Proceedings of the 5th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases, Madrid, Spain, February 15-17, 2006 (pp205-210)

Hybrid System for Extracting and Classifying Arabic Proper Names

SALEEM ABULEIL
Information Systems Department
Chicago State University
9501 S. King Drive, Chicago, IL 60628
USA

Abstract - Many applications such as information extraction systems, question answering systems, text summarization systems, information retrieval systems, etc. rely on proper names as one main tool to achieve their goals. In the Arabic language there is a big challenge for finding those proper names in the text because they do not start with capital letter as in many other languages, nor they have special sign to identify them and distinguish them from other words in the text. Little research has been conducted in this area; most efforts have been done based on a number of heuristic rules used to find names in the text, some used graphs to represent the words that might form a name and the relationships between them, some they use statistical methods for this reason. In this paper we describe a hybrid system built based on both statistical methods and predefined rules.

Key-words:- Arabic Language, Proper Names, Extraction, Classification, Rules, Statistical Methods

1. Introduction

Rau [11] argues that names not only account for a large percentage of the unknown words in a text, but are also recognized as a crucial source of information in a text for extracting contents, identifying a topic in a text, or detecting relevant documents in information retrieval systems. As defined in the Message Understanding Conference [5], names recognition consists in identifying and categorizing entity names (person, organization, location), temporal expressions (dates and times), and some types of numerical expressions (percentages, monetary values and so on), which are considered to constitute up to 10% of written texts [6]. Among the different techniques used to process these data, we find some systems based on statistics methods, such as Hidden Markov Models [4] some based on strictly linguistics methods which make use of grammar rules [8], and finally the ones that combine rules and statistics [10].

Many researchers have attacked this problem in a variety of languages but only a few limited research projects have focused on natural language processing problems for Arabic language. Wacholder et al. [12] analyzed the types of ambiguity - structural and semantic - that make the discovery of proper names in the text difficult. Kim and Evens [7] built a natural language processing system for extracting personal names and other proper nouns from the *Wall Street Journal*. Yangerber et al. [13] presented an algorithm, called NOMEN for learning generalized names in text. NOMEN uses a novel form of bootstrapping to grow sets of textual instances and of their contextual patterns. Abuleil and Evens [2] built a parser that use a set of rules to parse the Arabic text, tag the proper nouns, and extract information about them. Abuleil [1] uses the relationships between the words in the proper name phrases by building a directed graph that represents the words as nodes and the relationships between them as weights on the edges.

2. Proper Names in Arabic Language

The problem of identifying proper names is particularly difficult for Arabic, since names in the Arabic language do not start with capital letters so we can’t mark them in the text by looking at the first letter of the word. To tag proper names in Arabic text we use keywords to guide us to the place where we can find them in the text. By using keywords we mark name phrases that might contain a certain name then we process these phrases to tag names. We noticed from our analysis of the Arabic text that proper names and with respect to the way they appear next to the keyword can be classified into to different categories: people names, location names, organization names, etc.

3. Marking Proper Name Phrases in the Arabic Text

We generated a set of rules to predict where the names are located in the text. These rules are based on two things: special nouns and special verbs we will refer to the special nouns by n-keywords and to the special verbs by v-keyword in this paper. Well-known names seem to appear close to one of these noun keywords or verb keywords in Arabic text. We collected tens of keywords in a previous research project [2] and we classified them into different classes: people, locations, and organizations. Table 1 and 2 show some examples of these keywords.

Table 1 N-Keywords

Keyword	Main Type	Sub Type
مدير Manager	Person	Manager
رئيس President	Person	President
دولة Country	Location	Country
مدينة City	Location	City
صحيفة Newspaper	Organization	Newspaper
بنك Bank	Organization	Bank
ب / في in / at	Location	N/A
شمال North of	Location	N/A

Table 2 V-Keywords

Keyword	Main Type	Sub Type
تحدث Said	Person	N/A
صرح Announced	Person	N/A

Following are some rules we generated for this purpose:

Rule#1: n-keyword used to mark people name, organization name and location name while v-keyword used just to mark people name.

Rule #2: Person name comes either to the left or to the right of n-keyword. If it appears to the right it is attached direct to the n-keyword but if it appears to the left it could be separated by at most two words. We assume that the longest name is three words so we mark five words to the left of the n-keyword and three words to the right of the n-keyword to identify the proper name phrase.

←

w5 …w2 w1 [n-k/w (people)] w3 w2 w1

Rule #3: When person name attached to v-keyword it comes direct next to it and we assume that the longest name is three so we mark three words to the right and three words to the left of the v-keyword.

←

w3 w2 w1[v-k/w (people)] w3 w2 w1

Rule#4: Organization name comes direct to the left after n-keyword. We assume that the longest name is five words so we mark five words to the left of the n-keyword to identify the proper name phrase.

←

w5 w4 w3 w2 w1 [n-k/w (org)]

Rule#5: Location name comes direct to the left after n-keyword. We assume that the longest name is three words so we mark three words to the left of the n-keyword to identify the proper name phrases.

←

w3 w2 w1 [n-k/w (location)]

Rule#6: More than one keyword could be mentioned in the same proper noun phrase; like a people-keyword followed by an organization-keyword; in this case we mark three words to the right of the keyword and eight to the left of the keyword.

←

w8 …w2 w1 [n-k/w(org) n-k/w (people)] w3 w2 w1

Rule#7: Proper noun phrase terminated when it encounters a stop word: particle, verb, adverb, punctuation mark, etc. excluding the ones that used as keywords

Rule#8: n-keywords that mark people names two types: either start with the letters "ال" like "the" in English language (title keyword) or they do not (occupation keyword). If they start with "ال" most of the time the names comes after the keyword immediately but if does not most of the time an organization name appear between the keyword and the people name. Examples:المدير حسن Manager Hassan, مدير مصنع القدس حسن Manager of Al-Quds Factory Hassan.

Rule# 9: Location keyword and when it starts with the letters "ال" like "the" in English language it does not follow up with a proper name (location name) but instead it followed by an adjective or adjective derived from proper name such as الدولة الفلسطينية Palestinian State with few exceptional such as المملكة العربية السعودية Kingdom of Saudi Arabia.

Rule#10: Some keywords consist of two words. For example, the word “نائب” “Vice” is usually connected to the word "الرئيس" "President" to form the keyword "نائب الرئيس" "Vice President".

Rule#11: When organization name consists of more than one word and when the second word in the proper name starts with the letters "لل" like "for" in English language the first word is classified a primary proper name and the rest of the words in the organization name classified co-proper names. Example: مصنع القدس *للفرشات* Jerusalem Factory for *Mattresses*.

Rule#12: When a noun appears between keyword and proper name we classify it as co-keywords. Example: وزير *الزراعة* علي Minister of *Agriculture* Ali

Rule#13: Each word in person name represents an independent name while each word in other proper name types should be mentioned along with other words around it to classify the whole string as a proper name. Examples: السيد سليم ابوليل Mr. Saleem Abuleil, موسسة الارض المقدسة Holy Land Foundation. Saleem and Abuleil each one of them represents a proper name even if they do not mentioned together while the word "Holy" and the word "Land" they should appear together to classify them as one proper name for the "Foundation"

4. Tagging Proper Names Methods

After we extract proper noun phrases from the text, next step is to mark and extract proper names. Different methods implemented and used for this reason: Rule-based method, Graph-based method and statistical-based method.

Rules-based method [2] uses a bunch of heuristic rules to parse text to tag proper name. This technique has many limitations: it is hard to tell exactly where the name starts in the phrase and where it ends. We can not tell even if there is a proper name is attached to the keyword or not and if it is to the left of the keyword or to the right. No matter how many rules you add to the system you will never cover all the scenarios that you might face, since each person writes in a different way with a different style, so the same name phrase can be written in many different ways.

Graph-based method [1] uses the relationships between the words in the proper name phrases by building a directed graph that represents the words as nodes and the relationships between them as weights on the edges. The relationship (weight) between two words represents the number of times these two words appear attached to each other in the name phrases. This approach proved to give a better result than the rule-based method especially for organization and location names but after we process few hundreds of proper noun phrases the graph becomes complicated and as more proper noun phrases to process as more the graph becomes complicated and hard to maintain and manage.

5. Our Approach for Extracting Proper Names

In this paper we use a hybrid system to tag and extract proper names by combining three different techniques: rules, graphs, and statistics. We use rules to mark proper noun phrases, we use a technique similar to graph-based method to mark candidate full or partial proper names by breaking proper noun phrase into tokens each one is either an individual word or two adjacent words, and we use some rules and frequency of tokens to identify proper names. For this purpose we use two main files one to save tokens and one to save proper names as follows:

Tokens File

Token	PNP#	Status	Freq

Proper Nouns (PN) File

PN

Token: either individual word or two adjacent words mentioned in a proper name phrase "PNP"
PNP-Code: A sequence number generated and assigned to a new PNP.
Status: "Y" means proper name or part of proper name. "N" means not proper name or part of proper name.
Freq: number of times the token mentioned since the first time it is captured.

The system carries out the work in three steps: prepare proper noun phrase, update tokens file, and clean up tokens file. First, when we receive a new proper noun phrase we assign it a sequence unique code and break it down into tokens based on the keyword(s) mentioned in it as follows:

Keyword type: People
Tokens: W1, W2, W3,…,Wn
Keyword type: Organization and location
Tokens: W1,

W1 + W2, W2 + W3, …, Wn-1 + Wn

Then we check each token to see if it is previously marked as proper name or not:

```
For each token (Ti) do:
        If Ti is a PN → mark Ti "PN"
```

i: 1..n, n: number of tokens in the proper noun phrase

Second, we update tokens file by checking each new token with all tokens in the tokens file, if there is a mach we increment frequency by one if not we add it as new entry to the file. If there is match and if the result of dividing frequency of the token over number of words in the token is greater than a threshold value "n1" we change the status of the token to "Y".

```
For each Ti do
 For each entry "token" (Tj) in the tokens file do:
   If (Ti = Tj) →
        Increment Tj_Freq by one
        Mark Ti "Found"
   If (Ti = Tj)  and (Tj_frerq / | Tj |  > n1)
          and (Tj_status = "N") →
               Turn Tj_status to "Y"
   If (Ti is not marked "Found" and Ti is not
     marked "PN") →
         Create a new entry for Ti:
          (Ti, pnp-code, "N", 1)
   If (Ti is not marked "Found" and Ti is
    marked "PN") →
         Create a new entry for Ti:
          (Ti, pnp-code, "Y", 1)
```

Third, frequently we clean up the tokens file and update the proper names file by performing several tests:

1- For each proper noun phrase in the tokens file, we classify its tokens into two classes: proper names and none proper names. If frequency of none proper name is less than frequency of proper name by a threshold value "n2"we drop none proper noun token from the file:

```
For each PNPi in the tokens File
For each Token belongs to PNPi do:
    If freq (Token) / freq (Tk) <= n2 →
        Drop the entry "Token" from the tokens file
 Tk: Min [ Freq [All Tokens belong to PNPi
 with   status = "Y"] ]
```

2- If all tokens belong to one particular proper noun phrase are classified as proper names we use the rules as follows to identify final version of proper name, save it in proper names file and drop them all from tokens file:

Person name:

- If W1 is a person name (first name), W2 is a person name (last name) then Freq (W2) >= Freq (W1).
- If Wn-1 is a person name and Wn+1 is a person name then Wn is a person name.
- If a people name consist of two words the first word considered first name the second word considered last name and if people name consists of one word it is considered last name.

Location and Organization names:

- If W1 marked as proper name and W1 + W2 marked as proper name then ignore W1 and consider W1 + W2 as a proper name. Example:
  PNP: جامعة القدس المفتوحة Al-Quds Open Nniversity
  Tokens: القدس / القدس المفتوحة Alquds / Alquds Open.
- If Wn-1 + Wn marked as proper name and Wn + Wn+1 marked as proper name then ignore Wn-1 + Wn and Wn + Wn+1 and consider Wn-1 + Wn + Wn+1 as proper name. Example:
  PNP: دولة الأمارات العربية المتحدة United Arab Emirates
  Tokens: الأمارات العربية / العربية المتحدة United Arab / Arab Emirates

3-If none of the tokens that belong to the same proper noun phrase classified as proper noun after "r1" period then we drop them all from the tokens file. "r1" is a threshold value represents the difference between the code of the proper noun phrase we are checking and the code of the last proper noun phrase captured.

6. Proper Names Classification

Some names may be attached to different types of keywords and to more than one keyword in the same name phrase.
Examples:

السيد مبارك **رئيس دولة** مصر
Mr. Mubark the **President** of the **country** of Egypt

عميد كلية تكنولوجيا المعلومات حسن شاكر
Dean of IT **College** Hasan Shaker

After we find the name we classify it with respect to its major class: people, organization and location. We use the following equation to classify the names:

$$\text{pos (Name | KWi)} >= \text{R2}$$

and

$$\frac{\text{pos (Name | KWi)}}{\text{pos (Name | KWi) + neg (Name | KWi)}} >= \text{R3}$$

Where:
pos (Name | KWi): number of times the name found attached to the keyword KWi.
neg (Name | KWi): number of times the name is found attached to keywords other than KWi.

7. Experimental Results

We have tested our new system on 200 articles from the *Al-Quds* newspaper [3], published in Palestine. The system marked 3387 proper noun phrases classified into 1433 people name phrases, 1291 organization name phrases, 663 location name phrases. We tested both the Name Tagger method and the Name Classifier method. For the first method we used the following threshold values n1, n2, r1 respectively 2, 0.5, and 1000. The module identified 2084 names (258 distinguished names), missed 57 names, and extracted 31 names mistakenly out of 1303 garbage proper noun phrase (keyword with no proper name around it). We found that most of the proper name phrases marked to the right of the people keyword is garbage proper name phrases. Table 3 shows the extracted names, distinct extracted names and missing names for all proper name types. Table 4 shows the number and the percentage of names extracted and the number and the percentage of names missed by the Name Tagger Method. The reason for the missing names is that the number of times it mentioned does not qualify it as a name. When we checked the tokens file we found all the missing names there but their weight (frequency) was insufficient to qualify them as names. The system could not extract names mentioned in the document with no keywords attached to them.

In Figure 1 the proper noun phrases are grouped into ten groups, 338 proper noun phrases in each one to show the comparison between three methods for extracting the proper names in the text: the new technique “Hybrid System” we use in this paper, the system that built by Abuleil and Evens [2] based on heuristic rules, and the system that built by Abuleil [1] based on graphs to represent the relationships between words in the proper noun phrases. The figure shows the total number of extracted names by each method in each group

Table 3 Comparison between Different Types of Proper Names

PN Type	# of Names Extracted	# of Distinct Names Extracted	# of Names Missed
People	858	93	13
Location	312	48	21
Organization	914	117	23
Total	2084	258	57

Table 4 Comparisons between Captured and None Captured Proper Names

PN Type	# & % Distinct Names Captured	# & % Names Missed	Total
People	93 87.7%	13 22.3%	106
Location	48 69.5%	21 30.5%	69
Organization	117 83.6%	23 16.4%	140
Total	258 81.9%	57 18.1%	315

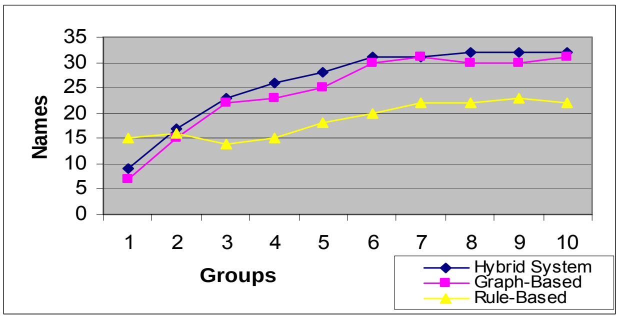

Fig.1 Comparison between Different Methods

We classified the names according to major classes (people, location and organization). We used the following values for r2 and r3 respectively 3 and, 0.7. Table 5 show the number of names classified correctly and the number of names not classified correctly. Different reasons behind misclassifying some names: a person that name appears with a phrase contains both a title and organization or location, many titles appear with the same person name and many different organizations (university, center, bank, etc.) use the same name.

Table 5 Classification with Respect to Major Class

Class	# & % names Captured	# & % Classified correctly	# & % Not Classified correctly
People	858	845 98.5%	13 1.5%
Organization	914	898 98.2%	16 1.8%
Location	312	312 100%	0 0%
Total	2084	2055 98.6%	29 1.4%

8. Conclusion

We have described a new system to extract names in the Arabic text by collecting information about the words in the text and building hybrid system uses three techniques: rules, statistics and relationship between words. We have tested our new system on 3387 proper noun phrases. We extracted 97.3% of all names and 81.9% of the distinguished names found in the text. We found all missing names in the tokens file that represent the words of the proper noun phrases so we believe that if we run more data that have these names the system will extract them.

References:

[1] Abuleil, Saleem, 2004. "Extracting Names From Arabic Text for Question-Answering Systems". RIAO'04, Proceeding of the 7th International Conference on Coupling Approaches, Coupling Media, and Coupling Languages For Information Retrieval. University of Avignon (Vaucluse), France April 26th-28th, 2004. pp 638-647.

[2] Abuleil, S. and Evens, M., 2002. Extracting an Arabic Lexicon from Arabic Newspaper Text. *Computers and the Humanities*, 36(2), pp. 191-221.

[3] Al-Quds Newspaper, 2005. Palestine.

[4] Bikel D., Miller S., Swatch R, Weischedel R. 1999, An Algorithm that learns what's in a name. Machine learning: special issue on Natural Language Learning, 34

[5] Chinchor N., 1998, Overview of MUC-7. in proceeding of the 7th message understanding conference (MUC-7).

[6] Coates-Stephens S., 1992, the analysis and acquisition of proper names for robust text understanding. Ph.D thesis, department of computer science city university London

[7] Kim, J-S., and Evens, M., 1995, "Extracting Personal Names from the Wall Street Journal", *Proceedings of the 6th Midwest Artificial Intelligence and Cognitive Science Society Conference*, Carbondale, IL, April 21-23, pp. 78-82.

[8] Magnini B, Negri M, Prevete R,Tanev H A 2002 WordNet Approach to name Entity recognition. Proceeding of the workshop semaNet'2002. Binding using semantic networks

[9] Mehdi, S. A. 1986. "Arabic Language Parser", *International Journal of Man-Machine Studies*. 2(5):593-611.

[10] Mikheev a, Grover C, Moens M. 1998 Description of the LTG system used for UMC-7 Proceeding of Message Understanding Conference (UMC-7).

[11] Rau, L. F., 1991, "Extracting Company Names from Text", *Proceedings of the Seventh Conference on Artificial Intelligence Applications*, Feb. 24-28, Miami Beach, Florida, pp.29-32.

[12] Wacholder, N., Ravin, Y., and Choi, M., 1997, "Disambiguation of Proper Names in Text", *Proceedings of the Fifth Conference on Applied Natural Language Processing*, Mar 31-Apr 3, Washington, DC, pp. 202-208.

[13] Yangerber, R., Winston, L, and Grishman, R., 2002, "Unsupervised Learning of Generalized Names", *COLING 2002*, Taipei. pp.1135-1141.