Efficient Processing of Underspecified Discourse Representations

Michaela Regneri (Saarland University)
Markus Egg (University of Groningen)
Alexander Koller (University of Edinburgh)

ACL-2008 (Short Paper Session) Columbus, OH
Discourse Structure and Underspecification

[1 I try to read a novel] [2 if I feel bored] [3 or I am unhappy.]

(Gardent & Webber 1998)
Discourse Structure and Underspecification

[1 I try to read a novel] [2 if I feel bored] [3 or I am unhappy.]

(Gardent & Webber 1998)
Discourse Structure and Underspecification

[1 I try to read a novel]  [2 if I feel bored]  [3 or I am unhappy.]

- represent genuine ambiguity

(Gardent & Webber 1998)
Discourse Structure and Underspecification

[1 I try to read a novel] [2 if I feel bored] [3 or I am unhappy.]

\( C_1 \) \( C_2 \) \( C_3 \)

- represent genuine ambiguity
- gain robustness in discourse parsing

(Gardent & Webber 1998)
Underspecified Discourse Parsing

Segmenter

[1] a unit
[2] a unit
[3] a unit
...
[n] a unit

Discourse Model

constraints on & preferences for relations / structure

(convert)

(lots of computation)

add

convenient data structure

(lots of computation)
Underspecified Discourse Parsing

Segmenter

[1] a unit
[2] a unit
[3] a unit
...
[n] a unit

Discourse Model

constraints on & preferences for relations / structure

this paper

(add)

(convert)

(lots of computation)

convenient data structure

(lots of computation)
Underspecified Discourse Representations

• assume binary trees as discourse structure

• create a dominance graph (Althaus et al. 2003) representing all possible binary trees connecting the discourse units

[1 I try to read a novel] [2 if I feel bored]
Underspecified Discourse Representations

- assume binary trees as discourse structure
- create a *dominance graph* (Althaus et al. 2003) representing all possible binary trees connecting the discourse units

\[1\text{ I try to read a novel}] [2\text{ if I feel bored}] [3\text{ or I am unhappy.}]

\[\text{if} \quad \begin{array}{c} 1 \quad 2 \\ C_1 \quad C_2 \end{array} \]
Underspecified Discourse Representations

- assume binary trees as discourse structure
- create a *dominance graph* (Althaus et al. 2003) representing all possible binary trees connecting the discourse units

[1 I try to read a novel] [2 if I feel bored] [3 or I am unhappy.]
Underspecified Discourse Representations

- assume binary trees as discourse structure
- create a *dominance graph* (Althaus et al. 2003) representing all possible binary trees connecting the discourse units

> [1 I try to read a novel] [2 if I feel bored]

> [3 or I am unhappy.]

> [4 You will notice this.]

---

Underspecified Discourse Representations

- assume binary trees as discourse structure
- create a *dominance graph* (Althaus et al. 2003) representing all possible binary trees connecting the discourse units

\[1 \text{ I try to read a novel} \] \[2 \text{ if I feel bored} \]
\[3 \text{ or I am unhappy.} \]
\[4 \text{ You will notice this.} \]
Underspecified Discourse Representations

- assume binary trees as discourse structure
- create a dominance graph (Althaus et al. 2003) representing all possible binary trees connecting the discourse units

[1] I try to read a novel] [2] if I feel bored]
[3] or I am unhappy.]
[4] You will notice this.]

(constrained) chain
Regular Tree Grammars

- describe set of trees by means of rewrite rules ($\approx$ context free grammars)
- recently introduced as a formalism for semantic underspecification (Koller et al. 2008)

S $\rightarrow$ f(B,C)
B $\rightarrow$ g
C $\rightarrow$ h

(Comon et al. 2007)
Regular Tree Grammars

- describe set of trees be means of rewrite rules ($\approx$ context free grammars)
- recently introduced as a formalism for semantic underspecification (Koller et al. 2008)

(S $\rightarrow$ f(B,C))

B $\rightarrow$ g
C $\rightarrow$ h

○ S $\rightarrow$

f

B

C
Regular Tree Grammars

- describe set of trees be means of rewrite rules (≈ context free grammars)
- recently introduced as a formalism for semantic underspecification (Koller et al. 2008)

\[
\begin{aligned}
S &\rightarrow f(B,C) \\
B &\rightarrow g \\
C &\rightarrow h
\end{aligned}
\]

(Comon et al. 2007)
Regular Tree Grammars

• describe set of trees by means of rewrite rules ($\approx$ context free grammars)

• recently introduced as a formalism for semantic underspecification (Koller et al. 2008)

\[
\begin{align*}
S & \rightarrow f(B,C) \\
B & \rightarrow g \\
C & \rightarrow h \\
\end{align*}
\]

(Comon et al. 2007)
Dominance Graphs and Regular Tree Grammars

(cf. Koller et al. 2008)

{1,2,3,4,5,6,7} ➔ if({1},{3,4,5,6,7})
{1,2,3,4,5,6,7} ➔ or({1,2,3},{5,6,7})
{1,2,3,4,5,6,7} ➔ after({1,2,3,4,5},{7})
{1,2,3,4,5} ➔ if({1},{3,4,5})
{1,2,3,4,5} ➔ or({1,2,3},{5})
{3,4,5,6,7} ➔ or({3},{5,6,7})
{3,4,5,6,7} ➔ after({3,4,5},{7})
{3,4,5} ➔ or({1},{3})
{1,2,3} ➔ if({1},{3})
{5,6,7} ➔ after({5},{7})
{1} ➔ C₁
{3} ➔ C₂
{5} ➔ C₃
{7} ➔ C₄
Dominance Graphs and Regular Tree Grammars

(cf. Koller et al. 2008)
Dominance Graphs and Regular Tree Grammars

\[\{1,2,3,4,5,6,7\} \rightarrow \text{if}\{1\},\{3,4,5,6,7\}\]
\[\{1,2,3,4,5,6,7\} \rightarrow \text{or}\{1,2,3\},\{5,6,7\}\]
\[\{1,2,3,4,5,6,7\} \rightarrow \text{after}\{1,2,3,4,5\},\{7\}\]
\[\{1,2,3,4,5\} \rightarrow \text{if}\{1\},\{3,4,5\}\]
\[\{1,2,3,4,5\} \rightarrow \text{or}\{1,2,3\},\{5\}\]
\[\{3,4,5,6,7\} \rightarrow \text{or}\{3\},\{5,6,7\}\]
\[\{3,4,5,6,7\} \rightarrow \text{after}\{3,4,5\},\{7\}\]
\[\{3,4,5\} \rightarrow \text{or}\{1\},\{3\}\]

\[\{1,2,3\} \rightarrow \text{if}\{1\},\{3\}\]
\[\{5,6,7\} \rightarrow \text{after}\{5\},\{7\}\]
\[\{1\} \rightarrow C_1 \quad \{3\} \rightarrow C_2 \quad \{5\} \rightarrow C_3 \quad \{7\} \rightarrow C_4\]

(cf. Koller et al. 2008)
Dominance Graphs and Regular Tree Grammars

(cf. Koller et al. 2008)
Underspecification and Regular Tree Grammars

- straightforward extension to \textit{weighted} RTGs for preference model inclusion

\[
\begin{align*}
S &\rightarrow f(B,C) \ [5] \quad B \rightarrow g \ [1] \\
S &\rightarrow r(C) \ [1] \quad C \rightarrow h \ [1]
\end{align*}
\]

- expressing complex discourse constraints
  \cite{EggRegneri2008}
Ambiguity in underspecification: Semantics vs. Discourse

MRSes from Rondane coprus, ERG July 2006

- Median #fragments: 13
- Median #RTG rules: 41
- Median #readings: 56
Ambiguity in underspecification: Semantics vs. Discourse

MRSes from Rondane corpus, ERG July 2006

<table>
<thead>
<tr>
<th>Median</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>#fragments</td>
<td>13</td>
</tr>
<tr>
<td>#RTG rules</td>
<td>41</td>
</tr>
<tr>
<td>#readings</td>
<td>56</td>
</tr>
</tbody>
</table>

constrained chains for texts in RST-Discourse Treebank (Carlson et al. 2002)
Ambiguity in underspecification: Semantics vs. Discourse

MRSes from Rondane corpus, ERG July 2006

median #fragments
- 13

median #RTG rules
- 41

median #readings
- 56

constrained chains for texts in RST-Discourse Treebank (Carlson et al. 2002)
Ambiguity in underspecification: Semantics vs. Discourse

Median #fragments: 13
Median #RTG rules: 41
Median #readings: 56

MRSes from Rondane corpus, ERG July 2006

Constrained chains for texts in RST-Discourse Treebank (Carlson et al. 2002)
Ambiguity in underspecification: Semantics vs. Discourse

MRSes from Rondane corpus, ERG July 2006

- median #fragments: 13
- median #RTG rules: 41
- median #readings: 9,139

Constrained chains for texts in RST-Discourse Treebank (Carlson et al. 2002)

4.6 \times 10^{19}
Optimizing RTG Algorithms for Discourse Representations

\[
\begin{align*}
\{1,2,3,4,5,6,7\} & \rightarrow \text{if}(\{1\},\{3,4,5,6,7\}) \\
\{1,2,3,4,5,6,7\} & \rightarrow \text{or}(\{1,2,3\},\{5,6,7\}) \\
\{1,2,3,4,5,6,7\} & \rightarrow \text{after}(\{1,2,3,4,5\},\{7\}) \\
\{1,2,3,4,5\} & \rightarrow \text{if}(\{1\},\{3,4,5\}) \\
\{1,2,3,4,5\} & \rightarrow \text{or}(\{1,2,3\},\{5\}) \\
\{3,4,5,6,7\} & \rightarrow \text{or}(\{3\},\{5,6,7\}) \\
\{3,4,5,6,7\} & \rightarrow \text{after}(\{3,4,5\},\{7\}) \\
\{3,4,5\} & \rightarrow \text{or}(\{1\},\{3\}) \\
\{1,2,3\} & \rightarrow \text{if}(\{1\},\{3\}) \\
\{5,6,7\} & \rightarrow \text{after}(\{5\},\{7\}) \\
\{1\} & \rightarrow C_1 \\
\{3\} & \rightarrow C_2 \\
\{5\} & \rightarrow C_3 \\
\{7\} & \rightarrow C_4
\end{align*}
\]
Optimizing RTG Algorithms for Discourse Representations

Grammar nonterminals:

subgraphs

integer pairs

\[
\begin{align*}
\{1,2,3,4,5,6,7\} & \rightarrow \text{if}\{(1,\{3,4,5,6,7\})
\\{1,2,3,4,5,6,7\} & \rightarrow \text{or}\{(1,2,3),\{5,6,7\}
\{1,2,3,4,5,6,7\} & \rightarrow \text{after}\{(1,2,3,4,5),\{7\}
\{1,2,3,4,5\} & \rightarrow \text{if}\{(1,\{3,4,5\})
\{1,2,3,4,5\} & \rightarrow \text{or}\{(1,2,3),\{5\}
\{3,4,5,6,7\} & \rightarrow \text{or}\{(3),\{5,6,7\}
\{3,4,5,6,7\} & \rightarrow \text{after}\{(3,4,5),\{7\}
\{3,4,5\} & \rightarrow \text{or}\{(1),\{3\}
\{1,2,3\} & \rightarrow \text{if}\{(1),\{3\}
\{5,6,7\} & \rightarrow \text{after}\{(5),\{7\}
\{1\} & \rightarrow C_1 \quad \{3\} \rightarrow C_2 \quad \{5\} \rightarrow C_3 \quad \{7\} \rightarrow C_4
\end{align*}
\]
Optimizing RTG Algorithms for Discourse Representations

Grammar nonterminals:

- subgraphs
- integer pairs

Graph:

1. if
   - C1
   - 1
2. or
   - C2
   - 3
3. after
   - C3
   - 5
4. or
   - C4
   - 7

Rules:

- {1;7} → if({1;1},{3;7})
- {1;7} → or({1;3},{5;7})
- {1;7} → after({1;5},{7;7})
- {1;5} → if({1;1},{3;5})
- {1;5} → or({1;3},{5;5})
- {3;7} → or({3;3},{5;7})
- {3;7} → after({3;5},{7;7})
- {3,4,5} → or({1;1},{3;3})
- {1;3} → if({1;1},{3;3})
- {5;7} → after({5;5},{7;7})
- {1;1} → C1
- {3;3} → C2
- {5;5} → C3
- {7;7} → C4
Optimizing RTG Algorithms for Discourse Representations

Grammar nonterminals:

subgraphs

↓

integer pairs
Optimizing RTG Algorithms for Discourse Representations

Grammar nonterminals:

subgraphs

\[
\downarrow
\]

integer pairs
Optimizing RTG Algorithms for Discourse Representations

Grammar nonterminals:
- subgraphs
  - integer pairs

Grammar generation algorithm:
- graph-based operations
  - integer comparisons
Runtime Evaluation: Method

• simulate a typical processing run:
  ♦ generate an underspecified RTG
  ♦ integrate a preference model (Egg & Regneri 2008)
  ♦ extract the best configuration (Knight & Graehl 2005)

• compare the optimized algorithm with the generic implementation for underspecified semantic descriptions: *Utool* (Koller & Thater 2005)
Runtime Evaluation: Results

runtime (ms)

no of discourse units

new total
utool total
Conclusions

- Underspecification in discourse processing
- RTGs as underspecification formalism
- Optimize algorithms for discourse RTGs
- Compute globally best configuration according to some weight model
- Making underspecified discourse processing computationally feasible