Using NLP Techniques For Tagging Events in Arabic Text

Saleem Abuleil
Department of MMIS
Chicago State University
9501 S. king Drive, Chicago, IL 60628
Email: sabuleil@csu.edu

Abstract

In any text Events represent a rich source of information, such as proper names that specify the people, organizations, locations, etc., that participate in the event, dates that specify the time of the event, noun phrases that describe the event, and keywords that indicate that an interesting event has occurred. In this paper our focus is on those elements in terms of how we use natural language processing techniques to capture, analyze and identify the role each element plays in the event and understand how they are linked to form events as well as how they form the relationships between related events.

1. Introduction

Information Extraction (IE), as defined in the Message Understanding Conferences has been traditionally defined as the extraction of information from a text in the form of text strings and processed text strings that are placed into slots labeled to indicate the kind of information that can fill them. In this paper the input to our information extraction system is a set of unclassified newswire articles, and the output is a set of filled slots which may represent an event with its attributes, i.e., proper names, dates, noun phrases, keywords, etc., and relationships between two or more related events.

Because of the importance of this topic and its influence on other fields in natural there is a serious need to keep developing it, and to improve it, to increase the correctness of the results and the efficiency of the system. This is especially challenging when it comes to the Arabic language, because it has several unique features that do not exist in other languages, and also because of the lack of research in the Arabic language. Although our research was carried out for the Arabic language, we believe that our algorithms will work for other languages too.

2. Previous Work

Some recent techniques for generating rules in the realm of text extraction are called "wrapper induction" methods. These techniques have proved to be fairly successful for IE tasks in their intended domains, which are collections of highly structured documents such as web pages generated from a template script (Muslea et al., 1999; Kushmerick, 2000). However, wrapper induction methods do not extend well to natural language documents because of the specificity of the induced rules. Boosted Wrapper Induction (BWI) is an IE technique that uses the AdaBoost algorithm to generate a general extraction procedure that combines a set of specific wrappers (Freitag et al., 2000). BWI has been shown to perform well on a variety of tasks with moderately structured and highly structured documents, but exactly how boosting contributes to this success has not been investigated. Furthermore, BWI has been proposed as an IE method for unstructured natural language documents, but little has been done to examine its performance in this challenging direction. KNOWITALL built by Etzioni et al. (2004) is an Unsupervised Information Extraction Algorithm (UIE). This means that it performs information extraction in the absence of hand-tagged training data. Because UIE systems do not require human intervention, they can recursively discover new relations, attributes, and instances in a rapid, scalable manner.

3. Our Approach

In this paper our target is made up of unstructured newsworthy classes of events, e.g., natural disasters, deaths, bombings, elections, and financial fluctuations. and the algorithm that we use to extract and understand those events is based on natural language processing techniques. We break events into their elements, analyze them, understand the syntax of each one, identify the role each plays in the event and understand how they are linked to form events as well as how they form relationships between related events.

1082-3409/07 $25.00 © 2007 IEEE
DOI 10.1109/ICTAI.2007.179

We assume that any event consists of one or more of the following elements:

- Keyword-Event: Events are marked in the text by looking for special words (keywords). We have collected dozens of keywords to mark certain events, e.g., natural disasters, deaths, bombings, elections, and financial fluctuations, including زلزال earthquakes اعصار hurricanes, مجزرة massacres, and death threats لقي...مصرعهم . We classify them into noun keywords and verb keywords and we use the morphology systems of nouns and verbs to analyze keywords and to generate their complete paradigms such as the number feature for nouns: singular, dual, and plural, and tense feature for verbs: past and present progressives.

- Proper Name: A proper name can refer to any person who participates in the event, any organizations involved in the event, the location of the event, the date of the event, and the date of the article. In the Arabic language there are some indicators we can use to guide us to the proper names associated with the event, such as keywords (special nouns, particles and verbs). We use a system built by Abuleil and Evens, (2002) to find proper names in the text and to classify them. The role each proper name type plays in events is recorded as follows: (People / Organization, Who), (Location, Where), (Time, When).

- Noun Phrases: They explain the event and play the role "what" in them. They come either after a verb Keyword of an event

  سقوط أكثر من 600 قتيل

  falling more than 600 casualty

  before a verb keyword of an event

  تسعة مدنيين قتلوا

  nine civilians have been killed

  or they come between two parts of a discontinuous keywords (the first one is a verb) for an event

  لقي حوالي مائتي شخص مصرعهم

  faced about two hundred people a *death*

  We terminate noun phrase when we see either a proper name or a particle.

- Event Names: Sometimes event are assigned a name and usually it comes right after the keyword of the event.

- Link Tools: Special Particles and Nouns for Confirming Roles: When some particles and nouns appear next to proper nouns and noun phrases in the text they confirm the role that each plays in the event. For example, when we encounter the particle "في" ("in" in English) followed by a location in a text that mentions an event, this confirm the location of the event. We have collected many such keywords to use in our system. Examples (Link Tool, Role) (في "in" + Location, Where), (في "in" + Organization, Who), (شرق "south" + Location, Where), (Event + ب "by" + NP, What).

- Relationship Tools: When special particles and nouns appear in related events and when they come right before a keyword of the second event in a context that mentions two or more related events, they play an important role in connecting events and in representing relationships between them. We have collected many of them to use in our system. We have classified the types of relationship between related events into five categories as follows:

1. Event2 *Makes/causes* Event1 to happen [Role: *How-why*]
   Example:

لقي 16 شخصا معظمهم من الأطفال مصرعهم في فيضانات نتجت عن أمطار غزيرة هطلت على جاكرتا اليوم 16 people, most of them children, were killed in a hurricane caused by heavy rain on Jakarta

2. Event1 and Event2 happened *simultaneously* [Role : *And*]
   Example:

على الاقل عشرة اشخاص قتلوا و الاف شردوا من بيوتهم نتيجة لاعصار ضرب باكستان At least 10 people have died and thousands have fled their homes as a powerful hurricane hit Pakistan

3. Event1 is a *subset / Part of* Event2 [Role: *Part-Of*]
   Example:

54 عراقيا قتلوا في أنحاء متفرقة من العراق من بينهم 35 في هجوم انتحاري في النجف جنوب العراق اليوم 54 Iraqi people were killed in different parts of Iraq among them 34 in a suicide attack in Najaf in southern Iraq

4. Event2 is result of Event1 [role: *result-of*]
   Example:

انفجار قنبله ادى الى مقتل شخصين A bomb's explosion leads to the death of two people

The different between "result-of" role and "causes" role is that the order of cause and results which one mentioned first and which one mentioned second.

By analyzing events elements, organize them and represent the role each one of them plays in the event, as well as the relationship between these elements and the relationship between related events we can obtain a good knowledge about event(s).

4. Example

In this section we will show an example obtained from *Aljazeera.net*, 2006 and explain how the algorithm works:

قتل 16 شخصا معظمهم من الأطفال في فيضانات نتجت عن أمطار غزيرة هطلت على جاكرتا
Sixteen people most of them are children have killed in a floods caused by a heavy rains that fell on Jakarta.
Step#1: tag events keywords

قتل 16 شخصا معظمهم من الأطفال في فيضانات نتجت عن أمطار غزيرة هطلت على جاكرتا
Sixteen people most of them are children have killed in a floods caused by a heavy rains that fell on Jakarta.

Step#2: identify the elements of the events where it is applicable

Keyword-First Event :**قتل killed**
sixteen people most of them children
Noun Phrase :**16 شخصا معظمهم من الأطفال**
Relationship Tool / Link Tool :**في in**
Keyword-Second Event :**فيضانات flood**
Relationship Tool :**نتجت عن caused by**
Keyword-Third Event :**أمطار غزيرة heavy rains**
Link Tool :**على on**
Proper Noun: Location:**جاكرتا Jakarta**

Step#3: apply the rules to analyze the elements of the events and extract the information about the event(s):

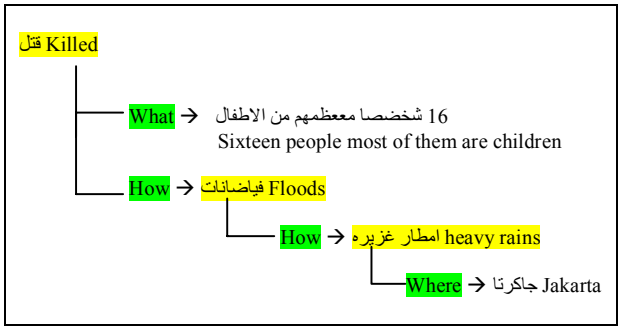
First event: قتل Killed
What "explain/Describe first event":
16 شخصا معظمهم من الأطفال
Sixteen people most of them are children

How "[the second event] caused the first event":
فيضانات Floods

Note: since "في in" followed by an event keyword and not a proper name (location) so we consider it as relationship tool and not link tool.

How "[the third event] caused the second event":
أمطار غزيرة Heavy Rains

Where "place of event": جاكرتا Jakarta



5. Preliminary Results

We have tested our system on 300 passages from *Aljazeera.net* published in Qatar, in 2006; each passage has one or more related events. We divided the articles into two sets, 150 passages in each, and we started with some seeds we collected randomly such as keywords, link tools and relationship tools. First, we ran the system on the first set of passages we recorded the results and identified the missing keywords and special particles and nouns and added them to the database. Second, we ran the system on the second set of passages. The system works very well if all the elements (keyword, noun phrases, special nouns and particles, and proper nouns) appear in the description of the event, but if one of them is missing then the algorithm fails to detect and understand the event. The algorithm also fails to detect and understand the event if it cannot recognize one of the elements in the event, even if it is present. This happens if the date appears in words and not numerals, if the keyword is not recognized (because it is not in the database), and if the proper noun is not tagged. Some times tagged noun phrases have extra words are not related to the topic and they need to be cleaned up, in different occasions the system tagged events while they are not and this due to keywords mentioned in the text and no certain events attached to them. The algorithm detected 439 out of the 467 events found in the text and it missed 28 events due to missing keywords (9 distinct keywords). Out of the 439 events found there are 381 events identified correctly and 58 incorrectly, the source of the incorrect information and improper understanding of events is due to one or more of the following: proper nouns, noun phrases, special nouns and particles used to identify relationships and roles so one event may affected by more than factor at the same time. The system also tagged 15 as events while they are not. Table 1 shows the problems in the system number of events the system could not identify, number of event tagged in correctly and number of events the system could not understand correctly due to missing

information, the table shows also the causes of each problem and how many time occurred in each set. The table shows how the results improved when we feed the system with keywords, proper names, link tools and relationship tools. The following is the recall and the precision of the algorithm from:

Set#	Recall	Precision
1	0.89	0.85
2	0.94	0.89
Average	0.92	0.87

Table 1 Source of Missing Events / Wrong Events/ Improper Understanding of Events

Problem	Causes	Set # 1	Set # 2	Total
Missing Events	Missing Keyword	17	11	28
Wrong Events	Keyword With No Event Attached	7	8	15
Improper Understanding of Events	Missing Proper Name	26	17	43
	Missing Link Tool	25	14	39
	Missing Relationship Tool	16	9	25

Table 2 shows the correct relationships between events identified by the system as well as the incorrect ones. Table 3 shows the roles of the elements in the event that identified by the system correctly and incorrectly. The problem of generating some incorrect results in both cases either because of missing keywords (particles and nouns) in the database or the keyword exist but it does not work as it planned in the algorithm.

Table 2 Relationship Types (Relationships Between Related Events)

Relationship	Correct # & %	Incorrect # & %	Total
Why-How	109 88.6%	14 11.4%	123
And	53 88.3%	7 11.7%	60
Part-of	7 77.8%	2 22.2%	9
Cause of	15 88.2%	2 11.8%	17
Average	**84.6%**	**15.4%**	

Table 3 Role Types

Role	Correct # & %	Incorrect # & %	Total
Who	25 86%	4 14%	29
Where	165 88.8%	21 11.2%	186
When	74 90.2%	8 9.8%	82
What	239 87.6%	34 12.4%	273
Average	**88.15%**	**11.85%**	

6. Conclusion

In this paper we proposed a new algorithm, to scan, extract and understand events in Arabic text based on natural language processing techniques and methods. The system evaluation used 300 passages, each one of which has one or more related events. It identified 439 events out of 467 and the recall and the precision were calculated as 0.92 and 0.87, respectively. Although our experiments were carried out on Arabic data, we believe that these techniques are language universal.

7. References

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

[2] Etzioni, O., Cafarella, M., Downey, D., Kok, S., Popescu, A., Shaked, T., Soderland, S., Weld, D., and Yates A. (2004). Web-Scale Information Extraction in System x: (Preliminary Results). *Proceedings of the 13th international conference on World Wide Web* 2004, New York, NY, USA May 17 - 20, 2004, pp. 100-110.

[3] Freitag, D., and Kushmerick, N. (2000). Boosted Wrapper Induction, *Proceedings of the Seventeenth National Conference on Artificial Intelligence,* pp. 577-583.

[4] Kushmerick, N. (2000). Wrapper Induction: Efficiency and Expressiveness, *Artificial* I*ntelligence* 2000, pp. 118.

[5] Muslea, I., Minton, S., and Knoblock, C. (1999). A Hierarchical Approach to Wrapper Induction. Proceedings of the Third International Conference on Autonomous Agents (Agents'99), Seattle, WA, pp. 190-197.

[6] www.alajazeera.net (2005-2006), Qatar.