Event Extraction as Dependency Parsing

David McClosky
Stanford University
4.21.2011

Joint work with Mihai Surdeanu and Chris Manning
(to appear in ACL 2011)
Goal: Determine which biological events occur within text
Goal: Determine which biological events occur within text
Why? Thousands of biomedical articles are published each month. Create databases of known interactions, better search
Goal: Determine which biological events occur within text

Why? Thousands of biomedical articles are published each month. Create databases of known interactions, better search

We have found that the HTLV-1 transactivator protein, tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription in that it can cooperate with TCR signals to mediate high level gene expression.
Goal: Determine which biological events occur within text

Why? Thousands of biomedical articles are published each month. Create databases of known interactions, better search

... tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription...
Event Extraction from Biomedical Text

**Goal:** Determine which biological events occur within text

**Why?** Thousands of biomedical articles are published *each month*. Create databases of known interactions, better search
Event Extraction from Biomedical Text

**Goal:** Determine which biological events occur within text

**Why?** Thousands of biomedical articles are published *each month.*
Create databases of known interactions, better search

... tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription ...
Prot Positive Regulation Prot Prot Transcription
Hierarchical Event Extraction from Biomedical Text

**Goal:** Determine which biological events occur within text

**Why?** Thousands of biomedical articles are published *each month*. Create databases of known interactions, better search
This talk in two slides...

... tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription...
Spoiler alert!

... tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription...

Root

Cause

Theme

Theme

Theme

root

Prot

Positive Regulation

Prot

Prot

Transcription
A little bit about the BioNLP 2009 shared task

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Arguments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple</td>
<td>Gene expression</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td>Transcription</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td>Protein catabolism</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td>Phosphorylation</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td>Localization</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td>Binding</td>
<td>Theme (Protein) +</td>
</tr>
</tbody>
</table>
A little bit about the BioNLP 2009 shared task

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Arguments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple</td>
<td><strong>Gene expression</strong></td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td><strong>Transcription</strong></td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td><strong>Protein catabolism</strong></td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td><strong>Phosphorylation</strong></td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td><strong>Localization</strong></td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td><strong>Binding</strong></td>
<td>Theme (Protein) +</td>
</tr>
<tr>
<td>Complex</td>
<td><strong>Regulation</strong></td>
<td>Theme / Cause (Protein / Event)</td>
</tr>
<tr>
<td></td>
<td><strong>Positive regulation</strong></td>
<td>Theme / Cause (Protein / Event)</td>
</tr>
<tr>
<td></td>
<td><strong>Negative regulation</strong></td>
<td>Theme / Cause (Protein / Event)</td>
</tr>
</tbody>
</table>
Some messy details

- Protein entities given for free
Some messy details

- Protein entities given for free
  - ...but event anchors must be detected by the model
Some messy details

- Protein entities given for free
  - ...but event anchors must be detected by the model
- Event anchors and proteins can participate in multiple events
Some messy details

- Protein entities given for free
  - ...but event anchors must be detected by the model
- Event anchors and proteins can participate in multiple events
- Events can span sentences ($\approx 7\%$ do)
Some messy details

- Protein entities given for free
  - ...but event anchors must be detected by the model
- Event anchors and proteins can participate in multiple events
- Events can span sentences ($\approx 7\%$ do)
- Actually the simplest BioNLP 2009 shared task (“Task 1”)
Some messy details

- Protein entities given for free
  - ...but event anchors must be detected by the model
- Event anchors and proteins can participate in multiple events
- Events can span sentences (≈ 7% do)
- Actually the simplest BioNLP 2009 shared task (“Task 1”)
  - ...and BioNLP 2011 task includes two new domains
Outline

1. BioNLP shared task
2. Previous approaches
   - Pipelined systems
   - Markov Logic
3. Event Parsing
4. Experiments
5. Future work
6. Conclusion
UTurku: Björne et al. (2009)

- Best scoring system in BioNLP 2009 shared task

[Björne et al., BioNLP 2009]
UTurku: Björne et al. (2009)

- Best scoring system in BioNLP 2009 shared task
- Pipelined classifiers:
  - Event anchor detection and classification

[Björne et al., BioNLP 2009]
UTurku: Björne et al. (2009)

- Best scoring system in BioNLP 2009 shared task
- Pipelined classifiers:
  1. Event anchor detection and classification
  2. Event linking

[Björne et al., BioNLP 2009]
UTurku: Björne et al. (2009)

- Best scoring system in BioNLP 2009 shared task
- Pipelined classifiers:
  1. Event anchor detection and classification
  2. Event linking
  3. Heuristic postprocessing rules

[Björne et al., BioNLP 2009]
UTurku: Björne et al. (2009)

- Best scoring system in BioNLP 2009 shared task
- Pipelined classifiers:
  1. Event anchor detection and classification
  2. Event linking
  3. Heuristic postprocessing rules
- 52.0% f-score

[Björne et al., BioNLP 2009]
Miwa et al. (2010)

- **Outperforms** best scoring system in BioNLP 2009 shared task
- Pipelined classifiers:
  1. Event anchor detection and classification
  2. Event linking
  3. **Learned** postprocessing rules
- **53.3%** $f$-score
- More domain specific features, multiple syntactic parsers

[Miwa et al., JBCB 2010]
Markov Logic

- Markov logic-based system using hard and soft constraints

[Poon and Vanderwende, NAACL 2010]
[Riedel et al., NAACL 2009]
Markov Logic

- Markov logic-based system using hard and soft constraints
- Example formula schema:

\[ \text{Token}(j, + \text{text}) \land \text{SyntacticDep}(i, j, dep) \implies \text{EventType}(i, + \text{type}) \]

[Poon and Vanderwende, NAACL 2010]
[Riedel et al., NAACL 2009]
Markov Logic

- Markov logic-based system using hard and soft constraints
- Example formula schema:

\[
\text{Token}(j, + \text{text}) \land \text{SyntacticDep}(i, j, \text{dep}) \implies \text{EventType}(i, + \text{type})
\]
\[
\text{SyntacticDep}(i, j, + \text{dep}) \land \text{Protein}(i) \implies \text{EventArg}(i, j, + \text{label})
\]

[Poon and Vanderwende, NAACL 2010]
[Riedel et al., NAACL 2009]
Markov Logic

- Markov logic-based system using hard and soft constraints
- Example formula schema:

\[
\begin{align*}
\text{Token}(j, +\text{text}) \land \text{SyntacticDep}(i, j, \text{dep}) & \implies \text{EventType}(i, +\text{type}) \\
\text{SyntacticDep}(i, j, +\text{dep}) \land \text{Protein}(i) & \implies \text{EventArg}(i, j, +\text{label})
\end{align*}
\]

- 50.0% f-score

[Poon and Vanderwende, NAACL 2010]
[Riedel et al., NAACL 2009]
Outline

1. BioNLP shared task
2. Previous approaches
3. Event Parsing
4. Experiments
5. Future work
6. Conclusion
Overview of our model

... tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription ...
Overview of our model

... tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription ...

**Preprocessing:** Segmentation, tokenization, syntactic parsing

[McClosky and Charniak, ACL 2008]
Overview of our model

Anchor classification: Essentially NER for event anchors
Overview of our model

Event parsing: Parse anchors and proteins using reranking parser
Anchor classification

- Anchors can be multiple words (13% have 2+ words)
Anchor classification

- Anchors can be multiple words (13% have 2+ words)
- Our anchor classifiers only operate on heads of anchors
Anchor classification

- Anchors can be multiple words (13% have 2+ words)
- Our anchor classifiers only operate on heads of anchors
- Logistic Regression works best for us ($\approx 65\%$ f-score)
Anchor classification

- Anchors can be multiple words (13% have 2+ words)
- Our anchor classifiers only operate on heads of anchors
- Logistic Regression works best for us (≈65% f-score)
- More recent work on boosting recall (distributional similarity features)
Event parsing with dependency parsers

... tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription ...

Prot Positive Regulation

GM-CSF Prot

IL-2 Prot Transcription

Cause

Theme

Theme

Theme
Event parsing with dependency parsers

- Root
- Cause
- Theme

... tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription ...

Prot | Positive Regulation | Prot

David McClosky (Stanford)
Event parsing with dependency parsers

(Not pictured: Unused entities linked to the root as well.)
Event parsing with dependency parsers
DAGnabbit!

root

Root

Cause

Root

Cause

Theme

Theme

Theme

Theme

tax

acts

GM-CSF

IL-2

transcription

Prot

Positive Regulation

Prot

Prot

Transcription
...but most duplicates can be merged

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Arguments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple</td>
<td>Gene expression</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td>Transcription</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td>Protein catabolism</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td>Phosphorylation</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td>Localization</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td>Binding</td>
<td>Theme (Protein) +</td>
</tr>
<tr>
<td>Complex</td>
<td>Regulation</td>
<td>Theme / Cause (Protein / Event)</td>
</tr>
<tr>
<td></td>
<td>Positive regulation</td>
<td>Theme / Cause (Protein / Event)</td>
</tr>
<tr>
<td></td>
<td>Negative regulation</td>
<td>Theme / Cause (Protein / Event)</td>
</tr>
</tbody>
</table>
...but most duplicates can be merged

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Arguments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple</td>
<td>Gene expression</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td>Transcription</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td>Protein catabolism</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td>Phosphorylation</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td>Localization</td>
<td>Theme (Protein)</td>
</tr>
<tr>
<td></td>
<td><strong>Binding</strong></td>
<td>Theme (Protein) +</td>
</tr>
<tr>
<td>Complex</td>
<td>Regulation</td>
<td>Theme / Cause (Protein / Event)</td>
</tr>
<tr>
<td></td>
<td>Positive regulation</td>
<td>Theme / Cause (Protein / Event)</td>
</tr>
<tr>
<td></td>
<td>Negative regulation</td>
<td>Theme / Cause (Protein / Event)</td>
</tr>
</tbody>
</table>

- Binding is the only ambiguous case.
Maximum-spanning tree based parsing

Why a dependency parser?

- Event structures are non-projective (non-planar)
Maximum-spanning tree based parsing

Why a dependency parser?
- Event structures are non-projective (non-planar)

Why MSTParser? [McDonald et al., EMNLP 2005]
- Handles non-projective trees naturally
Maximum-spanning tree based parsing

Why a dependency parser?
- Event structures are non-projective (non-planar)

Why MSTParser? [McDonald et al., EMNLP 2005]
- Handles non-projective trees naturally
- Easy to extend feature extractor
Maximum-spanning tree based parsing

Why a dependency parser?
- Event structures are non-projective (non-planar)

Why MSTParser?  [McDonald et al., EMNLP 2005]
- Handles non-projective trees naturally
- Easy to extend feature extractor
- Support for $n$-best parsing
Crash course in MSTParser

- Parse trees represented as a **labeled graph** \((G = (V, E))\)
- Words are nodes \((i, j, \cdots \in V)\), dependency relations are edges \((e_{ij} \in E)\)
Crash course in MSTParser

- Parse trees represented as a **labeled graph** \( G = (V, E) \)
- Words are nodes \((i, j, \cdots \in V)\), dependency relations are edges \((e_{ij} \in E)\)
- Each edge has a **feature vector** \( f(i, j) \) and **score**: \( s(i, j) = w \cdot f(i, j) \)
Crash course in MSTParser

- Parse trees represented as a **labeled graph** \((G = (V, E))\)
- Words are nodes \((i, j, \cdots \in V)\), dependency relations are edges \((e_{ij} \in E)\)
- Each edge has a **feature vector** \((f(i, j))\) and **score**: \(s(i, j) = w \cdot f(i, j)\)
- Find a subset of edges \(\pi = \{e_{ij}\} \subset E\) such that
  - Edges form a **tree**
Crash course in MSTParser

- Parse trees represented as a **labeled graph** \((G = (V, E))\)
- Words are nodes \((i, j, \cdots \in V)\), dependency relations are edges \((e_{ij} \in E)\)
- Each edge has a **feature vector** \((f(i, j))\) and **score**: \(s(i, j) = w \cdot f(i, j)\)
- Find a subset of edges \(\pi = \{e_{ij}\} \subset E\) such that
  - Edges form a **tree**
  - Edges have **maximal score**: \(\sum_\pi s(i, j)\)
Crash course in MSTParser

- Parse trees represented as a labeled graph \( G = (V, E) \)
- Words are nodes \((i, j, \cdots \in V)\), dependency relations are edges \((e_{ij} \in E)\)
- Each edge has a feature vector \(f(i, j)\) and score: \(s(i, j) = w \cdot f(i, j)\)
- Find a subset of edges \(\pi = \{e_{ij}\} \subset E\) such that
  - Edges form a tree
  - Edges have maximal score: \(\sum_\pi s(i, j)\)
- Can be solved in \(O(n^2)\) time

[Chu and Liu, 1965], [Edmonds, 1967], [Tarjan, 1977]
Crash course in MSTParser

- Parse trees represented as a **labeled graph** \((G = (V, E))\)
- Words are nodes \((i, j, \cdots \in V)\), dependency relations are edges \((e_{ij} \in E)\)
- Each edge has a **feature vector** \((f(i, j))\) and **score**: \(s(i, j) = w \cdot f(i, j)\)
- Find a subset of edges \(\pi = \{e_{ij}\} \subset E\) such that
  - Edges form a **tree**
  - Edges have **maximal score**: \(\sum_\pi s(i, j)\)
- Can be solved in \(O(n^2)\) time
  - [Chu and Liu, 1965], [Edmonds, 1967], [Tarjan, 1977]
- Features must be **edge-factored**
Edge-factored features

```
root  ... tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription ...
Prot Positive Regulation Prot Transcription

Root

Cause

Theme

Theme

Theme
```
Edge-factored features

... tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription ...

Prot Positive Regulation Prot Prot Transcription
Second-order edge-factored features

... tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription...

Prot Positive Regulation

Theme Theme
Feature spaces

... tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription ...

Positve Regulation

“Full”
Feature spaces

“Full”
(includes original syntactic tree)
Feature spaces

„Reduced”
Features for BioNLP

**Full sentence space:**

- Surface words features (distance, \(n\)-grams)
Features for BioNLP

Full sentence space:

- Surface words features (distance, $n$-grams)
- Constituency/dependency path features (length, $n$-grams, endpoints)
Features for BioNLP

Full sentence space:
- Surface words features (distance, \( n \)-grams)
- Constituency/dependency path features (length, \( n \)-grams, endpoints)
- Semantic graph features (\# and identities of children/siblings/parents)
Features for BioNLP

Full sentence space:
- Surface words features (distance, $n$-grams)
- Constituency/dependency path features (length, $n$-grams, endpoints)
- Semantic graph features (# and identities of children/siblings/parents)

Reduced sentence space:
- All the original MSTParser features
Features for BioNLP

Full sentence space:
- Surface words features (distance, $n$-grams)
- Constituency/dependency path features (length, $n$-grams, endpoints)
- Semantic graph features (# and identities of children/siblings/parents)

Reduced sentence space:
- All the original MSTParser features
- Generalized type features
  (e.g. *Positive Regulation* is a *Complex Event* is an *Event*)
Event parse reranking

- MSTParser is limited to highly local features (1–2 edges).
Event parse reranking

- MSTParser is limited to highly local features (1–2 edges).
- Rerankers work great for syntactic parsing, why not event parsing?
  [Ratnaparkhi, JML 1997], [Charniak and Johnson, ACL 2005]
Event parse reranking

- MSTParser is limited to highly local features (1–2 edges).
- Rerankers work great for syntactic parsing, why not event parsing? [Ratnaparkhi, JML 1997], [Charniak and Johnson, ACL 2005]
- Extend parser to $k$-best parser ($k = 50$ for us) [Hall, ACL 2007]
Event parse reranking

- MSTParser is limited to highly local features (1–2 edges).
- Rerankers work great for syntactic parsing, why not event parsing? [Ratnaparkhi, JML 1997], [Charniak and Johnson, ACL 2005]
- Extend parser to $k$-best parser ($k = 50$ for us) [Hall, ACL 2007]
- Given $k$ parses, rescore them and rerank
Event parse reranking

- MSTParser is limited to highly local features (1–2 edges).
- Rerankers work great for syntactic parsing, why not event parsing? [Ratnaparkhi, JML 1997], [Charniak and Johnson, ACL 2005]
- Extend parser to $k$-best parser ($k = 50$ for us) [Hall, ACL 2007]
- Given $k$ parses, rescore them and rerank
- Can optimize actual BioNLP $f$-score metric, use any features
Event parse reranking

- MSTParser is limited to highly local features (1–2 edges).
- Rerankers work great for syntactic parsing, why not event parsing? [Ratnaparkhi, JML 1997], [Charniak and Johnson, ACL 2005]
- Extend parser to $k$-best parser ($k = 50$ for us) [Hall, ACL 2007]
- Given $k$ parses, rescore them and rerank
- Can optimize actual BioNLP $f$-score metric, use any features
- Can combine output from multiple parsers [Johnson and Ural, NAACL 2010]
Event parse reranking

- MSTParser is limited to highly local features (1–2 edges).
- Rerankers work great for syntactic parsing, why not event parsing? [Ratnaparkhi, JML 1997], [Charniak and Johnson, ACL 2005]
- Extend parser to $k$-best parser ($k = 50$ for us) [Hall, ACL 2007]
- Given $k$ parses, rescore them and rerank
- Can optimize actual BioNLP $f$-score metric, use any features
- Can combine output from multiple parsers [Johnson and Ural, NAACL 2010]
- $k$-best decoding in $O(kn^2)$, reranking takes $O(k)$ time
Reranker features

- Root
- Cause
- Theme

*root*... tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription...
Reranker features

Paths to root

- **Root**
- **Cause**
- **Theme**

```
root: tax, acts as a costimulatory signal for GM-CSF and IL-2 gene transcription
Prot: Positive Regulation
Prot: Transcription
```
Reranker features

Event frames

Root → Cause

... tax, acts as a costimulatory signal for GM-CSF
Prot Positive Regulation Prot

... and IL-2 gene transcription
Prot Prot Transcription

David McClosky (Stanford)
Reranker features

MST score: 23.492

Score from parser
Relation to previous models

more global

slower
Relation to previous models

- more global
- Turku
- slower
Relation to previous models

more global

- Turku
- Markov Logic

slower
Relation to previous models

- Turku
- Event Parsing
- Markov Logic

more global

slower
Relation to previous models

- More global
- Event Parsing
- with reranker
- Markov Logic
- Turku
- Slower
Outline

1. BioNLP shared task
2. Previous approaches
3. Event Parsing
4. Experiments
5. Future work
6. Conclusion
Experimental setup

Corpora

- 800 articles for training, 150 for development, 260 for testing
Experimental setup

Corpora

- 800 articles for training, 150 for development, 260 for testing
- Training includes 8,597 events, 6,607 anchors, 9,300 proteins
Experimental setup

Corpora
- 800 articles for training, 150 for development, 260 for testing
- Training includes 8,597 events, 6,607 anchors, 9,300 proteins

Anchors
- Two scenarios: Gold or predicted
Experimental setup

Corpora

- 800 articles for training, 150 for development, 260 for testing
- Training includes 8,597 events, 6,607 anchors, 9,300 proteins

Anchors

- Two scenarios: Gold or predicted
- When predicted, train on the union of predicted and gold anchors
Performance of system components

<table>
<thead>
<tr>
<th>Anchors</th>
<th>Parser</th>
<th>RR</th>
<th>Conv.</th>
<th>Rec</th>
<th>Prec</th>
<th>F_1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gold</td>
<td>Gold</td>
<td>Gold</td>
<td>✓</td>
<td>81.6</td>
<td>93.4</td>
<td>87.1</td>
</tr>
</tbody>
</table>

(performance on development corpus)
Performance of system components

<table>
<thead>
<tr>
<th>Anchors</th>
<th>Parser</th>
<th>RR</th>
<th>Conv.</th>
<th>Rec</th>
<th>Prec</th>
<th>F₁</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gold</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>68.9</td>
<td>77.1</td>
<td>72.7</td>
</tr>
<tr>
<td>Gold</td>
<td>Gold</td>
<td>Gold</td>
<td>✓</td>
<td>81.6</td>
<td>93.4</td>
<td>87.1</td>
</tr>
</tbody>
</table>

(performance on development corpus)
# Performance of system components

<table>
<thead>
<tr>
<th>Anchors</th>
<th>Parser</th>
<th>RR</th>
<th>Conv.</th>
<th>Rec</th>
<th>Prec</th>
<th>F&lt;sub&gt;1&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gold</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>68.9</td>
<td>77.1</td>
<td>72.7</td>
</tr>
<tr>
<td>Gold</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>68.5</td>
<td>78.2</td>
<td>73.1</td>
</tr>
<tr>
<td>Gold</td>
<td>Gold</td>
<td>Gold</td>
<td>✓</td>
<td>81.6</td>
<td>93.4</td>
<td>87.1</td>
</tr>
</tbody>
</table>

(performance on development corpus)
### Performance of system components

<table>
<thead>
<tr>
<th>Anchors</th>
<th>Parser</th>
<th>RR</th>
<th>Conv.</th>
<th>Rec</th>
<th>Prec</th>
<th>F₁</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>45.9</td>
<td>61.8</td>
<td>52.7</td>
</tr>
<tr>
<td>Gold</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>68.9</td>
<td>77.1</td>
<td>72.7</td>
</tr>
<tr>
<td>Gold</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>68.5</td>
<td>78.2</td>
<td>73.1</td>
</tr>
<tr>
<td>Gold</td>
<td>Gold</td>
<td>Gold</td>
<td>✓</td>
<td>81.6</td>
<td>93.4</td>
<td>87.1</td>
</tr>
</tbody>
</table>

(Performance on development corpus)
## Performance of system components

<table>
<thead>
<tr>
<th>Anchors</th>
<th>Parser</th>
<th>RR</th>
<th>Conv.</th>
<th>Rec</th>
<th>Prec</th>
<th>F₁</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>45.9</td>
<td>61.8</td>
<td>52.7</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>48.7</td>
<td>59.3</td>
<td>53.5</td>
</tr>
<tr>
<td>Gold</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>68.9</td>
<td>77.1</td>
<td>72.7</td>
</tr>
<tr>
<td>Gold</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>68.5</td>
<td>78.2</td>
<td>73.1</td>
</tr>
<tr>
<td>Gold</td>
<td>Gold</td>
<td>Gold</td>
<td>✓</td>
<td>81.6</td>
<td>93.4</td>
<td>87.1</td>
</tr>
</tbody>
</table>

(Performance on development corpus)
## Oracle reranker scores

<table>
<thead>
<tr>
<th>Anchors</th>
<th>Parser(s)</th>
<th>1</th>
<th>2</th>
<th>10</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Gold</strong></td>
<td>2P</td>
<td>71.8</td>
<td>77.5</td>
<td>84.8</td>
<td>86.2</td>
</tr>
<tr>
<td></td>
<td>1P, 2P, 2N</td>
<td></td>
<td></td>
<td></td>
<td>86.7</td>
</tr>
<tr>
<td><strong>Predicted</strong></td>
<td>2P</td>
<td>52.7</td>
<td>60.7</td>
<td>70.1</td>
<td>72.5</td>
</tr>
<tr>
<td></td>
<td>1P, 2P, 2N</td>
<td></td>
<td></td>
<td></td>
<td>73.4</td>
</tr>
</tbody>
</table>

(Performance on development corpus)
## Oracle reranker scores

<table>
<thead>
<tr>
<th>Anchors</th>
<th>Parser(s)</th>
<th>1</th>
<th>2</th>
<th>10</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gold</td>
<td>2P</td>
<td>71.8</td>
<td>77.5</td>
<td>84.8</td>
<td>86.2</td>
</tr>
<tr>
<td></td>
<td>1P, 2P, 2N</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>86.7</td>
</tr>
<tr>
<td>Predicted</td>
<td>2P</td>
<td>52.7</td>
<td>60.7</td>
<td>70.1</td>
<td>72.5</td>
</tr>
<tr>
<td></td>
<td>1P, 2P, 2N</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>73.4</td>
</tr>
</tbody>
</table>

*(performance on development corpus)*
**Oracle reranker scores**

<table>
<thead>
<tr>
<th>Anchors</th>
<th>Parser(s)</th>
<th>1</th>
<th>2</th>
<th>10</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gold</td>
<td>2P</td>
<td>71.8</td>
<td>77.5</td>
<td>84.8</td>
<td>86.2</td>
</tr>
<tr>
<td></td>
<td>1P, 2P, 2N</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>86.7</td>
</tr>
<tr>
<td>Predicted</td>
<td>2P</td>
<td>52.7</td>
<td>60.7</td>
<td>70.1</td>
<td>72.5</td>
</tr>
<tr>
<td></td>
<td>1P, 2P, 2N</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>73.4</td>
</tr>
</tbody>
</table>

*(performance on development corpus)*
Oracle reranker scores

<table>
<thead>
<tr>
<th>Anchors</th>
<th>Parser(s)</th>
<th>1</th>
<th>2</th>
<th>10</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gold</td>
<td>2P</td>
<td>71.8</td>
<td>77.5</td>
<td>84.8</td>
<td>86.2</td>
</tr>
<tr>
<td></td>
<td>1P, 2P, 2N</td>
<td></td>
<td></td>
<td></td>
<td>86.7</td>
</tr>
<tr>
<td>Predicted</td>
<td>2P</td>
<td>52.7</td>
<td>60.7</td>
<td>70.1</td>
<td>72.5</td>
</tr>
<tr>
<td></td>
<td>1P, 2P, 2N</td>
<td></td>
<td></td>
<td></td>
<td>73.4</td>
</tr>
</tbody>
</table>

(Performance on development corpus)
### Oracle reranker scores

<table>
<thead>
<tr>
<th>Anchors</th>
<th>Parser(s)</th>
<th>1</th>
<th>2</th>
<th>10</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Gold</strong></td>
<td>2P</td>
<td>71.8</td>
<td>77.5</td>
<td>84.8</td>
<td>86.2</td>
</tr>
<tr>
<td></td>
<td>1P, 2P, 2N</td>
<td></td>
<td></td>
<td></td>
<td>86.7</td>
</tr>
<tr>
<td><strong>Predicted</strong></td>
<td>2P</td>
<td>52.7</td>
<td>60.7</td>
<td>70.1</td>
<td>72.5</td>
</tr>
<tr>
<td></td>
<td>1P, 2P, 2N</td>
<td></td>
<td></td>
<td></td>
<td>73.4</td>
</tr>
</tbody>
</table>

(Performance on development corpus)
Comparison with State-of-the-Art

<table>
<thead>
<tr>
<th>System</th>
<th>$f$-score</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\text{dev}_{GA}$</td>
</tr>
<tr>
<td>Event Parsing</td>
<td>73.1</td>
</tr>
<tr>
<td>[Björne et al., 2009]</td>
<td>72.1</td>
</tr>
<tr>
<td>[Poon and Vanderwende, 2010]</td>
<td>N/A</td>
</tr>
<tr>
<td>[Miwa et al., 2010]</td>
<td>—</td>
</tr>
</tbody>
</table>

(dev$_{GA}$ is the development section with gold anchors)
Outline

1. BioNLP shared task
2. Previous approaches
3. Event Parsing
4. Experiments
5. Future work
   - Document-level parsing
   - DAG parsing
6. Conclusion
Document-level parsing

- All existing systems are restricted to events within a sentence
Future work

Document-level parsing

- All existing systems are restricted to events within a sentence
- Recall: ≈ 7% of events cross sentences boundaries
Document-level parsing

- All existing systems are restricted to events within a sentence
- Recall: \( \approx 7\% \) of events cross sentence boundaries
- We can parse an entire document at once naturally
Document-level parsing

- All existing systems are restricted to events within a sentence
- Recall: $\approx 7\%$ of events cross sentences boundaries
- We can parse an entire document at once naturally
- Adjust features:
  - Need a notion of sentence distance between entities
Document-level parsing

- All existing systems are restricted to events within a sentence
- Recall: \( \approx 7\% \) of events cross sentences boundaries
- We can parse an entire document at once naturally
- Adjust features:
  - Need a notion of sentence distance between entities
  - Dependency paths can cross sentences
Document-level parsing

- All existing systems are restricted to events within a sentence
- Recall: $\approx 7\%$ of events cross sentences boundaries
- We can parse an entire document at once naturally
- Adjust features:
  - Need a notion of sentence distance between entities
  - Dependency paths can cross sentences
- Currently performs $\approx 3\%$ worse than sentence-level parsing
DAG parsing

- Event parse trees become DAGs in the presence of conjunctions
DAG parsing

- Event parse trees become DAGs in the presence of conjunctions
- Rule-based or learned heuristics currently handle this
DAG parsing

- Event parse trees become DAGs in the presence of conjunctions
- Rule-based or learned heuristics currently handle this
- Relatively little work on DAG parsing
DAG parsing

- Event parse trees become DAGs in the presence of conjunctions
- Rule-based or learned heuristics currently handle this
- Relatively little work on DAG parsing
- [Sagae and Tsujii, COLING 2008] shows how to do it in MaltParser
DAG parsing

- Event parse trees become DAGs in the presence of conjunctions
- Rule-based or learned heuristics currently handle this
- Relatively little work on DAG parsing
- [Sagae and Tsujii, COLING 2008] shows how to do it in MaltParser
  - New action adds an additional parent to nodes
DAG parsing

- Event parse trees become DAGs in the presence of conjunctions
- Rule-based or learned heuristics currently handle this
- Relatively little work on DAG parsing
  - [Sagae and Tsujii, COLING 2008] shows how to do it in MaltParser
    - New action adds an additional parent to nodes
  - Maybe TurboParser [Martins and Smith, ACL 2009] can do this by adjusting constraints
Summary

- New approach to event extraction
Summary

- New approach to event extraction
  - Parsing can be used for event extraction
Summary

- New approach to event extraction
  - Parsing can be used for event extraction
  - Reranker further improves performance

Conclusion

It’s over!
Summary

- New approach to event extraction
  - Parsing can be used for event extraction
  - Reranker further improves performance
- vs. pipelined systems: can handle more global features
Summary

- New approach to event extraction
  - Parsing can be used for event extraction
  - Reranker further improves performance
- vs. pipelined systems: can handle more global features
- vs. Markov Logic: faster inference, features instead of formulae
Summary

- New approach to event extraction
  - Parsing can be used for event extraction
  - Reranker further improves performance
- vs. pipelined systems: can handle more global features
- vs. Markov Logic: faster inference, features instead of formulae
- Performance close to state-of-the-art systems
Summary

- New approach to event extraction
  - Parsing can be used for event extraction
  - Reranker further improves performance

- vs. pipelined systems: can handle more global features
- vs. Markov Logic: faster inference, features instead of formulae
- Performance close to state-of-the-art systems

Questions?