MWE parsing as deduction

Peter Juel Henrichsen
DanCAST - Danish Center for Applied Speech Technology
Copenhagen Business School
pjh.ibc@cbs.dk

Abstract

This small paper is an off-spin of an ongoing project Computational Models of First Language Acquisition, a topic perhaps not usually associated with MWE parsing, however sharing some basic challenges. We thus present our learning algorithm GraSp in a cross-disciplinary spirit. GraSp, a variant of the Gentzen-Lambek calculus, was designed for inducing grammars from unlabeled transcripts of spontaneous speech rich in 'difficult' (discontinuous or incomplete) grammatical dependences. We here present GraSp in a new approach to the recognition and clustering of MWEs with slightly deviating surface forms with non-constituent words, non-paradigmatic inflexions, and meaning-preserving word order variation. Most examples are in Danish, but our observations generalize to all languages.

1. The MWE parsing challenge

A Danish language user will immediately recognize the expressions *a-d* as instances of the same lexical idiom.

a. der er ikke noget at rafle om der
b. der er ikke så meget at rafle om
c. der var sgu ikke noget som helst og rafle om
d. (..) at der ikke er noget at rafle om

They all refer to the lexical MWE "der er ikke noget at/og rafle om" found in any standard dictionary (word-by-word translation: *there is not anything to cast-dice about*, meaning: *this is not a subject of negotiations*), with a few largely meaning-preserving transformations in the form of non-constituent words (2nd 'der' in *a*; 'sgu' and 'som helst' in *c*), synonym replacements ('så meget' for 'noget' in *b*; 'og' for 'at' in *c*), inflections ('var'$_{\text{PAST}}$ for 'er'$_{\text{PRES}}$ in *c*), and reorderings ('der ikke er' for 'der er ikke' in *d*). Even though the identity of the idiom is clearly preserved in *a-d*, this fact is not easily grasped by an automatic parser. Indeed, MWE recognition is a non-trivial task not least due to superficial variations in the text data obscuring the identity of the underlying idioms.

2. The learning algorithm

By nature, a first language (L1) learning algorithm must be largely data-driven since to a newborn learner *all* idioms (and other lexical patterns) are unknown. We needed to design a 'humanoid' L1 parser robust enough to accept input strings with little or no hints of lexical structure (for the early stages of a learning session) while at the same time retaining the discriminating powers of a context-free parser (for the later stages). The learning algorithm GraSp ("**Gra**mmar of **Sp**eech") was one result.

GraSp belongs to the Lambek-Gentzen family of deductive systems (Morrill 2010). The core rules of GraSp are identical to the Lambek calculus except that antecedents may be empty (see fig. 1) while the non-classical addendum keeps non-recognized constituents parsable. More specifically, the Lambek rules (**\l**, **/l**, **\r, /r, *l, *r**) capture input parts interpretable as context-free constituents while **σl** and **σr** allow the parser to ignore currently indigestible parts.

A GraSp learning session begins from a state of complete grammatical ignorance. Each word type occurring in a training corpus *I* is mapped in the initial lexicon to an arbitrary unique category label (e.g. c_{12} or c_{987}). GraSp is then faced with the task of building lexical structure by changing the category assignments in the lexicon based on word order observations in *I* as governed by a notion of structural entropy. Henrichsen (2002) has more formal definitions and details.

2.1 GraSp at work - an example

Consider a typical spoken language utterance.

A: *right so let's er- let's let's look at the suggestions*

In the early stages of a learning session, *A* will map to a sequent A_{seq} of *basic* categories (c_n).

A_{seq}: $c_{29}\ c_{22}\ c_{81}\ c_5\ c_{81}\ c_{81}\ c_{215}\ c_{10}\ c_1\ c_{891} \Rightarrow c_0$

Label c_0 serves as the top symbol (like the S in rewrite grammars).

Classical part

$$\frac{}{\sigma \Rightarrow \sigma}\ \textbf{link}$$

$$\frac{\Delta_B \Rightarrow B \quad \Delta_1 A\, \Delta_2 \Rightarrow C}{\Delta_1\ A/B\ \Delta_B\ \Delta_2 \Rightarrow C}\ /\textbf{l} \qquad \frac{\Delta\, B \Rightarrow A}{\Delta \Rightarrow A/B}\ /\textbf{r}$$

$$\frac{\Delta_B \Rightarrow B \quad \Delta_1\ A\ \Delta_2 \Rightarrow C}{\Delta_1\, \Delta_B\ B\backslash A\ \Delta_2 \Rightarrow C}\ \backslash\textbf{l} \qquad \frac{B\, \Delta \Rightarrow A}{\Delta \Rightarrow B\backslash A}\ \backslash\textbf{r}$$

$$\frac{\Delta_1 A\ B\ \Delta_2 \Rightarrow C}{\Delta_1\ A*B\ \Delta_2 \Rightarrow C}\ *\textbf{l} \qquad \frac{\Delta_1 \Rightarrow A \quad \Delta_2 \Rightarrow B}{\Delta_1\ \Delta_2 \Rightarrow A*B}\ *\textbf{r}$$

A, B, C are categories; $\Delta_{(x)}$ are (possibly empty) strings of categories.

Non-classical part

$$\frac{\sigma^+\ \Delta_1\Delta_2\ \Rightarrow C}{\Delta_1\,\sigma\ \Delta_2\ \Rightarrow\ C}\ \sigma\textbf{l} \qquad \frac{\sigma^-}{\Rightarrow \sigma}\ \sigma\textbf{r}$$

σ is a basic category. $^{+\,-}$ show polarity of residuals.

Figure 1. The GraSp sequent rules.

Since A_{seq} shows no hints of constituency, the sequent is proved (i.e. parsed) by recursive calls of the non-structural rule **σl** (NB: read sequent proofs from bottom to top).

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{c_{81}^{+}}\quad \dfrac{}{c_{215}^{+}}\quad \dfrac{\dfrac{}{c_{10}^{+}}\ \dfrac{}{c_1^{+}}\ \dfrac{}{c_{891}^{+}}\quad \dfrac{}{c_0^{-}}}{c_{10}^{+}\ c_1^{+}\ c_{891}^{+} \Rightarrow c_0}\,\sigma\textbf{r}}{c_{215}^{+}\ c_{10}\ c_1\ c_{891} \Rightarrow c_0}\,\sigma\textbf{l}\ \ \ldots}{c_{81}^{+}\ c_{215}\ c_{10}\ c_1\ c_{891} \Rightarrow c_0}\,\sigma\textbf{l}}{(\ldots)\ \ c_{81}\ c_{215}\ c_{10}\ c_1\ c_{891} \Rightarrow c_0}\,\sigma\textbf{l}}{A_{seq}:}$$

In later stages, more lexical structure will have developed exemplified in A'_{seq} with e.g. lexeme 'look' now migrated to category $c_{215}/(c_{10}*c_{891})$.

$$\dfrac{\dfrac{\dfrac{}{c_1 \Rightarrow c_1}\,\textbf{link}\quad \dfrac{\dfrac{}{c_{10}\Rightarrow c_{10}}\,\textbf{link}\quad \dfrac{}{c_{891}\Rightarrow c_{891}}\,\textbf{link}}{c_{10}\ c_{891}\Rightarrow c_{10}*c_{891}}\,*\textbf{r}\quad \dfrac{}{c_{81}^{+}c_{215}^{+}c_0^{-}}\,\sigma\textbf{x}}{c_{81}\ c_{215}/(c_{10}*c_{891})\ \ c_{10}\ \ c_{891} \Rightarrow c_0}\,/\textbf{l}}{(\ldots)\ \ c_{81}\ \ c_{215}/(c_{10}*c_{891})\ \ c_{10}\ \ c_1\ \ c_1\backslash c_{891} \Rightarrow c_0}\,\backslash\textbf{l}}{A'_{seq}:}$$

In contrast to A_{seq} , this A'_{seq} proof contains three **link**s reducing the entropy wrt. A by 3 degrees.

Learning in GraSp, then, amounts to developing the lexical categories in pursuit of a steady decrease in (global) structural entropy.

3. GraSp as a MWE parser

Now reconsider the four expressions *a-d* (quoted from the Danish spoken language corpus BySoc, http://bysoc.dyndns.org). Using BySoc as training material, GraSp develops a highly complex category for the verb 'rafle'.

'rafle': ((c12\(c22\(**c8**\(c5\(c7\c5808))))/c7)/c42

This category reflects the occurrences of 'rafle' in contexts such as *a-d.* For example, the statistical fact that the word 'ikke' (category **c8**) occurs more often than not in the left context of 'rafle' in BySoc is reflected directly in the 'rafle' category; similarly for the types 'der' (c7), 'er' (c5), 'noget' (c22), 'meget' (c22), 'at' (c12), 'og' (c12) and 'om' (c42). The minimal context (*MinCon*) motivating the full 'rafle' category is now easily derived from the trained lexicon.

MinCon('rafle') =
- der - er - ikke - (noget|meget) - (at|og) - rafle - om

Observe that *MinCon*('rafle') is an almost verbatim image of the dictionary form. Literally hundreds of such MWE-like idioms can be derived from the final GraSp'ed lexicon by related reasoning. More examples include:

- det - kan - man - ikke - fortænke - <PRO> - i -
- det - (vil|ville) - <PRO> - blæse - på -
- (har|havde) - ikke - en - kinamands - chance -

4. Concluding remarks

There are of course simpler and faster algorithms than GraSp available for extracting MWEs from large corpora. Many parsers are however vulnerable to superficial variation in word order, synonym selection, etc. GraSp-parsing offers robustness against 'noisy' data such as spoken language transcripts, hastily produced blog posts, sms, email, poor translations, and soforth. With its advanced inference engine, GraSp is able to induce idiom templates from corpora without even a single verbatim occurrence, a property that might come in handy when parsing MWEs.

References

Morrill, G. (2010) Categorial Grammar: Logical syntax, semantics, and processing. Oxford Uni. Press

Boonkwan, P.; T. Supnithi (2008) Memory-Inductive Categorial Grammar: an approach to gap resolution in analytic-language translation; IJCNLP08

Henrichsen, P.J. (2002) Grammar learning from unlabeled speech corpora"; CoNLL02