

KNOWLEDGE MEDIA

KMi

I N S T I T U T E

Event Recognition on News Stories and Semi-Automatic Population of an Ontology

Tech Report kmi-04-14

Maria Vargas-Vera and David Celjuska



Event Recognition on News Stories and Semi-Automatic Population of an Ontology

Maria Vargas-Vera
Knowledge Media Institute (KMi),
The Open University,
Walton Hall, Milton Keynes, MK7 6AA,
United Kingdom
m.vargas-vera@open.ac.uk

David Celjuska
Department of Artificial Intelligence
and Cybernetics
Technical University of Kosice,
Letna 9/A, 04001 Kosice, Slovakia
celjuska@neuron.tuke.sk

Abstract

This paper describes a system which recognizes events on news stories. Our system classifies stories and populates a hand-crafted ontology with new instances of classes defined in it. Currently, our system recognizes events which can be classified as belonging to a single category and it also recognizes overlapping events within one article (more than one event is recognized). In each case, the system provides a confidence value associated to the suggested classification. Our system uses Information Extraction and Machine Learning technologies. The system was tested using a corpus of 200 news articles from an archive of electronic news stories describing the academic life of the Knowledge Media (KMi). In particular, these news stories describe events such as a project award, publications, visits, etc.)

1. Introduction

This paper focuses on the problem of semi-automatic population of an ontology. Our approach is based on Information Extraction and Machine Learning technologies. Essentially, information extraction can be seen as the task of pulling predefined relations from texts. Efforts have been made to apply information extraction to several domains, for instance, scientific articles such as MEDLINE [3] and medical records [9]. In designing an information extraction system for KMi, the system should be able to extract the name of KMi projects, KMi funding organizations, awards, dates, etc and ignore anything not clearly relevant to these pre-specified categories. Ontologies can be used in information extraction systems to help them extract relations from semi- or unstructured documents, statements or terms [8]. Also, recent work on semi-automatic ontology acquisition

by means of information extraction, supported by machine-learning methods, is described in [5, 4, 10, 11]. On similar lines, there is CMU's approach for extracting information from hypertext using machine learning techniques and making use of an ontology [2].

Our system, as most information extraction systems, uses some form of partial parsing to recognize syntactic constructs without generating a complete parse tree for each sentence. Such partial parsing has the advantages of greater speed and robustness. High speed is necessary to apply the information extraction to a large set of documents. The robustness achieved by allowing useful work to be done from a partial parsing is essential to deal with unstructured and informal texts.

The main contributions of our paper can be summarized as 1) identification of events in news stories by means of Information Extraction and Machine Learning technology and 2) semi-automatic population of a selected ontology.

The paper is organized as follows: Section 2 shows the event topology used in our event recognition system. Section 3 describes the classification of news stories. Section 4 presents the process model in our system. Section 5 presents the assignment of confidence values to the rules extracted using Crystal. Section 6 presents the evaluation carried out using the KMi archive of news stories. Finally, Section 7 gives conclusions and directions for future work.

2. Event topology

KMi planet is a newsletter covering events and activities taking place at KMi. Events are defined formally in our ontology as classes. Currently, the KMi ontology defines 40 different types of events. Firstly, the event topology is defined directly in the ontology. Then, for each event, we already have defined the slots which might be instantiated by an information extraction component. For the sake of space, we only present the structure of one type of events from

the event hierarchy, this is the (visiting-a-place-or-people) event type.

Class Event: visiting-a-place-or-people
slots:
visitor (list of person(s))
people-or-organisation-being-visited
(list of person(s) or organization)
has-duration (duration)
start-time (time-point)
end-time (time-point)
has-location (a place)
other agents-involved
(list of person(s))
main-agent (list of person(s))

The structure of event visiting-a-place-or-people describes a set of objects which might be encountered in a news story describing a visit event, e.g. visitor, people-or-organisation-being-visited, other agents-involved.

3. Classification of news stories

The main process in the classification is to take each sentence in the text (in our case a news story and see if it matches any of our domain-specific learned patterns. If no pattern applies to a sentence, then no information will be extracted; this means that irrelevant text can be processed very quickly.

We classify news stories or documents as belonging to any of the types of events according to the objects that are found in them. For each event type, we have predefined objects that should be found in a news story covering events of that type. For instance, for the event “visiting-a-place-or-people” the system might encounter objects of type: visitor, place and date.

In our system, classification is performed in the following steps:

- pre-process a news story
- find the objects in a news story using partial parsing
- provide classification of a news story with associated confidence value

Each event in our system has several patterns which can be used to recognize it. For instance, in the case of the “visiting-place-or-people” event, some of the patterns encountered were “X visited Y”, “X visits Y” and “Y visited by X” where X is a person and Y is a place/institution.

Problems might occur when more than one event is described in a news story. In this case, our system decides to classify the news story according to the following criteria:

the confidence value is computed as the number of slots the system was able to extract divided by the total number

of slots that an annotator/expert used during the annotation process on any news story from a given class.

The confidence of classification into a given class is shown below.

$$confidence_classification = \frac{n}{m}$$

where n is the number of items extracted and m is the number of slots used by annotator.

Then, the category which maximizes the sum of the filled slots is placed at the top of the window (i.e. the classification with the maximum confidence value). If none of the templates are able to be filled (during the extraction phase), then the news story is given the status of unclassified news story. The user will be presented with classification, associated confidence value and extracted objects. Once the user agrees (rejects) one (all) of the suggested classifications and extracted information, the ontology is updated with a new instance.

4. Process model

Within this work, we have focused on creating a generic process model for event recognition in news stories. In our system, we have devised four activities: mark-up, learning, extraction and population. We will provide more details of each of these activities in turn.

Mark-up

The activity of semantic tagging refers to the activity of annotating text documents (written in plain ASCII or HTML) with a set of tags defined in the ontology. In particular, we work with a KMi hand-crafted ontology. Our classification system provides means to browse the event hierarchy. In this hierarchy, each event is a class and the annotation component extracts the set of possible tags from the slots defined in the ontology. During the mark-up phase, as the text is selected, the system inserts the relevant XML tags into the document. Our system also offers the possibility of removing tags from a document.

Learning

This phase was implemented by integrating two tools: Marmot and a learning component called Crystal, both from UMass (full description can be found in [7]). Marmot is a natural language pre-processing tool that accepts ASCII files and produces an intermediate level of text analysis that is useful for information extraction applications. Sentences are separated and segmented into noun phrases, verb phrases and prepositional phrases. Marmot has several functionalities: it preprocesses abbreviations to guide sentence

segmentation, resolves sentences boundaries, identifies parenthetical expressions, recognizes entries from a phrasal lexicon and replaces them, recognizes dates and duration phrases, performs phrasal bracketing of noun, preposition and adverbial phrases and finally scopes conjunctions and disjunctions.

A second component is Crystal, a dictionary induction tool [7]. Crystal derives a dictionary of concept nodes from a training corpus. The first step in dictionary creation is the annotation of a set of training texts by a domain expert. Each phrase that contains information to be extracted is tagged (with XML tags).

Crystal initializes a concept node dictionary for each positive instance of each type of event. The initial concept node definitions are designed to extract the relevant phrases in the training instance that creates them but are too specific to apply to an unseen sentences. The main task of Crystal is to gradually relax the constraints on the initial definitions and also to merge similar definitions. Crystal finds generalizations of its initial concept node definitions by comparing definitions that are similar. This similarity is deduced by counting the number of relaxations required to unify two concept node definitions. Then a new definition is created with constraints relaxed. Finally, the new definition is tested against the training corpus to insure that it does not extract phrases that were not marked with the original two definitions. This means that Crystal takes similar instances and generalizes them into a more general rule by preserving the properties from each of the concept node definitions which are generalized.

The inductive concept learning in Crystal is similar to the inductive learning algorithm described in [6] a specific-to-general data-driven search to find the most specific generalization that covers all positive instances. Crystal finds the most specific generalization that covers all positive instances but uses a greedy unification of similar instances rather than breadth-first search.

Extraction

A third component called Badger (from UMass) was also integrated into our event recognition system. Badger makes the instantiation of templates. The main task of badger is to take each sentence in the text and see if it matches any of our concept node definitions. If no extraction concept node definition applies to a sentence, then no information will be extracted: thus irrelevant text can be processed very quickly.

It might occur that Badger obtains more than one type of event for a news story¹. Then our information extraction

system decides to classify the news story according to the criteria defined in section 3.

Populating the ontology

Building domain-specific ontologies often requires time-consuming expensive manual construction. Therefore, we envisage information extraction as a technology that might help us in the ontology maintenance process. During the population step, our information extraction system has to fill predefined slots associated with each event already defined in the ontology. Our goal is to automatically fill as many slots as possible. However, some of the slots will probably still require manual intervention. There are several reasons for this problem:

- implicit information, e.g. there is information that is not stated explicitly in the news story but is understood by human readers from its context,
- none of our patterns match with the sentence that might provide the information (incomplete library of patterns)

The extracted information could be validated using the ontology. This is possible because each slot of each class of the ontology has a type associated with it. Therefore, extracted information not matching the type definition of the slot in the ontology can be highlighted as incorrect.

5. Confidence values associated to the extraction rules

In the automatic construction of ontologies, precision is more important than recall since we want to populate an ontology. Therefore, our goal is to obtain high precision. Currently, we are focusing on associating a confidence value with the Crystal-induced rules in order to increase precision. The confidence value for each rule was computed by a three-fold cross-validation methodology on the training set. According to this methodology, the training set is split into three equally sized subsets and the learning algorithm is run three times. Each time, two of the three pieces are used for training and the third is kept as unseen data (test set) for the evaluation of the induced rules. The final result is the average over the three runs. At run time, each instance extracted by Badger will be assigned the precision value of that rule. The main feature of using confidence values is that, when presented with ambiguous instantiations, we can still choose the one with the highest estimated confidence. We believe that the confidence value could be used as one way to get rid of extraction rules which are below a given threshold.

¹ The first implementation of our event recognition System, which only recognizes single events, is described in [11]

6. Evaluation

We have tested our event recognition system applying a three-fold cross-validation methodology. In this evaluation, we have used standard metrics for computing precision and recall²

Previous work has reported that spurious patterns were deleted manually from the library of rules under the assumption that they were not likely to be of much value [7]. However, as a first phase in our experiments, we did not carry out any deletion of spurious rules. Overall precision for event “visiting-a-place-or-people” was 68% and overall recall is 52%. The experimental results suggested that precision could drop dramatically if the set of extraction rules were used as generated by Crystal. As second phase in our experiment, we associated confidence value to the extraction rules. The performance of the event “visiting-a-place-or-people” was precision 90% and recall 14%. Preliminary results seems encouraging. However, further research needs to be undertaken in the direction of the association of a confidence level with the extraction rules.

7. Conclusions and future work

We have built a system which recognizes events in news stories and extracts knowledge using an ontology. Currently, our system has been trained using KMi Planet, an archive of 200 news stories that we have collected in KMi.³ The training step was performed using typical examples of news stories belonging to each of the different type events defined in a hand-crafted ontology. Our system recognizes single events and overlapping events in one document. It is then able to suggest a likely classification for a news story. Currently, the population of the selected ontology is performed at the level of instances. Our system extracts instances of classes defined in the event ontology. However, in the future, we will explore the possibility of using the extracted information with Conceptool [1] in order to create new classes in a selected ontology. This will allow us to refine our ontology to a finer granularity.

The experiments showed that an automatic mechanism was needed in order to determine which extraction rules are spurious. We have outlined that this problem can be solved by associating confidence values to the extraction rules.

² As a reminder, precision and recall metrics are defined as follows:

$$\text{Precision} = \frac{\text{number of items correctly extracted}}{\text{total number of items extracted by system}}$$

$$\text{Recall} = \frac{\text{number of items correctly extracted}}{\text{total number of items need to be extracted by system}}$$

³ URL:<http://kmi.open.ac.uk/planet/>

Currently, our event recognition system works with the KMi Planet ontology. But, in the future, we plan to offer a selection of ontologies.

References

- [1] E. Compatangelo and H. Meisel. Reasonable support to knowledge sharing through schema analysis and articulation. *Journal of Engineering Intelligent Systems*, To appear 2003.
- [2] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, K. N. T. Mitchell, and S. Slattery. Learning to Construct Knowledge Bases from the World Wide Web. *Artificial Intelligence*, 1999.
- [3] M. Craven and J. Kumlien. Constructing Biological Knowledge Bases by Extracting Information from Text Sources. In *Proceedings of The 7th International Conference on Intelligent Systems for Molecular Biology (ISMB-99)*, 1999.
- [4] J.-U. Kietz, A. Maedche, and R. Volz. A method for semi-automatic ontology acquisition from a corporate intranet. In *Proceedings of the EKAW'00 Workshop on Ontologies and Text, Juan-Les-Pins, France*, oct 2000.
- [5] A. Maedche and S. Staab. Semi-automatic engineering of ontologies from texts. In *Proceedings of the 12th International Conference on Software Engineering and Knowledge Engineering, SEKE2000, Chicago, IL, USA*, pages 231–239, july 2000.
- [6] T. Mitchell. Generalization as search. *Artificial Intelligence*, 18:203–226, 1982.
- [7] E. Riloff. An Empirical Study of Automated Dictionary Construction for Information Extraction in Three Domains. *AI Journal*, 85:101–134, 1996.
- [8] C. Roux, D. Proux, F. Rechenmann, and L. Julliard. An Ontology Enrichment Method for a Pragmatic Information Extraction System gathering Data on Genetic Interactions. In *Proceedings of The 14th European Conference on Artificial Intelligence (Workshop on Ontology Learning ECAI-2000)*, 2000.
- [9] S. Soderland, D. Aronow, D. Fisher, J. Aseltine, and W. Lehnert. Machine Learning of Text Analysis Rules for Clinical Records. Tr 39, Center for Intelligent Information Retrieval, 1995.
- [10] M. Vargas-Vera, J. Domingue, Y. Kalfoglou, E. Motta, and S. B. Shum. Template-driven information extraction for populating ontologies. In *In proceedings of the Workshop Ontology Learning IJCAI-2001*, 2001.
- [11] M. Vargas-Vera, J. Domingue, E. Motta, S. B. Shum, and M. Lanzoni. Knowledge extraction by using an ontology-based annotation tool. In *In proceedings of the Workshop Knowledge Markup & Semantic Annotation, held in association with the First International Conference on Knowledge Capture (K-CAP 2001), Victoria Canada*, pages 5–12, 2001.