

Using Supertags and Encoded Annotation Principles for Improved Dependency to Phrase Structure Conversion

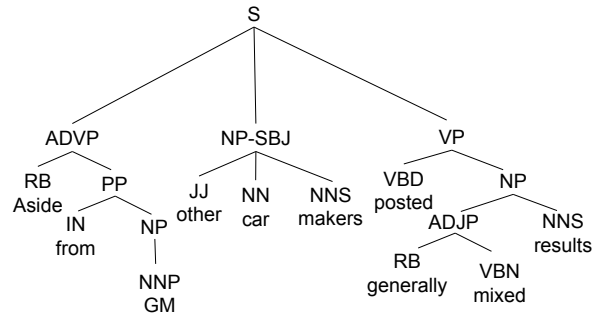
Seth Kulick, Ann Bies, Justin Mott {skulick,bies,jmott}@ldc.upenn.edu
Linguistic Data Consortium, University of Pennsylvania



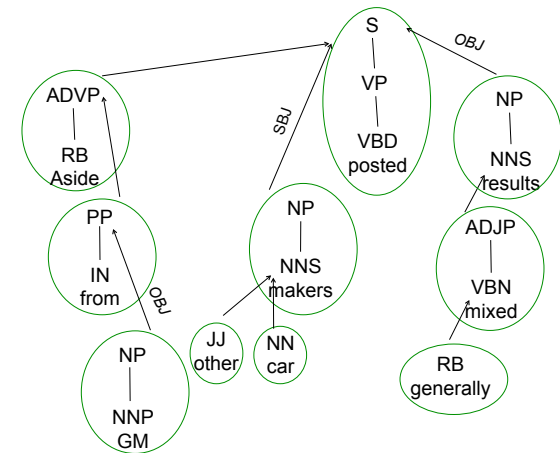
GOAL: CONVERT FROM DEPENDENCY STRUCTURE (DS) TO PHRASE STRUCTURE (PS)

- ◆ Phrase structure treebank construction – parser output as input for manual correction.
- ◆ To use dependency parsing, need to convert from DS parser output to PS for manual correction.
- ◆ Conflicting requirements: “minimal” DS for parser, but able to do high quality conversion to PS.
- ◆ Solution:
 - ◆ Tree Insertion Grammar (TIG) representation...
 - ◆ ... results in set of “supertags”...
 - ◆ ... allowing encoding of properties of guidelines.
- ◆ We start with Penn Treebank for development.
- ◆ Future goal: use this for Arabic Treebank construction.

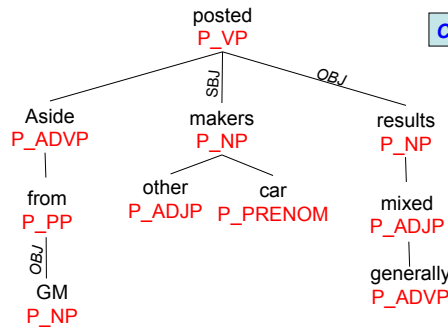
1. Penn Treebank Tree



2. Tree Insertion Grammar (TIG) Derivation Tree

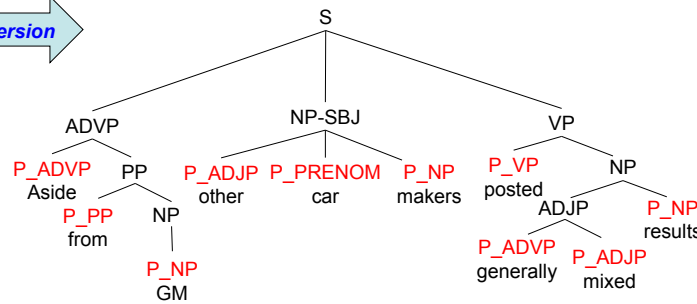


3. Dependency Structure



Conversion

4. Phrase Structure



Main Properties of Conversion from TIG to DS

- ◆ Conversion to DS results in set of 30 “supertags” replacing POS tags. (10 are for punctuation.)
- ◆ P_XP means projection of XP.
 - ◆ e.g. (ADVP Aside) -> P_ADVP supertag.
- ◆ “Minimal” tags not encoding entire structure.
 - ◆ e.g. (S (VP posted)) -> P_VP supertag, not P_S.
- ◆ 5 arc labels: SBJ, OBJ, PRN, COORD, APP.

Main Properties of Conversion from DS to PS

- ◆ Annotation principles encoded using supertags. e.g.:
 - ◆ P_NP projection reduces P_ADJP “other” to single word.
 - ◆ P_PP projection does not reduce P_NP complement.
- ◆ Supertags disambiguated by dependency structure. e.g.:
 - ◆ P_VP -> (S ...) or (SINV ...) or (SQ ...) or (SBAR ...) or (VP ...).
 - ◆ P_QP -> (QP ...) or (NP ...) or (QP (NP ...)) or (QP (ADJP ...)).
- ◆ Conversion to PS can be run in two ways:
 - ◆ USE-SUPER – use supertags.
 - ◆ USE-POS – use the POS tags to approximate the supertags.

EVALUATION

- ◆ Evaluation: (Xia et al. '09) – convert to DS for test data, algorithm converts back to PS, score with evalb.
- ◆ We also use evalb, but it is inadequate for the full problem (empty categories).
- ◆ Improvement over previous work (Xia et al. '09).
- ◆ (Wang & Zong '10) - gold dependency tree with parser.
 - ◆ Our interpretation – parser is providing roughly same information as in our conversion code, along with learning non-straightforward cases of POS tags.

Sec	System	rec	prec	f
22	Xia et al. '09	90.7	88.1	89.4
	USE-POS	95.0	95.5	95.3
	USE-SUPER	96.4	97.1	96.7
23	Wang & Zong '10	95.9	96.3	96.1
	USE-POS	94.8	95.7	95.3
	USE-SUPER	96.2	97.3	96.7
24 dev	USE-POS	94.0	94.7	94.4
	USE-SUPER	95.9	97.1	96.5