A clustering method for tree family induction for Tree Adjoining Grammars

Matías Guzmán Naranjo Laura Kallmeyer Simon Petitjean

Introduction A Lexicalized Tree Adjoining Grammar (TAG, Aravind K. Joshi and Schabes, 1997) consists of lexicalized elmentary trees where each tree roughly represents a specific syntactic configuration for a predicate and its arguments (Fig. 1 gives some examples). Unanchored elementary trees that represent the same subcategorization frame are grouped into tree families (see Fig. 2). The anchor (◇) is where the lexical item gets inserted.

We present an approach to tree family induction for LTAG, i.e., an approach that clusters the unanchored elementary trees of a TAG into families. The motivation is twofold: On the one hand, this work investigates the nature of tree families in hand-written precision grammars in the context of TAG (Crabbé, 2005; XTAG Research Group, 2001), by identifying features that are relevant for clustering the elementary trees into their respective families. On the other hand, the approach can also be applied to grammars induced from treebanks (Bladier et al., 2018; Kaeshammer, 2012; Xia and Palmer, 2006). This can improve supertagging results for TAG (Bladier et al., 2018; Aravind K Joshi and Srinivas, 1994; Kasai et al., 2017), that suffer from the large number of subertags, and thereby it can improve data-driven TAG parsing. The approach can be applied to other frameworks involving constructions in a more general sense (CxG, Goldberg, 2013). It can for example feed into parsing with induced construction grammars (Dunn, 2017).

Method We use an unsupervised clustering method based on tree distances to induce groups of similar trees. We represent each tree as a vector, where each position represents the number of times a given node appears in the tree. Because not all nodes are equally relevant to the subcategorization frame of a tree, we add extra weight to more important nodes (identifying more relevant nodes is part of the induction process). We calculate the distance between two trees as the Chebyshev distance between the vector representations of said trees. Based on the calculated distances, we find clusters using Ward's method for hierarchical clustering. We cut the cluster tree at the height of the largest weight used.

Results We applied this method to the XTAG grammar (XTAG Research Group, 2001), a hand written TAG grammar of English with 1219 trees, across 67 families. Results from comparing this family induction approach with the families defined for the XTAG grammar are promising. Overall, with our method we found 118 clusters, most of them containing a single family. While some families got split across two or three clusters, this was relatively systematic. Table 1 shows partial results for the intransitive, transitive, and two ditransitive families. Columns represent the cluster trees were assigned to. As we can observe, most clusters only contain trees from one family, and most families have their trees assigned to only one or two clusters. Interestingly, both ditransitive families have identical distribution of trees across clusters 27 and 39, and 29 and 43, respectively. Similar results apply to other families. Applied to automatically induced TAGs, the approach also yields meaningful results, but so far we do not have a way of evaluating the quality of these results.

Overall, we show that it is possible to automatically induce meaningful clusters of trees, which strongly correlate with tree families. Beyond its possible uses for LTAG grammars, we expect this approach should be useful for inducing constructions in a similar manner.

	3	10	11	22	25	27	29	31	34	39	43	44	61	62	63
Tnx0V	1	0	0	11	0	0	0	0	0	0	0	0	0	0	0
Tnx0Vnx1	2	26	10	1	0	0	0	0	0	0	0	0	0	0	0
Tnx0Vnx2nx1	1	0	0	0	0	0	0	29	0	0	0	10	6	0	0
Tnx0Vnx1Pnx2	1	0	0	0	1	35	0	0	0	10	0	0	0	5	0
Tnx0Vnx1_pnx2	1	0	0	0	0	0	35	0	1	0	10	0	0	0	5

Table 1: Parial results family induction for the XTag grammar.

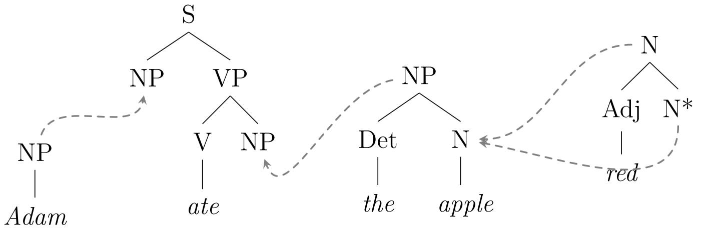

Figure 1: Sample LTAG elementary trees for *Adam ate the red apple* and their composition

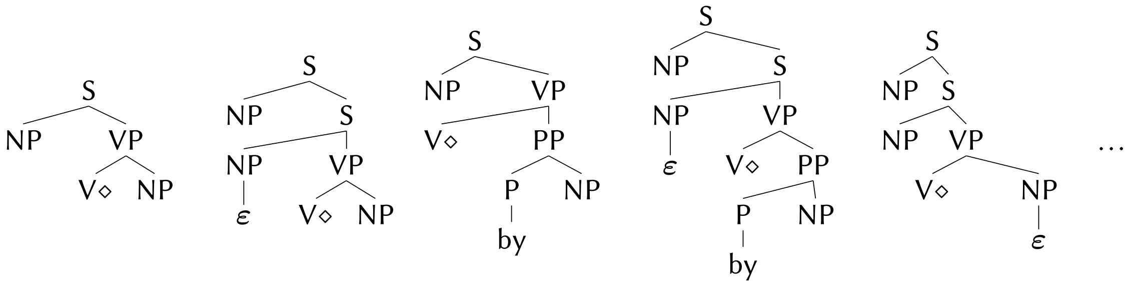

Figure 2: LTAG tree family for transitive verbs

References

Bladier, Tatiana et al. (2018). “German and French Neural Supertagging Experiments for LTAG Parsing”. In: *ACL Student Research Workshop*. Melbourne, Australia.

Crabbé, Benoit (2005). “Représentation informatique de grammaires d'arbres fortement lexicalisées : le cas de la grammaire d'arbres adjoints”. PhD thesis. Université Nancy 2.

Dunn, Jonathan (2017). “Computational learning of construction grammars”. In: *Language and Cognition* 9.2, pp. 254–292.

Goldberg, Adele (2013). “Constructionist Approaches”. In: pp. 15–31.

Joshi, Aravind K. and Yves Schabes (1997). “Tree-Adjoining Grammars”. In: *Handbook of Formal Languages. Vol. 3: Beyond Words*. Ed. by Grzegorz Rozenberg and Arto Salomaa. Berlin: Springer, pp. 69–123.

Joshi, Aravind K and Bangalore Srinivas (1994). “Disambiguation of super parts of speech (or supertags): Almost parsing”. In: *Proceedings of the 15th conference on Computational linguistics-Volume 1*. Association for Computational Linguistics, pp. 154–160.

Kaeshammer, Miriam (2012). “A German treebank and lexicon for tree-adjoining grammars”. PhD thesis. Master's thesis, Universität des Saarlandes, Saarlandes, Germany.

Kasai, Jungo et al. (2017). “Tag parsing with neural networks and vector representations of supertags”. In: *Proceedings of EMNLP*, pp. 1713–1723.

Xia, Fei and Martha Palmer (2006). “From Treebanks to Tree-Adjoining Grammars”. In: *Supertagging: using complex lexical descriptions in natural language processing*, pp. 1–39.

XTAG Research Group (2001). *A Lexicalized Tree Adjoining Grammar for English*. Tech. rep. Available from `ftp://ftp.cis.upenn.edu/pub/xtag/release-2.24.2001/tech-report.pdf`. Philadelphia: Institute for Research in Cognitive Science.