Pairing Model-Theoretic Syntax and Semantic Network for Writing Assistance

Jean-Philippe Prost and Mathieu Lafourcade

LIRMM, Université Montpellier 2

CSLP@Context, 2011
The Problem

Syntax/Semantics interface for Property Grammar (PG) through writing assistance

Example

*L’avocat le dossier de son client (The lawyer his client’s file)

- Case of (likely) missing word:
  L’avocat X le dossier de son client
  where X is of category V

- Expected surface realisation: plaide (pleads)
Outline

1. Introduction and Background
2. Error Detection
3. Re-generation
4. Surface Realisation by Network Exploration
   - The Lexical Network
   - Completion Message
   - Propagation Algorithm
5. Conclusion and Perspectives
Sketch of the Process

1. Error detection with approximated parse(s)
2. (unrealised) Re-generation by tree transduction
3. Surface realisation with lexical network
   - choice of functional and semantic roles
   - completion message
   - propagation in the network
Property Grammar (Blache, 2001)
Model-Theoretic Semantics for PG (Duchier et al., 2009)

- Models are labelled trees

- The grammar is a constraint system over tree nodes

\[
NP : D \prec N \\
VP : \triangle V \\
NP : N \Rightarrow D, \ldots
\]
Model-Theoretic Semantics for PG

- Strong semantics (i.e. well-formedness)

\[ \tau : \sigma \models G \]

a syntax tree \( \tau \) is a strong model of property grammar \( G \), with realization \( \sigma \), iff it is admissible and \( R_\tau(\varepsilon) = \sigma \) and \( I_{G,\tau}^- = \emptyset \)
Model-Theoretic Semantics for PG

- Loose semantics

\[
F_{G,\tau} = I_{G,\tau}^+ / I_{G,\tau}^0
\]

fitness

\[
\tau : \sigma \models G \quad \text{iff} \quad \tau \in \text{argmax}(F_{G,\tau'}) \quad \tau' \in A_{G,\sigma}
\]

loose models
Use of robust parsers’s combined output as set of models

Most robust parsers are not capable of deciding about the well-formedness of the input sentence
Characterising Approximated Parses

Characterisation of a model: set of pairs (pertinent instance of property, truth value)

\[ I^0_{G,\tau} = \{ r \in I_\tau[G] \mid P_\tau(r) \} \]
\[ I^+_{G,\tau} = \{ r \in I^0_{G,\tau} \mid S_\tau(r) \} \]
\[ I^-_{G,\tau} = \{ r \in I^0_{G,\tau} \mid \neg S_\tau(r) \} \]
Re-generation by Tree Transduction

- Basic tree operations, where $\tau$ is a tree, and $c, c_1, c_2$ are node labels (i.e. categories):
  - *Node insertion*, denoted by $\tau \downarrow c$
  - *Node deletion*, denoted by $\tau \uparrow c$
  - *Node permutation*, denoted by $c_1 \leftrightarrow c_2$
- Transduction:

<table>
<thead>
<tr>
<th>Property</th>
<th>Violated instances</th>
<th>Tree operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirement</td>
<td>$\mathcal{I}_\tau [c_0 : c_1 \Rightarrow s_2]$</td>
<td>$\tau \downarrow s_2$</td>
</tr>
<tr>
<td>Obligation</td>
<td>$\mathcal{I}_\tau [c_0 : \triangle c_1]$</td>
<td>$\tau \downarrow c_1$</td>
</tr>
<tr>
<td>Linearity</td>
<td>$\mathcal{I}_\tau [c_0 : c_1 \prec c_2]$</td>
<td>$c_1 \leftrightarrow c_2$</td>
</tr>
<tr>
<td>Uniqueness</td>
<td>$\mathcal{I}_\tau [c_0 : c_1 !]$</td>
<td>$\tau \uparrow c_1$</td>
</tr>
</tbody>
</table>
| Exclusion    | $\mathcal{I}_\tau [c_0 : c_1 \not
ddagger c_2]$ | $\tau \uparrow c_1 \cup \tau \uparrow c_2$ |
Les employés rendent rapport complète très complet.

The employees deliver report complete very complete.
The Lexical Network

- Nodes and directed relations
- Weights and types

**Example**

<table>
<thead>
<tr>
<th>Relation Type</th>
<th>Node 1</th>
<th>Node 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>isa</td>
<td>cat</td>
<td>animal</td>
</tr>
<tr>
<td>loc</td>
<td>cat</td>
<td>sofa</td>
</tr>
<tr>
<td>can</td>
<td>cat</td>
<td>pur</td>
</tr>
<tr>
<td>part</td>
<td>cat</td>
<td>claw</td>
</tr>
</tbody>
</table>

- Many relation types including semantic roles
  - AGENT, PATIENT, INSTRUMENT
- Other relations
  - Typical location, manner, entailment
  - isa, partof, substance, synonym, antonym, ...
A game for building the network

JeuxDeMots

http://jeuxdemots.org/

- Users associate terms given a relation
- A popular consensus filtered by pairs of players
- In 4 years time
  - over 1,200,000 relations among 230,000 terms
  - evaluation through a guessing game (AKI, *tip-of-the-tongue*)
    - term found in more than 75% of cases
    - while the typical human score is around 46%
DONNER DES IDEES ASSOCIEES AU TERME QUI SUIVIT :

grippe aviaire

Dernier terme proposé : épidémie
Ce terme a plusieurs sens ou il en manque ? Demandez de l'aide à vos amis
Donner des idées associées au terme qui suit :

**grippe aviaire**

Réponses données par kaput : poulet (volaille) - poulet - H1N1 - oiseau - oiseau (animal) - grippe - maladie - épidémie

Réponses données par BackInBusiness : fièvre - tousser - Asie - H1N1 - AH1N1 - grippe

AH1N1 - oiseau (animal) - volaille - grippe espagnole - virus - grippe

H1N1 - oiseau (animal) 🦅 - grippe

Vous gagnez 241 crédits et 5 point(s) d'honneur.

J-P. Prost and M. Lafourcade
Pairing MTS and Semantic Network
Completion Message

\[ \langle a, :R, b \rangle \]

:R denotes an oriented semantic relation, and a and b its oriented elements.
Gathering Functional and Semantic Roles

- NP2 is in a Patient relationship with V
- NP2’s head is inherited from NP3’s: dossier (file)

Message $\#1 = \langle X, :PAT, dossier \rangle$
Gathering Functional and Semantic Roles

- NP1 is in an Agent relationship with VP
- by inheritance, NP1’s head is *avocat* (*lawyer*)

Message $\#2 = \langle avocat, :AGT, X \rangle$
Completion Messages

\{\langle X, :PAT, dossier \rangle, \langle avocat, :AGT, X \rangle \}
The Problem

Given an underspecified input, such as
\langle avocat, :AGT, X \rangle and \langle X, :PAT, dossier \rangle
can \( X \) be filled in?

- Propagation in the lexical network
  - iterative and globally convergent (3 runs)
  - what is the most activated node for \( X \)?
    - plaider, . . . , étudier
  - what are the most activated nodes for the constraints?
    - avocat \( \rightarrow \) avocat > justice
    - dossier \( \rightarrow \) dossier > affaire
Conclusion and Perspectives

- Near-lack of Syntax/Semantics interface in PG prevents its use for deep processing
- Yet the PG formal properties are well-suited to address Grammar Checking problems
- The linguistic information provided by PG properties (i.e. characterisation) allows building a detailed information structure about non-canonical sentences
- Work in progress: completion of a characterised syntax tree through the exploration of a semantic network.