Automatic Event and Relation Detection with Seeds of Varying Complexity

Feiyu Xu, Hans Uszkoreit and Hong Li

Language Technology Lab, DFKI GmbH
Stuhlsatzenhausweg 3
D-66123 Saarbrücken, Germany
{feiyu, uszkoreit, lihong}@dfki.de

Abstract

In this paper, we present an approach for automatically detecting events in natural language texts by learning patterns that signal the mentioning of such events. We construe the relevant event types as relations and start with a set of seeds consisting of representative event instances that happen to be known and also to be mentioned frequently in easily available training data. Methods have been developed for the automatic identification of event extents and event triggers. We have learned patterns for a particular domain, i.e., prize award events. Currently we are systematically investigating the criteria for selecting the most effective patterns for the detection of events in sentences and paragraphs. Although the systematic investigation is still under way, we can already report on first very promising results of the method for learning of patterns and for using these patterns in event detection.

Introduction

In the last two decades, information extraction research area has grown into a major subfield of natural language processing. The most relevant steps in this development (Grishman and Sundheim 1996, Appelt and Israel 1999, Appelt 2003) brought us from attempts to use the methods of full text understanding to shallow text processing (Hobbs et al. 1996), from pure knowledge-based hand-coded systems to (semi-) automatic systems using machine learning methods (e.g., Riloff 1996, Califf and Mooney 1999, Pierce and Cardie 2001, Mann and Yarowsky 2005, McDonald et al. 2005), from complex domain-dependent event extraction to standardized domain-independent elementary entity, simple semantic relation and event extraction (e.g., Fleischman et al. 2003).

The ACE program[1] is a further approach to standardization of the information extraction subtasks since MUC-6: entity recognition, relation extraction and event extraction, aiming to develop a more systematic grounded approach to semantics by focusing on domain independent elementary entities, relations, and events. In the last few years, extensive research has been dedicated to entity recognition and binary relation recognition with quite significant results (e.g., Bikel et al. 1999; Zelenko et al. 2003; etc.). However, the event extraction is still considered as one of the most challenging tasks, because an event mention can be expressed by several sentences and several linguistic expressions.

In this paper, we will describe a general and domain independent method that can identify event extent, event trigger[2] and event arguments automatically, starting with some seed events. A study is being carried out to learn more about the nature and effects of seeds, the size and locality of start patterns and the interaction among patterns, which together contribute to the extraction of an event. As part of this investigation, we tried to estimate the effectiveness of seed relations of different arity to see how much they can contribute to the learning of successful patterns. Our approach belongs to the bootstrapping methods. We choose prize-winning events as a domain for our experiments because this domain exhibits certain typical properties of application relevant event detection tasks. Events are sparsely represented in large text selections, in our case in freely available news texts. We find the typical skewed frequency distribution, i.e., some prize events such as Nobel and Pulitzer Prize awardings are covered in the text base with great redundancy, many other, less prestigious prizes are mentioned only once or twice. The most prominent prizes give us reliable databases of seeds whereas there are no databases comprising information on all prizes and their recipients.

Fully in line with this reasoning we have started with the Nobel Prize subdomain since it is a domain for which complete records can be obtained of all awarded prizes in structured formats and in addition a large number of free texts about awards and laureates can be found on the web.

[1] http://projects.ldc.upenn.edu/ace/

[2] An *Event extent* is a sentence within which a taggable Event is described. Its *trigger* is the word that most clearly expresses its occurrence. See *ACE English Annotation Guidelines for Events.* Version 5.4.3 2005.07.01. http://projects.ldc.upenn.edu/ace/docs/English-Events-Guidelines_v5.4.3.pdf.

Furthermore the data are manageable in size, authoritative and can be used for the creation of a gold-standard for seed selection and evaluation.

In the following, we will introduce the main aspects of our semantically oriented approach to modeling the prize-winning domain. Then we will explain seed selection and construction. We will provide the results of a study showing the frequency distribution of the possible seed types according to their arity and locality. We will show the relationship between the complexity of seeds and the precision of identification of event extent and trigger. Then we will describe the pattern learning component and the evaluation of their performance based on the complexity of arguments. We close off with a conclusion and indicate future research directions.

Relation and Event Representation

We propose a pragmatic approach to relation representation for the relation/event detection task. All relations can be represented via a set of binary relations. We choose a neo-davidsonian style to represent events. For instance, the event relation *nominate* is defined by ACE with five arguments:

nominate(person, agent, position, time, place)

The neo-Davidsonian style of this event relation is then extended by one event argument:

nominate(event, person, agent, position, time, place)

It can be transformed into a set of binary relations:

{nominate_person(event, person),
nominate_agent(event, agent),
nominate_position(event, position),
nominate_time(event, time),
nominate_place(event, place)}

In the award-winning domain, the event type is called "receive_award". Its arguments are

reason : achievement (accomplishment, service, skills, ...)
award : award_type (medal, prize, title, ...)
recipients : person, organization, GPE, ...
time : time interval or time point
location : place

"receive_prize" is a subtype of "receive_award". Therefore, its arguments can be more specified:
reason : achievement (accomplishment, service, skills, ...)
award: prize name
recipients: person, organization or GPE
time: time interval or time point
location: place
area: area
prize amount: currency or percentage

"receive_Nobel_Prize" is a subtype of "receive_prize". Its arguments are then
reason: achievement (scientific contribution)
award: prize name (Nobel Prize)
recipients: laureate (person or organization)
time: year
area: area (Nobel Prize discipline)
prize amount: currency or percentage

The neo-Davidsonian style is
{receive_Nobel_Prize_reason (event, achievement),
receive_Nobel_Prize_award (event, prize name),
receive_Nobel_Prize_recipients (event, person|organization),
receive_Nobel_Prize_time (event, year),
receive_Nobel_Prize_area (event, area),
receive_Nobel_Prize_amount (event, currency|percentage)}

Data Resources

For the evaluation and the seed construction, we have collected the complete Nobel Prize winner list from Nobel e-Museum[3] and store them into a relational database. The list contains the following information about the winner, such as the name, the gender, the award year, the monetary amount of the prize, the position, the affiliation, country, nationality, the prize area. The training and test data for our information extraction task contains

- texts from New York Times from June 1998 to September 2000 (part of the AQUAINT data)
- online news texts from BBC[4] (November 1997 to December 2005), CNN[5] (October 1995 to January 2006), New York Times[6] (October 1981 to January 2006)
- reports from Nobel e-Museum

We include other news texts in addition to the AQUAINT data, because there is not sufficient data in the AQUAINT corpus about the Nobel Prize. In our initial experiments, we use an ad-hoc classification method to identify the domain relevant documents, by applying an information retrieval engine to find our collection containing only documents about the Nobel Prize, using a query with "Nobel" as its major keyword. In our bootstrapping framework, the identification of relevant documents is dependent on the seeds and its size will grow with the growth of seeds. Of course, not all of these texts report on prize events. Our domain relevant data collection has finally 3027 documents with a total size of 18 MB. Then we selected the years 1998, 1999 as our training partition, taking 10.348864 MB as our training data. The rest of data we reserved for testing.

[3] http://www.nobel.se/
[4] http://news.bbc.co.uk
[5] http://search.edition.cnn.com
[6] http://www.nytimes.com

System Description

Our system learns extraction rules of relations and events from un-annotated news texts, taking some seed relations or events in the initialization. The learned extraction rules will be used to extract more relation and event instances. The whole learning and extraction process is embedded in a bootstrapping framework, containing the following steps:

1. Given a set of free text documents and a set of seeds
2. Detect seeds in the input corpus and annotate the corpus with event arguments of seeds. A document is relevant, if its text fragments contain the event arguments of a seed and the distance among the arguments does not exceed three sentences.
3. Learn event extraction rules from the text fragments containing seeds.
4. Apply event extraction rules to the relevant documents and obtain more potential seeds.
5. Go to 1. using extracted events as new seeds.

It is still open for us how to valid the quality of the extracted new seeds.

Seed Construction

Many bootstrapping-oriented unsupervised machine learning IE systems are initialized with so-called seeds. In ExDisco (Yangarber 2001), some example patterns of the management succession domain are chosen as the seeds, e.g., *subject(company) v("appoint") object(person)*, for learning more relevant patterns incrementally via bootstrapping. A disadvantage of this pattern-oriented seed approach is that it is too closely bound to the linguistic representation of the seed. It is well known that semantic relations and events could be expressed via different levels of linguistic representations that do not restrict the realizations to one or several patterns such as *subject v object* constructions. Furthermore, an event can be more complex than be expressed by one single pattern. In most cases, several relations extracted by different patterns can contribute to one event. Thus, we favor a semantics-oriented notion of seed construction, using relation and event instances as our seeds, like the DIPRE system (Brin 1998) and the Snowball system series (Agichtein and Gravano 2000). The advantages of this seed construction method are

- domain independence: it can be applied to all relation and event instances
- flexibility of the relation and event complexity: it allows n-ary relations and events
- processing independence: the seeds can lead to patterns in different processing modules, thus also supporting hybrid systems, voting approaches etc.

The seed in our approach should fulfill the functions of

- detection of relevant sentences, which describe the seed events. These relevant sentences can be used as potential event extent
- detection of relevant linguistic expressions, which can be used as event/relation triggers
- learning patterns and their interaction rules for event extraction

In our experiment, we start from the entire list of Nobel Prize winners of 1998 and 1999. In our first experiment, our Nobel-Prize winning event seed for now contains four arguments: *recipients*, *prize name*, *year* and *area*. Since the seed is a semantic relation, we can also map any slot value to a number of patterns. Thus, we have generated all variants of the potential mentions of person names or areas, in order to boost the matching coverage of our seeds with the texts. For example, for the person name, *Alan J. Heeger*, its mentions can be *Alan J. Heeger*, *Alan Heeger*, *Heeger*, and *A. J. Heeger*. We did it same with the Prize area, e.g., the mention variants of *Chemistry* can be *chemical*, sometimes, the professional description *Chemist* gives also an indication of the area. Then a seed instance looks like follows:

{receive_Nobel_Prize_award (event, Prize_Name: "Nobel"), receive_Nobel_Prize__recipients (event, Person: "Alan Heeger" | "Alan J. Heeger" | "A. J. Heeger"| "Heeger"), receive_Nobel_Prize_time (event, Year: "2000"), receive_Nobel_Prize_area (event, Area:"Chemistry" | "chemical" | "chemist")}

Seed Complexity and Event Extraction

In the DIPRE, ExDisco and Snowball systems, the seeds are about relations between two entities. However, an event has often more than two entities as its arguments. It is important for an unsupervised learning system to know how complex an event seed should be, in order to find good candidates for learning good extraction patterns and their interaction. Most linguistic patterns only extract one or two arguments of an event. Therefore, it is important for event extraction to learn the rules how relevant patterns contribute to event extraction.

Thus, we annotated our training texts with the entity mentions of the seed events automatically, using the *SProUT* (Shallow Processing with Unification and Typed feature structures) system (Drozdzynski et al. 2004). *SProUT* is a platform for the development of multilingual text processing systems. In comparison to other finite-state centered systems, it supports unification-based grammars. The transduction rules in *SProUT* do not rely on simple atomic symbols, but instead on typed feature structures (TFSs), where the left-hand side of a rule is a regular expression over TFSs representing the recognition pattern, and the right-hand side (RHS) is a TFS specifying the output structure. We have extended the existing general entity classes with the prize names and the area names. Then all sentences containing entity mentions of the seeds are extracted by our system. The extracted sentences are sorted by the number of contained event arguments: quaternary, ternary and binary complexity. A quaternary complexity sentence contains all entity mentions of one event seed. Within ternary complexity and binary complexity, we classify them into different groups according to the entity class combination, e.g., person-area-time, person-prize-area, person-area, etc. We have evaluated whether these sentences are about the Nobel-Prize-winning event. In Table 1, we show the distribution of the seed complexity in the sentences describing the events.

complexity	matched sentence	event sentence	Relevant sentences in %
4-ary	36	34	94%
3-ary	110	96	87%
2-ary	495	18	3.6%

Table 1. distribution of the seed complexity

For the entity-class combinations of 3-ary and 2-ary, we have also carried out a distribution count, presented in Table 2.

combination (3-ary, 2-ary)	matched sentence	event sentence	Relevant sentences in %
person, prize, area	103	91	82%
person, prize, time	0	0	0%
person, area, year	1	1	100%
prize, area, year	6	4	68%
person, prize	40	15	37.5%
person, area	123	0	0 %
person, year	8	3	37.5%
prize, area	286	0	0 %
prize, year	25	0	0 %
area, year	12	0	0 %

Table 2. distribution of entity combinations

Table 1 tells us that the more event arguments a sentence contains, the high is the probability that the sentence is an event extent. Table 2 shows the difference between different entity-class combinations with respect to the event identification. We can potentially regard these values as additional validation criteria of event extraction rules.

Whereas the first table helps us to pre-estimate the contribution of the arity classes for successful event extraction, the latter shows us which types of incomplete seeds might be most useful. Both distributions, especially the second one will be very much dependent on the kind of relations to extract. Such seed analyses could be used to better characterize a given relation-extraction task.

Automatic Pattern Extraction

Given the automatically annotated sentences with mentions of the named entity seed arguments, our next goal is to learn event extraction rules automatically from these data. Each event extraction rule potentially contains a list of relation extraction rules that detect the individual event arguments. In the current experimentation stage, we use MINIPAR (Lin 1998) for the sentence analysis. We have concrete plans to test additional NLP tools for the sentence analysis. An event extraction rule is a conjunction of relation extraction rules written as a list. It starts with a rule that serves as the event trigger and potentially call other extraction rules. In the following example, the event trigger rule uses the trigger verb *win*. Its subject fills the event argument *person* and a relation trigger rule *relation_trigger_prize* is applied to its object to extract the event arguments *year* and *prize_name*. The object should have the head word *Prize*. Its modifier can use the rule *relation_trigger_for* to extract the event argument *area*.

```
rule_id: 1
rule_name: event_trigger_win
rule_body: {head("win", v),
              subject([person]),
              object(relation_trigger_prize)}

rule_name: relation_trigger_prize
rule_body:{head("prize", n),
              binary_relation([year], [prize_name]),
              mod(relation_trigger_for)}

rule_name: relation_trigger_for
rule_body: {head("for", prep),
              pcom-n([area])}
```

The *binary_relation* in the above rule is extracted by our IE system *SProUT*. *SProUT* can also identify simple binary relations within a noun phrase construction.

The input of our rule learning system is a list of sentences containing seeds. We choose sentences where at least three event arguments are identified. Then we apply MINIPAR to these sentences and obtain dependency trees. The highest node in the tree that dominates all event arguments is defined as our event trigger node. Then we extract the patterns top-down recursively. In this first run, we have 96 sentences with three event arguments and 34 sentences with four event arguments. We have learned from each sentence a rule instance. We have classified rules depending on the number of extracted arguments: 17 rules for two arguments, 83 rule instances for three arguments and 28 rule instances for four arguments. Because of the failure of parsing, some rules can extract fewer arguments than the sentences contain. In Table 3, we show the distribution of three argument rules with respect to event argument combinations:

combination: 3-ary	number of rules
person, prize, area	77
person, prize, time	1
person, area, year	4
prize, area, year	1

Table 3. distribution of three argument rules

In all these rules, the head of the event triggers is distributed as follows: 61 are verbs, 64 are mentions about persons, 3 are verb *be*. Following verbs have the highest frequency: *win* (21), *award* (10), *share* (9), *receive* (3), *accept* (2) and *present* (1).

The next step to do is to develop methods, which cluster the rules with the same trigger words, because these rules can differ from the their embedded extraction rules. For example, we have different variation of *win* rules:

```
rule_id: 26
rule_name: event_trigger_win
rule_body: {head("win", v),
                    subject([person]),
                    object(relation_trigger_prize)}

rule_name: relation_trigger_prize
rule_body:{head("prize", n),
                    binary_relation([year], [prize_name]),
                    mod(relation_trigger_in)}

rule_name: relation_trigger_for
rule_body: {head("in", prep),
                    pcom-n([area])}
```

Rule 1 and Rule 26 only differ from their third relation extraction rule because of the different prepositions *in* and *for*.

Initial Evaluation

In our first evaluation, we apply the rule instances learned above to the test data of year 2002 with a size of around 0,5 MB. We choose the rules triggered by the most frequent verbs "win" and "award".

In fact, the news texts in 2002 contain also Nobel Prize winning mentions of previous years too. As discussed in (Fleischman et al. 2003), the evaluation problem of an unsupervised IE system is the lack of annotation for recall calculation. In our experiment domain, we can show easily the precision and the completeness of the extraction results. In the following table, we present the first evaluation results of the precision of our rules

extraction rule	correct events	incorrect events	precision
win[v]	32	7	82%
award[v]	36	7	84%

Table 4. initial evaluation

In our test run, we have observed the following problems:

1. Many rules can be applied to the same event extent
2. Different rules have extracted event arguments with different completeness

Therefore, it is important to find methods to select rules and solve the conflicts. In most cases, the results are compatible. For the following sentence, six rules with different arities can be triggered and deliver event mentions with different argument coverage.

Known as the "Saint of the Gutters" for her unending work and compassion for the poor, Mother Teresa was awarded the Nobel Peace Prize in 1979.:

```
rule_id = 32 arity= 3   [nobel, peace, 1979, ]
rule_id = 66 arity= 3   [nobel, peace,  , (Teresa, mother )]
rule_id = 69 arity= 3   [nobel, peace,  , (Teresa, mother )]
rule_id = 123 arity= 2 [nobel, peace,  , ]
rule_id = 124 arity= 2 [nobel, peace,  , ]
rule_id = 125 arity= 2 [nobel, peace, , ]
```

The first evaluation tells us that the complex event-based pattern ensures the expected high precision. Therefore, it is important to investigate the recall behavior of these patterns. We will also evaluate the binary patterns and investigate whether and how they can contribute to the improvement of recall, especially if two or more binary patterns fire within one paragraph since they both detect different parts of the mention.

Conclusion and Future Work

In this paper, we have addressed the problem of the complexity of event-based relations and the unsupervised pattern learning for event extraction. We started with the investigation of the seed construction, since seeds play a central role for the performance in this kind of unsupervised systems. Our experiments showed us that the more complex a seed is, the higher is the precision of finding relevant event sentences. Furthermore, sentences triggered by complex seeds are better candidates for automatic pattern learning. Analog to the seed complexity, our first evaluation tells us that frequent patterns with higher arity are more relevant event-based patterns.

However, the main contribution of our work is not the empirical findings in this special case of event relation extraction but rather the novel systematic approach for producing the optimal combination of seeds with varying arity for the learning of the most effective extraction patterns. This approach is domain and task independent and therefore easily adaptable to relation types and text sorts with different extraction complexity.

At this stage, we could apply our learned patters to one-year of test data only. We are going to test our system with the entire volume of test data before the Workshop and plan to report the results there. We will also be able to evaluate the detection of other prize events through our learned patterns. A further challenge is to work out a new automatic evaluation method of new extracted seeds on top of the principles presented by ExDisco and Snowball, to improve the performance of the bootstrapping process.

In our prize-winning domain, the event can often be found within a sentence. However, this is not the standard case for all event extraction tasks. Therefore, we plan to take paragraphs as our future investigation unit. We want to find out how different patterns with different complexity interact with each other to build an event, and how to evaluate their connectivity confidence. It should be an extension of the work presented by (McDonald et al. 2005): the extraction of a complex relation should not only rely on the simple connectivity of the elementary relations, but also on their semantic and discourse relationship in the texts. Furthermore, the current pattern presentation uses very limited linguistic constraints. It works fine for the complex patterns. However, we are concerned about the binary patterns and will try to find out whether more linguistic constraints are needed to ensure their precision.

References

Agichtein, E. and Gravano, L. 2000. *Snowball: Extracting Relations from Large Plain-Text Collections*. In Proceedings of the 5th ACM International Conference on Digital Libraries (DL'00).

Appelt, D. and Israel, D. 1999. *Introduction to Information Extraction Technology*. IJCAI-99 Tutorial.

Appelt, D. 2003. *Semantics and Information Extraction*. Preworkshop Lecture. Workshop on Language Engineering. Johns Hopkins University, Baltimore, 2003.

Bikel, D.M.; Schwartz, R.; and Weischedel, R.M. 1999. An algorithm that learns what's in a name. *Machine Learning Journal Special Issue on Natural Language Learning*, 34(1/3):221—231.

Califf, M. E. and Mooney, R. J. 1999. *Relational Learning of Pattern-Match Rules for Information Extraction*. In Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99), pp. 328—334.

Drozdzynski, W.; Krieger, H.-U.; Piskorski, J.; Schäfer, U.; and Xu, F. 2004. Shallow Processing with Unification and Typed Feature Structures — Foundations and Applications. *Künstliche Intelligenz* 1:17—23.

Fleischman, M.; Hovy, E.; and Echihabi, A. 2003.*Offline Strategies for Online Question Answering: Answering Questions Before They Are Asked,* in R. Erhard Hinrichs and Dan, ed., In Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics.

Grishman, R. and Sundheim, B. 1996. *Message Understanding Conference –6: A Brief History*. In Proceedings of the 16th International Conference on Computational Linguistics, Copenhagen.

Hobbs, J.; Appelt, D.; Bear; Israel; Kameyama; Stickel; and Tyson. 1996. FASTUS: A Cascaded Finite-State Transducer for Extracting Information from Natural-Language Text, in Roche and Schabes, eds., *Finite State Devices for Natural Language Processing*, MIT Press, Cambridge MA.

Lin, D. 1998. *Dependency-Based Evaluation of MINIPAR*. In Proceedings of the Workshop on the Evaluation of Parsing Systems, First International Conference on Language Resources and Evaluation, Granada, Spain.

McDonald, R.; Pereira, F.; Kulick, S.;Winters, S.; Jin, Y.; and White, P. *Simple Algorithms for Complex Relation Extraction with Applications to Biomedical IE*. In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05). pp 491—498.

Mann, G. and Yarowsky, D. 2005. *Multi-Field Information Extraction and Cross-Document Fusion*. In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05). pp 483—490.

Miller, G. A.; Beckwith, R.; Fellbaum, C.; Gross, D.; K. Miller, K. 1993. *Five Papers on WordNet*. Technical Report, Cognitive Science Laboratory, Princeton.

Niles, I. and Pease, A. 2001. *Origins of the Standard Upper Merged Ontology: A Proposal for the IEEE Standard Upper Ontology*. In Proceedings of IJCAI-2001 Workshop on the IEEE Standard Upper Ontology.

Niles, I. and Pease, A. 2003. *Linking Lexicons and Ontologies: Mapping WordNet to the Suggested Upper Merged Ontology*. In Proceedings of the 2003 International Conference on Information and Knowledge Engineering.

Niles, I. and Terry, A. 2004. *The MILO: A general-purpose, mid-level ontology*. In Proceedings of 2004 International Conference on Information and Knowledge Engineering, Las Vegas, NV.

Pierce, D. and Cardie, C. 2001. User-oriented machine learning strategies for information extraction: Putting the human back in the loop. IJCAI-2001 Workshop on Adaptive Text Extraction and Mining, 2001, pp. 80-81.

Riloff, E. 1996. Automatically Generating Extraction Patterns from Untagged Text. *In Proceedings of Thirteenth National Conference on Artificial Intelligence (AAAI-96)*, The AAAI Press/MIT Press, pp. 1044-1049

Yangarber, R. 2001. Scenarion Customization for Information Extraction. Ph.D. diss., Department of Computer Science, Graduate School of Arts and Science, New York University.

Zelenko, D.; Aone, C.; and Richardella, A. 2003. Kernel methods for relation extraction. *JMLR*.