A powerful and versatile XML format for representing role-semantic annotation

Katrin Erk and Sebastian Padó

Computational Linguistics
Saarland University
Saarbrücken, Germany
{erk,pado}@coli.uni-sb.de

Abstract

We present two XML formats for the description and encoding of semantic role information in corpora. The TIGER/SALSA XML format provides a modular representation for semantic roles and syntactic structure. The Text-SALSA XML format is a lightweight version of TIGER/SALSA XML designed for manual annotation with an XML editor rather than a special tool. Both formats can deal with underspecification, roles crossing the sentence boundary, compound splitting, and whole-sentence tags for meta-level comments.

1. Introduction

The last years have seen increasing interest in the task of semantic role labelling, which mirrors the need for semantic information in NLP applications. This interest is manifest for example in the choice of semantic role labelling as the CoNLL 2004 shared task, and the inclusion of a semantic role labelling track in SENSEVAL-3. Crucial for the training of automatic systems for semantic role labelling are large role-annotated corpora. To represent these corpora, a multi-level annotation format which integrates semantic role annotation with other annotation levels is necessary.

In this paper we present such a format, TIGER/SALSA XML. Based on XML, it stores syntax and semantics independently and allows semantics to refer to syntax through a well-defined interface. It is a modular extension of TIGER XML (Mengel and Lezius, 2000), a largely theory-neutral description format for syntactic structure. In addition, we present a second format for semantic role assignment, Text-SALSA XML. As a lightweight version of TIGER/SALSA XML, it has the same expressivity, but is optimised for manual annotation.

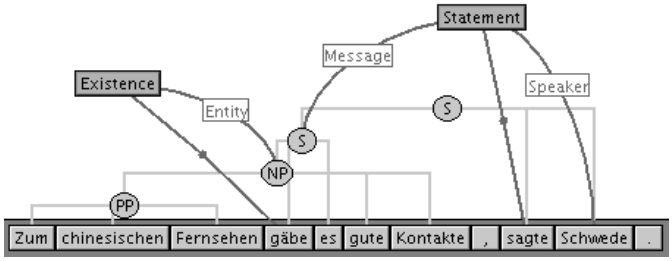
After sketching in Section 2 the project in which TIGER/SALSA XML is used, we present the two formats for role-semantic annotation, TIGER/SALSA XML (Sec. 3) and Text-SALSA XML (Sec. 4). We continue with a comparison of the two formats (Sec. 5) and of TIGER/SALSA XML with other formats for the representation of semantic roles (Sec. 6), closing with a discussion of further uses of TIGER/SALSA XML (Sec. 7).

2. Role-semantic annotation in the SALSA project

The two formats presented in this paper were developed within the SALSA project (Erk et al., 2003b), which is tagging a German corpus manually and semi-automatically with semantic roles in order to derive a large domain-independent lexical semantic resource. The corpus used is TIGER (Brants et al., 2002), a 1.5 Million word corpus of newspaper text with manually annotated syntactic structure, and the semantic annotation is performed using FrameNet (Johnson et al., 2002) frame semantic roles.

In FrameNet, expressions that introduce semantic roles (the *frame-evoking elements (FEEs)* or *targets*) are organised into *frames*, conceptual structures describing situations. The roles, or *frame elements (FEs)*, are local to particular frames and express participants and concepts involved in the described situations. FrameNet currently contains about 400 frames. While it was constructed for English, we found that the frames can be used for German without major problems.

Figure 1 shows a screenshot of the SALSA annotation tool (Erk et al., 2003a), a TIGER sentence annotated with syntax and semantic roles. The straight edges represent syntactic, the curved edges semantic annotation. This sentence contains two frame-evoking elements, namely *gäbe* (exist), which evokes `Existence`, and *sagte* (said), which evokes `Statement`. The SALSA annotation directly produces TIGER/SALSA XML as its export format.



(There were good connections to Chinese television, Schwede said.)

Figure 1: Semantic annotation in SALSA

3. TIGER/SALSA XML

In this section we present the TIGER/SALSA XML format. First we discuss the abstract model of syntactic and semantic information underlying the format, then we describe the XML representation.

3.1. The underlying model

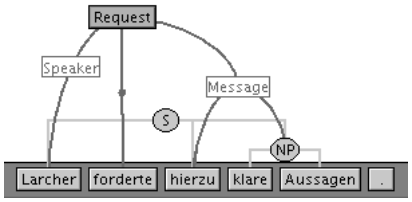
The syntactic level of TIGER/SALSA XML corresponds to the syntactic representation of TIGER XML (Mengel and Lezius, 2000). The syntactic structure is a tree with both

node and edge labels[1]. Trees can contain crossing edges to encode discontinuous constituents, like in Figure 1, where the daughters *gäbe es* of *S* are embedded within the *NP*.

On the role-semantic level, we model each frame instance as a *frame tree* of depth one with a root labelled with the frame name. A frame tree has at least one edge that points to the frame-evoking element, which is unlabelled in the graphical representation. All other edges point to frame elements and are labelled accordingly.

In order to make the annotation more flexible, we keep all frame trees separate. This means that leaves of frame trees are nodes of the syntactic structure. In principle, however, frame tree leaves could also figure as roots of other frame trees, resulting in a nested semantic structure.

In addition, a model for exhaustive semantic annotation must also be able to encode the following complications:



(Larcher demands clear statements concerning this issue.)

Figure 2: One semantic role, two constituents

- A frame element may consist of more than one constituent. In the sentence in Fig. 2, the modifier *hierzu* (concerning this issue) can be understood as modifying the object *klare Aussagen* and not the whole sentence. In this reading the `Message` frame element consists of two syntactic constituents, "clear statements" and "concerning this issue". Consequently, our model allows for frame trees in which multiple edges bear the same label.
- A frame element or target may consist of only part of a word in the case of (German) compound nouns. For example, the German compound *Mietrechtsdiskussion* (tenant law discussion) contains both the target (*diskussion*) that introduces a `Conversation` frame and its `Topic` role (*Mietrecht*). Therefore, our model is able to make reference to sub-word units.
- A frame element may be situated in a different sentence than the target, as often happens with conversation frames. Hence, frame trees can refer to entities in adjacent sentences.
- At times, the meaning of a sentence is ambiguous or vague, and annotators cannot commit to a single tag. For these situations, the model allows to tag multiple annotation referring to the same entity as *underspecified*, both on the level of frames and the level of frame elements. Consistent with (Kilgarriff and Rosenzweig, 2000), it is left to the user how to interpret this representation (e.g. as disjunction or conjunction).

[1]Secondary edges, used to model ellipsis, raise the proper descriptive power to DAGs.

3.2. The representation

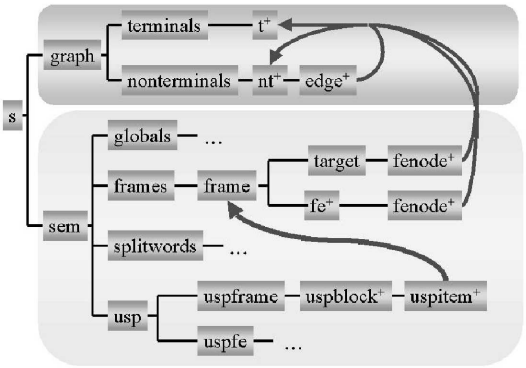


Figure 3: Structure of a TIGER/SALSA XML sentence

The tree in Figure 3 shows the implementation of this model in TIGER/SALSA. The nodes correspond to XML elements, and the edges to permissible embeddings. Elements that may be repeated are marked with a "+".

A sentence (an `<s>` element) has two parts, one for the syntactic structure, (the `<graph>` element in the upper part) and one for the semantic roles (the `<sem>` element in the lower part). In TIGER XML without semantic annotation, a sentence has only one child, `<graph>`. It lists each terminal node as a `<t>` element below `<terminals>`, and each nonterminal node as a `<nt>` element below `<nonterminals>`. Edges are realised not via XML element embedding, but with explicit `<edge>` elements that refer to nodes via their unique identifiers, depicted as arrows in Fig. 3 . This allows for crossing edges and hence for a uniform treatment of continuous and discontinuous constituents (see e.g. the NP in Figure 1).

TIGER/SALSA XML adds a layer of semantic information by introducing the additional semantics element `<sem>`, leaving the syntactic representation in `<graph>` unchanged. `<sem>` contains a straightforward representation of the semantic annotation for the current sentence, as modelled in Section 3.1.. Again, all references to (either syntactic or semantic) entities are expressed in terms of identifiers to keep the levels of representation separate.

The `<frames>` element contains the role-semantic information proper. Similar to syntax, nodes and edges of frame trees are represented as explicit elements `<frame>`, `<target>` and `<fe>`. For all semantic nodes and edges, we introduce new globally unique IDs, such that semantic roles crossing sentence boundaries do not need special treatment, as reference to unique

The `<globals>` element contains tags such as 'is metaphoric' or '(needs) reexamination'. In the `<splitwords>` element, we record the treatment of German compound nouns, effectively introducing new terminal nodes "below" the original terminals. Underspecification is recorded in `<usp>`. One `<uspblock>` inside `<uspframe>` describes one case of frame underspecification, each `<uspitem>` child referring to one frame involved in the underspecification (by its unique ID). `<uspfe>` handles frame element underspecification in the same manner.

In TIGER/SALSA XML, the different annotation levels are kept in two separate blocks. The format is not standoff in the strictest sense, as all information about a sentence collected within one `<s>` element. However, the annotation levels could in principle be decoupled completely because all reference between annotation levels is via identifiers that are unique throughout the corpus.

4. Text-SALSA XML

Text-SALSA XML is a lightweight version of TIGER/SALSA XML for use on bare text. Optimised for human readability and ease of use, it is simple enough to be suitable for manual annotation with an XML editor or even just a text editor.

In Text-SALSA XML, frame and frame element names translate directly to XML element names. The `Statement` frame of Figure 1 is encoded in Text-SALSA XML as follows:

```
<STATEMENT>
  <MESSAGE> Zum chinesischen Fernsehen gaexbe es
  gute Kontakte </MESSAGE> , <FEE> sagte </FEE>
  <SPEAKER> Schwede </SPEAKER> .
</STATEMENT>
```

Representing frame and frame element names as XML element names, rather than attributes, makes DTD maintenance cumbersome, but is much easier to read and write manually in a fast and reliable fashion.

Despite its simplicity, Text-SALSA XML has in principle the same expressivity as TIGER/SALSA XML: Discontinuous FEs can be tagged by using the same element label twice. FE assignment across sentence boundaries is possible if annotators have a window of context sentences available. Frame element underspecification uses multiple occurrences of elements, too, but embeds them into an underspecification element:

```
<USPFE>
 <SPEAKER> <MEDIUM> the motion </MEDIUM> </SPEAKER>
</USPFE>
asks for a policy change
```

Compounds can be annotated part by part:

```
<SPLITWORD>
  <TOPIC Mietrechts </TOPIC> <FEE> diskussion </FEE>
</SPLITWORD>
```

By representing the frame as an XML element enclosing the whole sentence, Text-SALSA XML assumes that only one frame at a time is annotated for a sentence. Annotation that involves multiple frames (such as frame underspecification and ellipsis) is possible but laborious, as it requires copying the whole sentence.

Using DTDs, any validating XML parser can check annotated Text-SALSA XML files for adherence to the annotation scheme. Furthermore, we have software to convert Text-SALSA XML into full TIGER/SALSA XML, given a syntactic description of the annotated sentences.

5. Comparison of the two formats

In this section we compare TIGER/SALSA XML and Text-SALSA XML with respect to annotation and processing.

Annotation. With regard to usability in the annotation task, we cannot directly compare the two formats per se; rather, we can only compare the two annotation scenarios the formats have been designed for.

Text-SALSA XML is meant to be produced directly by annotators, using a text or XML editor. Manual XML creation is slow and error-prone (e.g. annotators may accidentally cut through words), and if the syntactic structure is not shown in the file that is being annotated, annotators are not guaranteed to respect syntactic bracketing. However, the expressivity of this format is the same as for TIGER/SALSA XML (although the annotation of complex cases of coordination and ellipsis is infeasible in practice).

TIGER/SALSA XML is not (easily) human-readable and thus requires an annotation tool – but this overhead usually pays off, since a task-specific annotation tool can speed up the annotation and prevent many of the errors we found in Text-SALSA XML annotation. We found that the introduction of the SALSA Annotation Tool increased annotation speed by a factor of at least two, and reduced inter-annotator disagreement to one third the previous number.

Processing. Computation of inter-annotator agreement for Text-SALSA XML is rather complicated, since the semantic markup must be compared on the level of parts of words. TIGER/SALSA XML, on the other hand, is optimised for automatic evaluation: All frames for a sentence are grouped under the `<frames>` element, and they all refer to the syntactic structure via unique identifiers. Also, TIGER/SALSA XML encodes information redundantly at crucial places; for example, underspecification is stored both locally in elements and globally in the `<usp>` element, which makes processing very efficient.

Conclusion. The two formats are designed for different annotation scenarios. TIGER/SALSA XML is a modular, very general format for encoding semantic role information, optimised for efficient generation and evaluation. However, it can only be reasonably produced by an annotation tool. If such a tool is available, it is clearly the better choice. Nevertheless, there are situation in which this involves too much overhead, for example if there is no syntactic markup available for a text, or if the annotation is carried out as a kind of "rapid prototyping" to test the appropriateness of a new annotation scheme. For such cases, when roles are annotated on bare text, Text-SALSA XML can be used advantageously; for processing, it can then be transformed into TIGER/SALSA XML.

6. TIGER/SALSA XML and other formats for corpora with semantic role markup

In this section we compare TIGER/SALSA XML to other formats used for storing semantic role information.

FrameNet (Johnson et al., 2002) and PropBank (Kingsbury et al., 2002) both use stand-off annotation formats that specifies target and semantic roles by character offsets in the sentence. Here is an example in FrameNet XML:

```
<layer name="Target">
 <labels>
   <label name="Target" start="21" end="24" />
   <label name="Target" start="40" end="47" />
 </labels>
</layer>
<sentence ID="1242945">
 <text> Sometimes we have to give the officials the slip
 so that we can go out to fish , '' Samala said . </text>
</sentence>
```

The representation characterises the target *give the slip* (of frame `Evading`) as covering characters 21 to 24 and 40 to 47 in the sentence.

Whether semantic roles should refer to syntactic structure, or both syntactic structure and semantic roles should refer to the words of the sentence, is a design decision. For languages with freer word order, such as German, an advantage of reference to syntactic structure is that the semantic annotation does not have to deal with discontinuous constituents in special ways. Recall the example in Fig. 1, where the `Entity` frame element is realised by the discontinuous *Zum chinesischen Fernsehen. . . gute Kontakte*.

The Prague Dependency Treebank (Hajičová, 1998) stores its annotation in an SGML file. It uses not a standoff but a word-centered format, assembling all levels of annotation for a word in the SGML element for that word. Like in TIGER/SALSA XML, tree edges are realised as references to unique node IDs, however here we have a single tectogrammatical structure for the whole sentence, while in the SALSA scheme frame trees are independent.

7. The Question of Flexibility

Due to its very general underlying model, TIGER/SALSA XML is limited neither to the present corpus (the TIGER corpus) not the present task (annotating semantic roles in the FrameNet paradigm).

TIGER XML describes trees with arbitrary node and edge labels and crossing edges. It has been designed with the express purpose of being able to encode different linguistic frameworks. Transformation filters exist for several corpus formats, e.g. Penn Treebank, SWITCHBOARD, Susanne, Christine, and Negra.

TIGER/SALSA XML extends TIGER XML by a second annotation level, again describing trees with node and edge labels and crossing edges. The leaves of these trees can be tree nodes in the syntactic level or in this second level. As there is no restriction on possible node and edge labels (i.e. the frame and semantic role names), TIGER/SALSA XML can encode semantic roles that are verb-specific (as in PropBank), verb-group-specific (as in FrameNet), or general (as in the Prague Treebank).

To recode a range-based format like the ones of PropBank and FrameNet in TIGER/SALSA XML, we would refer to the words of the sentence (rather than character offsets) included in the target or FE via unique IDs. The example from Sec. 6. could be translated to

```
<frame>
 <target> <fenode idref="1242945_5/>
          <fenode idref="1242945_8/>
          <fenode idref="1242945_9/> </target>
</frame>
```

using sentence ID plus word index as node ID (*give the slip* are the 5th, 8th and 9th word of the sentence).

To encode Praguian tectogrammatical structure, where a single, deep tree describes the semantic roles for the whole sentence. To encode this structure in TIGER/SALSA XML, we can use the possibility of frame tree leaves to refer to roots of other frame trees (cf. Sec. 3.1).

It may even be interesting to encode in TIGER/SALSA XML other kinds of information besides semantic roles,

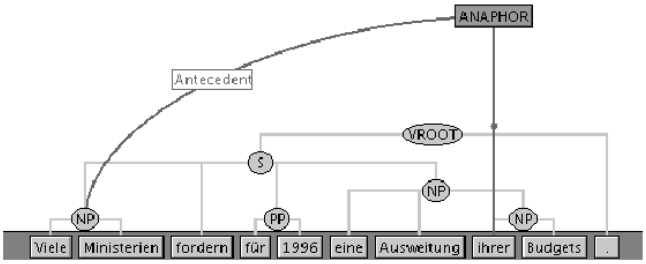


Figure 4: TIGER/SALSA XML for anaphora

since this makes it possible to use the SALSA Annotation Tool for the annotation tasks. The XML format offers, among other things, references beyond sentence boundaries and below word boundaries. So it could be used for example for coreference annotation, as shown in Fig. 4.

8. Conclusion

We have presented two XML formats for the representation of corpora with semantic role information. One, Text-SALSA XML is a lightweight format suitable for manual annotation on bare text. It can be automatically transformed into TIGER/SALSA XML, a powerful and highly modular format that can be efficiently processed and subsumes other formats for role-semantic annotation.

9. References

Brants, Sabine, Stefanie Dipper, Silvia Hansen, Wolfgang Lezius, and George Smith, 2002. The TIGER treebank. In *Proceedings of the Workshop on Treebanks and Linguistic Theories*. Sozopol.

Erk, Katrin, Andrea Kowalski, and Sebastian Pado, 2003a. The SALSA annotation tool. In Denys Duchier and Geert-Jan Kruijff (eds.), *Proceedings of the Workshop on Prospects and Advances in the Syntax/Semantics Interface*. Nancy, France.

Erk, Katrin, Andrea Kowalski, Sebastian Pado, and Manfred Pinkal, 2003b. Towards a resource for lexical semantics: A large German corpus with extensive semantic annotation. In *Proceedings of ACL-03*. Sapporo, Japan.

Hajičová, E., 1998. Prague Dependency Treebank: From Analytic to Tectogrammatical Annotation. In *Proceedings of TSD'98*. Brno, Czech Republic.

Johnson, C. R., C. J. Fillmore, M. R. L. Petruck, C. F. Baker, M. J. Ellsworth, J. Ruppenhofer, and E. J. Wood, 2002. FrameNet: Theory and Practice. `http://www.icsi.berkeley.edu/~framenet/book/book.html`.

Kilgarriff, Adam and Joseph Rosenzweig, 2000. Framework and results for English Senseval. *Computers and the Humanities*, 34(1-2).

Kingsbury, Paul, Martha Palmer, and Mitch Marcus, 2002. Adding semantic annotation to the Penn TreeBank. In *Proceedings of HLT*. San Diego.

Mengel, Andreas and Wolfgang Lezius, 2000. An XML-based encoding format for syntactically annotated corpora. In *Proceedings of LREC-2000*. Athens, Greece.