Universal Semantic Parsing with Neural Networks

Daniel Hershcovitch
Advisors: Ari Rappoport and Omri Abend

PhD Lecture

February 5, 2019
Machine translation:

After graduation, John moved to Copenhagen

ג'ון עבר לקלופנגן אחריו שסיים את הלימודים
Natural Language Processing: What’s It Good For?

Named entity recognition:

ג’ון עבר לקופנגן אחר,绅士们, at the conference.

Location  Person
Text simplification:

ג'ן עבר לקופנגן אחריו شسمي את הלימודים

ג'ן סים את הלימודים. ג'ן עבר לקופנגן.
After graduation, John moved to Copenhagen

ג"孫 עבר לברוקנCodeGen אחריו, שם סיים את הלימודים

ג"孫 סיים את הלימודים. ג"孫 עבר לברוקנCodeGen.

Sequence-to-sequence sometimes works, but lacks inductive bias.
Linguistic Structured Representations

Model explicit relations between words or concepts.

Example: syntactic/semantic bi-lexical dependencies.
Semantic Representations

Abstract away from detail that does not affect meaning:

\[
\begin{align*}
\text{rest} & \approx \text{take a break} \\
\text{.graduation} & \approx \text{סימ אט הלימודים}
\end{align*}
\]
After graduation, John moved to Copenhagen.
Outline

1. Background: The UCCA Semantic Representation Scheme

2. A Transition-Based DAG Parser for UCCA (ACL’17)

3. Multitask Parsing across Semantic Representations (ACL’18)

4. Content Differences between Syntactic and Semantic Representations (under submission)
Universal Conceptual Cognitive Annotation (UCCA)

Supports rapid and intuitive annotation of linguistic semantic phenomena. [Abend and Rappoport, 2013]

Diagram: After graduation, John moved to Copenhagen.
Universal Conceptual Cognitive Annotation (UCCA)

Supports rapid and intuitive annotation of linguistic semantic phenomena. Cross-linguistically applicable and stable [Sulem et al., 2015].
Universal Conceptual Cognitive Annotation (UCCA)

Supports rapid and intuitive annotation of linguistic semantic phenomena. Cross-linguistically applicable and stable [Sulem et al., 2015].
Semantics-based **evaluation** of

- Machine translation [Birch et al., 2016].
- Text simplification [Sulem et al., 2018a].
- Grammatical error correction [Choshen and Abend, 2018].
UCCA Applications

Semantics-based **evaluation** of

- Machine translation [Birch et al., 2016].
- Text simplification [Sulem et al., 2018a].
- Grammatical error correction [Choshen and Abend, 2018].

Sentence splitting for text simplification [Sulem et al., 2018b].
UCCA structures are directed acyclic graphs (DAGs) with labeled edges. Text tokens are terminals, complex units are non-terminal nodes.
UCCA structures are directed acyclic graphs (DAGs) with labeled edges. Text tokens are terminals, complex units are non-terminal nodes. Phrases may be discontinuous.
UCCA structures are directed acyclic graphs (DAGs) with labeled edges. Text tokens are terminals, complex units are non-terminal nodes. Phrases may be discontinuous. Remote edges enable reentrancy.
Structural Properties

(1) non-terminal nodes

John and Mary went home

(2) discontinuity

John gave everything up

(3) reentrancy

After graduation, John moved to Copenhagen
UCCA Data

- English Wikipedia articles (Wiki).
- English-French-German parallel corpus from *Twenty Thousand Leagues Under the Sea* (20K).
- Reviews from the English Web Treebank (EWT).
## Data Statistics

<table>
<thead>
<tr>
<th></th>
<th>Wiki</th>
<th>20K</th>
<th>EWT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>en</td>
<td>en</td>
<td>fr</td>
</tr>
<tr>
<td># sentences</td>
<td>5,141</td>
<td>492</td>
<td>492</td>
</tr>
<tr>
<td># tokens</td>
<td>158,739</td>
<td>12,638</td>
<td>13,021</td>
</tr>
<tr>
<td># non-terminal nodes</td>
<td>62,002</td>
<td>4,699</td>
<td>5,110</td>
</tr>
<tr>
<td>% discontinuous</td>
<td>1.71</td>
<td>3.19</td>
<td>4.64</td>
</tr>
<tr>
<td>% reentrant</td>
<td>1.84</td>
<td>0.89</td>
<td>0.65</td>
</tr>
<tr>
<td># edges</td>
<td>208,937</td>
<td>16,803</td>
<td>17,520</td>
</tr>
<tr>
<td>% primary</td>
<td>97.40</td>
<td>96.79</td>
<td>97.02</td>
</tr>
<tr>
<td>% remote</td>
<td>2.60</td>
<td>3.21</td>
<td>2.98</td>
</tr>
</tbody>
</table>
Outline

1. Background: The UCCA Semantic Representation Scheme

2. A Transition-Based DAG Parser for UCCA (ACL’17)

3. Multitask Parsing across Semantic Representations (ACL’18)

4. Content Differences between Syntactic and Semantic Representations (under submission)
Parses text $w_1 \ldots w_n$ to graph $G$ incrementally by applying transitions to the parser state, consisting of: stack, buffer and constructed graph.
TUPA: Transition-based UCCA Parser

Parses text $w_1 \ldots w_n$ to graph $G$ incrementally by applying transitions to the parser state, consisting of: stack, buffer and constructed graph.

Initial state:

<table>
<thead>
<tr>
<th>stack</th>
<th>buffer</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Initial Symbol]</td>
<td>They thought about taking a short break</td>
</tr>
</tbody>
</table>
Parses text $w_1 \ldots w_n$ to graph $G$ incrementally by applying transitions to the parser state, consisting of: stack, buffer and constructed graph.

**Initial state:**

<table>
<thead>
<tr>
<th>stack</th>
<th>buffer</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="bullet.png" alt="" /></td>
<td>They thought about taking a short break</td>
</tr>
</tbody>
</table>

**TUPA transitions:**

\{\texttt{Shift}, \texttt{Reduce}, \texttt{Node}_X, \texttt{Left-Edge}_X, \texttt{Right-Edge}_X, \texttt{Left-Remote}_X, \texttt{Right-Remote}_X, \texttt{Swap}, \texttt{Finish}\}

These transitions enable non-terminal nodes, reentrancy and discontinuity.
Example: TUPA Transition Sequence

$\Rightarrow$ **SHIFT**

<table>
<thead>
<tr>
<th>stack</th>
<th>buffer</th>
</tr>
</thead>
<tbody>
<tr>
<td>●</td>
<td>thought about taking a short break</td>
</tr>
<tr>
<td>They</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>graph</th>
</tr>
</thead>
<tbody>
<tr>
<td>●</td>
</tr>
</tbody>
</table>
Example: TUPA Transition Sequence

⇒ \textsc{Right-Edge}_A
Example: TUPA Transition Sequence

\[ \Rightarrow \text{SHIFT} \]

stack

|  ●  | They | thought |

buffer

| about | taking | a | short | break |

graph

They

A
Example: TUPA Transition Sequence

⇒ Swap

stack

<table>
<thead>
<tr>
<th></th>
<th>thought</th>
</tr>
</thead>
</table>

| They  | about   | taking | a  | short | break |

buffer

graph

They

A
Example: TUPA Transition Sequence

⇒ \textbf{RIGHT-EDGE}_P

stack

\begin{tabular}{c}
\textbf{thought} \\
\end{tabular}

buffer

\begin{tabular}{c c c c c}
They & about & taking & a & short & break \\
\end{tabular}

graph

\begin{tabular}{c}
They \\
\end{tabular}

\begin{tabular}{c}
thought \\
\end{tabular}
Example: TUPA Transition Sequence

$\Rightarrow$ REDUCE

stack

buffer

They about taking a short break

They thought

graph

A P
Example: TUPA Transition Sequence

⇒ Shift

stack

buffer

They

about taking a short break

graph

They thought
Example: TUPA Transition Sequence

\[ \Rightarrow \text{SHIFT} \]

<table>
<thead>
<tr>
<th>stack</th>
<th>buffer</th>
</tr>
</thead>
<tbody>
<tr>
<td>● They about</td>
<td>taking a short break</td>
</tr>
</tbody>
</table>

graph

They thought

A P
Example: TUPA Transition Sequence

⇒ NODE"R

stack

<table>
<thead>
<tr>
<th></th>
<th>They</th>
<th>about</th>
</tr>
</thead>
</table>

buffer

|   | taking | a | short | break |

graph

They

thought

about

Daniel Hershcovich
February 5, 2019
Example: TUPA Transition Sequence

⇒ REDUCE

stack

| ● | They |

buffer

| ● | taking | a | short | break |

daniel hershcovich
Example: TUPA Transition Sequence

⇒ **SHIFT**

```
stack
● | They | ●

buffer
taking | a | short | break

graph
They

A
P

thought

R

about
```
$\Rightarrow$ \textsc{Left-Remote}_A
Example: TUPA Transition Sequence

\[ \Rightarrow \text{SHIFT} \]

<table>
<thead>
<tr>
<th>stack</th>
<th>buffer</th>
</tr>
</thead>
<tbody>
<tr>
<td>● They ● taking</td>
<td>a short break</td>
</tr>
</tbody>
</table>

Graph:
- They
- thought
- about
- taking a short break
Example: TUPA Transition Sequence

\[ \Rightarrow \text{NODE}_C \]
Example: TUPA Transition Sequence

⇒ REDUCE

stack

<table>
<thead>
<tr>
<th></th>
<th>They</th>
<th></th>
</tr>
</thead>
</table>

buffer

|   | a    | short | break |

They took a short break about taking a thought.
Example: TUPA Transition Sequence

⇒ \textbf{SHIFT}

stack

\begin{tabular}{|c|c|c|}
\hline
\textbullet & They & \textbullet \hspace{0.5cm} \textbullet \\
\hline
\end{tabular}

buffer

\begin{tabular}{|c|c|c|}
\hline
a & short & break \\
\hline
\end{tabular}

graph

They \hspace{0.5cm} thought \hspace{0.5cm} about \hspace{0.5cm} taking

Daniel Hershcovich
February 5, 2019 15 / 39
Example: TUPA Transition Sequence

\[ \Rightarrow \text{RIGHT-EDGE}_P \]

<table>
<thead>
<tr>
<th>stack</th>
<th>buffer</th>
</tr>
</thead>
<tbody>
<tr>
<td>● They</td>
<td>● ○ short break</td>
</tr>
</tbody>
</table>

They thought about taking a short break.
Example: TUPA Transition Sequence

⇒ **SHIFT**

<table>
<thead>
<tr>
<th>stack</th>
<th>buffer</th>
</tr>
</thead>
<tbody>
<tr>
<td>● They</td>
<td>● short</td>
</tr>
<tr>
<td>● a</td>
<td>● break</td>
</tr>
</tbody>
</table>

They a short break

They thought about taking

Graph representation of the transition sequence.
Example: TUPA Transition Sequence

⇒ \text{RIGHT-EDGE}_F

\begin{itemize}
  \item stack
    \begin{tabular}{c|c|c|c}
      & They & a \hline
    \end{tabular}
  \item buffer
    \begin{tabular}{l|l}
      short & break \hline
    \end{tabular}
\end{itemize}
Example: TUPA Transition Sequence

 ⇒ REDUCE

stack

<table>
<thead>
<tr>
<th></th>
<th>They</th>
<th></th>
<th></th>
</tr>
</thead>
</table>

buffer

| short | break  |

They short break

A thought about taking a

A

F

P

R

F

graph
Example: TUPA Transition Sequence

⇒ \textbf{SHIFT}

<table>
<thead>
<tr>
<th>stack</th>
<th>buffer</th>
</tr>
</thead>
<tbody>
<tr>
<td>●</td>
<td>They</td>
</tr>
<tr>
<td>○</td>
<td>●</td>
</tr>
<tr>
<td>● red</td>
<td>short</td>
</tr>
<tr>
<td></td>
<td>break</td>
</tr>
</tbody>
</table>

graph

They
thought
about
taking

Daniel Hershcovich
February 5, 2019
Example: TUPA Transition Sequence

⇒ SWAP

stack

<table>
<thead>
<tr>
<th></th>
<th>They</th>
<th>short</th>
</tr>
</thead>
</table>

buffer

|     | break |

graph

They thought about F a

taking
Example: TUPA Transition Sequence

$\Rightarrow \text{RIGHT-EDGE}_D$

<table>
<thead>
<tr>
<th>stack</th>
<th>buffer</th>
</tr>
</thead>
<tbody>
<tr>
<td>They</td>
<td>short</td>
</tr>
</tbody>
</table>

They thought about taking a short break.

Diagram showing the transition sequence with nodes representing words and transitions.
Example: TUPA Transition Sequence

$\Rightarrow$ REDUCE

<table>
<thead>
<tr>
<th>stack</th>
<th>buffer</th>
</tr>
</thead>
<tbody>
<tr>
<td>⬤ They</td>
<td>⬤ break</td>
</tr>
</tbody>
</table>

They break

Thought

About

Taking a short

daniel hershcovich
Example: TUPA Transition Sequence

⇒ SWAP

stack

```
[ ] [ ]
```

buffer

```
[They] [break]
```

graph

```
They
thought
about
taking
```

```
A
P
```

```
D
R
P
F
F
```

Daniel Hershcovitch
Example: TUPA Transition Sequence

$\Rightarrow \text{RIGHT-EDGE}_A$
Example: TUPA Transition Sequence

$\Rightarrow \text{REDUCE}$

- **Stack:**
  - They

- **Buffer:**
  - They
  - break

Graph:

- They
- thought
- about
- taking
- a
- short
- A
- P
- A
- R
- P
- D

Diagram:

```
They
+-- thought
    +-- about
        +-- taking
            +-- a
                +-- short
                    +-- A
                        +-- P
                            +-- A
                                +-- R
                                    +-- P
                                        +-- D
                                            +-- F
                                                +-- F
```
Example: TUPA Transition Sequence

⇒ REDUCE

stack

buffer

They

break

diagram

They

thought

about

taking

a

short

A

P

A

R

P

D

A

F

F
Example: TUPA Transition Sequence

⇒ SHIFT

stack
They

buffer
break

They

A

thought

about

taking

a

short

A

A

P

R

P

D

F

F

graph

February 5, 2019 15 / 39
Example: TUPA Transition Sequence

⇒ REDUCE

They thought about taking a short break.

Diagram:

- Stack: [ ]
- Buffer: break
- Graph:
  - They
  - thought
  - about
  - taking
  - a short
Transition-based UCCA Parser

Example: TUPA Transition Sequence

⇒ Shift

They thought about taking a short break

graph

They

thought

about

taking

a

short

stack

buffer

break

Daniel Hershcovitch
February 5, 2019 15 / 39
Example: TUPA Transition Sequence

⇒ \textbf{RIGHT-EDGE}_C

They thought about taking a short break

daniel hershcovich
Example: TUPA Transition Sequence

⇒ FINISH

They thought about taking a short break.
An *oracle* provides the transition sequence given the correct graph:

\[
\text{They} \xrightarrow{\text{Shift}} \text{thought} \xrightarrow{\text{Shift}} \text{about} \xrightarrow{\text{Shift}} \text{taking} \xrightarrow{\text{Shift}} \text{break}
\]

**SHIFT, RIGHT-EDGE}_A, \text{SHIFT, SWAP, RIGHT-EDGE}_P, \text{REDUCE, SHIFT, SHIFT, NODE}_R, \text{REDUCE, LEFT-REMOTE}_A, \text{SHIFT, SHIFT, NODE}_C, \text{REDUCE, SHIFT, RIGHT-EDGE}_P, \text{SHIFT, RIGHT-EDGE}_F, \text{REDUCE, SHIFT, SWAP, RIGHT-EDGE}_D, \text{REDUCE, SWAP, RIGHT-EDGE}_A, \text{REDUCE, REDUCE, SHIFT, REDUCE, SHIFT, RIGHT-EDGE}_C, \text{FINISH}
TUPA Model

Learns to greedily predict transition based on current state. Experimenting with three classifiers:

- **Sparse**: Perceptron with sparse features.
- **MLP**: Word embeddings + MLP.
- **BiLSTM**: Word embeddings + bidirectional RNN + MLP.

Features include:
{words, parts of speech, syntactic dependencies, existing edge labels} from the stack and buffer + parents, children, grandchildren.
Learns to greedily predict transition based on current state. Experimenting with three classifiers:

**Sparse** Perceptron with sparse features.
**MLP** Word embeddings + MLP.
**BiLSTM** Word embeddings + bidirectional RNN + MLP.
Learns to greedily predict transition based on current state. Experimenting with three classifiers:

- **Sparse**: Perceptron with sparse features.
- **MLP**: Word embeddings + MLP.
- **BiLSTM**: Word embeddings + bidirectional RNN + MLP.

They thought about taking a short break.
Transition-based UCCA Parser

TUPA Model

Learns to greedily predict transition based on current state.
Experimenting with three classifiers:

**Sparse**  Perceptron with sparse features.

**MLP**  Word embeddings + MLP.

**BiLSTM**  Word embeddings + **bidirectional** RNN + MLP.
Learns to greedily predict transition based on current state. Experimenting with three classifiers:

**Sparse**  Perceptron with sparse features.
**MLP**  Word embeddings + MLP.
**BiLSTM**  Word embeddings + bidirectional RNN + MLP.
They are taking a short break.
Comparing to Existing Methods

Using conversion-based approximation as baseline, with bi-lexical DAG parsers and transition-based tree parsers.

They thought about taking a short break

UCCA bi-lexical DAG approximation.
Bi-lexical Graph Approximation

1. Convert UCCA to bi-lexical DAGs.
2. Train bi-lexical parsers.
3. Parse test set.
4. Convert to UCCA.
5. Evaluate.

After graduation, John moved to Copenhagen.
Evaluation

1. Match primary edges between the graphs by terminal yield and label.
2. Calculate precision, recall and F1 scores.
3. Repeat for remote edges.
1. Match primary edges between the graphs by terminal yield and label.
2. Calculate **precision, recall and F1** scores.
3. Repeat for remote edges.

**Primary**

<table>
<thead>
<tr>
<th>P</th>
<th>R</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{6}{9} = 67%$</td>
<td>$\frac{6}{10} = 60%$</td>
<td>64%</td>
</tr>
</tbody>
</table>

**Remote**

<table>
<thead>
<tr>
<th>P</th>
<th>R</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{1}{2} = 50%$</td>
<td>$\frac{1}{1} = 100%$</td>
<td>67%</td>
</tr>
</tbody>
</table>
**Results**

**TUPA\textsubscript{BiLSTM}** outperforms all other methods on the English Wiki test set:

<table>
<thead>
<tr>
<th></th>
<th>Primary</th>
<th>Remote</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>English Wiki</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TUPA</td>
<td>64.1</td>
<td>16</td>
</tr>
<tr>
<td>Sparse</td>
<td>64.9</td>
<td>16.9</td>
</tr>
<tr>
<td>MLP</td>
<td>73.2</td>
<td>46.8</td>
</tr>
<tr>
<td>BiLSTM</td>
<td>73.2</td>
<td>46.8</td>
</tr>
<tr>
<td><strong>Baselines</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DAGParser</td>
<td>58.6</td>
<td>1</td>
</tr>
<tr>
<td>TurboParser</td>
<td>51.2</td>
<td>3.7</td>
</tr>
<tr>
<td>MaltParser</td>
<td>60.2</td>
<td></td>
</tr>
<tr>
<td>StackLSTM</td>
<td>69.9</td>
<td></td>
</tr>
<tr>
<td>UPARSE</td>
<td>61.1</td>
<td></td>
</tr>
</tbody>
</table>
...and also on the **out-of-domain** English 20K:

<table>
<thead>
<tr>
<th></th>
<th>English Wiki</th>
<th></th>
<th>English 20K</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Primary</td>
<td>Remote</td>
<td>Primary</td>
<td>Remote</td>
</tr>
<tr>
<td></td>
<td>F1</td>
<td>F1</td>
<td>F1</td>
<td>F1</td>
</tr>
<tr>
<td>TUPA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sparse</td>
<td>64.1</td>
<td>16</td>
<td>59.8</td>
<td>11.5</td>
</tr>
<tr>
<td>MLP</td>
<td>64.9</td>
<td>16.9</td>
<td>62.5</td>
<td>9.7</td>
</tr>
<tr>
<td>BiLSTM</td>
<td><strong>73.2</strong></td>
<td><strong>46.8</strong></td>
<td><strong>67.9</strong></td>
<td><strong>23.0</strong></td>
</tr>
<tr>
<td>Baselines</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DAGParser</td>
<td>58.6</td>
<td>1</td>
<td>53.4</td>
<td></td>
</tr>
<tr>
<td>TurboParser</td>
<td>51.2</td>
<td>3.7</td>
<td>43.1</td>
<td>0.8</td>
</tr>
<tr>
<td>MaltParser</td>
<td>60.2</td>
<td></td>
<td>55.3</td>
<td></td>
</tr>
<tr>
<td>StackLSTM</td>
<td>69.9</td>
<td></td>
<td>63.5</td>
<td></td>
</tr>
<tr>
<td>UPARSE</td>
<td>61.1</td>
<td></td>
<td>52.8</td>
<td></td>
</tr>
</tbody>
</table>
## Results

<table>
<thead>
<tr>
<th></th>
<th>English Wiki</th>
<th>English 20K</th>
<th>French 20K</th>
<th>German 20K</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Primary F1</td>
<td>Remote F1</td>
<td>Primary F1</td>
<td>Remote F1</td>
</tr>
<tr>
<td>TUPA</td>
<td>64.1</td>
<td>16</td>
<td>59.8</td>
<td>11.5</td>
</tr>
<tr>
<td>Sparse</td>
<td>64.9</td>
<td>16.9</td>
<td>62.5</td>
<td>9.7</td>
</tr>
<tr>
<td>BiLSTM</td>
<td><strong>73.2</strong></td>
<td><strong>46.8</strong></td>
<td><strong>67.9</strong></td>
<td><strong>23.0</strong></td>
</tr>
<tr>
<td>Baselines</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DAGParser</td>
<td>58.6</td>
<td>1</td>
<td>53.4</td>
<td></td>
</tr>
<tr>
<td>TurboParser</td>
<td>51.2</td>
<td>3.7</td>
<td>43.1</td>
<td>0.8</td>
</tr>
<tr>
<td>MaltParser</td>
<td>60.2</td>
<td></td>
<td>55.3</td>
<td></td>
</tr>
<tr>
<td>StackLSTM</td>
<td>69.9</td>
<td>63.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UPARSE</td>
<td>61.1</td>
<td>52.8</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Structured meaning representation benefits language understanding.

UCCA’s semantic distinctions require a graph structure including non-terminals, reentrancy and discontinuity.

TUPA is an accurate transition-based UCCA parser, and the first to support UCCA and any DAG over the text tokens.

Outperforms strong conversion-based baselines.
Structured meaning representation benefits language understanding.

UCCA’s semantic distinctions require a graph structure including non-terminals, reentrancy and discontinuity.

TUPA is an accurate transition-based UCCA parser, and the first to support UCCA and any DAG over the text tokens.

Outperforms strong conversion-based baselines.

Up next:
- Parsing other semantic representations.
- Comparing representations through conversion.
Outline

1. Background: The UCCA Semantic Representation Scheme
2. A Transition-Based DAG Parser for UCCA (ACL'17)
3. Multitask Parsing across Semantic Representations (ACL’18)
4. Content Differences between Syntactic and Semantic Representations (under submission)
After graduation, John moved to Copenhagen.
After graduation, John moved to Copenhagen.
After graduation, John moved to Copenhagen.
After graduation, John moved to Copenhagen.
After graduation, John moved to Copenhagen

UD (Universal Dependencies)
### Data

**UCCA training data is scarce**

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Sentences</th>
</tr>
</thead>
<tbody>
<tr>
<td>UCCA</td>
<td>5,141</td>
</tr>
<tr>
<td>AMR</td>
<td>36,521</td>
</tr>
<tr>
<td>DM</td>
<td>33,964</td>
</tr>
<tr>
<td>UD</td>
<td>17,062</td>
</tr>
</tbody>
</table>

(English)
UCCA training data is scarce

(English)

<table>
<thead>
<tr>
<th>Data Source</th>
<th>UCCA</th>
<th>AMR</th>
<th>DM</th>
<th>UD</th>
</tr>
</thead>
<tbody>
<tr>
<td>UCCA</td>
<td>5,141 sentences</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMR</td>
<td></td>
<td>36,521 sentences</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DM</td>
<td></td>
<td></td>
<td>33,964 sentences</td>
<td></td>
</tr>
<tr>
<td>UD</td>
<td></td>
<td></td>
<td></td>
<td>17,062 sentences</td>
</tr>
</tbody>
</table>

and domains are limited.

- UCCA: Wikipedia, books
- AMR: blogs, news, emails, reviews
- DM: news
- UD: blogs, news, emails, reviews, Q&A
After graduation, John moved to Copenhagen.
After graduation, John moved to Copenhagen.
After graduation, John moved to Copenhagen.
### Results

<table>
<thead>
<tr>
<th></th>
<th>Primary F1</th>
<th>Remote F1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>English Wiki (in-domain)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Single-task</td>
<td>73.2</td>
<td>46.8</td>
</tr>
<tr>
<td>+AMR</td>
<td>72.7</td>
<td>52.7</td>
</tr>
<tr>
<td>+DM</td>
<td><strong>74.0</strong></td>
<td><strong>53.8</strong></td>
</tr>
<tr>
<td>+UD</td>
<td>72.2</td>
<td>48.0</td>
</tr>
<tr>
<td>+AMR+DM</td>
<td>73.6</td>
<td>48.5</td>
</tr>
<tr>
<td>+AMR+UD</td>
<td>73.3</td>
<td>51.2</td>
</tr>
<tr>
<td>+DM+UD</td>
<td>73.9</td>
<td>52.2</td>
</tr>
<tr>
<td>All</td>
<td>73.8</td>
<td>52.1</td>
</tr>
</tbody>
</table>
## Results

<table>
<thead>
<tr>
<th>English Wiki (in-domain)</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary F1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Remote F1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Single-task</td>
<td>73.2</td>
<td>46.8</td>
</tr>
<tr>
<td>+AMR</td>
<td>72.7</td>
<td>52.7</td>
</tr>
<tr>
<td>+DM</td>
<td>74.0</td>
<td>53.8</td>
</tr>
<tr>
<td>+UD</td>
<td>72.2</td>
<td>48.0</td>
</tr>
<tr>
<td>+AMR+DM</td>
<td>73.6</td>
<td>48.5</td>
</tr>
<tr>
<td>+AMR+UD</td>
<td>73.3</td>
<td>51.2</td>
</tr>
<tr>
<td>+DM+UD</td>
<td>73.9</td>
<td>52.2</td>
</tr>
<tr>
<td>All</td>
<td>73.8</td>
<td>52.1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>English 20K (out-of-domain)</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary F1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Remote F1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Single-task</td>
<td>67.9</td>
<td>23.0</td>
</tr>
<tr>
<td>+AMR</td>
<td>67.0</td>
<td>31.2</td>
</tr>
<tr>
<td>+DM</td>
<td>69.1</td>
<td>27.5</td>
</tr>
<tr>
<td>+UD</td>
<td>67.4</td>
<td>23.9</td>
</tr>
<tr>
<td>+AMR+DM</td>
<td>68.9</td>
<td>25.4</td>
</tr>
<tr>
<td>+AMR+UD</td>
<td>68.3</td>
<td>31.4</td>
</tr>
<tr>
<td>+DM+UD</td>
<td>68.6</td>
<td>29.1</td>
</tr>
<tr>
<td>All</td>
<td>69.1</td>
<td>25.8</td>
</tr>
</tbody>
</table>
## Results

### Primary F1 Remote F1

<table>
<thead>
<tr>
<th></th>
<th>English Wiki (in-domain)</th>
<th>English 20K (out-of-domain)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Primary F1</td>
<td>Remote F1</td>
</tr>
<tr>
<td>Single-task</td>
<td>73.2</td>
<td>46.8</td>
</tr>
<tr>
<td>+AMR</td>
<td>72.7</td>
<td>52.7</td>
</tr>
<tr>
<td>+DM</td>
<td><strong>74.0</strong></td>
<td><strong>53.8</strong></td>
</tr>
<tr>
<td>+UD</td>
<td>72.2</td>
<td>48.0</td>
</tr>
<tr>
<td>+AMR+DM</td>
<td>73.6</td>
<td>48.5</td>
</tr>
<tr>
<td>+AMR+UD</td>
<td>73.3</td>
<td>51.2</td>
</tr>
<tr>
<td>+DM+UD</td>
<td>73.9</td>
<td>52.2</td>
</tr>
<tr>
<td>All</td>
<td><strong>73.8</strong></td>
<td>52.1</td>
</tr>
</tbody>
</table>

### French 20K (in-domain)

<table>
<thead>
<tr>
<th></th>
<th>Primary F1</th>
<th>Remote F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single-task</td>
<td>44.0</td>
<td>3.8</td>
</tr>
<tr>
<td>+UD</td>
<td><strong>49.6</strong></td>
<td><strong>1.6</strong></td>
</tr>
</tbody>
</table>

### German 20K (in-domain)

<table>
<thead>
<tr>
<th></th>
<th>Primary F1</th>
<th>Remote F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single-task</td>
<td><strong>73.9</strong></td>
<td>47.2</td>
</tr>
<tr>
<td>+UD</td>
<td><strong>80.1</strong></td>
<td><strong>59.8</strong></td>
</tr>
</tbody>
</table>
TUPA output:
(Single-task)

No transoceanic navigational undertaking has been conducted

Multitask TUPA output:
(+AMR+DM+UD)

No transoceanic navigational undertaking has been conducted with more ability no business dealings have been crowned with greater success
Outline

1. Background: The UCCA Semantic Representation Scheme
2. A Transition-Based DAG Parser for UCCA (ACL’17)
3. Multitask Parsing across Semantic Representations (ACL’18)
4. Content Differences between Syntactic and Semantic Representations (under submission)
Many formal differences.

Semantic representation:

UCCA

Syntactic representation:

UD

What about content?
Many formal differences.

Semantic representation:

**UCCA**

```
After graduation, John moved to Copenhagen
```

Syntactic representation:

**UD**

```
After graduation, John moved to Copenhagen
```

Daniel Hershcovich

February 5, 2019
Many formal differences.

What about \textit{content}?
After graduation, John moved to Copenhagen.

UD

Now we can evaluate by matching edges (UCCA unlabeled evaluation):

F1 = 89%

F1 = 80%

F1 = 84%
Now we can evaluate by matching edges (UCCA unlabeled evaluation)
Now we can evaluate by matching edges (UCCA unlabeled evaluation)
Now we can evaluate by matching edges (UCCA unlabeled evaluation)

<table>
<thead>
<tr>
<th>P</th>
<th>R</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{8}{9} = 89%$</td>
<td>$\frac{8}{10} = 80%$</td>
<td>$84%$</td>
</tr>
</tbody>
</table>
### Confusion Matrix

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>L</th>
<th>N</th>
<th>P</th>
<th>Q</th>
<th>R</th>
<th>S</th>
<th>T</th>
<th>∅</th>
</tr>
</thead>
<tbody>
<tr>
<td>acl</td>
<td>8</td>
<td>101</td>
<td>2</td>
<td>15</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>49</td>
<td>97</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>advcl</td>
<td>2</td>
<td>2</td>
<td>103</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>advmod</td>
<td>61</td>
<td>9</td>
<td>51</td>
<td>12</td>
<td>33</td>
<td>3</td>
<td>61</td>
<td>2</td>
<td>10</td>
<td>6</td>
<td>5</td>
<td>117</td>
<td>71</td>
<td></td>
<td></td>
</tr>
<tr>
<td>amod</td>
<td>1</td>
<td>33</td>
<td>99</td>
<td>197</td>
<td>2</td>
<td>7</td>
<td>3</td>
<td>27</td>
<td>97</td>
<td>2</td>
<td>60</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>appos</td>
<td>1</td>
<td>10</td>
<td>8</td>
<td>5</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>aux</td>
<td>96</td>
<td>285</td>
<td>48</td>
<td>6</td>
<td>1</td>
<td>1</td>
<td>489</td>
<td>50</td>
<td>75</td>
<td>11</td>
<td>41</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>case</td>
<td>1</td>
<td>5</td>
<td>2</td>
<td>14</td>
<td>34</td>
<td>1</td>
<td>305</td>
<td>71</td>
<td>11</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>164</td>
<td></td>
</tr>
<tr>
<td>ccomp</td>
<td>78</td>
<td>8</td>
<td>1</td>
<td>41</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>90</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>compound</td>
<td>23</td>
<td>24</td>
<td>8</td>
<td>176</td>
<td>2</td>
<td>265</td>
<td>3</td>
<td>1</td>
<td>24</td>
<td>3</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>conj</td>
<td>2</td>
<td>88</td>
<td>1</td>
<td>333</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cop</td>
<td>1</td>
<td>2</td>
<td>19</td>
<td>763</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>csubj</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dep</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>det</td>
<td>2</td>
<td>1</td>
<td>993</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>19</td>
<td>2</td>
<td>26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>discourse</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>expl</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>iobj</td>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>list</td>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mark</td>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>nmod</td>
<td>100</td>
<td>1</td>
<td>1</td>
<td>233</td>
<td>1</td>
<td>1</td>
<td>134</td>
<td>1</td>
<td>53</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>18</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>nsubj</td>
<td>993</td>
<td>4</td>
<td>7</td>
<td>6</td>
<td>3</td>
<td>2</td>
<td>9</td>
<td>1</td>
<td>3</td>
<td>24</td>
<td>1</td>
<td>37</td>
<td>37</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>nummod</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>8</td>
<td>1</td>
<td>6</td>
<td>4</td>
<td>92</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>obj</td>
<td>439</td>
<td>1</td>
<td>21</td>
<td>7</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>39</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>obl</td>
<td>247</td>
<td>4</td>
<td>7</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>orphan</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>parataxis</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>vocative</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>xcomp</td>
<td>44</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>116</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>head</td>
<td>125</td>
<td>1402</td>
<td>152</td>
<td>37</td>
<td>91</td>
<td>18</td>
<td>652</td>
<td>2</td>
<td>1</td>
<td>961</td>
<td>18</td>
<td>9</td>
<td>353</td>
<td>1</td>
<td>524</td>
</tr>
<tr>
<td>∅</td>
<td>329</td>
<td>172</td>
<td>34</td>
<td>56</td>
<td>6</td>
<td>5</td>
<td>466</td>
<td>29</td>
<td>141</td>
<td>27</td>
<td>7</td>
<td>98</td>
<td>11</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
After graduation, John moved to Copenhagen.
After graduation, John moved to Copenhagen.
After graduation, John moved to Copenhagen.
They thought about taking a short break.
Multi-word Expressions

UCCA

Converted UD

Daniel Hershcovich
February 5, 2019
Multi-word Expressions

UCCA

Converted UD

They

thought

about
taking

a
short
break

Daniel Hershcovich

February 5, 2019 36 / 39
Linkage between Scenes

UCCA

From the moment you enter, you know.

UD

From the moment you enter, you know.
Linkage between Scenes

UCCA

From the moment you enter, you know

UD

From the moment you enter, you know
Conclusion

- Meaning representation is valuable for language understanding.
Meaning representation is valuable for language understanding.

TUPA, an accurate UCCA parser, is suited to many representations.
Conclusion

- Meaning representation is valuable for language understanding.
- TUPA, an accurate UCCA parser, is suited to many representations.
- Multitask learning allows useful shared generalizations to emerge.
Conclusion

- Meaning representation is valuable for language understanding.
- TUPA, an accurate UCCA parser, is suited to many representations.
- Multitask learning allows useful shared generalizations to emerge.
- Divergences limit inter-scheme gain, but highlight relative strengths.
Conclusion

- Meaning representation is valuable for language understanding.
- TUPA, an accurate UCCA parser, is suited to many representations.
- Multitask learning allows useful shared generalizations to emerge.
- Divergences limit inter-scheme gain, but highlight relative strengths.
Conclusion

- Meaning representation is valuable for language understanding.
- TUPA, an accurate UCCA parser, is suited to many representations.
- Multitask learning allows useful shared generalizations to emerge.
- Divergences limit inter-scheme gain, but highlight relative strengths.

Ongoing work:
- Complement syntax with *lexical* semantics to make up for differences.
Meaning representation is valuable for language understanding.

TUPA, an accurate UCCA parser, is suited to many representations.

Multitask learning allows useful shared generalizations to emerge.

Divergences limit inter-scheme gain, but highlight relative strengths.

Ongoing work:

Complement syntax with *lexical* semantics to make up for differences.

Establish cross-framework meaning representation parsing as a task.
Conclusion

- Meaning representation is valuable for language understanding.
- TUPA, an accurate UCCA parser, is suited to many representations.
- Multitask learning allows useful shared generalizations to emerge.
- Divergences limit inter-scheme gain, but highlight relative strengths.

Ongoing work:
- Complement syntax with *lexical* semantics to make up for differences.
- Establish cross-framework meaning representation parsing as a task.

Long term goal: learning semantic parsing as a means to learn language.
Universal Conceptual Cognitive Annotation (UCCA).
In Proc. of ACL, pages 228–238.


Reference-less measure of faithfulness for grammatical error correction.
In Proc. of NAACL-HLT.

Conceptual annotations preserve structure across translations: A French-English case study.
In Proc. of S2MT, pages 11–22.

Semantic structural annotation for text simplification.
In Proc. of NAACL.

Simple and effective text simplification using semantic and neural methods.
In Proc. of ACL.