Integrating Syntax with Semantics using a Psycholinguistically motivated TAG

Ioannis Konstas

Joint work with Frank Keller, Mirella Lapata and Vera Demberg

Institute for Language, Cognition and Computation
University of Edinburgh

March 2014
Text and speech are perceived sequentially.

Sentence comprehension proceeds incrementally:

- the interpretation of a sentence is built word by word;
- each new word is integrated as fully as possible into a representation of the sentence thus far;
- processing effort depends on the properties of the word and its relationship to the preceding context.
Not only is processing word-by-word, it is also predictive:

- comprehenders anticipate upcoming linguistic material;
- and thus have more time to keep up with the input and compensate for noise or ambiguity.

van Berkum et al. (2005) show that contextual information is used to predict lexical items; processing difficulty arises if input is incompatible with prediction (ERP study).
Syntactic Prediction

Words that are syntactically compatible with prior context are named more quickly, even when they are not semantically related to the context (Wright & Garrett, 1984)

Words that are **syntactically compatible** with prior context are named more quickly, even when they are not semantically related to the context (Wright & Garrett, 1984).

Staub & Clifton (2006) show that the sentence processor can also make **structural predictions**:

1. Peter read *either* a book or an essay in the school magazine.
2. Peter read *a* book or an essay in the school magazine.

The presence of *either* leads to shorter reading times on *or* and on the NP that follows it (eye-tracking study).
Semantic Prediction

A word that is preceded by a *semantically related* prime word or a semantically congruous sentence fragment is processed faster (e.g., Stanovich & West 1981).
A word that is preceded by a semantically related prime word or a semantically congruous sentence fragment is processed faster (e.g., Stanovich & West 1981).

Altmann & Kamide (1999) use the visual paradigm to provide evidence for semantic prediction. They presented sentences such as:

(1) The boy will eat . . .
(2) The boy will move . . .

Together with a scene that contained one edible but several movable objects.
When participants heard *eat*, they looked more at the cake. Evidence for prediction induced by semantic restrictions of the verb.
Challenge: develop a model of prediction in sentence processing that accounts for syntactic prediction. Assumptions:
Challenge: develop a model of prediction in sentence processing that accounts for **syntactic prediction**. Assumptions:

- structures are built incrementally (word by word);
Modelling Syntactic Prediction

**Challenge:** develop a model of prediction in sentence processing that accounts for syntactic prediction. Assumptions:

- structures are built incrementally (word by word);
- partial structures do not contain unconnected nodes;
Challenge: develop a model of prediction in sentence processing that accounts for syntactic prediction. Assumptions:

- structures are built incrementally (word by word);
- partial structures do not contain unconnected nodes;
- upcoming syntactic material is predicted.
Challenge: develop a model of prediction in sentence processing that accounts for *syntactic prediction*. Assumptions:

- structures are built incrementally (word by word);
- partial structures do not contain unconnected nodes;
- upcoming syntactic material is predicted.

Evidence for *connectedness*: Sturt & Lombardo (2005). Existing incremental parsers don’t build fully connected structures.

Our approach: devise a *grammar formalism* that supports incrementality and connectedness; prediction then follows.
We propose Psycholinguistically Motivated TAG (PLTAG), a variant of tree-adjoining grammar (Demberg et al., 2014):
We propose Psycholinguistically Motivated TAG (PLTAG), a variant of tree-adjoining grammar (Demberg et al., 2014):

- in standard TAG, the lexicon consists of initial trees and auxiliary trees (both are lexicalized);
We propose Psycholinguistically Motivated TAG (PLTAG), a variant of tree-adjoining grammar (Demberg et al., 2014):

- in standard TAG, the lexicon consists of initial trees and auxiliary trees (both are lexicalized);
- we add unlexicalized predictive trees to achieve connectivity;
We propose Psycholinguistically Motivated TAG (PLTAG), a variant of tree-adjoining grammar (Demberg et al., 2014):

- in standard TAG, the lexicon consists of initial trees and auxiliary trees (both are lexicalized);
- we add unlexicalized predictive trees to achieve connectivity;
- the standard TAG operations are substitution and adjunction;
We propose Psycholinguistically Motivated TAG (PLTAG), a variant of tree-adjoining grammar (Demberg et al., 2014):

- in standard TAG, the lexicon consists of initial trees and auxiliary trees (both are lexicalized);
- we add unlexicalized predictive trees to achieve connectivity;
- the standard TAG operations are substitution and adjunction;
- we add verification to verify predictive trees;
We propose Psycholinguistically Motivated TAG (PLTAG), a variant of tree-adjoining grammar (Demberg et al., 2014):

- in standard TAG, the lexicon consists of initial trees and auxiliary trees (both are lexicalized);
- we add unlexicalized predictive trees to achieve connectivity;
- the standard TAG operations are substitution and adjunction;
- we add verification to verify predictive trees;

PLTAG supports parsing with incremental, fully connected structures.
Lexicon:
- Standard TAG lexicon
- Predictive lexicon (PLTAG)

Operations:
- Substitution
- Adjunction
- Verification (PLTAG)
Lexicon:
- Standard TAG lexicon
- Predictive lexicon (PLTAG)

Operations:
- Substitution
- Adjunction
- Verification (PLTAG)

Example:

*Initial Tree:*
```
NP
   ↓
VP
```
```
Peter
```
```
S
```
```
NP
```
```
VP
```
```
sleeps
```

*Auxiliary Tree:*
```
VP
```
```
AP
```
```
often
```
```
VP*
```
Modelling Syntactic Prediction

PLTAG

Lexicon:
- Standard TAG lexicon
- Predictive lexicon (PLTAG)

Operations:
- Substitution
- Adjunction
- Verification (PLTAG)

Example:

```
NP
Peter

\[\downarrow\]

VP
sleeps

\[\downarrow\]

S

\[NP\]

\[\downarrow\]

\[VP\]

\[\downarrow\]

\[sleeps\]

resulting in

```

NP
Peter

\[\downarrow\]

VP
sleeps

\[\downarrow\]

S
Lexicon:
- Standard TAG lexicon
- Predictive lexicon (PLTAG)

Operations:
- Substitution
- Adjunction
- Verification (PLTAG)
PLTAG

Lexicon:
- Standard TAG lexicon
- Predictive lexicon (PLTAG)

Operations:
- Substitution
- Adjunction
- Verification (PLTAG)

Example

Prediction Tree:

\[ S_k \]
\[ NP^k \downarrow \]
\[ VP^k \]

Index \( k \) marks predicted node.
Lexicon:
- Standard TAG lexicon
- Predictive lexicon (PLTAG)

Operations:
- Substitution
- Adjunction
- Verification (PLTAG)
Comparison with TAG

TAG derivations are not always incremental.

Example

NP ↓ S VP
  sleeps  subst  NP Peter VP sleeps adj NP Peter VP
S VP
sleeps

NP Peter VP

S AP VP
often sleeps
Comparison with TAG

PLTAG derivation are always incremental and fully connected.

Example

```
NP  | subst  NP1 | VP1  | S1  | adj  NP1 | VP1  | S1  | verif  NP  
  Peter  |       Peter |       |       |       |       |       |       Peter 
          |       |       |       |       |       |       |       often 
          |       |       |       |       |       |       |       VP1  
          |       |       |       |       |       |       |       sleeps
```
In order to construct an incremental parser for PLTAG, we need to:

1. convert the Penn Treebank into PLTAG format;
In order to construct an incremental parser for PLTAG, we need to:

1. convert the Penn Treebank into PLTAG format;
2. induce a lexicon from it;
In order to construct an incremental parser for PLTAG, we need to:

1. **convert** the Penn Treebank into PLTAG format;
2. induce a **lexicon** from it;
3. develop an incremental **parsing algorithm**;
In order to construct an incremental parser for PLTAG, we need to:

1. convert the Penn Treebank into PLTAG format;
2. induce a lexicon from it;
3. develop an incremental parsing algorithm;
4. devise a probability model;
In order to construct an incremental parser for PLTAG, we need to:

1. **convert** the Penn Treebank into PLTAG format;
2. **induce** a **lexicon** from it;
3. **develop** an incremental **parsing algorithm**;
4. **devise** a **probability model**;
5. **formulate** a **linking theory**.
An Incremental Parser for PLTAG

In order to construct an incremental parser for PLTAG, we need to:

1. convert the Penn Treebank into PLTAG format;
2. induce a lexicon from it;
3. develop an incremental parsing algorithm;
4. devise a probability model;
5. formulate a linking theory.

Evaluation:

- WSJ section 23 (≤ 40 words) F-1 score: 78.65
What about Modelling Semantic Prediction?

Garden Path examples (Frazier and Rayner 1982)

Example 1

ARG0          ARG1
When Mary was knitting the socks
What about Modelling Semantic Prediction?

Garden Path examples (Frazier and Rayner 1982)

### Example 1

<table>
<thead>
<tr>
<th>ARG0</th>
<th>ARG1 &gt; ARG0</th>
</tr>
</thead>
</table>

When Mary was knitting the socks fell to the floor
Garden Path examples (Frazier and Rayner 1982)

<table>
<thead>
<tr>
<th>Example 1</th>
<th>ARG0</th>
<th>ARG1 &gt; ARG0</th>
</tr>
</thead>
<tbody>
<tr>
<td>When Mary was knitting the socks fell to the floor</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Example 2</th>
<th>ARG0</th>
<th>(ARG0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>When Mary was knitting the vase</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
What about Modelling Semantic Prediction?

Garden Path examples (Frazier and Rayner 1982)

Example 1

ARG0
ARG₁ > ARG₀

When Mary was knitting the socks fell to the floor

Example 2

ARG₀
ARG₀

When Mary was knitting the vase fell to the floor
What about Modelling Semantic Prediction?

Garden Path examples (Frazier and Rayner 1982)

Example 1

ARG0  ARG1 > ARG0
When Mary was **knitting** the **socks** fell to the floor

Example 2

ARG0  ARG0
When Mary was **knitting** the **vase** fell to the floor

Semantic relationship between words
Integrating Syntax with Semantics

Integration of syntactic and semantic processing in a single framework
Modelling Semantic Prediction

Integrating Syntax with Semantics

Integration of syntactic and semantic processing in a single framework

Diagram:
- Sentence
- PLTAG
- SRLs
- Incremental Semantic Role Labelling
- Model of Semantics
- Parse Tree Semantic Representation
Integrating Syntax with Semantics

Integration of syntactic and semantic processing in a single framework
Semantic Role Labels

- PLTAG parser builds fully connected prefix trees for a given input sentence
Semantic Role Labels

- PLTAG parser builds fully connected prefix trees for a given input sentence
- Transduce a semantically labelled dependency graph in parallel
Semantic Role Labels

- PLTAG parser builds fully connected prefix trees for a given input sentence
- Transduce a semantically labelled dependency graph in parallel
- Semantic Roles on arcs between predicates (verbs, nouns) and arguments
Semantic Role Labels

- PLTAG parser builds fully connected prefix trees for a given input sentence
- Transduce a semantically labelled dependency graph in parallel
- Semantic Roles on arcs between predicates (verbs, nouns) and arguments
- PropBank (Palmer et al., 2005), NomBank (Meyers et al., 2004)
Semantic Role Labels

- PLTAG parser builds fully connected prefix trees for a given input sentence
- Transduce a semantically labelled dependency graph in parallel
- Semantic Roles on arcs between predicates (verbs, nouns) and arguments
- PropBank (Palmer et al., 2005), NomBank (Meyers et al., 2004)

- Use a compositional model of semantics pivoting around role labels on prefix trees to construct a semantic representation of the input sentence incrementally
Augmented PLTAG Lexicon
Incremental Semantic Role Labelling

- Output fully-connected **partial dependency graphs** on word-level synchronously with prefix trees
Incremental Semantic Role Labelling

- Output fully-connected **partial dependency graphs** on word-level synchronously with prefix trees
- Generative approach to SRL task including **probabilistic model** for disambiguating multiple roles on elementary trees
Incremental Semantic Role Labelling

- Output fully-connected **partial dependency graphs** on word-level synchronously with prefix trees.
- Generative approach to SRL task including **probabilistic model** for disambiguating multiple roles on elementary trees.
- Precision/recall metrics for measuring **incomplete** labelled predicate-argument arcs (Beuck et al., 2011).
Incremental Semantic Role Labelling

- Output fully-connected partial dependency graphs on word-level synchronously with prefix trees
- Generative approach to SRL task including probabilistic model for disambiguating multiple roles on elementary trees
- Precision/recall metrics for measuring incomplete labelled predicate-argument arcs (Beuck et al., 2011)
- First approach to tackle task incrementally
Incremental Semantic Role Labelling

- Output fully-connected partial dependency graphs on word-level synchronously with prefix trees
- Generative approach to SRL task including probabilistic model for disambiguating multiple roles on elementary trees
- Precision/recall metrics for measuring incomplete labelled predicate-argument arcs (Beuck et al., 2011)
- First approach to tackle task incrementally

Preliminary Results (SRL-only Task - CoNLL 2009)

<table>
<thead>
<tr>
<th></th>
<th>Precision</th>
<th>Recall</th>
<th>F-measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>76.79</td>
<td>58.47</td>
<td>66.39</td>
</tr>
<tr>
<td>I-SRL</td>
<td>79.50</td>
<td>61.50</td>
<td>69.30</td>
</tr>
</tbody>
</table>
Take into account role labelling *violations* (e.g., ignore upcoming TAG trees that do not conform to the already predicted dependency graph structure)
Interaction between Syntax and Semantics

- Take into account role labelling *violations* (e.g., ignore upcoming TAG trees that do not conform to the already predicted dependency graph structure)
- Score of attaching a candidate TAG tree to the prefix tree, by querying a model of semantics
Interaction between Syntax and Semantics

- Take into account role labelling violations (e.g., ignore upcoming TAG trees that do not conform to the already predicted dependency graph structure)
- Score of attaching a candidate TAG tree to the prefix tree, by querying a model of semantics
- As a result, low-probability syntactic analyses can be excluded from semantic composition, as well
Which Model of Semantics?

There are many formalisms and theories of semantics. Which is most suitable to our needs?
Which Model of Semantics?

There are many formalisms and theories of semantics. Which is most suitable to our needs?

- Compositional: represent meaning of phrases and sentences, not only words.
There are many formalisms and theories of semantics. Which is most suitable to our needs?

- Compositional: represent meaning of phrases and sentences, not only words.
- Ideally, factor against Semantic Role Labels
Which Model of Semantics?

There are many formalisms and theories of semantics. Which is most suitable to our needs?

- Compositional: represent meaning of phrases and sentences, not only words.
- Ideally, factor against Semantic Role Labels
- Syntactic relations (CoNLL-style) can do too! Mapping to most frequent SRLs is trivial.
Which Model of Semantics?

There are many formalisms and theories of semantics. Which is most suitable to our needs?

- Compositional: represent meaning of phrases and sentences, not only words.
- Ideally, factor against Semantic Role Labels
- Syntactic relations (CoNLL-style) can do too! Mapping to most frequent SRLs is trivial.
- This is a good place for scouting...
Conclusions

- Human sentence processing is incremental and predictive
- We developed a version of TAG that models these properties
- PLTAG attains comparable to state-of-the-art parsing accuracy (compared to other TAG parsers)
- Integrate Syntax with Semantics using PLTAG and Semantic Role Labels as a proxy
- Incremental Semantic Role Labelling
Thank you