Why Parsing?

- Who did what to whom?
- Required for natural language understanding (most likely).
- Syntactic/semantic connection.
- Key task in the Natural Language Processing (NLP) community.

Syntacto-semantic Dependencies

![syntacto-semantic dependencies diagram]

- Transition-based Dependency Parsing

![transition-based dependency parsing diagram]

Previous Work and Conceptual Problems

- Vector composition and constituency parsing:
  - Recursive Neural Network (RNN) model (Socher et al. 2010).
  - Works within the constituency tree.
  - Produces phrase representations and constituency trees.
- "Vanilla" RNN approach not applicable to dependency trees:
  - Different number of parents for non-sources.
  - Can not handle non-projectives.

A Compositional Vector Framework

- Vector representations as opposed to words/trees.
- Compose the representations and predict a transition.
- Compositional and non-compositional transitions.
- Results in a Transition Directed Acyclic Graph (DAG).

Model

- Arc-Standard algorithm cast in our framework.
- Replace the head with the composition of the head/dependant.
- Greedy search and global weight updates.
- observes top 3 representations of the stack/buffer (horizon).
- Single composition matrix $W_C \in \mathbb{R}^n \times 6n$.
- SoftMax weights $W_s \in \mathbb{R}^3 \times n$.
- Only word representations as input $a_i \in \mathbb{R}^n$.

Why Parsing?

- Who did what to whom?
- Required for natural language understanding (most likely).
- Syntactic/semantic connection.
- Key task in the Natural Language Processing (NLP) community.

Composition and Parsing Example

![composition and parsing example diagram]

Training

- Generate gold Transition DAGs using oracle transitions.
- 200-dimensional word representations by Turian et al. (2010).
- Diagonal version of AdaGrad for optimisation.

Quantitative Results

<table>
<thead>
<tr>
<th>Model</th>
<th>UAS</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) This work</td>
<td>86.25%</td>
</tr>
<tr>
<td>(2) Comparable Feature-based System</td>
<td>88.06%</td>
</tr>
<tr>
<td>(3) Shared-task Top System</td>
<td>92.45%</td>
</tr>
</tbody>
</table>

| Table: Unlabeled Attachment Score (UAS) for our model. |

Qualitative Results

- Examples for non-projectives and other algorithms can be found in the paper.

Conclusions and Future Work

- Conclusions:
  - First Deep Learning-based approach to dependency parsing.
  - Performs within 2% UAS to a comparable feature-based model.
  - Produces similar phrase representations as Socher et al. (2010).
- Future work:
  - Compositional vector parsing for "any" language.
  - "Horizon-Free" dependency parsing.
  - Untied weights and other improvements to reach for the state-of-the-art.