Representing and Querying Standoff XML

Stefanie Dipper*, Michael Götze*, Uwe Küssner†, Manfred Stede*

*Department of Linguistics
University of Potsdam
14415 Potsdam, Germany
{dipper,goetze,stede}
@ling.uni-potsdam.de

† Uwe Küssner IT-Consulting
Markgrafendamm 24 Haus 16
10245 Berlin, Germany
uwekuessner@web.de

Representing and Querying Standoff XML

In recent years, focus in corpus-based work has switched from corpora annotated by part-of-speech and syntactic annotations only (treebanks) to corpora that are annotated by properties beyond the (morpho-) syntactic level. Often this information is added to already-annotated corpora, which allows the user to combine the different types of annotation by posing cross-level queries and to search for putative interactions between different linguistic domains.

This scenario presupposes that the new information can be integrated into the corpus. That is, the representation of the corpus must be flexible and general enough to accommodate all kinds of annotations. At the same time, the format must support complex, cross-level and efficient querying.

In this abstract, we present our standoff XML format for data representation, which comes with various import filters for tool-specific formats (TIGER XML, RST Tool, MMAX2, Exmaralda) and export filters to statistical analysis (by WEKA) as well as our linguistic database. We discuss and evaluate our standoff format as well as an inline variant derived from this format, by testing their performance with regard to a testsuite of representative queries, and compare them to other generic XML-based representation formats.

1. XML Representations

Standoff Representation To integrate annotations from different sources, we developed a standoff XML format which uses generic elements and attributes: <mark> elements ('markables') denote units of annotations; <feat> elements, which are anchored to <mark> elements by means of XPointer expressions, specify the features that are annotated to the markables. <struct> elements specify structured markables, for representing trees or graphs.

The highly modularized nature and generality of our standoff representation allows us to incorporate all kinds of annotations and to successively supplement already-existing corpora with new layers. Moreover, it easily accommodates layers that define 'contradictory' information, such as overlapping segments, conflicting hierarchies, incompatible feature annotations of the same kind from different sources, etc.

However, the generality has its price: further processing of the data becomes expensive. Besides the standoff format, which serves as our interchange format, we therefore compute a supplementary internal *inline* representation to support efficient querying of the data.

Inline Representation In the inline version, all annotations referring to the same token or markable are collected and annotated as attributes of one element. Spans of markables and hierarchical structures are represented by embedding. To differentiate between mere alignment of spans vs. 'real' embedding structures such as trees, we introduce <_relations> elements to explicitly encode hierarchical embedding.

For the representation of overlapping segments and hierarchies, which cannot be represented in XML via embedding, we use the strategy of *fragmentation* (cf. Sperberg and Burnard (1994, ch.31), Barnard *et al.* (1995)): the 'less important' nesting element is broken into smaller units and an attribute '_gid' ('group id') is added to the fragmentation representation to explicitly mark elements that belong together.

2. Evaluation

For the evaluation, we chose one of the standard XML query languages, *XQuery*, and defined two test scenarios: in the first, the query expressions are processed by the XQuery processor *Saxon*, in the second by the native XML database *eXist*. Inspired by Bird *et al.* (2006), our testsuite consists of 7 queries, with queries involving hierarchical, pointing and overlapping relations of varying complexity, and a data set of 2300 sentences with multiple and overlapping annotations.

First results show that ‘simple’ queries are processed faster with the standoff format, whereas hierarchical queries perform better with the inline representation. Furthermore, query evaluation by *eXist* is considerably slower than by *Saxon*; however, we have not yet exploited all ways of optimization (like indexing) that are offered by *eXist*.

References

Barnard, D., Burnard, L., Gaspart, J.-P., Price, L. A., Sperberg-McQueen, C. M., and Varile, G. B. (1995). Hierarchical encoding of text: Technical problems and SGML solutions. *Text Encoding Initiative: Background and Context. Special Issue of Computers and the Humanities*, **29**(211–231).

Bird, S., Chen, Y., Davidson, S., Lee, H., and Zheng, Y. (2006). Designing and evaluating an xpath dialect for linguistic queries. In *Proceedings 22nd International Conference on Data Engineering (ICDE)*, Atlanta, USA.

Sperberg, C. M. and Burnard, L., editors (1994). *Guidelines for Electronic Text Encoding and Interchange (TEI P3)*. Text Encoding Initiative, Chicago, Oxford.