First Order Hidden Markov Model for Automatic Arabic Name Entity Recognition

Fadl Dahan
Computer Science Department, College of Computer and Information Sciences, King Saud University

Ameur Touir
Computer Science Department, College of Computer and Information Sciences, King Saud University

Hassan Mathkour
Dean of Computer Science Department, College of Computer and Information Sciences, King Saud University

ABSTRACT
Name Entity Recognition (NER) is an important process used for several type of applications such as Information Extraction, Information Retrieval, Question Answering, text clustering, etc. It is intended to identify and classify name entities from a given text. NER is performed by using a rule-based approach that relies on human intuitive or machine learning methods such as Hidden Markov Model (HMM), Maximum Entropy (ME), and Decision tree (DT). In this paper, we describe a model based on the first order HMM to recognize name entity in the Arabic language. The model is based on stemming process that solves Arabic's inflection problem and ambiguity. To the best of our knowledge, no work uses this approach for the Arabic language has been reported.

General Terms
Natural Language Processing (NLP), Arabic language.

Keywords
Hidden Markov Model (HMM), Name Entity Recognition (NER), Bigram Model.

1. INTRODUCTION
NER is the process of identification and classification of name entities from a given text. The term "Name Entity (NE)" is widely used in Information Extraction (IE), Question Answering (QA) and other Natural Language Processing (NLP) applications. It appeared in the Message Understanding Conferences (MUC) which influenced IE research in the U.S. in the 1990's [1]. The study of English and French newspapers proved that these entities represent 10% of the articles [2]. NER consists of two main processes:

- ✓ Identification: Locating the name entity in a text.
- ✓ Classification: Determining the semantic of the name entity.

Benajiba et. Al [3] defines the NER as having three types:

- ✓ ENAMEX (Entity Name Expression): for the proper names.
- ✓ TIMEX (Time Expression): for temporal expressions of time and dates.
- ✓ NUMEX (Numeric Expression): for numeric expression of percentage, height, monetary expression, etc.

In this paper, we concern about the ENAMEX. ENAMEX deals with the extraction of the proper names and their classification. This classification places each extracted name into one of the following categories:

- ✓ Organization (ORG): name corporate, governmental, or other organizational entity.
- ✓ Location (LOC): name of politically or geographically defined locations.
- ✓ Person (PERS): name of persons or families.
- ✓ Others (O): defining entities that are not in one of above.

NER may be processed using two different approaches. First, rule-based approach that relies on human intuitive. It is based on linguistic knowledge e.g., grammar rules. The required resources for the rule-based approach are usually gazetteers and rules [4]. The linguistic knowledge-based model achieves better results in specific domains. Gazetteers can be adapted very precisely, and it is able to detect complex entities. Rules can be tailored to meet nearly any requirement machine.

The second approach is based on machine learning methods like HMM, ME, DT, etc. This approach requires an annotated corpus. This corpus gives learn the system about the name entity and their context.

HMM is a probabilistic Markov function process. The first use of HMM was for a linguistic purpose modeling. It models the letter sequences of Russian literature. Since, Markov model were developed as a general statistical tool [5].

Like other Semitic writing systems, Arabic does not exhibit differences in orthographic case. Unlike the English-language which is mixed-case texts, in the Arabic language there is no obvious clue such as initial capitalized letters to indicate the presence of a name constituent. It is obvious that this makes the NER more complex. In this paper, we will use first order HMM approach to classify name entities for Arabic language.

The paper is organized as the follows: Section 2 describes related works. Section 3 discusses the proposed solution that deals with the NER for Arabic language. In section 4, we discuss some results that we obtained through our approach. Section 5 concludes the paper.

2. RELATED WORKS
Daniel et.al. [6] developed variant of HMM . Their system called IdentiFinder. IdentiFinder processes the English and the Spanish languages. IdentiFinder shows good results for the English and Spanish languages. For the Arabic language (to the best of our knowledge), only a few researches address this problem. Most of the proposed researches were based on rule-based method.

Maloney and Niv [7] developed a system called TAGARB for Arabic name entity recognition. It is based on combination between pattern matching engine and supporting data with a morphological analysis component. The morphological analyzer used for making the distinction between likely and unlikely name constituents. This is particularly important when deciding where a name ends and the non-name context begins. The pattern-matching engine used data consisting of a set of pattern-action rules supported by a list of words to search for the name.

Abuleil [8] presented a technique to extract names from Arabic text based on database and graphs. This technique represents the words that might form names and the relationships between them. They used directed graphs to represent these words in the name phrases, the relative frequency (weight) of each of them, and the relationship between them.

Shaalan and Raza [4] developed a system for person name entity recognition for Arabic language called (PERA). They used a rule-based approach. The system consists of lexicon in the form of gazetteer name lists, and a grammar in the form of regular expressions. Regular expressions are responsible for recognizing person name entities.

Mesfar [9] described a system for Arabic name entity recognition that combines a morphological parser and a syntactic parser. It is built with the NOOJ linguistic development environment. Morphological analyzer uses finite state technology to parse vowelled as well as partially vowelled and non-vowelled text.

Benajiba et al. [3, 10] applied an automatic approach to work with NER problem for Arabic language. The system is called ANERsys. They used two different approaches. In the first one [3], they used Bigram and Maximum Entropy (ME); this is boosted by gazetteers (List of names). They developed their own resources (training and testing Corpus ANERcorp) which are freely available on their website [11]. In the second one [10], they improved the system with by using of Part Of Speech (POS) and the two steps approach. These improvements enhanced the performance and the result of the system.

3. SYSTEM ARCHITECTURE

We propose a system that deals with the name entity recognition for the Arabic language based on the HMM approach. Figure 1 shows the system process, which involves training and a classification phases.

The input text is tokenized and normalized text. It is used in the training phase if it involves a known token or it is for classification process. A sample of training input is shown in Figure 2. In the sequel, we focus on the different part of the system process.

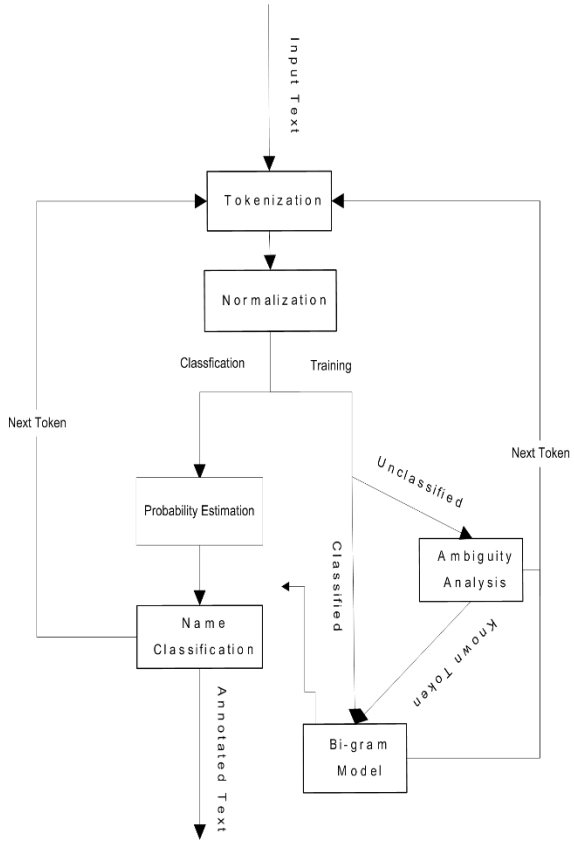

Fig 1: System process

3.1 Tokenization

It is responsible for 'cleaning' the input text from stop words and other special characters. Then, it splits into tokens. A token is defined as the couple of word and the class it belongs to.

Fig 2: Training input sample

3.2 Normalization

One of the characteristic of the Arabic language is it fills by inflections. This leads to harmonize between the names token and the joint letters (only). Table 1 shows an example of some letters that uses as word prefix. The stemming is processed in two steps:

- Removing these letters from the tokenized word.
- Adding these letters to the tokenized word.

Table 1: Word inflection letters

و	ب	ل	ت	ك
ف	وب	ول	فل	فب

3.3 Ambiguity Analysis

Ambiguity problem comes from the tokens which have more than one class. The token may be considered as having a class and may be considered as having no class (O). In the first case (a multi-class token), we consider each situation apart. In the second case, we consider every token having an O class along with its supposed additional class.

3.4 Bi-gram process

Predicting the words' class depends on the information of some previous words. For this reason, information such as the class, the features, and the number of occurrences of the word are stored. The bi-gram model uses a single previous word to predict the current words' class. The process of the training consists of storing this history and building a bi-gram. The following information is used in this phase:

W: current word.

NC: current word's class.

F: current word feature.

W_{-1}*: previous word.*

NC_{-1}*: previous word's class.*

F_{-1}*: previous word feature.*

4. EXPERIMENTAL RESULTS

In the training phase, we used a corpus composed of 200,000 words from different resources. Half of this corpus was selected from the freely available corpus [11]. We checked the corpus as it contains some words that were not annotated such as some ministries, some colleges, and stadiums. Other words were wrongly annotated such as (ألمانيا /Germany) in some position it is classified as "Other", and (الولايات المتحدة /United State) in some position it is classified as "PERS" and other position as "ORG" so we correct them before.

In a second step, we built our own corpus taken from Arabic Gigaword Third Edition. Table 2 shows the source of the corpus and the distribution of the words over the different classes along with their percentage.

We tested the corpus that we collect from different sources such as the websites of Alarabiya, Aljazeera, and Alhayat. We also collected other data from sport news, international news, Arabic news, and economic news.

Table 2: Arabic Gigaword Third Edition Corpus

Source	M/ Y	Words	PERS	LOC	ORG
France Press Agency	1/2006	12208	647 (5.3%)	502 (4.1%)	323 (2.6%)
Assabah newspaper	1/2005	41578	874 (2.1%)	800 (1.9%)	310 (0.7%)
Al Hayat newspaper	1/2006	49882	2083 (4.2%)	1527 (3%)	690 (1.4%)

The system is implemented using Java Netbeans 6.0.1 and SQL Server 2000.

We evaluated the system by measuring the precision (P) and the recall (R). The precision measures the correct responses with respect to the total responses. The recall measures the correct responses with respect to the total correct annotation. In other words, precision and recall come from comparing the system outputs with the correct annotation manually done by a linguist. Then, the numbers are used to compute P and R.

$$P = \frac{\#of\ correct\ responses}{\#of\ total\ responses}$$

$$R = \frac{\#of\ correct\ responses}{\#of\ correct\ annotations}$$

The combination of precision and recall is the F-measure which is the harmonic mean of precision and recall:

$$F - measure = \frac{2\ (PR)}{P + R}$$

Table 3 illustrates our system performance using a test corpus of 18,000 words.

In the testing phase, our systems achieved a combined precision and recall score of 77% and 73%, respectively. These results outperform those of Benajiba et al. [3, 10], as shown on table 3. In addition, the system is fully automated where the classification process dependents only on the information extracted from the training corpus without any supported name list. When compared with the results of Benajiba's systems, we found that Benjajiba's results were supported with a name list called ANERgazet. ANERgazet consists of three different gazetteers which are built manually using web resources [3, 10]:

- Location Gazetteer consists of 1,950 names;
- Person Gazetteer contains of 2,309 names;
- Organizations Gazetteer consists of a list of 262 names.

Table 3: System performance

	Precision	Recall	F-measure
Person	0.79	0.80	0.79
Organization	0.72	0.63	0.67
Location	0.82	0.75	0.78
Total	0.77	0.73	0.752

5. CONCLUSION

In this paper, we describe a model based on the first order HMM to recognize name entity in the Arabic language. The model is based on a stemming process that helps solve the Arabic's inflection and ambiguity problems. To the best of our knowledge, no work using this approach for the Arabic language has been reported.

With this in mind, we develop a system that is fully automated to recognize name entities in texts written in the Arabic language. We experimented the system by using a corpus composed of 200,000 words that were collected from different resources.

We evaluated the system by measuring precision, recall, and the harmonic mean of precision and recall. Proposed system achieved a combined precision and recall score of 77% and 73%, respectively. These results outperform existing results. The system is fully automated. The classification process is dependent only on the information extracted from the training corpus without any name list.

6. ACKNOWLEDGMENT

This work is partially supported by the research center in the college of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia. The Authors wish to thank KACST (king Abdulaziz City for Science and Technology, Riyadh, Saudi Arabia) for providing part of the corpus. We also thank Yassine Benajiba from the Universidad Politécnica de Valencia for providing valuable resources.

7. REFERENCES

[1] R. Grishman and B. Sundheim, “Message Understanding Conference - 6: A Brief History”. COLING-96.

[2] A. Borthwick, "A Maximum Entropy Approach to Named Entity Recognition". New York University, September, 1999.

[3] Y. Benajiba, P. Rosso, J. Miguel and B. Ruiz "ANERsys: An Arabic Named Entity Recognition System Based on Maximum Entropy". Proceeding of 8th International Conference on Computational Linguistics and Intelligent Text Processing, CICLing, Mexico City,Mexico, February , 2007. Vol 4394/2007,pp.143-153.

[4] K. Shaalan and H. Raza, "Person Name Entity Recognition for Arabic". Proceedings of the 5th Workshop on Important Unresolved Matters. Prague, Czech Republic, June 2007. pp. 17–24.

[5] Marie-Francine Moens, "Information Extraction: Algorithms and Prospects in a Retrieval Context", 1st edition Springer, 2006.

[6] Daniel M. Bikel, Scott Miller, Richard Schwartz, and Ralph Weischedel, "Nymble: A high performance learning name-finder". Proceedings of the 5th Conference on Applied Natural Language Processing, pages 194–201.

[7] J. Maloney and M. Niv, "TAGARAB: A Fast, Accurate Arabic Name Using High-Precision Morphological Analysis". Proceeding of the Workshop on Computational Approaches to Semitic Language, August 1998. pp. 8-15.

[8] S. Abuleil, "Extracting Names From Arabic Text For Question-Answering Systems". Proceedings of Coupling approaches, coupling media and coupling languages for information retrieval (RIAO2004), Avignon, France. pp. 638- 647.

[9] S. Mesfar, "Name Entity Recognition for Arabic Using Syntactic Grammars". Proceeding 12th International Conference on Applications of Natural Language to Information Systems, NLDB, Paris, France, June, 2007. Vol. 4592/2007, pp.305-316.

[10] Y. Benajiba and P. Rosso, "ANERsys 2.0: Conquering the NER Task for Arabic Language by combining the Maximum Entropy with POS-tag information". Proceeding of the 3rd Indian International Conference on Artificial Intelligence (IICAI-0 December 17-19 2007. pp.1814-1823.

[11] Y. Benajiba. Natural Language Engineering Lab http://www.dsic.upv.es/~ybenajiba.