Type-driven Incremental Semantic Parsing with Polymorphism

Kai Zhao            Liang Huang
Dept. EECS, Oregon State University
From Language to Meaning

Information Extraction

Extracts information about a set of pre-specified relations and entities

Example:
Relation Extraction

is_a(Barack_Obama, US_President)

(Artzi et al., 2013)
From Language to Meaning

more informative

Broad-Coverage Semantics

Focuses on specific phenomena (e.g., verb-argument matching)

Example: Summarization

Barack Obama is a president.

(Artzi et al., 2013)
From Language to Meaning

Semantic Parsing
Extracts complete meaning representation
Example: Database Query from Natural Language

Which states border Texas?

\[ \lambda x. \text{state}(x) \land \text{borders}(x, \text{texas}) \]

execution over database

LA, AR, OK, NM

(Artzi et al., 2013)
Semantic Parsing

• **fully supervised**
  • analogous to MT

  **Input**
  What states border Texas?

  **Output**
  \( \lambda x. \text{state}(x) \land \text{borders}(x, \text{texas}) \)

• **weakly supervised**
  • aka. parsing from Q/A pairs

  **Input**
  What states border Texas?

  **Output**
  \{LA, AR, OK, NM\}
Challenges

✓ Unknown Derivation
  i) which parsing tree leads to the correct MR?
  ii) treated as Latent Variable

✓ Unknown Grammar
  iii) i.e., the correspondences b/w English phrases & predicates

✓ Learn both derivation & grammar
From Bottom-Up to Incremental

- **Conventional Parsing Algorithms:**
  - CKY-based bottom-up parsing
  - cubic time

- **Incremental Parsing**
  - popular in constituent/dependency parsing
  - linear time
Our Contributions

✓ **incremental** parsing (aka shift-reduce)

✓ abandon CCG, use **types** to guide parsing

- CCG: Combinatory Categorical Grammar
  A synchronous grammar b/w syntax & semantics

- type-driven: uses type checking to avoid unnecessary branching in searching
✓ Type-driven Incremental Parsing

i) maintains **Stack + Queue**

ii) Actions:

- **SHIFT**: pops a word from queue, pushes its grounded semantic expr. onto stack
  use templates triggered by POS tags/patterns

- **REDUCE**: function application (type-driven)

- **SKIP**
A Running Example

INIT

Stack

Queue

What is the capital of the largest state by area

WP VBZ DT NN IN DT JJS NN IN NN
What is the capital of the largest state by area?

<table>
<thead>
<tr>
<th>Stack</th>
<th>Queue</th>
</tr>
</thead>
<tbody>
<tr>
<td>What</td>
<td>is the capital of the largest state by area</td>
</tr>
<tr>
<td>WP</td>
<td>VBZ DT NN IN DT JJS NN IN NN</td>
</tr>
</tbody>
</table>
A Running Example

**Stack**

What is

WP VB

\( \phi \)

**Queue**

the capital of the largest state by area

DT NN IN DT JJS NN IN NN
What is the capital of the largest state by area?
What is the capital of the largest state by area?

<table>
<thead>
<tr>
<th>Stack</th>
<th>Queue</th>
</tr>
</thead>
<tbody>
<tr>
<td>WP is the capital</td>
<td>of the largest state by area</td>
</tr>
<tr>
<td>WP VBZ DT NN</td>
<td>IN DT JJS NN IN NN</td>
</tr>
<tr>
<td>capital</td>
<td></td>
</tr>
<tr>
<td>e → e</td>
<td></td>
</tr>
</tbody>
</table>

**POS Tag NN triggers templates:**

\[ \lambda P : e \to e . P \]
\[ \lambda P : e \to t . P \]
\[ \lambda P : e \to i . P \]

Predicate capital in database:

<table>
<thead>
<tr>
<th>New York</th>
<th>Albany</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Jersey</td>
<td>Trenton</td>
</tr>
<tr>
<td>Pennsylvania</td>
<td>Harrisburg</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
A Running Example

**Stack**
What is the capital of?
WP VBZ DT NN IN

capital
e → e

**Queue**
the largest state by area
DT JJS NN IN NN
A Running Example

Stack

What is the capital of the
WP VBZ DT NN IN DT

Queue

largest state by area
JJS NN IN NN

capital
e → e
A Running Example

**SHIFT:** largest (JJS)

Stack

What is the capital of the largest state by area

Queue

capital

argmax

\( e \rightarrow e \quad (e \rightarrow t) \rightarrow (e \rightarrow i) \rightarrow e \)

POS Tag JJS triggers template:

\[ \lambda P : (e \rightarrow t) \rightarrow (e \rightarrow i) \rightarrow e \cdot P \]

\[ \text{argmax} \quad f \quad g \overset{\Delta}{=} \text{arg max} \quad g(x) \]

\( e \rightarrow t \quad e \rightarrow i \quad x : f(x) \)
A Running Example

TRY REDUCE?

Stack

What is the capital of the largest state by area?

WP VBZ DT NN IN DT JJS

Queue

NN IN NN

capital

argmax

e → e (e → t) → (e → i) → e

type checking:

• left-reduce?

  • e → e does not match e → t

• right reduce?

  • (e → t) → (e → i) → e does not match e
A Running Example

**SHIFT:** state (NN)

Stack

What is the capital of the largest state

WP VBZ DT NN IN DT JJS NN

Queue

by area
IN NN

capital
argmax
state

e → e
(e → t) → (e → i) → e
e → t

POS Tag NN triggers templates:

\[ \lambda P : e \rightarrow e \cdot P \]
\[ \lambda P : e \rightarrow t \cdot P \]
\[ \lambda P : e \rightarrow i \cdot P \]

predicate state in database

New York State TRUE
New York City FALSE
Pennsylvania TRUE
...
...
A Running Example

TRY REDUCE?

Stack

What is the capital of the largest state by area

WP VBZ DT NN IN DT JJS NN

capital argmax state

\[ e \rightarrow e \quad (e \rightarrow t) \rightarrow (e \rightarrow i) \rightarrow e \quad e \rightarrow t \]

type checking:

- left-reduce?
  - \[(e \rightarrow t) \rightarrow (e \rightarrow i) \rightarrow e\] does not match \[e\]

- right reduce?
  - \[e \rightarrow t\] does match \[e \rightarrow t\]
What is the capital of the largest state by area?

Stack:

- What (WP)
- is (VBZ)
- the (DT)
- capital (NN)
- of (IN)
- the (DT)
- largest (JJ)
- state (NN)

Queue:

- by (IN)
- area (NN)

REDUCE:

- capital (e → e)
- (argmax state) (e → i) → e
What is the capital of the largest state by area?
A Running Example

Stack

What is the capital of the largest state by area

Queue

SHIFT: area (NN)

capital

\( e \rightarrow e \)

\( \text{argmax state) size e} \)

\( e \rightarrow i \)

POS Tag NN triggers templates:

\( \lambda P : e \rightarrow e . P \)

\( \lambda P : e \rightarrow t . P \)

\( \lambda P : e \rightarrow i . P \)

<table>
<thead>
<tr>
<th>State</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>New York State</td>
<td>54,556</td>
</tr>
<tr>
<td>New York City</td>
<td>304.6</td>
</tr>
<tr>
<td>Pennsylvania</td>
<td>46,055</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
What is the capital of the largest state by area?
What is the capital of the largest state by area?

(capital (argmax state size))
Grounding Ambiguity

Who is the mayor of New York?

Who is the governor of New York?
Subtyping

Type Hierarchy

Typed Function Application

\[ f : t_1 \rightarrow t_2 \text{ takes argument } x : t_3 \]

iff.

\[ t_3 \text{ is a subtype of } t_1 \]

\[ t_3 <: t_1 \]

e.g.

- population
  - au → i
  - new york city → ct
  - new york state → st

- population
  - au → i
  - hudson river → rv

- population
  - au → i
  - new york state → st
Who is the **mayor** of New York?

Who is the **governor** of New York?
Think again about argmax

\[
\text{argmax } f \quad g \triangleq \text{arg max } g(x) \\
\text{argmax } (e \to t) \to (e \to i) \to e
\]

✓ argmax is defined to accommodate the context

i) returns \texttt{ct} in largest city; \texttt{rv} in largest river

ii) can be defined as polymorphic type

\[
\text{argmax } ('a \to t) \to ('a \to i) \to 'a
\]

iv) type is bound at the parsing time on-the-fly
Our Contributions

✓ incremental parsing (aka shift-reduce)
✓ abandon CCG, use type to guide parsing
✓ subtyping hierarchy
✓ polymorphic functions
A Running Example

**TRY REDUCE?**

<table>
<thead>
<tr>
<th>Stack</th>
<th>Queue</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is the capital of the largest state by area</td>
<td>NN IN NN</td>
</tr>
</tbody>
</table>

**capital**

\[ \text{st} \rightarrow \text{ct} \quad (\text{'}a \rightarrow \text{t}) \rightarrow (\text{'}a \rightarrow \text{i}) \rightarrow \text{'a} \]

**argmax**

**type checking:**

- left-reduce?
  - \[ \text{st} \rightarrow \text{ct} \text{ does not match } \text{'}a \rightarrow \text{t} \]
  - although \text{'}a can be bound to \text{st}, \text{ct does not match } \text{t}

- right reduce?
  - \[ (\text{'}a \rightarrow \text{t}) \rightarrow (\text{'}a \rightarrow \text{i}) \rightarrow \text{'a} \text{ does not match } \text{st} \]
Running Example Revisited

TRY REDUCE?

Stack

What is the capital of the largest state

WP VBZ DT NN IN DT JJS NN

Queue

by area

IN NN

capital argmax state

st → ct (‘a → t) → (‘a → i) → ‘a st → t

type checking:

• left-reduce?

  • (‘a → t) → (‘a → i) → ‘a does not match st

• right reduce?

  • st → t can match ‘a → t as long as ‘a is bound to st
Running Example Revisited

TRY REDUCE?

Stack

What is the capital of the largest state by area

WP VBZ DT NN IN DT JJS NN

capital argmax state
st → ct (st → t) → (st → i) → st st → t

type checking:

- left-reduce?
  - (‘a → t) → (‘a → i) → ‘a does not match st

- right reduce?
  - st → t can match ‘a → t as long as ‘a is bound to st
Running Example Revisited

Stack

What is the capital of the largest state

WP VBZ DT NN IN DT JJS NN

capital

(st → ct)

Queue

by area

IN NN

REDUCE

(argmax state)

(st → i) → st
A Running Example

What is the capital of the largest state by area?

Stack

What is the capital of the largest state by area?

WP  VBZ DT  NN  IN  DT  JJS  NN  IN  NN

capital

(size)

(right-reduce?)

(type checking:

• right-reduce?

• does lo → i match st → i

YES, due to the contravariant rule in type theory

\[
\frac{A <: B}{B \rightarrow C <: A \rightarrow C}
\]
What is the capital of the largest state by area?

Stack:

What is the capital of the largest state by area?

Queue:

REDUCE

Stack:

capital

st → ct

Queue:

(argmax state size)

st
A Running Example

What is the capital of the largest state by area?

Stack

What is the capital of the largest state by area?

Queue

(capital (argmax state size))

capital

ct
Learning

✓ Both derivation/Grammar are Unknown

✓ Spurious Ambiguity
  i) Various derivations/groundings lead to the same logical form

✓ Latent Variable
  ii) Structured Perceptron => Latent Variable Structured Perceptron
What is the capital of the largest state by area?

- \((\text{size} \ (\text{argmin} \ \text{city} \ \text{population}))\)

- \((\text{capital} \ (\text{argmax} \ \text{state} \ \text{size}))\)
Learning

\[ w \leftarrow w + \Phi(x, d^*) - \Phi(x, \hat{d}) \]

What is the capital of the largest state by area?

- **Reward** correct
- **Penalize** wrong

\[ \text{(capital (argmax state size))} \]
Experiments

✓ Datasets

i) GeoQuery
   - which state is dallas in?
   - what are the populations of the states through which the mississippi run?
   - what states border states that border states that border states that border texas?

ii) Jobs
   - are there any jobs using cpp with dell?
   - are there any jobs in the us with the title verification engineer?

iii) ATIS
   - show me the united flights from denver to baltimore
   - what flights do you have in the morning of september twentieth on united airlines from pittsburgh to san francisco and a stopover in denver
Experiments

✓ High decoding speed; Linear in theory & practice

  i) 0.5 sec/sentence

✓ recall (# correct parses / # sents)

<table>
<thead>
<tr>
<th></th>
<th>Z&amp;C '05</th>
<th>Z&amp;C '07</th>
<th>UBL</th>
<th>TISP (simple types)</th>
<th>TISP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recall</td>
<td>79</td>
<td>81.5</td>
<td>84</td>
<td>86.5</td>
<td>89</td>
</tr>
<tr>
<td>1.0</td>
<td>90</td>
<td>85</td>
<td>90</td>
<td>85</td>
<td>90</td>
</tr>
<tr>
<td>Recall</td>
<td>84</td>
<td>82.75</td>
<td>80</td>
<td>82.75</td>
<td>85</td>
</tr>
<tr>
<td>GeoQuery</td>
<td>86.5</td>
<td>85</td>
<td>80</td>
<td>85</td>
<td>90</td>
</tr>
<tr>
<td>Jobs</td>
<td>84</td>
<td>80.5</td>
<td>80</td>
<td>80</td>
<td>80</td>
</tr>
<tr>
<td>ATIS</td>
<td>81.5</td>
<td>85</td>
<td>75</td>
<td>80</td>
<td>80</td>
</tr>
</tbody>
</table>
Conclusion

✓ Polymorphic typing guides the parsing
✓ Linear time incremental parsing
✓ Learning w/ Latent Variable Structured Perceptron

✓ Future Work:
  ▶ Open Domain (Freebase)
  ▶ Learning from Q/A pairs