*, vol. 8, no. 11, pp. 613-620, 1975.

[19] Samy D., Moreno A., and Guirao J., "A Proposal for an Arabic Named Entity Tagger Leveraging a Parallel Corpus," *in Proceedings of the International Conference on Recent Advances in Natural Language Processing*, Borovets, pp. 459-465, 2004.

[20] Shaalan K., "A Survey of Arabic Named Entity Recognition and Classification," *Journal of Computational Linguistics*, vol. 40, no. 2, pp. 469-510, 2014.

[21] Shaalan K. and Oudah M., "A Hybrid Approach to Arabic Named Entity Recognition," *Journal of Information Science*, vol. 40, no. 1, pp. 67-87, 2014.

[22] Shaalan K. and Raza H., "NERA: Named Entity Recognition for Arabic," *The Journal of the American Society for Information Science and Technology*, vol. 60, no. 8, pp. 1652-1663, 2009.

[23] Quinlan J., *C4.5: Programs for Machine Learning*, Morgan Kaufmann Publishers, 1993.

[24] Zaghouani W., "RENAR: A Rule-Based Arabic Named Entity Recognition System," *Journal of ACM Transactions on Asian Language, Information Processing*, vol. 11, no. 1, pp. 1-13, 2012.

[25] Zribi I., Hammami S., and Belguith L., "L'apport d'une Approche Hybride Pour la Reconnaissance des Entités Nommées en Langue Arabe," *in Proceeding of the International Conference: Traitement Automatique des Langues Naturelles*, Montréal, pp. 1-6, 2010.

**Wahiba Karaa** she is currently an assistant professor in the Department of Computer Science at Taif University, Saudi Arabia. She received the Master Degree from Paris III, New Sorbonne, France, and PhD, from Paris 7 Jussieu France. Her research interest includes Natural language processing, document annotation, information retrieval, Text Mining, Data Mining, and Image Mining. She is a member of the Editorial Board of several International Journals, and Editor in Chief of the International Journal of Image Mining (inderscience publishers).

**Thabet Slimani** got a PhD in Computer Science from the University of Tunisia. He is currently an Assistant Professor in Computer Science department at Taif University of Saudia Arabia and a LARODEC Labo member (University of Tunisia). His research interests are mainly related to Semantic Web, Data Mining, Text Mining, Business Intelligence, Knowledge Management and Web services. He has published his research through international conferences and peer reviewed journals. He also serves as journals reviewer.