Annals of the University of Craiova, Mathematics and Computer Science Series
Volume 37(3), 2010, Pages 64–70
ISSN: 1223-6934

XML Semantic Schema Annotation for Dependency Relationships

Mihaela Colhon

Abstract. This paper shows that a semantic network structure, named *semantic schema*, can be used for adequate representation of the syntactic dependency relations existing between the words of a natural language phrase. Such a structure can be further used in various Natural Language Processing applications that make use of the words interactions inside a phrase. It is also described a wrapper program, that, as a proof of concept, converts the output data of a dependency parser (MINIPAR parser) into semantic schema XML format.

2010 Mathematics Subject Classification. Primary 60J05; Secondary 60J20.
Key words and phrases. morpho-syntactic annotation, MINIPAR parser, dependency relation, semantic network.

1. Introduction

Syntatic dependency relationships are successfully used in Natural Language Processing (NLP) tasks such as: *interlingual annotation* in text corpora ([5]), *extracting collocational knowledge* from text corpus ([9]), *information extraction*, *parsing tools* ([11]), *textual alignment* ([8]), *machine translation*, etc.

Dependency structures are usually represented by a special kind of graphs, described in the ongoing ISO LAF/GrAF initiative ([6], [7]). LAF (Linguistic Annotation Framework) provides the general framework for representing linguistic annotations. The data model belonging to GrAF (Graph Annotation Format) is based on a directed graph named *syntax graph* or *dependency tree*. Both structures are directed acyclic graphs with a single root node. The nodes and the edges may be labelled by feature structures.

We propose an XML annotation for dependency relations, mapped on an abstract semantic network structure named *semantic schema*, that is not restricted to a particular natural language syntax and furthermore can be interpreted with constructions from a variety of natural languages. In line with SynAF initiative ([3]) we take two types of textual units that can be annotated with dependency information: words/tokens and dependency relationships based on which dependency groups can be constructed.

The mechanism presented in this paper is endowed with morpho-syntactic annotation that can describe elementary linguistic data necessary in many natural language tasks, such as a contextual translation mechanism ([2]). Indeed, in the context of a translation mechanism the resulted representations will have two basic functions:

Received June 29, 2010. Revision received August 23, 2010.
This work was supported by the strategic grant POSDRU/89/1.5/S/61968, Project ID 61986 (2009), co-financed by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007-2013.

- to describe the morpho-syntactic words' data: the part-of-speech (POS) feature, like *verb, adjective, noun*, and morphological and grammatical features (*number, gender, person, verbal tense*)
- to describe the dependency relations, like *head-modifier* relation existing within the sentence boundaries.

To summarize, the model we introduce in this paper is about the syntactic constituencies and dependencies that characterizes every natural language phrase, in this way provind a proper interface between syntactic and semantic phrase annotation.

2. Semantic Schema Representation for Dependency Relations

2.1. Semantic Schema. A semantic schema is an abstract structure that extend the concept of the semantic network and which is formalized by means of a tuple of symbolic entities. This structure becomes a description of real data only if some interpretation is considered. Two aspects can be relieved in connection with a semantic schema $\mathcal{S}$ ([13]):
1) A formal aspect in $\mathcal{S}$ by which some formal computations in a Peano algebra are obtained.
2) An evaluation aspect with respect to an interpretation.

The proposed semantic schema annotation for dependency relations belongs to the formal aspect of the schema. As consequence, in what follows we will concentrate on this aspect. The evaluation one will be treated in our future works, in the context of a machine translation that will be defined to used this kind of representations.

Consider θ a symbol of arity 2 and a non-empty set A_0. Starting from A_0 we construct the set $\overline{A_0}$ as the Peano θ-algebra:

$$\overline{A_0} = \bigcup_{n\geq 0} A_n,$$
$$A_{n+1} = A_n \cup \{\theta(u,v) \mid u,v \in A_n\} \tag{1}$$

A semantic θ-schema is a system $\mathcal{S} = (X, A_0, A, R)$ where ([14]):
- X is a non-empty set of node symbols
- A_0 is a finite non-empty set of symbolic names used to label the dependency relations
- $A_0 \subseteq A \subseteq \overline{A_0}$, where $\overline{A_0}$ is the Peano θ-algebra generated by A_0
- $R \subseteq X \times A \times X$ is a non-empty set of relations such that the tuples satisfy the following conditions:
  $(R1)$ $(x, \theta(u,v), y) \in R$ then $\exists z \in X$: (x,u,z), $(z,v,y) \in R$
  $(R2)$ $\theta(u,v) \in A$, (x,u,z), $(z,v,y) \in R$ then $(x, \theta(u,v), y) \in R$
  $(R3)$ $pr_2 R = A$, for $pr_2 R = \{u \in A \mid (x,u,y) \in R\}$

We note the set of the dependency relations mapped on the schema $\mathcal{S}$ by:

$$R_0 = \{(x,u,y) \in R \mid u \in A_0\} \tag{2}$$

In what follows, the elements of R_0 will be named the *initial relations*. Every element of the set R is a triple $(x,u,y) \in X \times A \times X$ such that x and y represent some nodes of the schema while u is the symbolic name of an arc existing between the two nodes.

The elements of $A \setminus A_0$ denote the compound relations obtained in the semantic schema structure by fulfilling the conditions $(R1) \div (R3)$. Accordingly to the semantic schema defintion, every relation $r \in R \setminus R_0$ can be broken in two relations of $r_1, r_2 \in R$ that fulfill the composition condition: the final node of the first relation r_1 is the initial node of r_2 (results from the condition $R1$). Conversely, if two relations $r_1, r_2 \in R$,

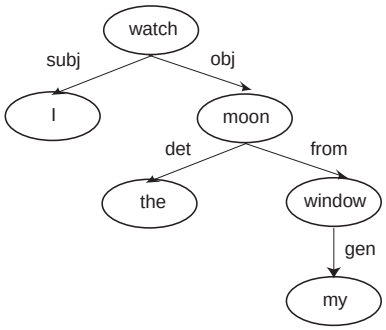

FIGURE 1. MINIPAR dependency tree

$r_1 = (x, u, z)$ and $r_2 = (z, v, y)$ fulfill the composition condition, then there must be $\theta(u, v) \in A$ in order to have $(x, \theta(u, v), y) \in R$ (condition $R2$). The last condition $R3$ "says" that all the symbolic names of A are used to label the relations from R.

As it can be seen, such a structure can offer the representation model for a path-based reasoning processes constructed by means of binary relations composition.

2.2. MINIPAR Dependency Parser. MINIPAR[1] ([10]) is a well-known parser for the English language. It represents its grammar as a network of nodes and links, where the nodes represent grammatical categories and the links represent types of dependency relationships. The grammar is manually constructed, based on the Minimalist Program ([1]).

An evaluation with the SUSANE[2] corpus (Surface and Underlying Structural Analysis of Naturalistic English) shows that MINIPAR achives about 88% precision and 80% recall with respect to dependency relationships.

The output produced by MINIPAR is a tree in which the nodes are the words from the analysed sentence labelled with their grammatical categories and the edges are dependency relations between the words. The dependency tree is given by a list of tuples, each tuple having the following format:

`(word category [head] [relationship])`

where `word` is the word represented by the node, followed by its grammatical `category`, `head`- the parent label from the dependency tree and the dependecy `relationship` that connects the two nodes (e.g. `subj` for subject, `cmpl` for complement, `spec` for specifier).

This is the theory. In practice, there are exceptions from it in order to ensure the linguistic fidelity and the usability of the representation. For example, let us consider the sentence "*I watch the moon from my window*". The dependency tree generated by MINIPAR is illustrated in Fig. 1.

As it can be seen, each node correspond to a word of the sentence and each edge, except the one labeled by "from" represent a dependency relationship between words. In this representation, there is no node for the word "from"; instead a dependency label is introduced and is labelled with this word. Also, the case we take also relieve an issue of the MINIPAR parser known as *attachment error*. In our case the prepositional

[1]http://www.cs.ualberta.ca/ lindek/minipar.htm

[2]The texts of the SUSANNE Corpus are a subset of the texts included in the (unannotated) Brown University Corpus

phrase "from my window" that should be attached to the verb "watch" is wrongly attached to the noun "moon". These errors belong to the *semantic ambiguity* of the natural language constructions and do not make the subject of this paper.

2.3. Dependency Relations mapped in Semantic Schema. Dependency relations provide a way of representing the syntactic properties of a natural language utterance by means of the directed binary relations (dependencies) existing between two elements of a phrase: one element is called the *head* or *governor* and the other the *modifier* or *dependent*. A dependency relation is usually labelled with a specific type of grammatical relation and annotated by some other additional information such as importance degree from a, let's say, disambiguation mechanism point of view.

The dependency links corresponding to a natural language phrase can be formally mapped on a semantic schema structure, where a node in the schema represents a linguistic unit and a directed edge linking two nodes denotes a dependency relation existing between a head word and a dependent one.

By mapping the dependency relations of a natural language phrase on a semantic schema structure $\mathcal{S} = (X,\ A_0,\ A, R)$, the closure of these relations can be obtained. Of course, this closure is restricted to the labels of A (condition $R2$). The relations obtained by composition, based on the initial relations will connect not only the words that are in a direct dependency relation, but words that are in a dependency relation with a common word. Thus, the dependency links that exist between the words of a phrase are automatically generated.

In order to exemplify the semantic schema representations for dependency relationships let us reconsider the sentence "*I watch the moon from my window*". The MINIPAR dependency tree for this sentence is given in Fig. 1.

We have that the sentence's words set is: {*I*, *watch*, *the*, *moon*, *from*, *my*, *window*}. The word ids together with the POS information as they result from the MINIPAR analysis are given below:

word	id	POS
I	w1	N(Noun or Noun Phrase)
watch	w2	V(Verb)
the	w3	Det(Determiner)
moon	w4	N
from	w5	Prep(Preposition)
my	w6	N
window	w7	N

Also we list the MINIPAR dependency relationships for this sentence together with their meanings:

relation id	name	dependent	head	meaning
r1	s	w1	w2	surface object of the clause
r2	det	w3	w4	determiner and its head noun
r3	obj	w4	w2	object
r4	mod	w5	w4	adjunct modifier and its noun
r5	gen	w6	w7	genitive modifier and its noun
r6	pcomp-n	w7	w5	nominal complement of preposition
r7	from	w7	w4	resulted from r6 and r4

In the semantic schema notations, these dependecy relations have the following form (we eliminate r6 and r4 relations): (w1, s, w2), (w3, det, w4), (w4,obj, w2), (w6, gen, w7), (w7, from, w4). Results that the set of labels for the initial relations is:

$$A_0 = \{\ \texttt{s}, \texttt{det}, \texttt{obj}, \texttt{gen}, \texttt{from}\}$$

If we take the set:

$$A = A_0 \cup \{\ \theta(\texttt{obj}, \texttt{from}), \theta(\texttt{from}, \texttt{gen}), \theta(\texttt{obj}, \theta(\texttt{from}, \texttt{gen}))\ \}$$

then the semantic schema structure for the sentence "*I watch the moon from my window*" becomes:

$$\mathcal{S} = (X, A_0, A, R)$$

where the set of nodes, that is the set of the word ids is:

$$X = \{\ \texttt{w1}, \texttt{w2}, \texttt{w3}, \texttt{w4}, \texttt{w6,w7}\}$$

and the set of all dependency relations (the initial ones, components of the subset R_0, together with the compound ones) is: $R = R_0 \cup \{(\texttt{w2}, \theta(\texttt{obj}, \texttt{from}), \texttt{w7}),(\texttt{w4}, \theta(\texttt{from}, \texttt{gen}), \texttt{w6}), (\texttt{w2}, \theta(\texttt{obj}, \theta(\texttt{from}, \texttt{gen})), \texttt{w6})\}$

3. XML Semantic Schema Annotation

As a proof of concept, we use MINIPAR Parser Wrapper for Java freely available on the RESuLT Project site[3] in order to convert the output of MINIPAR parser into a semantic schema structure that is further stored in a XML document. The benefit of such output format in the Web service context is obviously, improving the re-usability and interoperability of this parser results.

In order to represent a semantic schema structure using XML data we need two node types: a node type for the elements of X set and a node type for dependency relations, the elements of the R set. According to the GrAF specifications, these elements can be annotated by attaching feature structures. To a word/token one can assign morpho-syntactic information, while the dependency relations can be enriched with other information such as the estimated probability of occurence ([4]).

Our mechanism annotates the sentence words with morpho-syntactic information such as `lemma` and `POS`. In a future version these information can be decorated with additional data such as `gender`, `number`, `person`, `case`, `article`, `degree,mode`, `tense` and `type` data. The morphological information for a word are collected in the `parameters` component.

```
<node>
    <id> word id< /id>
    <token> word form< /token>
    <parameters>
        <parameter>
          <name> lemma < /name>
          <value> word ground form < /value>
        < /parameter>
        <parameter>
          <name> POS < /name>
          <value> word grammatical category < /value>
        < /parameter>
    < /parameters>
< /node>
```

[3] http://nlp.shef.ac.uk/result/software.html

A relation node denotes a dependency relation occuring in the semantic schema structure. It is identified by a tuple of three elements (or sub-types): one marks the head of the dependency relation, another marks the dependent while the label element codifies the symbolic name of the relation. As it will be shown, the head and the dependent component of such relations are designated by word indices (`id` attribute).

<**relation**> `dependency relation`
<**from**> `head word id`< /**from**>
<**to**> `dependent word id`< /**to**>
<**label**> `relation symbolic name`< /**label**>
< /**relation**>

For the sentence taken as the case study of this paper, the XML semantic schema data are (the t symbol replace the θ notation):

<root>
<nodes>
<node>
<id> w1< /id>
<token> i< /token>
<POS> N< /POS>
< /node>
<node>
<id> w3< /id>
<token> the< /token>
<POS> Det< /POS>
< /node>
<node>
<id> w6< /id>
<token> my< /token>
<POS> N< /POS>
< /node>
<node>
<id> w7< /id>
<token> window< /token>
<POS> N< /POS>
< /node>
<node>
<id> w4< /id>
<token> moon< /token>
<POS> N< /POS>
< /node>
<node>
<id> w2< /id>
<token> watch< /token>
<POS> V< /POS>
< /node>
< /nodes>
<relations>
<relation>
<from> w2< /from>
<to> w1< /to>
<label> subj< /label>
< /relation>
<relation>
<from> w4< /from>
<to> w3< /to>
<label> det< /label>
< /relation>
<relation>
<from> w7< /from>
<to> w6< /to>
<label> gen< /label>
< /relation>
<relation>
<from> w4< /from>
<to> w7< /to>
<label> from< /label>
< /relation>
<relation>
<from> w2< /from>
<to> w4< /to>
<label> obj< /label>
< /relation>
<relation>
<from> w2< /from>
<to> w7< /to>
<label> t(obj,from)< /label>
< /relation>
<relation>
<from> w4< /from>
<to> w6< /to>
<label> t(from,gen)< /label>
< /relation>
<relation>
<from> w2< /from>
<to> w6< /to>

<label> t(obj,t(from,gen))
< /label>
< /relation>
< /relations>
< /root>

4. Conclusion

The mechanism presented in this paper can facilitate the development of a standardized dependency relations XML annotation that could be more of a semantic type rather than purely syntactical. Thus, every output of an existing dependency parser can be stantardized by a data format conversion mechanism. Following this idea, we intend to align our future works with the present concerns ([4]) regarding the development of a more semantic orientated mechanism for data representation, in which the XML annotations are linked with ontological knowledge.

Also, the formal aspects of the semantic schemas should be more explored in order to provide for each dependency relation, all the information about the basic units from which it has been derived.

References

[1] N. Chomsky, *Minimalist Program*, MIT Press, 1995.
[2] M. Colhon and D. Dănciulescu, Semantic Schema for Natural Language Generation in Multilingual Systems, *Journal of Knowledge, Communications and Computing Technologies* **II** (2010), no. 2, 10–18.
[3] T. Declerck, SynAF: Towards a Standard for Syntactic Annotation, *Proceedings of LREC2006* (2006), 229–233.
[4] Y. Hayashi, T. Declerck and C. Narawa, LAF/GrAF-grounded Representation of Dependency Structures, *Proceedings of LREC 2010 - 7th Language Resources and Evaluation Conference*, (2010).
[5] S. Helmreich et al., Interlingual Annotation of Multilingual Text Corpora, *HLT-EACL Workshop on Frontiers in Corpus Annotation*, (2004).
[6] N. Ide and H. Bunt, Anatomy of Annotation Schemes: Mapping to GrAF, *Proceedings of the Fourth Linguistic Annotation Workshop, ACL 2010*, Uppsala, Sweden (2010), 247-255.
[7] N. Ide and L. Romary, International standard for a linguistic annotation framework. *Journal of Natural Language Engineering* **10** (2004), no. 34, 211-225.
[8] A. Iftene, Textual Entailment (Ph.D. Thesis), Technical Report TR 09-02, *Alexandru Ioan Cuza* University, Iaşi, ISSN 1224-9327, (2009).
[9] D. Lin, Extracting collocations from text corpora, *Proceedings of the First Workshop on Computational Terminology*, Montreal, Canada (1998), 57-63.
[10] D. Lin, Dependency-based Evaluation of Minipar, *Workshop on the Evaluation of Parsing Systems*, Spain, (1998).
[11] T. Nakazawa, S. Kurohashi, Statistical Phrase Alignment Model Using Dependency Relation Probability, *Proceedings of SSST-3, Third Workshop on Syntax and Structure in Statistical Translation* (2009), 10-18.
[12] V. Tsang and S. Stevenson, A Graph - Theoretic Framework for Semantic Distance, *Computational Linguistics* **36** (2010), no. 1, 31-69.
[13] N. Ţăndăreanu and M. Ghindeanu, Properties of derivations in a Semantic Schema, *Annals of University of Craiova, Math. Comp. Sci. Ser.* **33** (2006), 147–153.
[14] N. Ţăndăreanu, Semantic Schemas and Applications in Logical representation of Knowledge, *Proceedings of the 10th International Conference on Cybernetics and Information Technologies, Systems and Applications (CITSA2004)* **III** (2004), Orlando, Florida, USA, 82–87.

(Mihaela Colhon) Department of Computer Science, University of Craiova, 13 A.I. Cuza Street, Craiova, 200585, Romania
E-mail address: mcolhon@inf.ucv.ro