Neural Semantic Parsing with Type Constraints

Jayant Krishnamurthy, Pradeep Dasigi, Matt Gardner
Semantic Parsing: Language → Programs

**Question**: Which player was from South Korea?

**Semantic Parsing**

**Logical Form**

\[ ((\text{row2cell athlete}) \ (\text{cell2row south\_korea})) \]

**Knowledge Source**

<table>
<thead>
<tr>
<th>Athlete</th>
<th>Nation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sonja Henie</td>
<td>Norway</td>
</tr>
<tr>
<td>Kim Yu-Na</td>
<td>South Korea</td>
</tr>
</tbody>
</table>

**Execution**

**Answer**: Kim Yu-Na

[Zelle and Mooney, 1996]
If all the frogs died the population of raccoon would _ ?

NP : frog \( \lambda x. \text{decrease}(x) \)  
NP : raccoon \( \lambda x. \lambda e. \lambda f. \text{cause}(e, f(x)) \)

S : decrease(frog)  
(S\S) : \lambda e. \lambda f. \text{cause}(e, f(\text{raccoon}))

S : \lambda f. \text{cause}(\text{decrease}(\text{frog}), f(\text{raccoon}))
Semantic Parsing via Machine Translation

[Andreas et al., 2013]  [Jia and Liang, 2016]  [Dong and Lapata, 2016]  [Locascio et al., 2016]  [Ling et al., 2016]  [Yin and Neubig, 2017]  [Rabinovich et al., 2017]
## Traditional vs Neural Semantic Parsing

<table>
<thead>
<tr>
<th></th>
<th>Guaranteed to produce valid logical form</th>
<th>No grammar induction</th>
<th>Train on Question/Answer Pairs</th>
<th>Joint Entity Linking</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Traditional Semantic Parsing</strong></td>
<td>✔</td>
<td>✗</td>
<td>✔</td>
<td>✗/✔</td>
</tr>
<tr>
<td><strong>Neural Semantic Parsing</strong></td>
<td>✗</td>
<td>✔</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td><strong>This Work</strong></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>
Outline

- WikiTableQuestions
- Neural Semantic Parser
  - Type Constraints
  - Joint Entity Linking
  - QA Supervision
- Experimental Results
<table>
<thead>
<tr>
<th>Athlete</th>
<th>Nation</th>
<th>Olympics</th>
<th>Gold</th>
<th>Silver</th>
<th>Bronze</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gillis Grafström</td>
<td>Sweden</td>
<td>1932</td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Evgeni Plushenko</td>
<td>Russia</td>
<td>2014</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>Sonja Henie</td>
<td>Norway</td>
<td>1936</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Kim Yu-na</td>
<td>S Korea</td>
<td>2014</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Q: Which athlete was from South Korea after the year 2010?
A: Kim Yu-Na

Q: How many more silver medals did Gillis Grafström have than Sonja Henie?
A: 1
<table>
<thead>
<tr>
<th>Athlete</th>
<th>Nation</th>
<th>Olympics</th>
<th>Gold</th>
<th>Silver</th>
<th>Bronze</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gillis Grafström</td>
<td>Sweden</td>
<td>1932</td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Evgeni Plushenko</td>
<td>Russia</td>
<td>2014</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>Sonja Henie</td>
<td>Norway</td>
<td>1936</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Kim Yu-na</td>
<td>S Korea</td>
<td>2014</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Q: Which athlete was from South Korea after the year 2010?

\[
((\text{reverse athlete}) \ (\text{and} \ (\text{nation south_korea}) \\
\text{year} \ ((\text{reverse date}) \ (>= \ 2010-mm-dd))))
\]

Q: How many more silver medals did Gillis Grafström have than Sonja Henie?

\[
(- \ (\text{num} \ (\text{silver} \ (\text{athlete} \ gillis\_grafstrom))) \\
\ (\text{num} \ (\text{silver} \ (\text{athlete} \ sonja\_henie))))
\]
Outline

- WikiTableQuestions
- Neural Semantic Parser
  - Type Constraints
  - Joint Entity Linking
  - QA Supervision
- Experimental Results
which athlete?
Contribution 1: Generate actions from a typed grammar instead of tokens from the logical form.
<table>
<thead>
<tr>
<th>Generated Actions</th>
<th>Logical Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td>c</td>
</tr>
</tbody>
</table>

Which athlete was from South Korea after the year 2010?
Which athlete was from South Korea after the year 2010?

**Generated Actions**

\[ c \]
\[ c \rightarrow (r, c, r) \]

**Logical Form**

\( (r, c, r) \)
Which athlete was from South Korea after the year 2010?

Generated Actions

\[
\begin{align*}
  c \\
  c \rightarrow ( \langle r, c \rangle \ r )
\end{align*}
\]

Logical Form

\[
( \langle r, c \rangle \ r )
\]
Which athlete was from South Korea after the year 2010?

Generated Actions

\[
c \rightarrow (\langle r, c \rangle \ r)
\]

\[
\langle r, c \rangle \rightarrow (\langle \langle c, r \rangle, \langle r, c \rangle \rangle \ <c, r>)
\]

Logical Form

\[
((\langle \langle c, r \rangle, \langle r, c \rangle \rangle \ <c, r>) \ r)
\]
Which athlete was from South Korea after the year 2010?

Generated Actions

c
c→(<r,c> r)

Logical Form

((<<c,r>,<r,c>> <c,r>) r)
Which athlete was from South Korea after the year 2010?

**Generated Actions**

\[
\begin{align*}
\text{c} \\
\text{c} \rightarrow (&r, c \rightarrow r) \\
&c, r \rightarrow (<<c, r>, &r, c>>, &c, r) \\
<<c, r>, &r, c>> \rightarrow \text{reverse}
\end{align*}
\]

**Logical Form**

\[
((\text{reverse } &c, r) \rightarrow r)
\]
Which athlete was from South Korea after the year 2010?

Generated Actions

\[
\begin{align*}
&c \\
&c \rightarrow (<r, c>, r) \\
&<r, c> \rightarrow (<<c, r>, <r, c>>, <c, r>) \\
&<<c, r>, <r, c>> \rightarrow reverse
\end{align*}
\]

Logical Form

\[
((\text{reverse} \ <c, r>) \ r)
\]
Which athlete was from South Korea after the year 2010?

**Generated Actions**

\[
c
\rightarrow (<r, c> \ r)
\rightarrow (<<c, r>, <r, c>> \ <c, r>)
\rightarrow reverse
\rightarrow athlete
\]

**Logical Form**

\[
((reverse \ athlete) \ r)
\]
Type-constrained Decoding

Which athlete was from South Korea after the year 2010?

Generated Actions

\[
\begin{align*}
&c \\
&c \rightarrow (\langle r, c \rangle \ r) \\
&\langle r, c \rangle \rightarrow (\langle \langle c, r \rangle, \langle r, c \rangle \rangle \langle c, r \rangle) \\
&\langle \langle c, r \rangle, \langle r, c \rangle \rangle \rightarrow \text{reverse} \\
&\langle c, r \rangle \rightarrow \text{athlete} \\
&r \rightarrow (\langle r, \langle r, r \rangle \rangle \ r \ r) \\
&\langle r, \langle r, r \rangle \rangle \rightarrow \text{and} \\
&r \rightarrow (\langle c, r \rangle \ c) \\
&\langle c, r \rangle \rightarrow \text{nation} \\
&c \rightarrow \text{south_korea} \\
&r \rightarrow (\langle c, r \rangle \ c) \\
&\langle c, r \rangle \rightarrow \text{year} \\
&c \rightarrow (\langle r, c \rangle \ r) \\
&\langle r, c \rangle \rightarrow (\langle \langle c, r \rangle, \langle r, c \rangle \rangle \langle c, r \rangle) \\
\end{align*}
\]

Logical Form

\[
((\text{reverse athlete}) \\
\quad (\text{and} \ (\text{nation} \ \text{south_korea}) \\
\quad (\text{year} \ ((\langle \langle c, r \rangle, \langle r, c \rangle \rangle \langle c, r \rangle) \ r)))))
\]
Which athlete was from South Korea after the year 2010?

**Generated Actions**

\[
\begin{align*}
c & \rightarrow (<r, c> \ r) \\
<r, c> & \rightarrow ((<c, r>, <r, c>) \ \text{reverse} \\
<c, r> & \rightarrow \text{athlete} \\
<r, <r, r>> & \rightarrow \text{and} \\
<r, <r, r>> & \rightarrow \text{and} \\
<c, r> & \rightarrow \text{nation} \\
c & \rightarrow \text{south_korea} \\
r & \rightarrow (<c, r> \ c) \\
<c, r> & \rightarrow \text{year} \\
c & \rightarrow (<r, c> \ r) \\
<r, c> & \rightarrow ((<c, r>, <r, c>) \ \text{reverse} \\
<c, r> & \rightarrow \text{reverse}
\end{align*}
\]

**Logical Form**

\[
\begin{align*}
((\text{reverse} \ \text{athlete}) \\
(\text{and} \ (\text{nation} \ \text{south_korea}) \\
(\text{year} ((\text{reverse} \ \text{date}) \\
(\geq 2010\text{-mm-dd})))
\end{align*}
\]
Which athlete was from South Korea after the year 2010?

**Generated Actions**

- \( c \rightarrow (\langle r, c, r \rangle) \)
- \( r \rightarrow (<c, r>) \)
- \( c \rightarrow (\langle c, r \rangle) \)
- \( r \rightarrow (\langle c, r \rangle) \)
- \( c \rightarrow (\langle r, c \rangle) \)
- \( r \rightarrow (\langle c, r \rangle) \)
- \( c \rightarrow (\langle c, r \rangle) \)
- \( r \rightarrow (\langle c, r \rangle) \)
- \( c \rightarrow (\langle c, r \rangle) \)
- \( r \rightarrow (\langle c, r \rangle) \)
- \( c \rightarrow (\langle c, r \rangle) \)
- \( r \rightarrow (\langle c, r \rangle) \)

**Logical Form**

\[
((\text{reverse athlete}) \\
(\text{and} (\text{nation south_korea}) \\
(\text{year} ((\text{reverse date}) \\
(\geq 2010-mm-dd)))))
\]

Four transformation rules:
- Constant
- Application
- Lambda
- Variable

Instantiations with concrete types automatically induced from corpus
Our Model

Contribution 1: Generate actions from a typed grammar instead of tokens from the logical form.
Our Model

Contribution 2: Joint entity linking with entity embeddings
Entity Embedding and Linking

Word Embeddings

LSTM

which

athlete

LSTM

?
Entity Embedding and Linking

which athlete

LSTM LSTM ...

Word Embeddings

Link Embeddings

l(\textit{which}) l(\textit{athlete}) l(\textit{?})
Link Embedding

athlete
nation
gillis
2014

1(was)
Soft Entity Linking

Which athlete was from South Korea?

E + F = S_{athlete}

Embedding Similarity

Featurized Similarity

Link Embedding

athlete nation
gillis 2014

1(was)
Knowledge Graph Embedding

![Diagram of Knowledge Graph Embedding]

Soft Entity Linking

Which athlete was from South Korea?

![Diagram of Soft Entity Linking]

Link Embedding

![Diagram of Link Embedding]
Moving towards our model

Contribution 1: Generate actions from a typed grammar instead of tokens from the logical form.

Contribution 2: Embed KG and define a soft entity linking.

Entity Embedding
Our model

Contribution 1: Generate actions from a typed grammar instead of tokens from the logical form.

Contribution 2: Embed KG and define a soft entity linking

Contribution 3: Train on QA pairs, using logical forms from dynamic programming on denotations

Training Data:
Q: Which athlete was from South Korea after the year 2010?
A: Kim yu-na
Q: Which athlete was from South Korea after the year 2010?
A: Kim Yu-na
Q: Which athlete was from South Korea after the year 2010?
A: Kim yu-na

\[
((\text{reverse athlete}) \ (\text{and} \ (\text{nation south\_korea}) \ (\text{year} \ ((\text{reverse date}) \ (\geq 2010\text{-mm-dd}))))) \\
((\text{reverse athlete}) \ (\text{and} \ (\text{nation south\_korea})) \\
((\text{reverse athlete}) \ (\text{argmax ind})) \\
\ldots
\]
Q: Which athlete was from South Korea after the year 2010?
A: Kim yu-na

\[
O(\theta) = \sum_{i=1}^{n} \log \sum_{\ell \in L^i} P(\ell | q_i, T_i; \theta)
\]
Outline

- WikiTableQuestions
- Neural Semantic Parser
  - Type Constraints
  - Joint Entity Linking
  - QA Supervision
- Experimental Results
Comparison with Prior Work

![Graph comparing test accuracy with different works and methods.](image)

- **Pasupat and Liang, 2015**: 37.1
- **Neelakantan et al., 2017**: 34.2
- **Haug et al., 2017**: 34.8
- **This work**: 43.3
Comparison with Prior Work

<table>
<thead>
<tr>
<th></th>
<th>Single</th>
<th>Ensemble</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pasupat and Liang, 2015</td>
<td>37.1</td>
<td></td>
</tr>
<tr>
<td>Neelakantan et al., 2017</td>
<td>34.2</td>
<td>37.7</td>
</tr>
<tr>
<td>Haug et al., 2017</td>
<td>34.8</td>
<td>38.7</td>
</tr>
<tr>
<td>This work</td>
<td>43.3</td>
<td>45.9</td>
</tr>
</tbody>
</table>
Comparison with Prior Work

<table>
<thead>
<tr>
<th></th>
<th>Single</th>
<th>Ensemble</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pasupat and Liang, 2015</td>
<td>37.1</td>
<td></td>
</tr>
<tr>
<td>Neelakantan et al., 2017</td>
<td>34.2</td>
<td>38.7</td>
</tr>
<tr>
<td>Haug et al., 2017</td>
<td>34.8</td>
<td></td>
</tr>
<tr>
<td>This work</td>
<td>43.3</td>
<td>45.9</td>
</tr>
<tr>
<td>Zhang et al., 2017</td>
<td>43.7</td>
<td></td>
</tr>
</tbody>
</table>
Do Type Constraints Help?

![Bar chart showing performance comparison:]

- **Seq2Seq**: 31.3
- **Seq2Tree**: 31.6
- **Our model**: 42.7
Entity Linking Ablations

Dev Accuracy

- similarity only, no features: 27.5
- token features, no similarity: 28.1
- all features, no similarity: 37.8
- full model: 42.7
Error Analysis

Precision Errors (41%)

- Select an answer from a given list of options (15%)
  
  Q: Which David D'or album was the bigger seller - Changing Heights or In The Soul?

- Complex background knowledge (13%)
  
  Q: Which train ran for the longest time? 
  (time = Year Discontinued - Year Began)
Error Analysis

Recall Errors (36%)

- Complex cell values (15%)

<table>
<thead>
<tr>
<th>Name</th>
<th>Fate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jervis</td>
<td>Sold for scrap</td>
</tr>
<tr>
<td>Janus</td>
<td>Torpedoed and sunk off Anzio</td>
</tr>
</tbody>
</table>

Q: How many of these ships were sunk?

- Unusual table structures (10%)

  Tables with a total row

- Unsupported Operations (11%)

  (Average, Percents, String operations)
Contribution 1: Generate actions from a typed grammar instead of tokens from the logical form.

Contribution 2: Embed KG and define a soft entity linking

Contribution 3: Train on QA pairs, using logical forms from dynamic programming on denotations

Training Data:
Q: Which athlete was from South Korea after the year 2010?
A: Kim yu-na

State-of-the-art on WikiTableQuestions
Contributions

1. Neural semantic parsing with type-constrained decoding to generate valid logical forms
2. Novel soft entity linking trained jointly with semantic parser
3. Training with QA supervision using dynamic programming on denotations

State of the art onWikitableQuestions dataset
Error Analysis

Precision Errors (41%)

- Selecting an answer from a given list of options (15%)
  
  Which David D’or album was the bigger seller - Changing Heights or In The Soul?

- Requiring background knowledge (13%)
  
  Which train ran for the longest time? (time = Year Discontinued - Year Began)

Recall Errors (36%)

Table representation failures (25%)

- Complex cell values (15%)

<table>
<thead>
<tr>
<th>Name</th>
<th>Fate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jervis</td>
<td>Sold for scrap</td>
</tr>
<tr>
<td>Janus</td>
<td>Torpedoed and sunk off Anzio</td>
</tr>
</tbody>
</table>

Q: How many of these ships were sunk?

- Unusual table structures (10%)

  Tables with a total row

Unsupported operations (11%)

- Average
- Percentages
- String operations
Training using DPD

- Dynamic Programming on Denotations (Pasupat and Liang, 2016)
- Generate **consistent** logical forms given denotation
- Maximize marginal likelihood of all consistent logical forms, using SGD

\[
O(\theta) = \sum_{i=1}^{n} \log \sum_{\ell \in L_i} P(\ell | q^i, T^i; \theta)
\]