Lexical Disambiguation with Polarities and Automata

Guillaume Bonfante, Joseph Le Roux, and Guy Perrier

Universités de Nancy

Abstract. We propose a method for lexical disambiguation based on polarities for Interaction Grammars (IGs), well suited for coordination.

1 Introduction

We deal with lexical disambiguation using lexicalized IGs[2]. An IG is defined by a lexicon which associates to every word a set of lexical items specifying its grammatical behaviors. The number of lexical selections for a sentence is the product of the number of lexical entries for each word.

Lexical items are polarized. They may be seen as bags of polarized features and this simplification, as an abstraction. In the abstract grammar, parsing amounts to counting polarities and we use it to filter the initial grammar because of a homomorphism, presented in [1], from the initial to the abstract grammar: every parse in the former is transposed in a parse in the latter.

2 Interaction Grammars

IGs are based on *underspecification*, expressed by using *tree descriptions* rather than trees, and *polarities*. Polarized features decorating nodes express valences: positive (resp. negative) features represent available (resp. expected) resources. Syntactic composition consists of superposing tree descriptions while respecting polarities: a negative feature must encounter a dual positive feature to be neutralized. A feature is a triple (f, p, v) such that f is a feature name taken from $\mathcal{F}$, v is a finite disjunction $(v_1 | \dots | v_n)$ of atoms and p is a polarity from $\{\rightarrow, \leftarrow, =\}$.

3 Polarity Automata

We first need a function p_D that count polarities in a description for particular f and v. We assign +1 to $\rightarrow$, -1 to $\leftarrow$ and 0 to $=$ or if a the value is not present in a description. Feature values being disjunctions, this function returns the set of all possible countings. It can be shown that it is a $\mathbb{Z}$ interval.

Let $w_1 \dots w_n$ be a sentence to parse with an IG G given by its lexicon Lex_G. For each word w_i we know $Lex_G(w_i) = \{D_{i,1} \dots D_{i,k_i}\}$. A *lexical selection* is a sequence $S = D_{1,s_1} \dots D_{n,s_n}$, where $D_{i,s_i} \in Lex_G(w_i)$. We extend function p to selections as the sum of p_D for all D in S.

Here is the a *global neutrality criterion* (GNC) verified by valid selections: if a selection S is valid, for every f and v then $0 \in p_S(f, v)$.

O.H. Ibarra and H.-C. Yen (Eds.): CIAA 2006, LNCS 4094, pp. 283–284, 2006.

Polarity Automata. For any $f \in \mathcal{F}$ and value v, the automaton $A(f,v)$ is defined as follows. States are pairs (i,p), where i represents the position between w_i and w_{i+1} and p is an interval of $\mathbb{Z}$ which represents the counting of polarities. Transitions have the form $(i,p) \stackrel{D_{i+1,s_k}}{\longrightarrow} (i+1,q)$, where q is a $\mathbb{Z}$ interval of the sums of any element of p added to any element of $p_{D_{i+1,s_k}}(f,v)$. The initial state is $(0,\{0\})$ and accepting states are (n,p) such that $0 \in p$.

A lexical selection accepted by $A(f,v)$ verifies GNC. Hence, the intersection of polarity automata contains the good solutions. Furthermore, a (bad) lexical selection not contained in all initial automata will disappear from the intersection. Actually, this process pursues the filtering.

Selection of Feature Values. If a value does not appear with an active ($\rightarrow$ or $\leftarrow$) polarity in any description, the automaton will not filter. So, the first optimization is to consider only values with active polarity within some descriptions.

Then, the size of the automaton depends on the choice of the value. If $v \subseteq v'$ the automaton for v will be larger than the one for v'. So we order feature values.

Let us pay attention to maximal values for that order. If $v \cap v' \neq \emptyset$ then $A(f, v \cup v')$ may be smaller than $A(f,v)$ and $A(f,v')$. As a conclusion, we add any value $v1 \cup v2$ such that $v_1 \cap v_2 \neq \emptyset$ until we reach a fix point.

Refinement. Coordination shows so much ambiguity that GNC is not sufficient but we can take advantage of the syntactic modelisation. Two conjoinable segments must be on the left and on the right of a coordination and have the same active polarities. We can show that if D_i is associated with a coordination for two segments between position h and j then we have the following invariants: $\sum_{n=1}^{i} p_{D_n}(f,v) = \sum_{n=1}^{h-1} p_{D_n}(f,v)$ and $\sum_{n=1}^{j} p_{D_n}(f,v) = \sum_{n=1}^{i-1} p_{D_n}(f,v)$.

Our invariants can be applied on states. For every transition t labelled with a coordination from (i,p) to $(i+1,q)$ in $A(f,v)$ we check that: (1) there exists (h,q) in the path from the initial state to (i,p) and (2) there exists (k,p) in the path from $(i+1,q)$ to a final state. If these states cannot be found, the transition t should be removed.

4 Conclusion

We presented a symbolic method for lexical selection. We used IGs but this method can be extended to other formalisms, see [1]. We also go beyond a simple counting of polarities by incorporating syntactical information for coordination.

References

1. G. Bonfante, B. Guillaume, and G. Perrier. Polarization and abstraction of grammatical formalisms as methods for lexical disambiguation. In *20th Conference on Computational Linguistics, CoLing'2004, Genève, Switzerland*, pages 303–309, 2004.
2. G. Perrier. La sémantique dans les grammaires d'interaction. *Traitement Automatique des Langues*, 45(3):123–144, 2004.