Using Logics for Querying Treebanks

Stephan Kepser
Collaborative Research Centre 441, Tübingen University, Germany
kepser@sfs.uni-tuebingen.de

The last fifteen years saw a steady rise in the availability of treebanks with syntactic annotation for many languages. The two most important treebanks for German are the Tiger treebank (Brants et al., 2002) with more than 50.000 sentences (as of December 2005) and the TüBa-D/Z (Telljohann et al., 2005) with more than 27.000 sentences (July 2006), both being based on newspaper texts and both providing complete syntactic analyses – although of differing style. Offering such rich information, treebanks have been gaining importance as sources of evidence for diverse linguistic inquiries. It is especially questions of the existence and frequency of complicated linguistic constructions that treebanks help answering. But such constructions have to be found, and treebanks of today's size prohibit manual inspection. Rather some kind of query mechanism is required here. And it would be desirable that a linguist may specify rather closely and exactly what type of construction she or he is looking for to retain small answer sets. It is the aim of this paper to demonstrate the usefulness of employing logics as a query language to search for involved linguistic constructions by means of an example.

The example we use is embedded verb-initial (V1) clauses in standard German which are not embedded yes/no-questions. The prototypical example is

(1) *Regnet es, bleiben wir zuhause.*
rains it, stay we at home.
'When/if it rains, we stay at home.'

Embedded V1-clauses which are not questions are relatively rare. Their standard linguistic interpretation is that of a conditional clause, but this interpretation has come under some doubt recently. There are also embedded V1-clause in sentence-final position, but these are even rarer to find. In order to determine the exact status of these embedded V1-clauses it would be desirable to have access to their use in practise and this means to find examples of such sentences in German treebanks.

We will stepwise show how to search for sentence-initial embedded V1-clauses in the Tiger and the TüBa-D/Z treebank using a logical query language. This language, basically first-order predicate logic, talks about nodes in a tree, their properties and relations. Properties are linguistic labels like VVFIN or NP or grammatical functions like HD for head. The individual labels depend of course of

the particular label set in use in the treebank. Node relations are dominance (denoted by $\lhd$) and left-to-right order (denoted by $\prec$). Now, the matrix clause can be described by two properties. It has a sentence node – a node labelled S – and a finite verb, labelled VVFIN, which is the head of the sentence, and hence dominated by the sentence node. These properties can be expressed directly as a query:

$$\exists x\, \mathsf{S}(x) \wedge \exists y\, \mathsf{VVFIN}(y) \wedge \mathsf{HD}(y) \wedge x \lhd y$$

The query for a V1-clause is similar to the one for a matrix clause, but we have to ensure that the verb is the initial element of the clause. That means there cannot be any node to the left of the verb and dominated by the S-node of the clause. Hence the query looks like this:

$$\exists x.\, \mathsf{S}(x) \wedge \exists y.\, \mathsf{VVFIN}(y) \wedge \mathsf{HD}(y) \wedge x \lhd y \wedge \neg \exists z.\, x \lhd z \wedge z \prec y$$

We now compose these two queries to search for embedded sentence-initial V1-clauses. If the V1-clause is sentence-initial it must be to the left of the finite verb of the matrix clause under the standard assumption of the matrix clause to be a V2-clause. This is expressed naturally in the following query:

$$\begin{aligned} &\exists a.\, \mathsf{S}(a) \wedge \exists b.\, \mathsf{VVFIN}(b) \wedge \mathsf{HD}(b) \wedge a \lhd b \wedge \\ &\quad \exists x.\, \mathsf{S}(x) \wedge a \lhd x \wedge x \prec b \wedge \\ &\qquad \exists y.\, \mathsf{VVFIN}(y) \wedge \mathsf{HD}(y) \wedge x \lhd y \wedge \neg \exists z.\, x \lhd z \wedge z \prec y \end{aligned}$$

Here, variable *a* represents the S-node of the matrix clause, *b* the finite verb of the matrix clause, *x* the embedded S-node of the V1-clause to the left of *b*, and *y* the finite verb of the V1-clause. Using a powerful query system that allows for logical queries (fsq, cf. Kepser, 2003) we applied the above query to both the Tiger and TüBa-D/Z treebank. There are 394 hits for this query in the Tiger treebank and 236 in the TüBa-D/Z. There are some perfect matches like

(2) *Glaubt man Plakaten, jagt neuerdings ein Großereignis das nächste.*
Believes one billboards, hunts lately one major event the next.
'If you believe the billboards, one major event follows the next one.'

But quite a proportion of the hits are mismatches falling mainly into three categories: embedded questions, conjunctions, and artefacts of the annotation. A further refinement of the query brings the number of hits down to 286 for the Tiger treebank and 165 for the TüBa-D/Z, most of which now being proper matches. Figures this low should allow for manual inspection. Furthermore it has to be said that a further refinement with the intend to yield even smaller answer sets is a difficult task now, because any further restriction is under thread of excluding proper matches, which is very undesirable considering how few proper matches exist. Linguists prefer slightly larger answer sets over the risk of loosing proper matches.

The aim of this example is to demonstrate the power and naturality of using logics to formulate treebank queries. We showed that the typical notions used by linguists can be translated rather straight forwardly into logical queries. And the power of the query language is shown by the relatively small number of hits for the queries, small enough for manual inspection. The choice of first-order logic is a natural one since it is a powerful logic, which is familiar to many linguists. Indeed there seems to have been no need for a more powerful query language so far, although contenders exist. The particular syntax for the logic chosen is on the other hand to some degree a matter of taste. There exists a variable-free purely path-based query language, namely Conditional LPath (Bird et al., 2005), which is proven to be equivalent to first-order logic (Lay, 2005). We therefore think richly annotated treebanks can indeed be a valuable source of evidence provided they are matched with powerful query systems. And we argued that logics can indeed be a natural and powerful query language for such a query system.

References

Bird, S., Y. Chen, S. Davidson, H. Lee, Y. Zheng (2005). Extending XPath to Support Linguistic Queries**.** In *Proceedings of Programming Language Technologies for XML (PLANX)*, pp 35-46.

Brants, S., S. Dipper, S. Hansen, W. Lezius, and G. Smith (2002). The TIGER Treebank. In *Proceedings of the Workshop on Treebanks and Linguistic Theories*. (http://www.ims.uni-stuttgart.de/projekte/TIGER/)

Kepser, S. (2003). Finite Structure Query - A Tool for Querying Syntactically Annotated Corpora. In *EACL 2003*, Ann Copestake and Jan Hajic (eds.), pp. 179-186. (http://tcl.sfs.uni-tuebingen.de/fsq)

Lai, C. (2005). *A Formal Framework for Linguistic Tree Query*. M.Sc. thesis, University of Melbourne, Australia.

Telljohann, H., E. Hinrichs, S. Kübler, and H. Zinsmeister (2005): *Stylebook for the Tübingen Treebank of Written German (TüBa-D/Z)*. Tübingen University. (http://www.sfs.uni-tuebingen.de/en_tuebadz.shtml)