Learning Dependency-Based Compositional Semantics

Semantic Representations for Textual Inference Workshop – Mar. 10, 2012

Percy Liang

Google/Stanford

joint work with Michael Jordan and Dan Klein

Motivating Problem: Question Answering

Motivating Problem: Question Answering

What is the largest city in California?

Motivating Problem: Question Answering

What is the largest city in California?

What is the largest city in a state bordering California?

Semantic Interpretation

Semantic Interpretation

What is the largest city in a state bordering California?

Phoenix

Semantic Interpretation

What is the largest city in a state bordering California?

?

Phoenix

Semantic Interpretation

What is the largest city in a state bordering California?

$\texttt{city}(c)$

Phoenix

Semantic Interpretation

What is the largest city in a state bordering California?

$\texttt{city}(c) \wedge \exists s.\texttt{state}(s) \wedge \texttt{loc}(c, s)$

Phoenix

Semantic Interpretation

What is the largest city in a state bordering California?

$$\texttt{city}(c) \wedge \exists s.\texttt{state}(s) \wedge \texttt{loc}(c, s) \wedge \texttt{border}(s, \texttt{CA})$$

Phoenix

Semantic Interpretation

What is the largest city in a state bordering California?

$$\texttt{argmax}(\{c : \texttt{city}(c) \wedge \exists s.\texttt{state}(s) \wedge \texttt{loc}(c, s) \wedge \texttt{border}(s, \texttt{CA})\}, \texttt{population})$$

Phoenix

Semantic Interpretation

What is the largest city in a state bordering California?

$$\texttt{argmax}(\{c : \texttt{city}(c) \wedge \exists s.\texttt{state}(s) \wedge \texttt{loc}(c, s) \wedge \texttt{border}(s, \texttt{CA})\}, \texttt{population})$$

$$\Big\Downarrow \textbf{computation}$$

Phoenix

Semantic Interpretation

What is the largest city in a state bordering California?

?

computation

Phoenix

Supervision for Semantic Interpretation

Supervision for Semantic Interpretation

Detailed Supervision (current)

What is the largest city in California?

$$\texttt{argmax}(\{c : \texttt{city}(c) \wedge \texttt{loc}(c, \texttt{CA})\}, \texttt{population})$$

Supervision for Semantic Interpretation

Detailed Supervision (current)

What is the largest city in California?

$\Downarrow$ **expert**

$$\texttt{argmax}(\{c : \texttt{city}(c) \wedge \texttt{loc}(c, \texttt{CA})\}, \texttt{population})$$

Supervision for Semantic Interpretation

Detailed Supervision (current)

- doesn't scale up

What is the largest city in California?

expert

$$\texttt{argmax}(\{c : \texttt{city}(c) \wedge \texttt{loc}(c, \texttt{CA})\}, \texttt{population})$$

Supervision for Semantic Interpretation

Detailed Supervision (current)
- doesn't scale up

What is the largest city in California?

↓ **expert**

$\texttt{argmax}(\{c : \texttt{city}(c) \wedge \texttt{loc}(c, \texttt{CA})\}, \texttt{population})$

Natural Supervision (new)

What is the largest city in California?

↓

Los Angeles

Supervision for Semantic Interpretation

Detailed Supervision (current)

- doesn't scale up

What is the largest city in California?

expert

$$\texttt{argmax}(\{c : \texttt{city}(c) \wedge \texttt{loc}(c, \texttt{CA})\}, \texttt{population})$$

Natural Supervision (new)

What is the largest city in California?

non-expert

Los Angeles

Supervision for Semantic Interpretation

Detailed Supervision (current)
- doesn't scale up

What is the largest city in California?

expert

$$\texttt{argmax}(\{c : \texttt{city}(c) \wedge \texttt{loc}(c, \texttt{CA})\}, \texttt{population})$$

Natural Supervision (new)
- scales up

What is the largest city in California?

non-expert

Los Angeles

Supervision for Semantic Interpretation

Detailed Supervision (current)
- doesn't scale up
- representation-dependent

What is the largest city in California?

$\Big\downarrow$ **expert**

$\texttt{argmax}(\{c : \texttt{city}(c) \wedge \texttt{loc}(c, \texttt{CA})\}, \texttt{population})$

Natural Supervision (new)
- scales up

What is the largest city in California?

$\Big\downarrow$ **non-expert**

Los Angeles

Supervision for Semantic Interpretation

Detailed Supervision (current)

- doesn't scale up
- representation-dependent

What is the largest city in California?

expert

$$\texttt{argmax}(\{c : \texttt{city}(c) \land \texttt{loc}(c, \texttt{CA})\}, \texttt{population})$$

Natural Supervision (new)

- scales up
- representation-independent

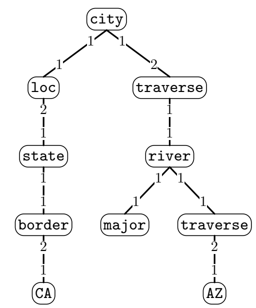
What is the largest city in California?

non-expert

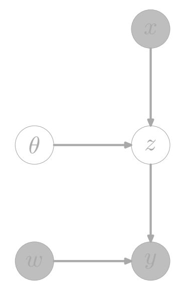
Los Angeles

Outline

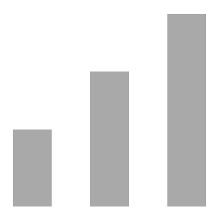
Representation



Learning



Experiments



Considerations

Computational: how to efficiently search exponential space?

Considerations

Computational: how to efficiently search exponential space?

What is the most populous city in California?

Los Angeles

Considerations

Computational: how to efficiently search exponential space?

What is the most populous city in California?

$$\lambda x.\texttt{state}(x)$$

Los Angeles

Considerations

Computational: how to efficiently search exponential space?

What is the most populous city in California?

$$\lambda x.\mathtt{city}(x)$$

Los Angeles

Considerations

Computational: how to efficiently search exponential space?

What is the most populous city in California?

$$\lambda x.\texttt{city}(x) \wedge \texttt{loc}(x, \texttt{CA})$$

Los Angeles

Considerations

Computational: how to efficiently search exponential space?

What is the most populous city in California?

$$\lambda x.\mathtt{state}(x) \wedge \mathtt{border}(x, \mathtt{CA})$$

Los Angeles

Considerations

Computational: how to efficiently search exponential space?

What is the most populous city in California?

`population(CA)`

Los Angeles

Considerations

Computational: how to efficiently search exponential space?

What is the most populous city in California?

$$\texttt{argmax}(\lambda x.\texttt{city}(x) \wedge \texttt{loc}(x, \texttt{CA}), \lambda x.\texttt{population}(x))$$

Los Angeles

Considerations

Computational: how to efficiently search exponential space?

What is the most populous city in California?

$\cdots$ LF LF LF LF LF $\boxed{\text{LF}}$ LF LF LF LF LF LF LF LF LF LF $\boxed{\text{LF}}$ $\cdots$

Los Angeles

Considerations

Computational: how to efficiently search exponential space?

What is the most populous city in California?

$\cdots$ LF LF LF LF LF $\boxed{\text{LF}}$ LF LF LF LF LF LF LF LF LF LF $\boxed{\text{LF}}$ $\cdots$

Los Angeles

Statistical: how to parametrize mapping from sentence to logical form?

What is the most populous city in California?

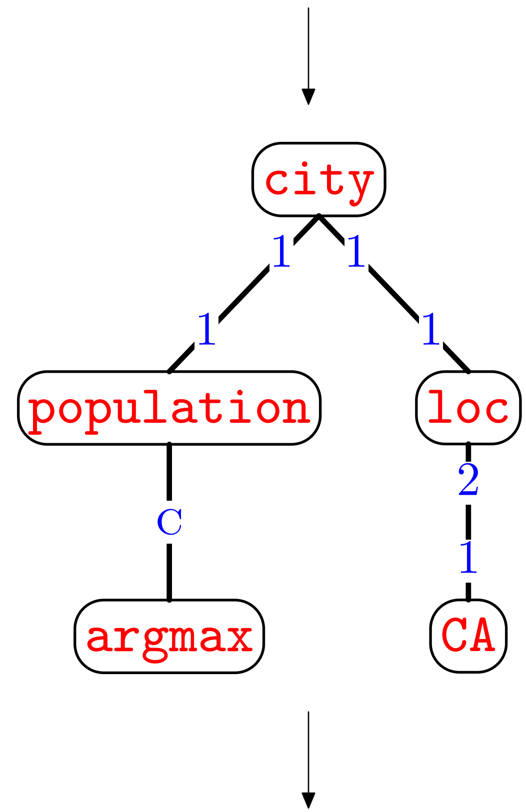
$$\texttt{argmax}(\lambda x.\texttt{city}(x) \wedge \texttt{loc}(x, \texttt{CA}), \lambda x.\texttt{population}(x))$$

Dependency-Based Compositional Semantics (DCS)

What is the most populous city in California?

Dependency-Based Compositional Semantics (DCS)

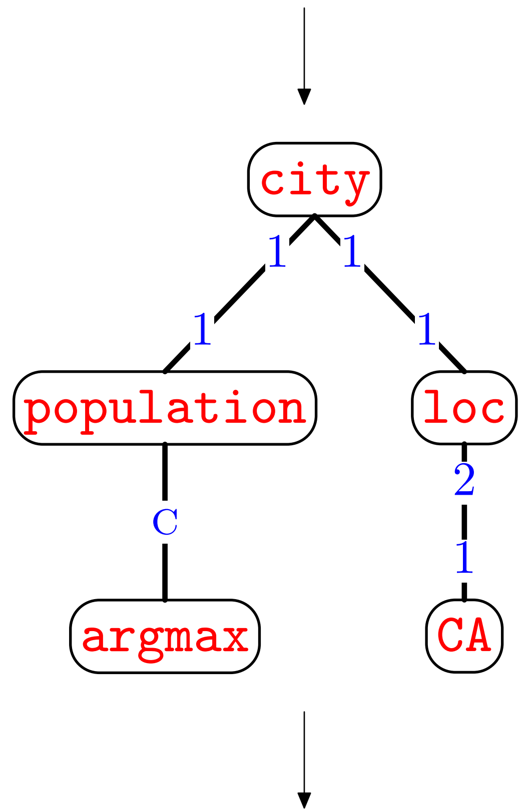
What is the most populous city in California?



Los Angeles

Dependency-Based Compositional Semantics (DCS)

What is the most populous city in California?

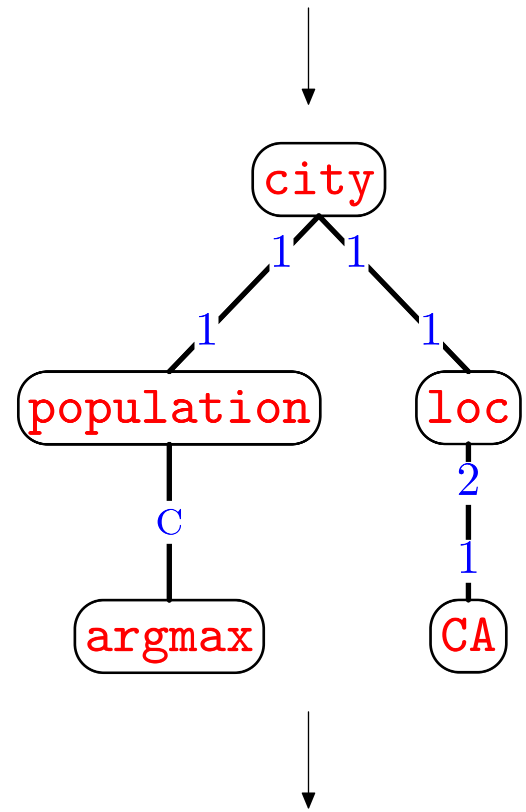


Los Angeles

Advantages of DCS: nice computational, statistical, linguistic properties

Where do the answers come from?

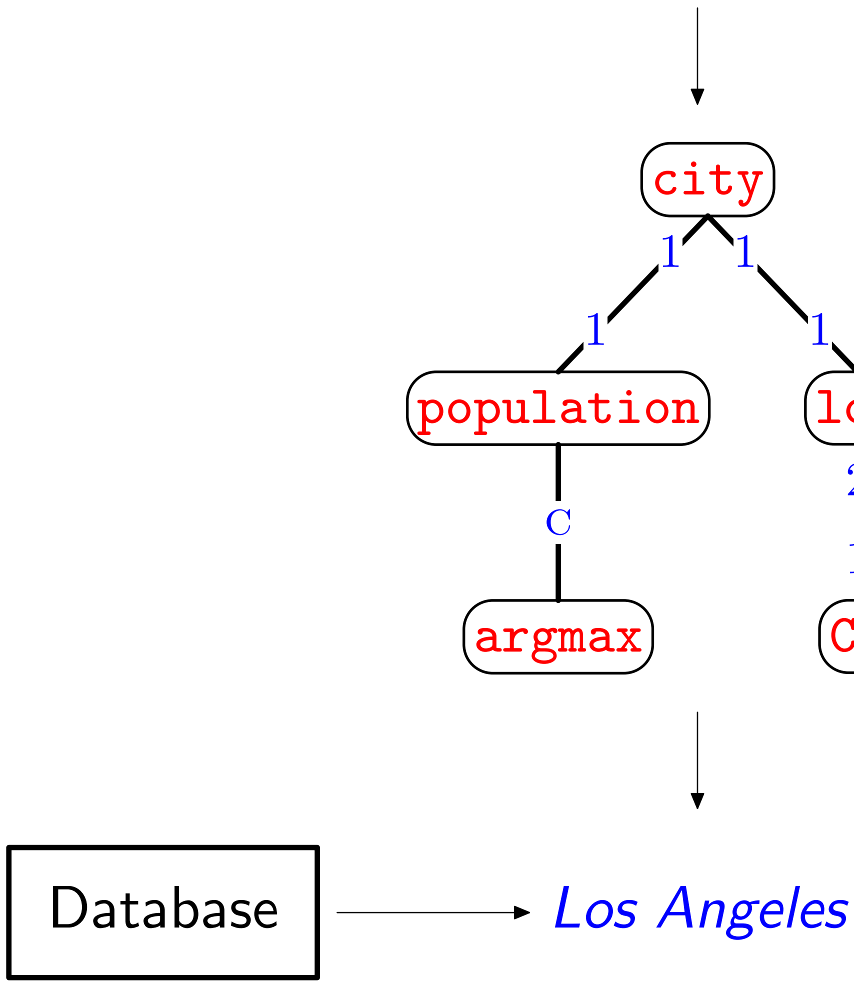
What is the most populous city in California?



Los Angeles

Where do the answers come from?

What is the most populous city in California?



Database

city

San Francisco
Chicago
Boston
...

state

Alabama
Alaska
Arizona
...

loc

Mount Shasta	California
San Francisco	California
Boston	Massachusetts
...	...

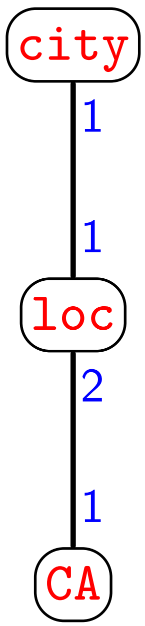
border

Washington	Oregon
Washington	Idaho
Oregon	Washington
...	...

... ...

Basic DCS Trees

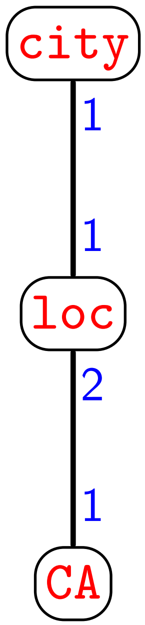
DCS tree



Database

Basic DCS Trees

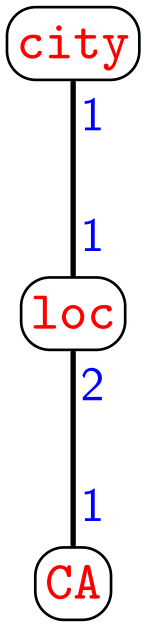
DCS tree **Constraints** **Database**



A DCS tree encodes a **constraint satisfaction problem** (CSP)

Basic DCS Trees

DCS tree



Constraints

$c \in \texttt{city}$

Database

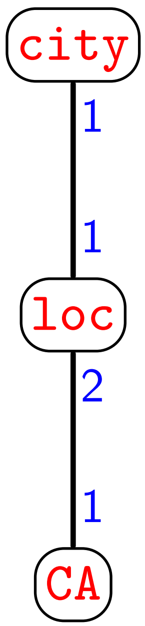
`city`

San Francisco
Chicago
Boston
...

A DCS tree encodes a **constraint satisfaction problem** (CSP)

Basic DCS Trees

DCS tree



Constraints

$c \in \texttt{city}$

$\ell \in \texttt{loc}$

Database

`city`

San Francisco
Chicago
Boston
...

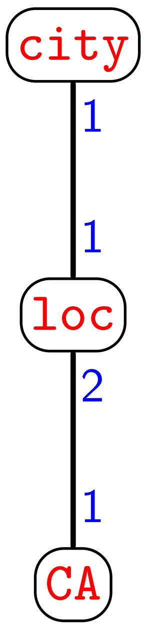
`loc`

Mount Shasta	California
San Francisco	California
Boston	Massachusetts
...	...

A DCS tree encodes a **constraint satisfaction problem** (CSP)

Basic DCS Trees

DCS tree



Constraints

$c \in \texttt{city}$

$\ell \in \texttt{loc}$

$s \in \texttt{CA}$

Database

`city`

San Francisco
Chicago
Boston
...

`loc`

Mount Shasta	California
San Francisco	California
Boston	Massachusetts
...	...

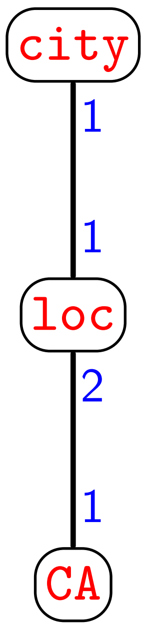
`CA`

California

A DCS tree encodes a **constraint satisfaction problem** (CSP)

Basic DCS Trees

DCS tree



Constraints

$c \in \texttt{city}$

$c_1 = \ell_1$

$\ell \in \texttt{loc}$

$s \in \texttt{CA}$

Database

`city`

San Francisco
Chicago
Boston
...

`loc`

Mount Shasta	California
San Francisco	California
Boston	Massachusetts
...	...

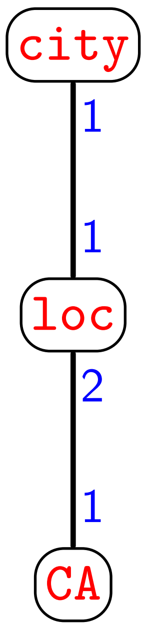
`CA`

California

A DCS tree encodes a **constraint satisfaction problem** (CSP)

Basic DCS Trees

DCS tree



Constraints

$c \in \texttt{city}$

$c_1 = \ell_1$

$\ell \in \texttt{loc}$

$\ell_2 = s_1$

$s \in \texttt{CA}$

Database

`city`

San Francisco
Chicago
Boston
...

`loc`

Mount Shasta	California
San Francisco	California
Boston	Massachusetts
...	...

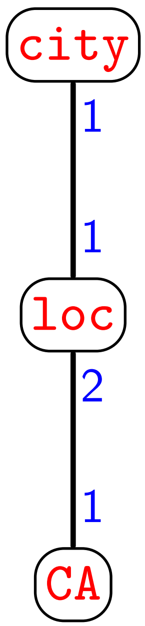
`CA`

California

A DCS tree encodes a **constraint satisfaction problem** (CSP)

Basic DCS Trees

DCS tree



Constraints

$c \in \texttt{city}$

$c_1 = \ell_1$

$\ell \in \texttt{loc}$

$\ell_2 = s_1$

$s \in \texttt{CA}$

Database

`city`

San Francisco
Chicago
Boston
...

`loc`

Mount Shasta	California
San Francisco	California
Boston	Massachusetts
...	...

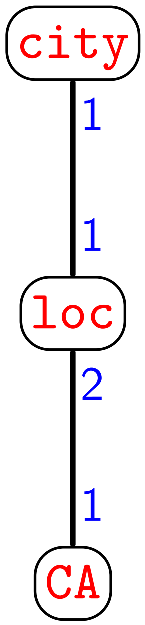
`CA`

California

A DCS tree encodes a **constraint satisfaction problem** (CSP)

Basic DCS Trees

DCS tree



Constraints

$c \in \texttt{city}$

$c_1 = \ell_1$

$\ell \in \texttt{loc}$

$\ell_2 = s_1$

$s \in \texttt{CA}$

Database

`city`

San Francisco
Chicago
Boston
...

`loc`

Mount Shasta	California
San Francisco	**California**
Boston	Massachusetts
...	...

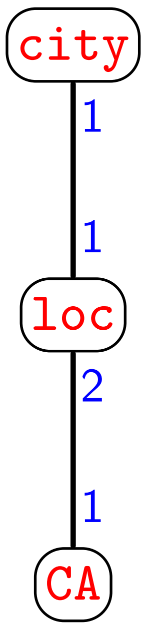
`CA`

California

A DCS tree encodes a **constraint satisfaction problem** (CSP)

Basic DCS Trees

DCS tree



Constraints

$c \in \texttt{city}$

$c_1 = \ell_1$

$\ell \in \texttt{loc}$

$\ell_2 = s_1$

$s \in \texttt{CA}$

Database

`city`

San Francisco
Chicago
Boston
...

`loc`

Mount Shasta	California
San Francisco	**California**
Boston	Massachusetts
...	...

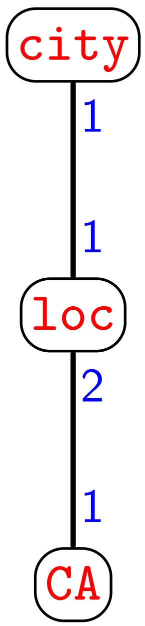
`CA`

California

A DCS tree encodes a **constraint satisfaction problem** (CSP)

Basic DCS Trees

DCS tree



Constraints

$c \in \texttt{city}$

$c_1 = \ell_1$

$\ell \in \texttt{loc}$

$\ell_2 = s_1$

$s \in \texttt{CA}$

Database

`city`

San Francisco
Chicago
Boston
...

`loc`

Mount Shasta	California
San Francisco	**California**
Boston	Massachusetts
...	...

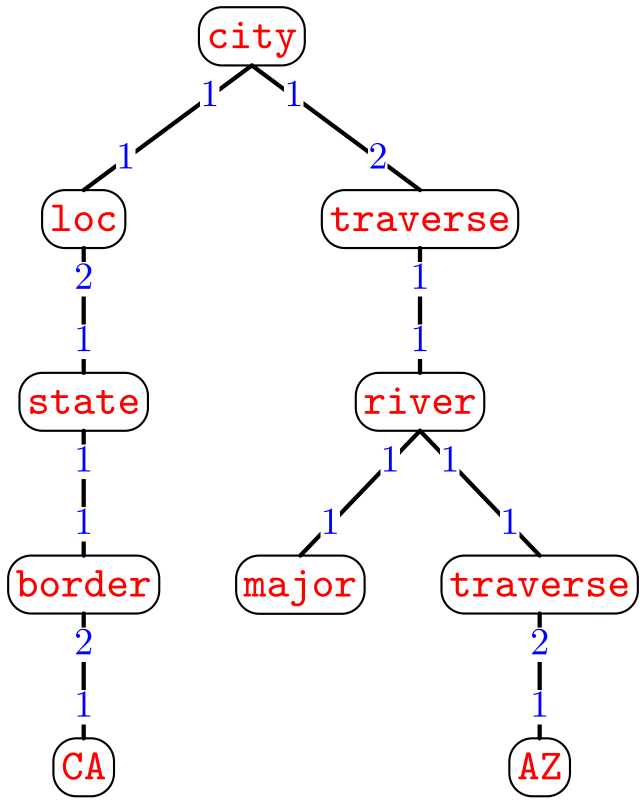
`CA`

California

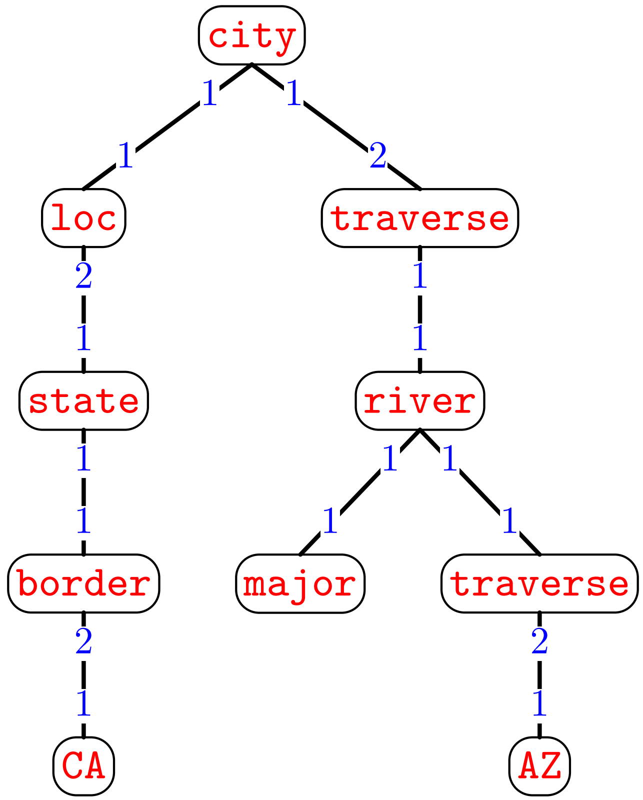
A DCS tree encodes a **constraint satisfaction problem** (CSP)

Computation: dynamic programming $\Rightarrow$ time $= O(\#$ nodes$)$

Properties of DCS Trees

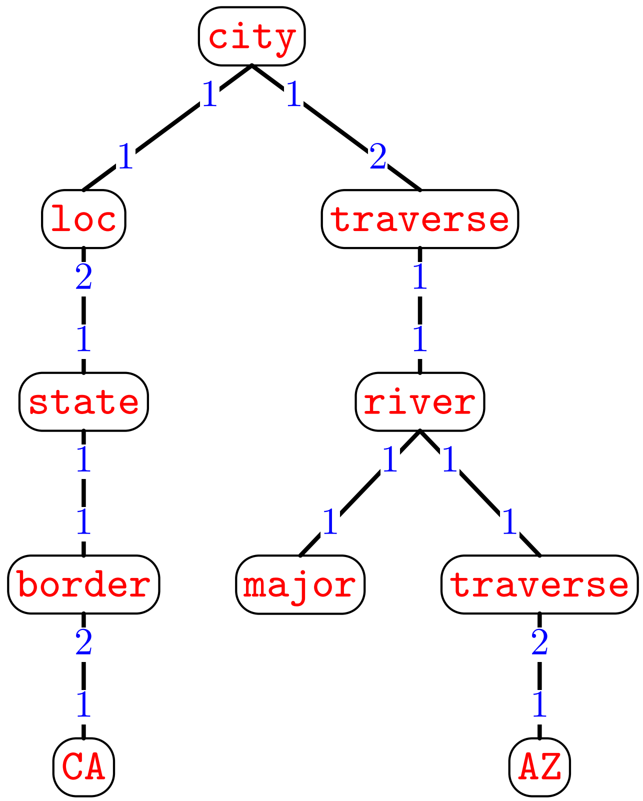


Properties of DCS Trees



Trees

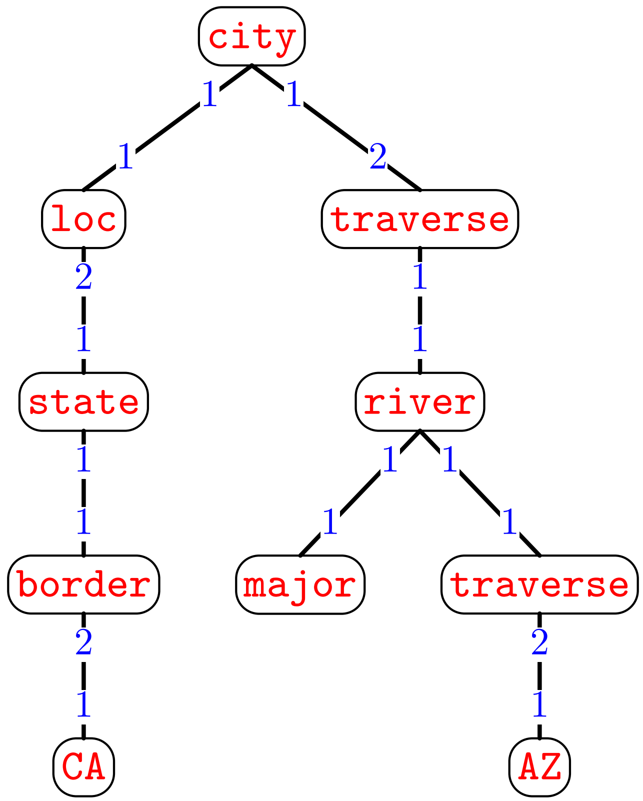
Properties of DCS Trees



Linguistics .... Trees

syntactic locality

Properties of DCS Trees



Linguistics ·············· Trees ·············· Computation

syntactic locality — efficient interpretation

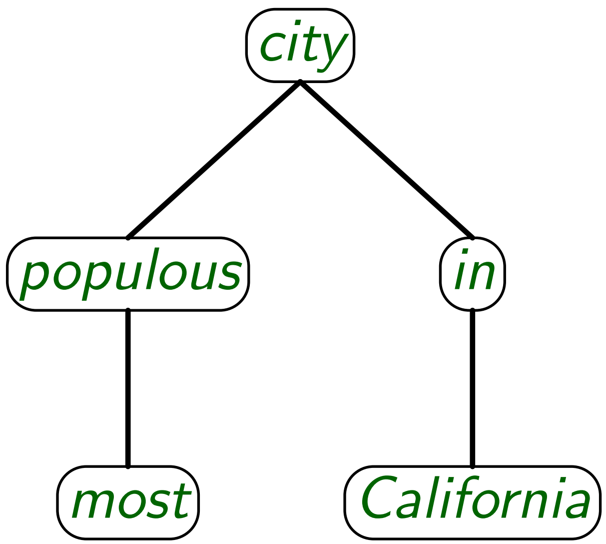
Divergence between Syntactic and Semantic Scope

most populous city in California

Divergence between Syntactic and Semantic Scope

most populous city in California

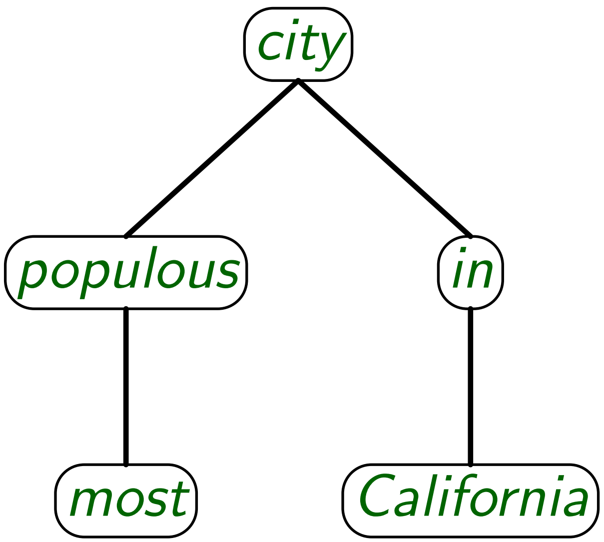
Syntax



Divergence between Syntactic and Semantic Scope

most populous city in California

Syntax



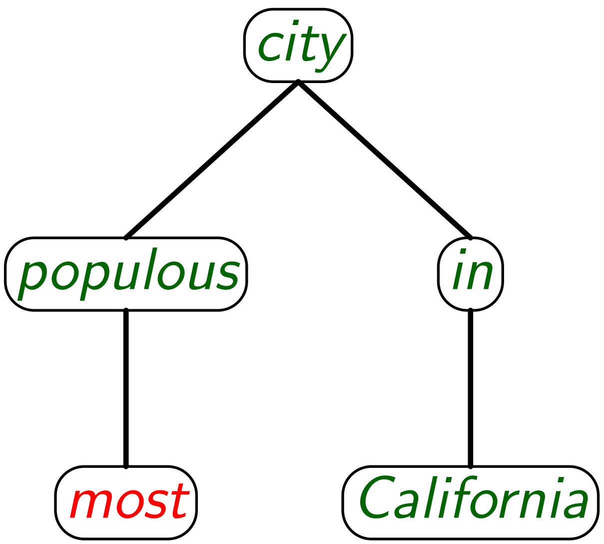
Semantics

$$\texttt{argmax}(\lambda x.\texttt{city}(x) \wedge \texttt{loc}(x, \texttt{CA}), \lambda x.\texttt{population}(x))$$

Divergence between Syntactic and Semantic Scope

most populous city in California

Syntax



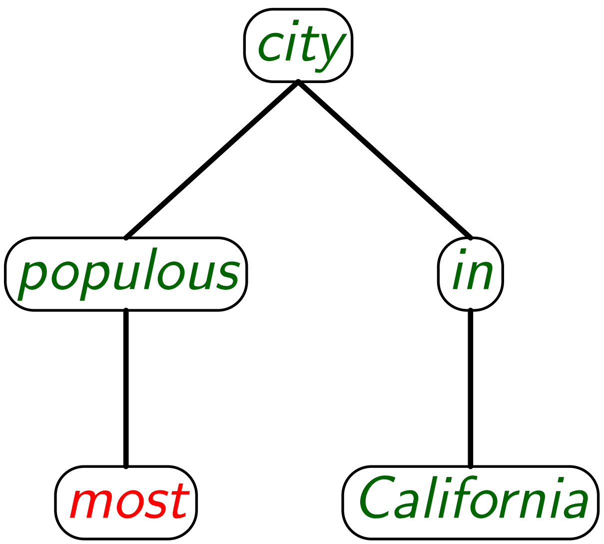
Semantics

$$\texttt{argmax}(\lambda x.\texttt{city}(x) \wedge \texttt{loc}(x, \texttt{CA}), \lambda x.\texttt{population}(x))$$

Divergence between Syntactic and Semantic Scope

most populous city in California

Syntax



Semantics

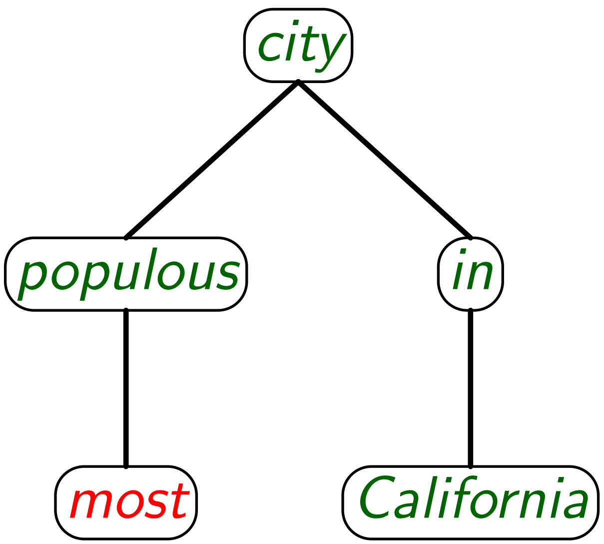
$\texttt{argmax}(\lambda x.\texttt{city}(x) \wedge \texttt{loc}(x, \texttt{CA}), \lambda x.\texttt{population}(x))$

Problem: syntactic scope is lower than semantic scope

Divergence between Syntactic and Semantic Scope

most populous city in California

Syntax



Semantics

$$\texttt{argmax}(\lambda x.\texttt{city}(x) \land \texttt{loc}(x, \texttt{CA}), \lambda x.\texttt{population}(x))$$

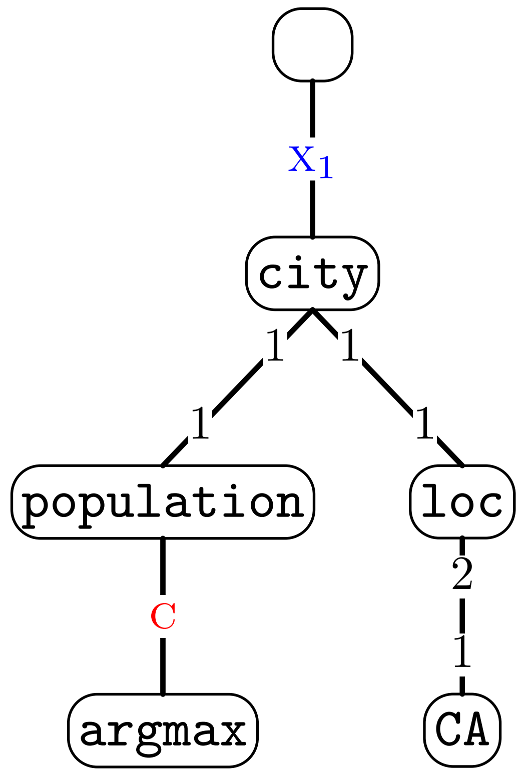
Problem: syntactic scope is lower than semantic scope

If DCS trees look like syntax, how do we get correct semantics?

Solution: Mark-Execute

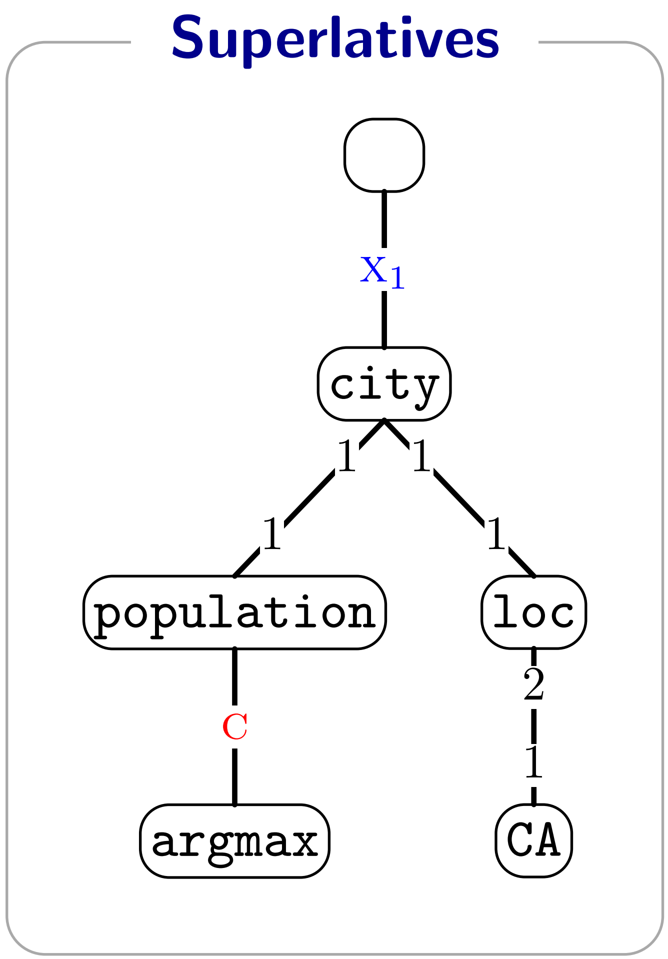
most populous city in California

Superlatives



Solution: Mark-Execute

most populous city in California



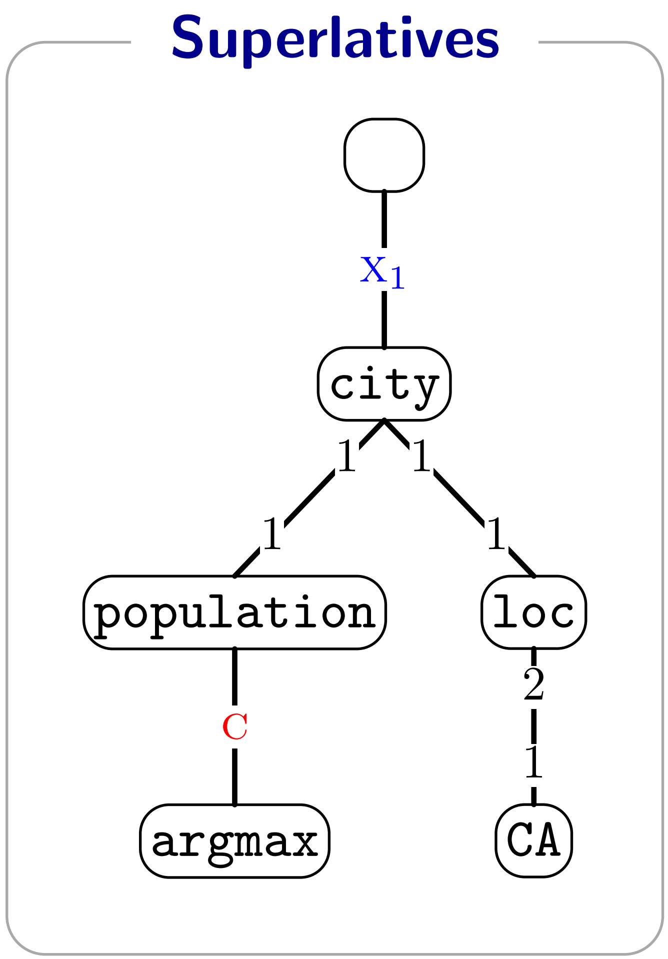
Mark at syntactic scope

Solution: Mark-Execute

most populous city in California

Execute at semantic scope

Mark at syntactic scope

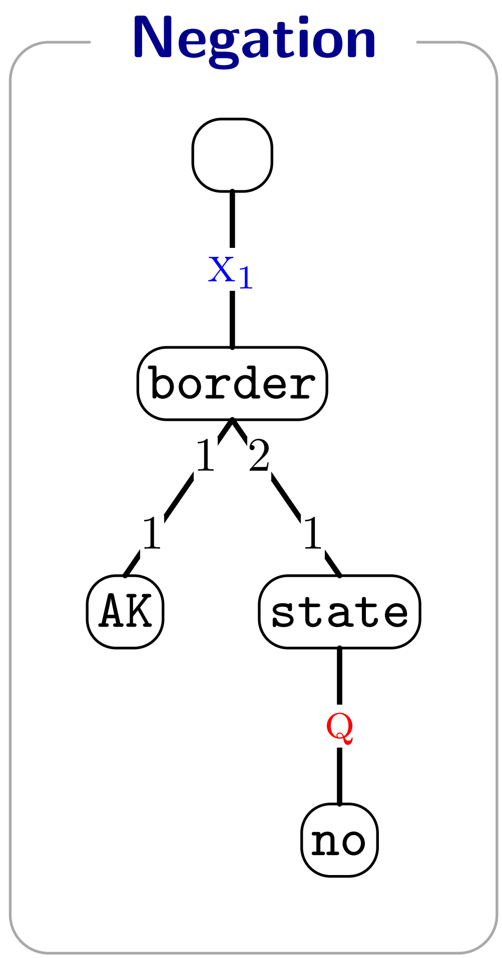


Solution: Mark-Execute

Alaska borders no states.

Execute at semantic scope

Mark at syntactic scope



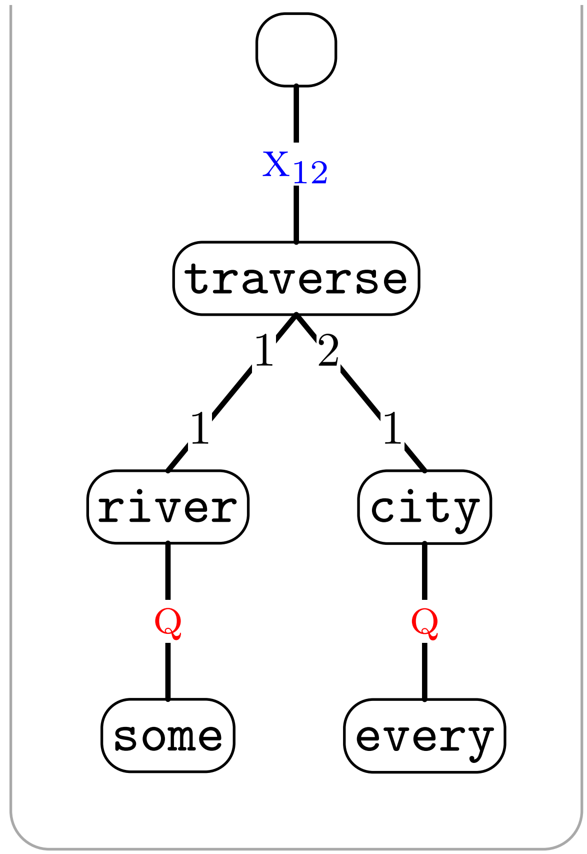
Solution: Mark-Execute

Some river traverses every city.

Quantification (narrow)

Execute at semantic scope

Mark at syntactic scope



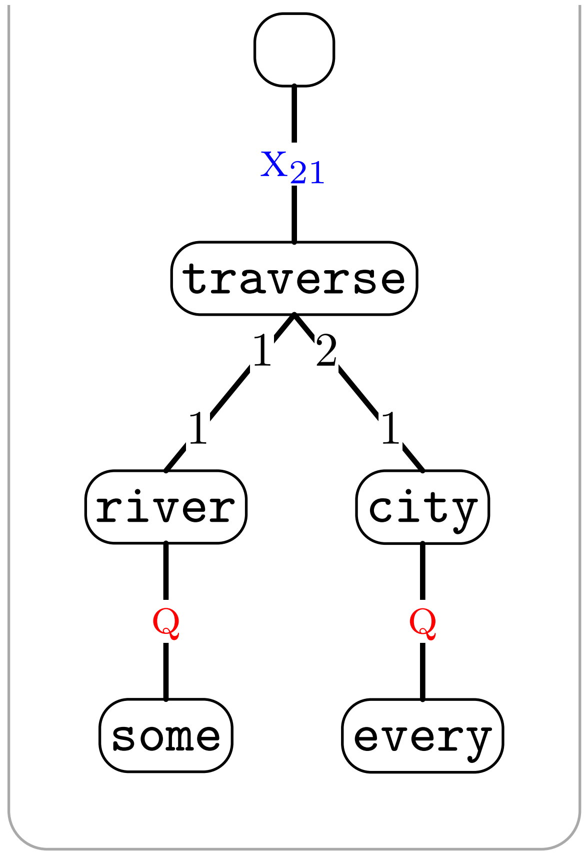
Solution: Mark-Execute

Some river traverses every city.

Quantification (wide)

Execute at semantic scope

Mark at syntactic scope



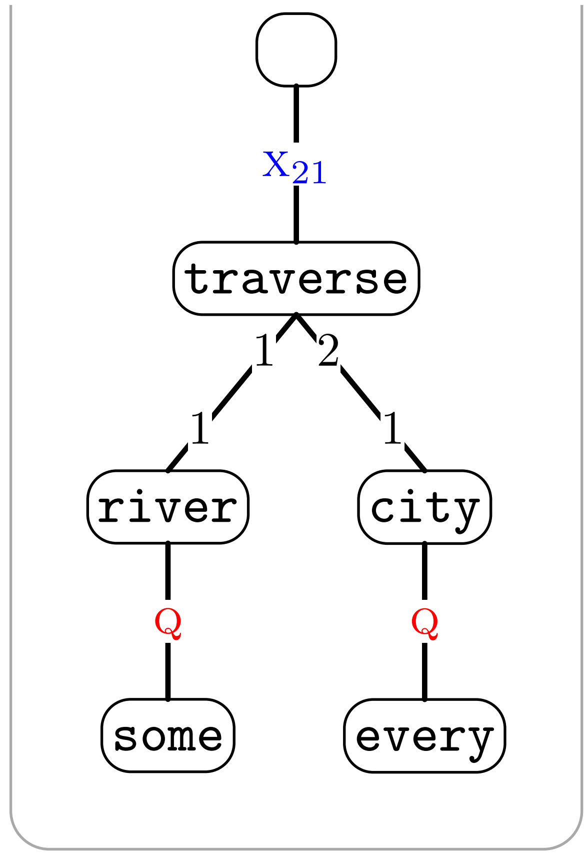
Solution: Mark-Execute

Some river traverses every city.

Quantification (wide)

Execute at semantic scope

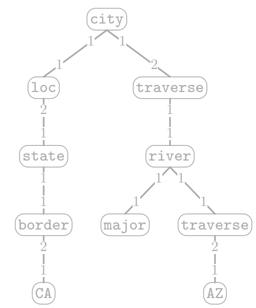
Mark at syntactic scope



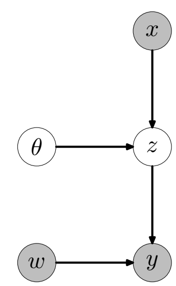
Analogy: Montague's quantifying in, Carpenter's scoping constructor

Outline

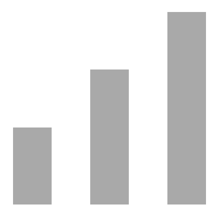
Representation



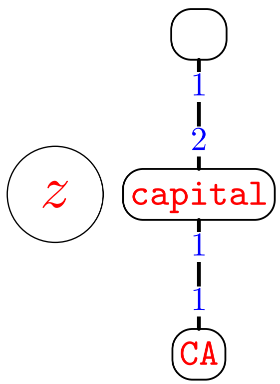
Learning

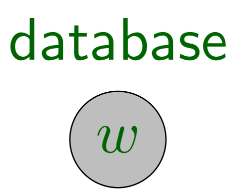


Experiments

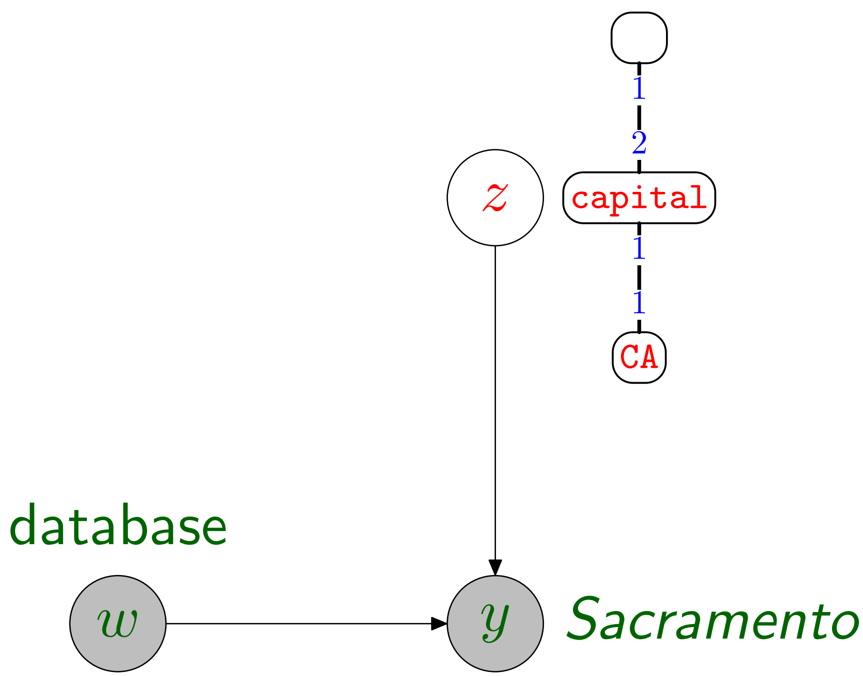


Graphical Model

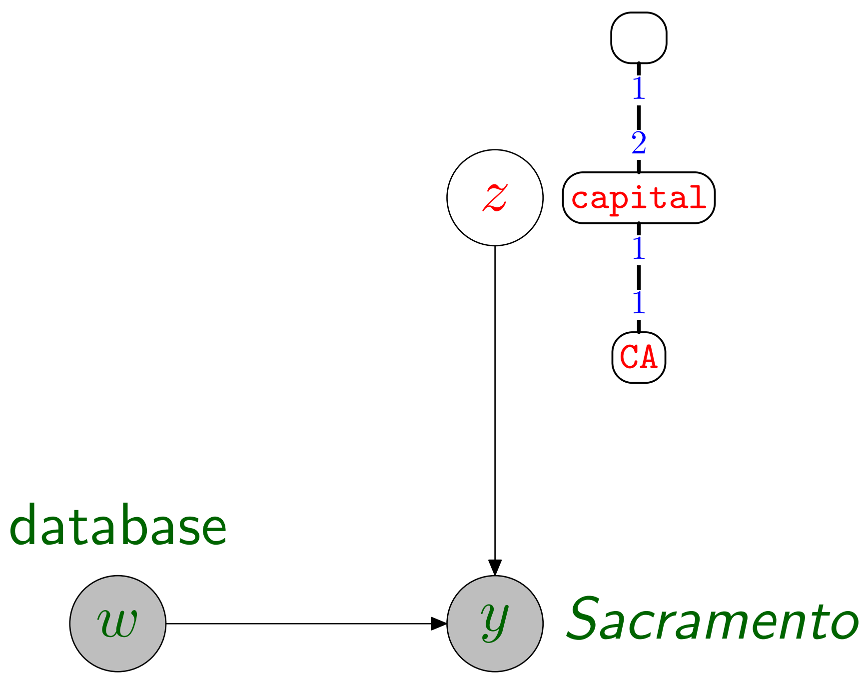




Graphical Model



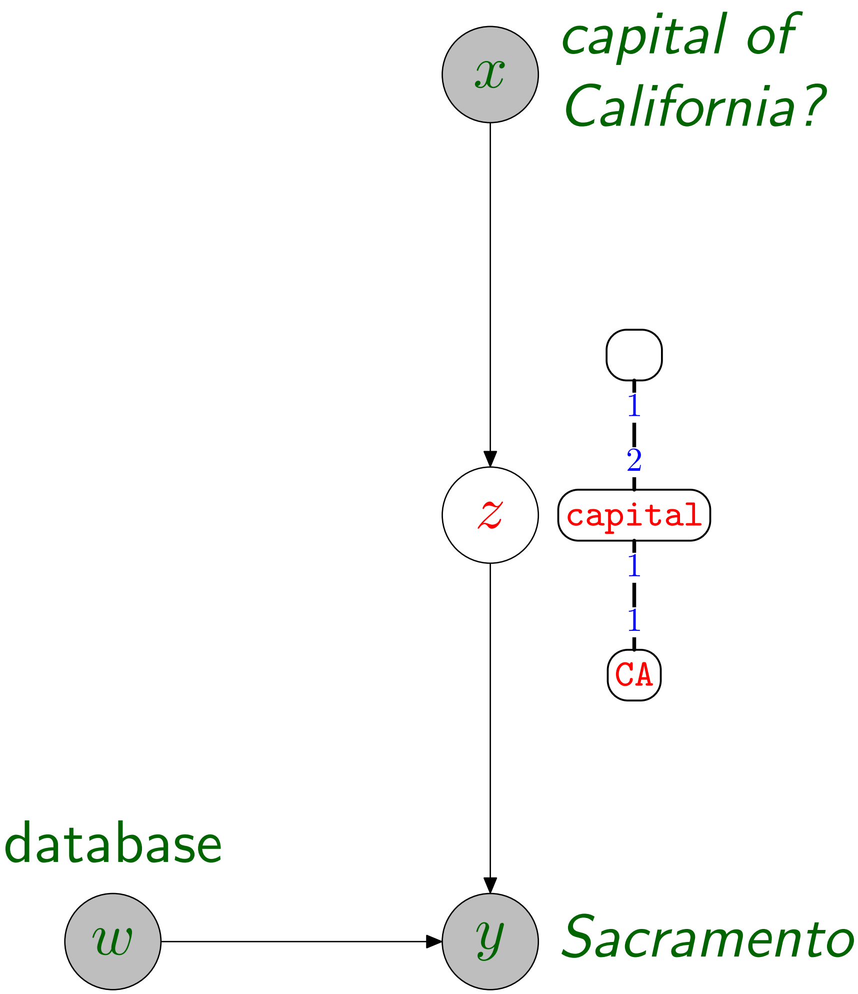
Graphical Model



Interpretation: $p(y \mid z, w)$

(deterministic)

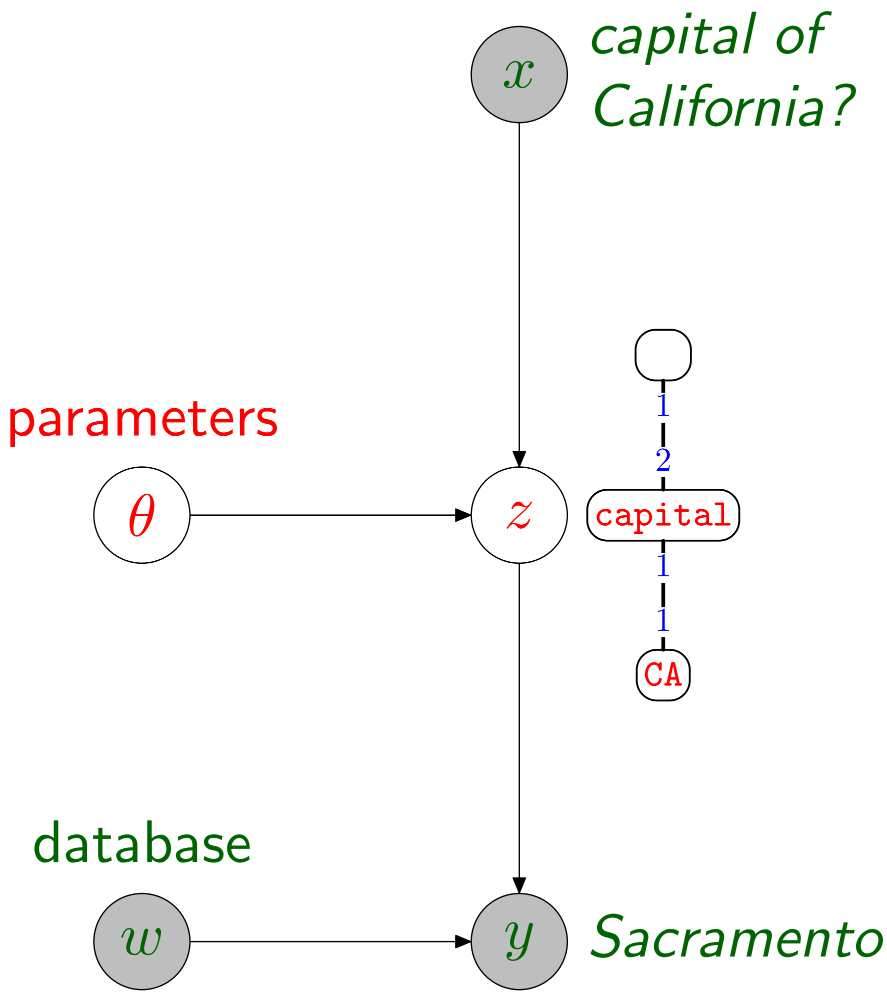
Graphical Model



Interpretation: $p(y \mid z, w)$

(deterministic)

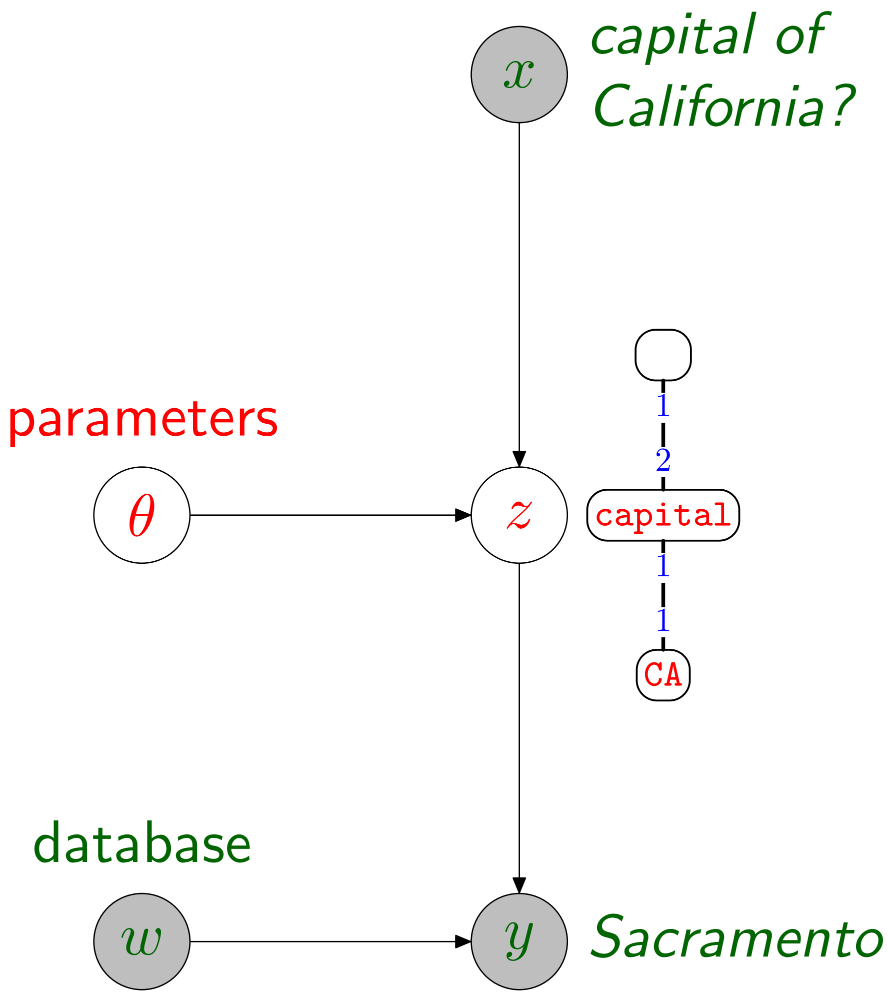
Graphical Model



Interpretation: $p(y \mid z, w)$

(deterministic)

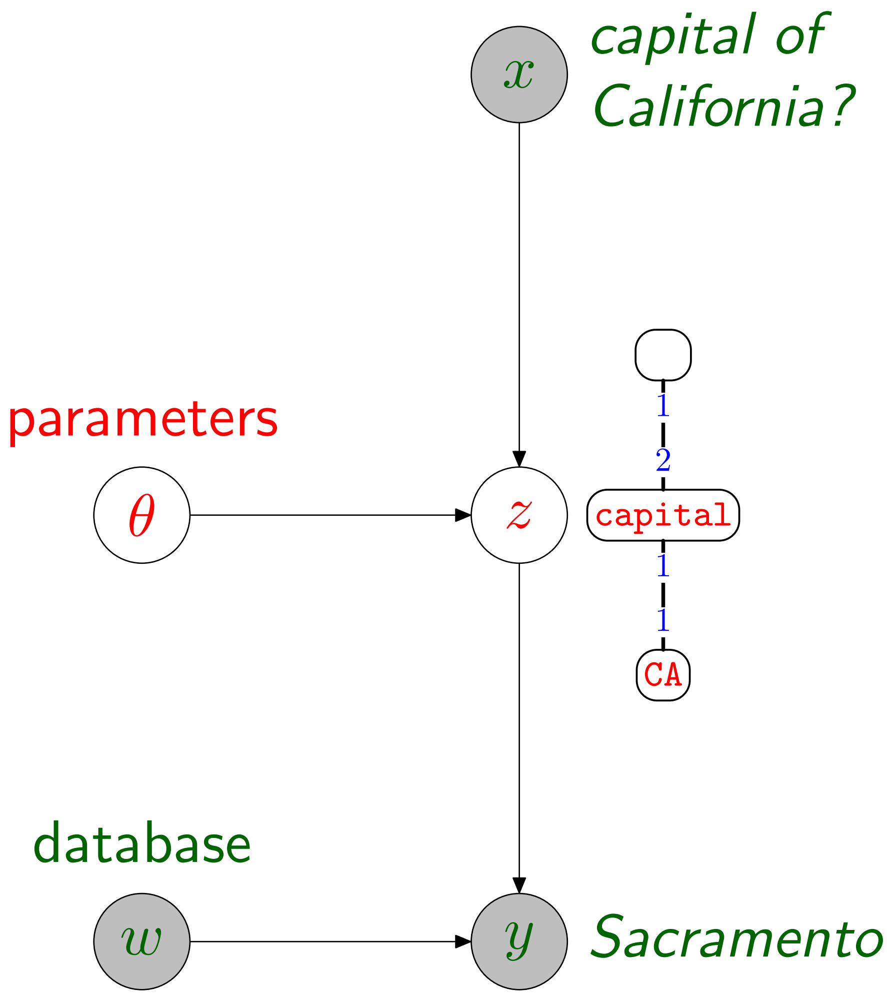
Graphical Model



Semantic Parsing: $p(z \mid x, \theta)$
(probabilistic)

Interpretation: $p(y \mid z, w)$
(deterministic)

Plan



- What's **possible**? $z \in \mathcal{Z}(x)$
- What's **probable**? $p(z \mid x, \theta)$
- **Learning** θ from (x, y) data

Words to Predicates (Lexical Semantics)

What is the most populous city in CA ?

Words to Predicates (Lexical Semantics)

$\texttt{CA}$

What is the most populous city in CA ?

Lexical Triggers:

1. String match $\quad$ *CA* $\Rightarrow$ $\texttt{CA}$

Words to Predicates (Lexical Semantics)

			`argmax`				`CA`	
What	*is*	*the*	*most*	*populous*	*city*	*in*	*CA*	*?*

Lexical Triggers:

1. String match *CA* $\Rightarrow$ `CA`
2. Function words (20 words) *most* $\Rightarrow$ `argmax`

Words to Predicates (Lexical Semantics)

				`city`	`city`			
				`state`	`state`			
				`river`	`river`			
			`argmax`	`population`	`population`		`CA`	
What	*is*	*the*	*most*	*populous*	*city*	*in*	*CA*	*?*

Lexical Triggers:

1. String match *CA* $\Rightarrow$ `CA`
2. Function words (20 words) *most* $\Rightarrow$ `argmax`
3. Nouns/adjectives *city* $\Rightarrow$ `city state river population`

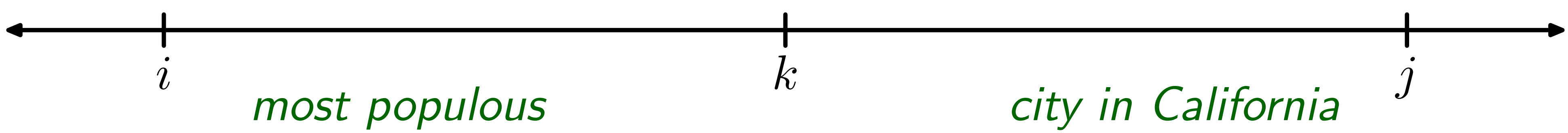
Predicates to DCS Trees (Compositional Semantics)

$C_{i,j}$ = set of DCS trees for span $[i, j]$

i *most populous* *city in California* j

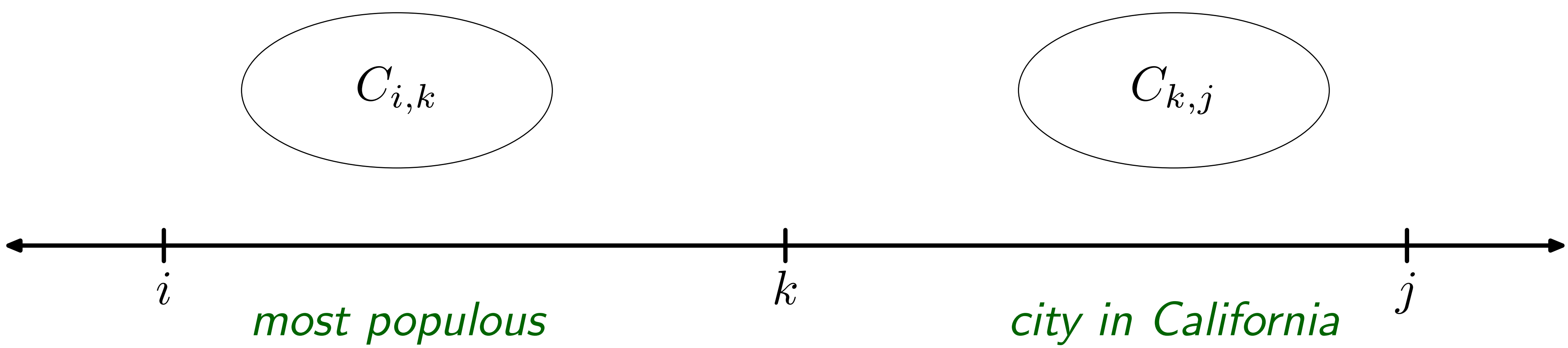
Predicates to DCS Trees (Compositional Semantics)

$C_{i,j}$ = set of DCS trees for span $[i, j]$



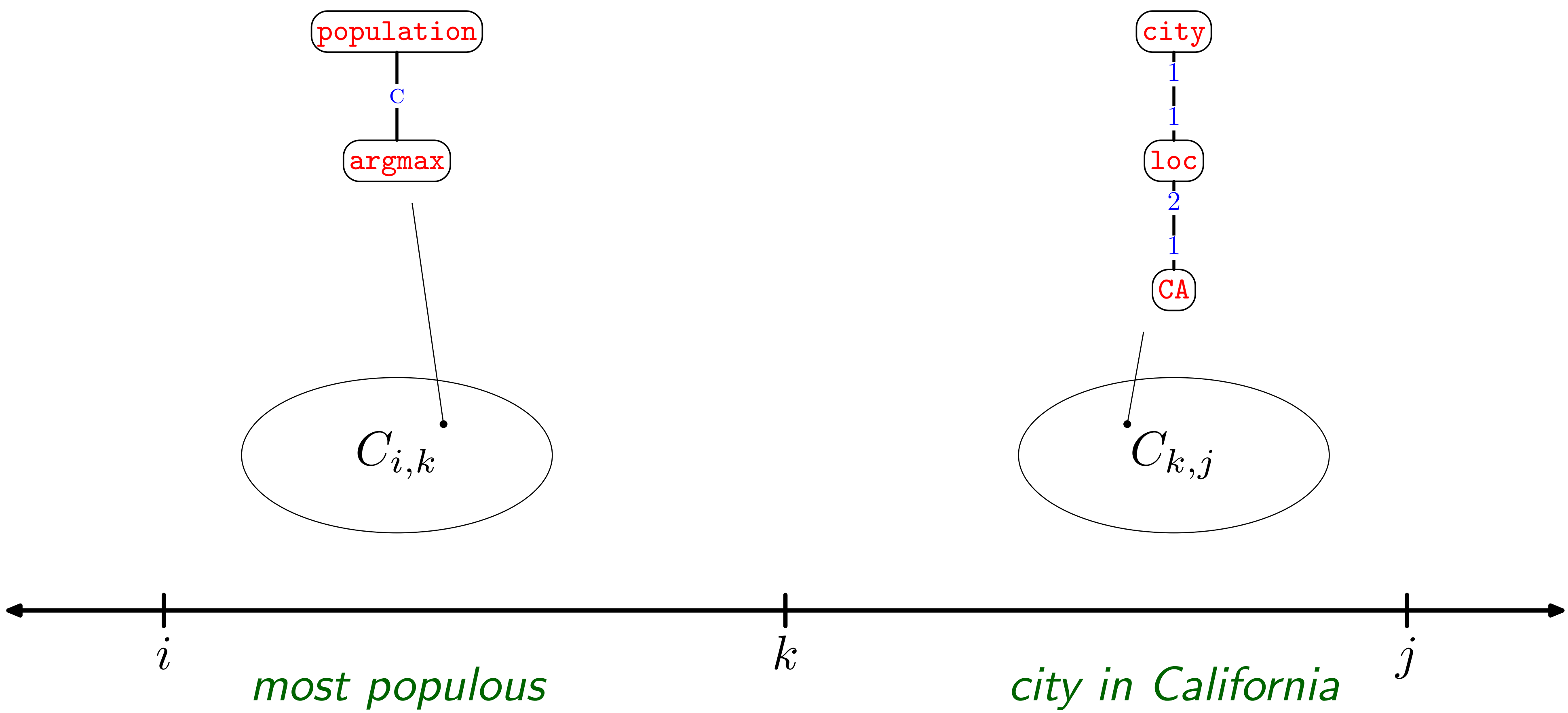
Predicates to DCS Trees (Compositional Semantics)

$C_{i,j}$ = set of DCS trees for span $[i, j]$



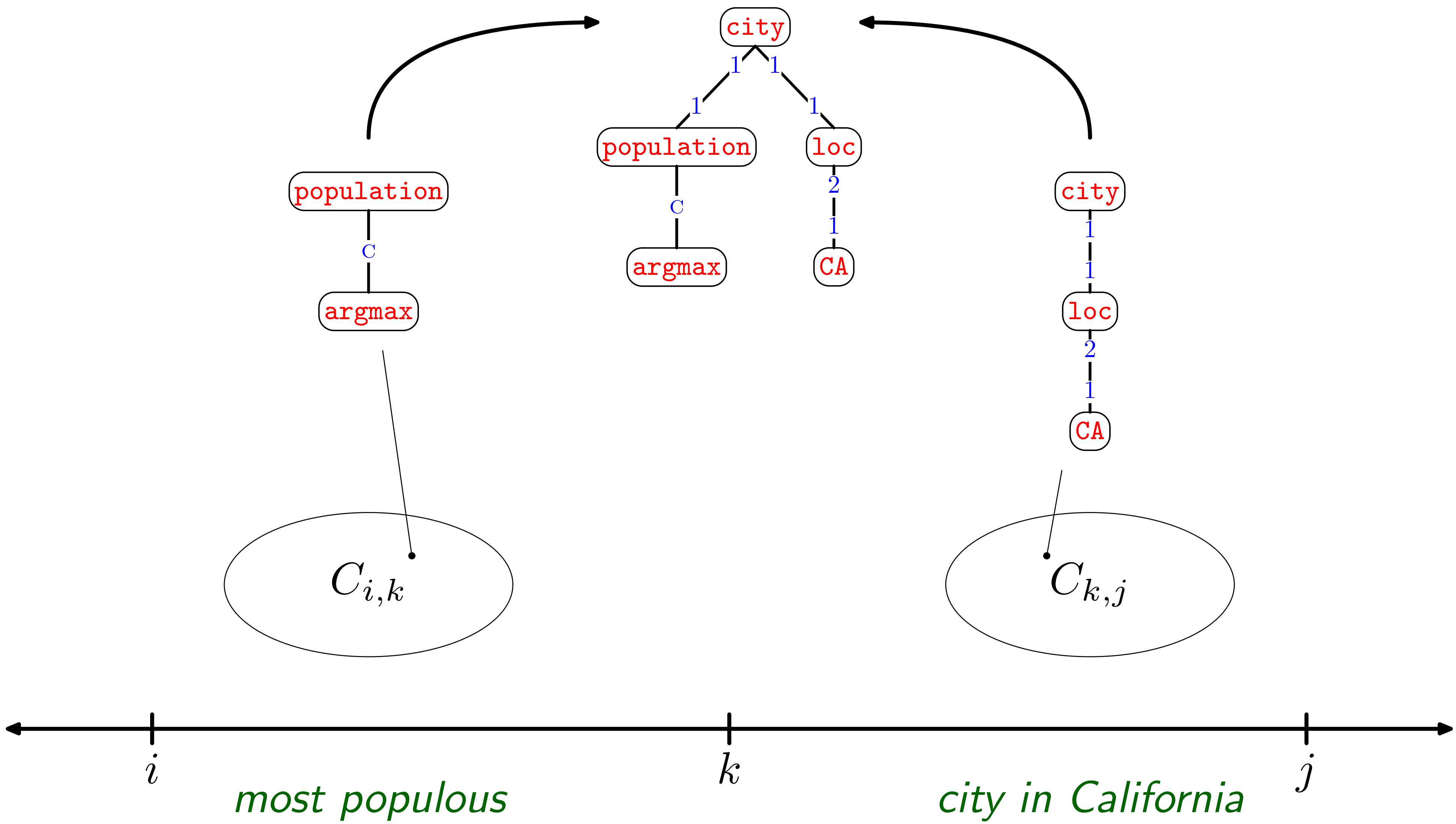
Predicates to DCS Trees (Compositional Semantics)

$C_{i,j}$ = set of DCS trees for span $[i, j]$



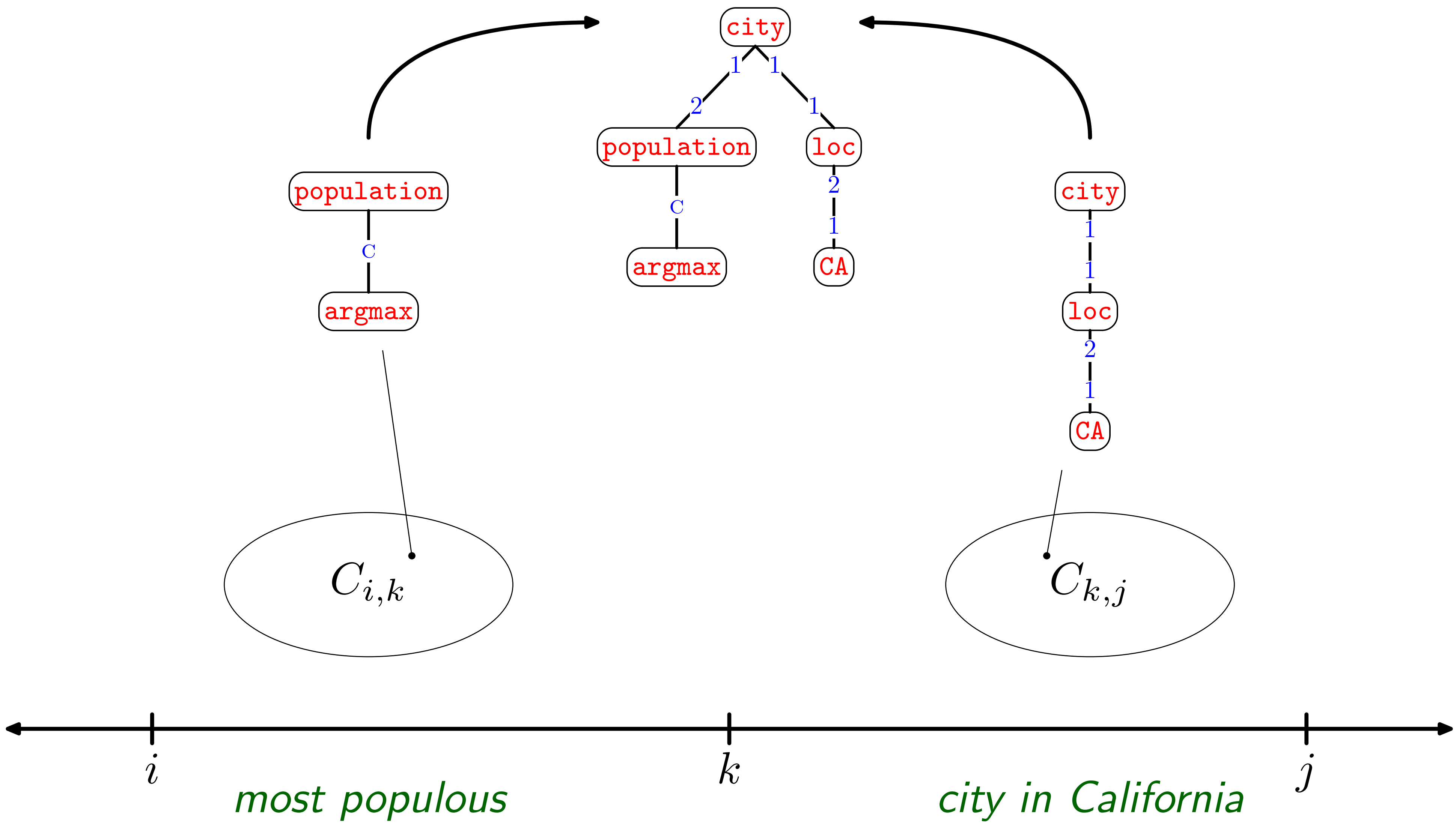
Predicates to DCS Trees (Compositional Semantics)

$C_{i,j}$ = set of DCS trees for span $[i, j]$



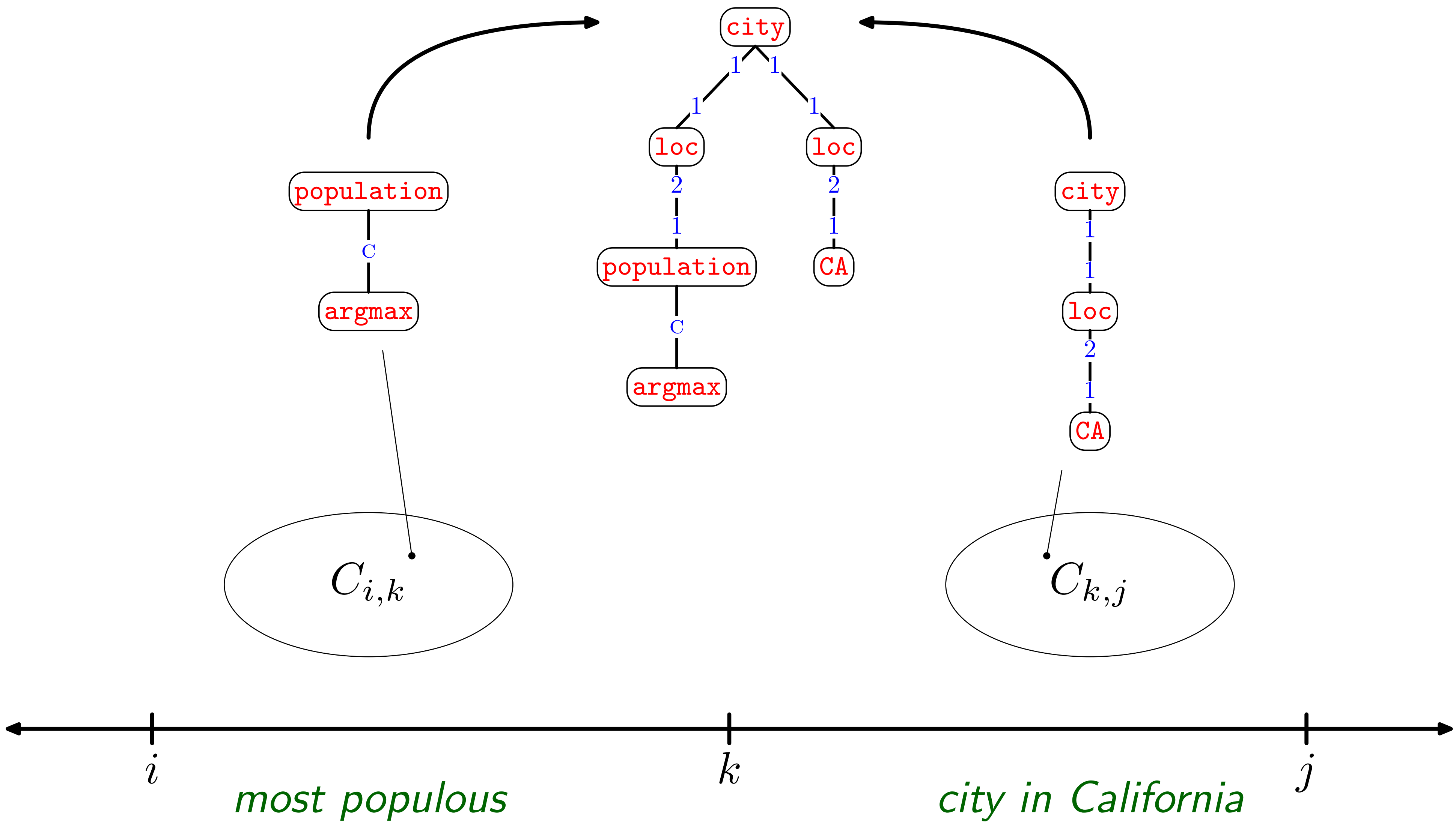
Predicates to DCS Trees (Compositional Semantics)

$C_{i,j}$ = set of DCS trees for span $[i, j]$



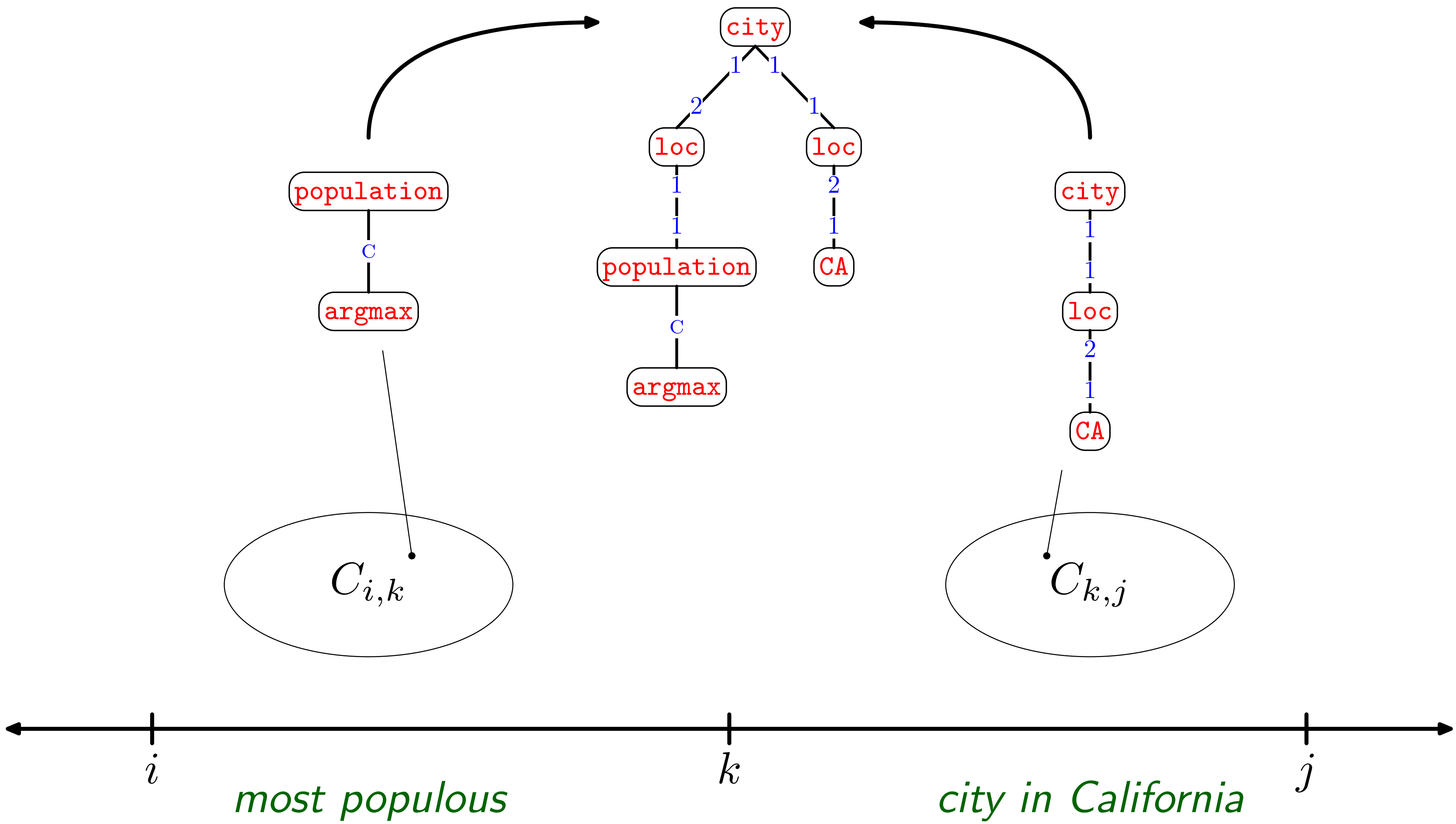
Predicates to DCS Trees (Compositional Semantics)

$C_{i,j}$ = set of DCS trees for span $[i, j]$



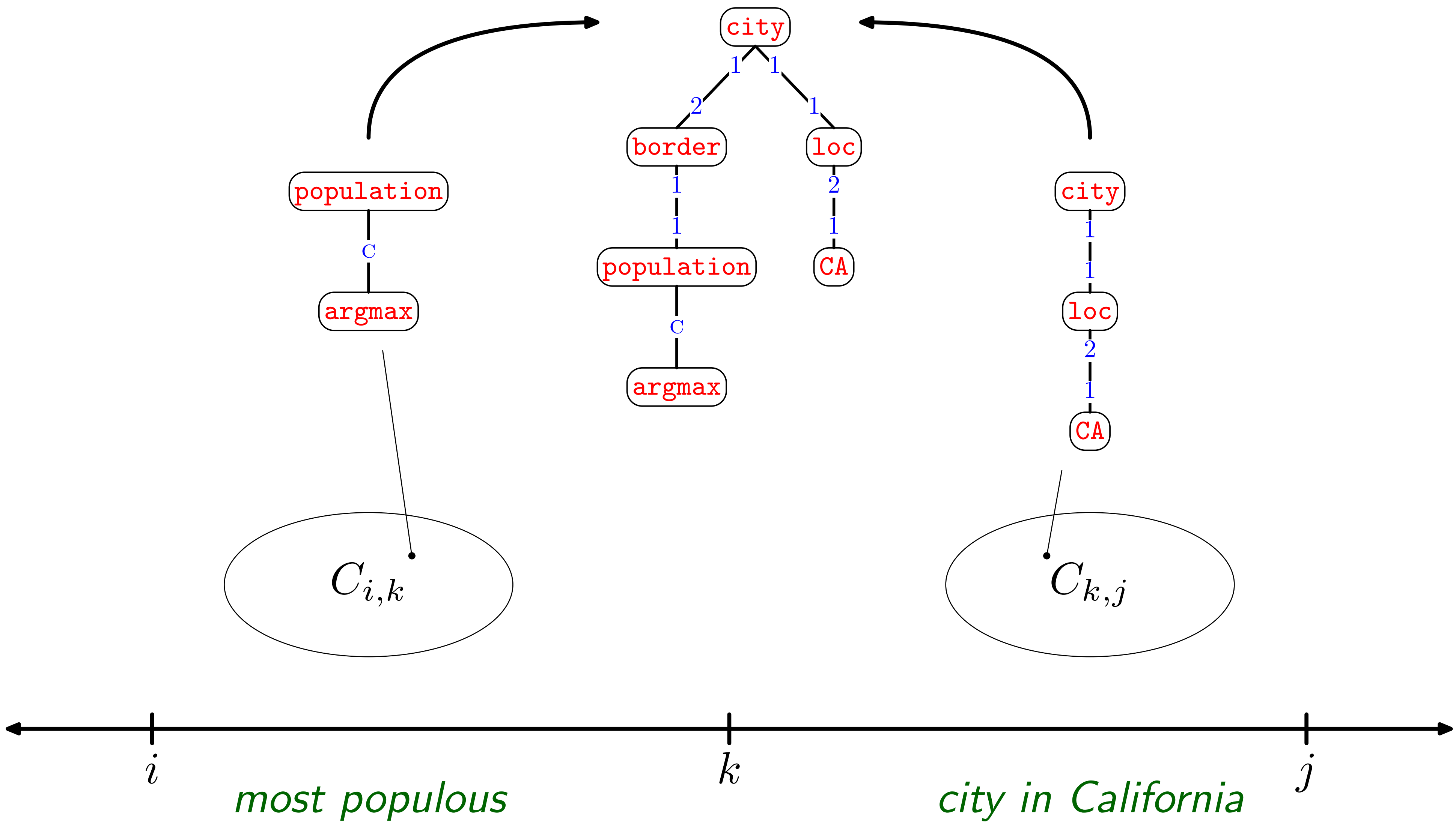
Predicates to DCS Trees (Compositional Semantics)

$C_{i,j}$ = set of DCS trees for span $[i, j]$



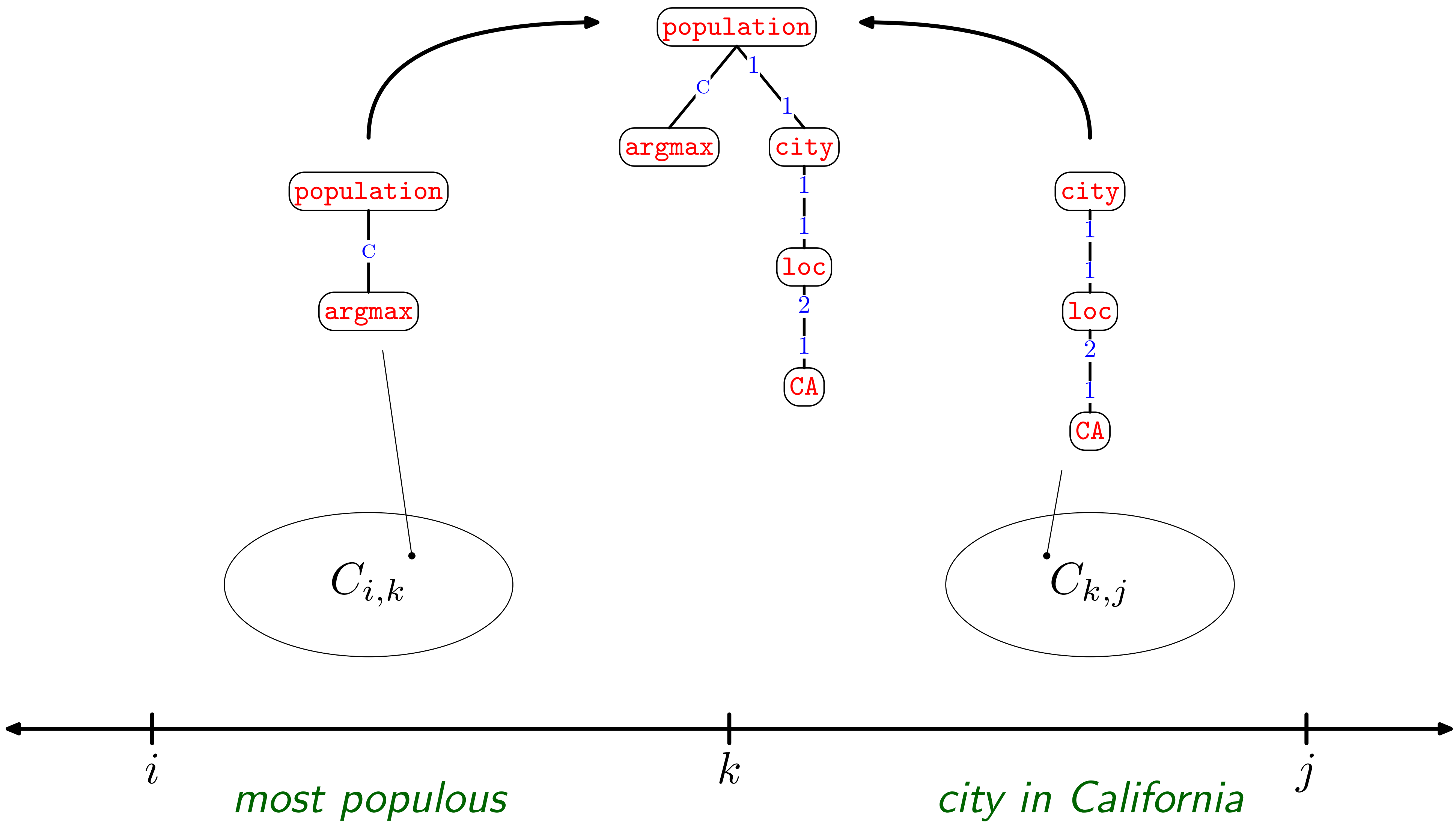
Predicates to DCS Trees (Compositional Semantics)

$C_{i,j}$ = set of DCS trees for span $[i, j]$

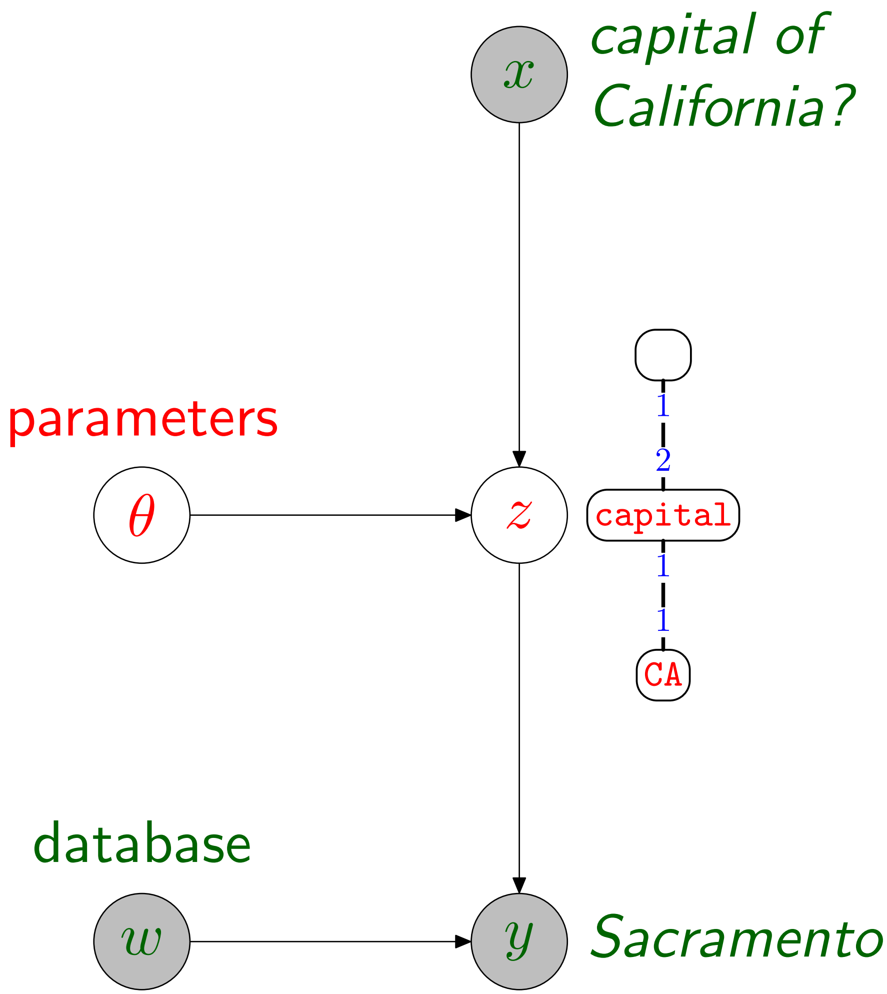


Predicates to DCS Trees (Compositional Semantics)

$C_{i,j}$ = set of DCS trees for span $[i, j]$

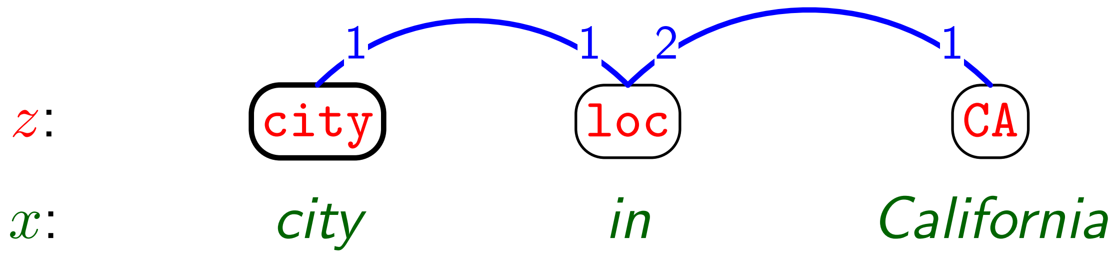


Plan



- What's **possible**? $z \in \mathcal{Z}(x)$
- What's **probable**? $p(z \mid x, \theta)$
- **Learning** θ from (x, y) data

Log-linear Model



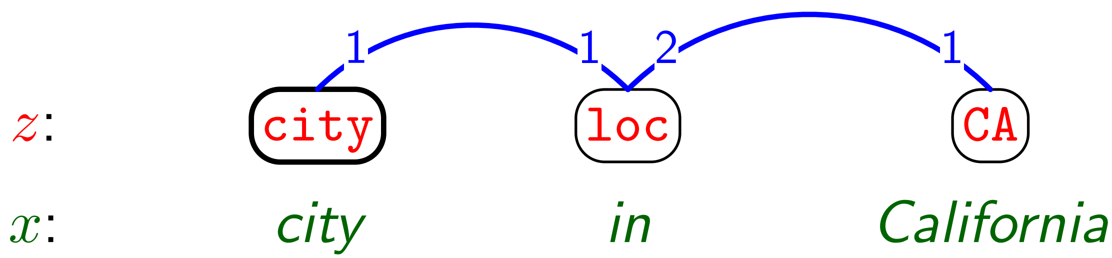
Log-linear Model

z: `city` `loc` `CA`

x: *city* *in* *California*

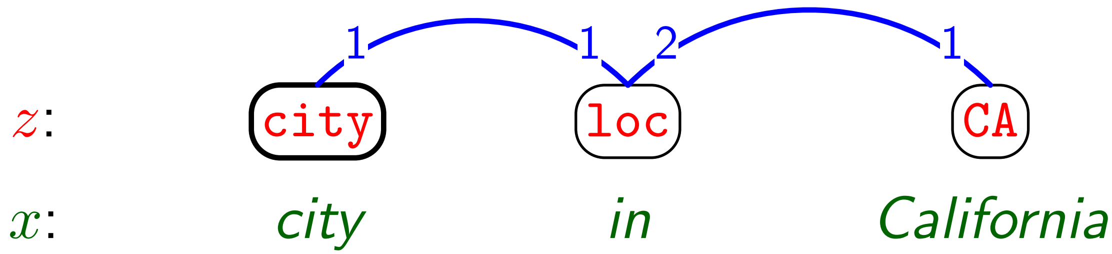
$$\text{features}(x, z) = \left(\phantom{\begin{matrix} \\ \\ \\ \end{matrix}} \right) \in \mathbb{R}^d$$

Log-linear Model



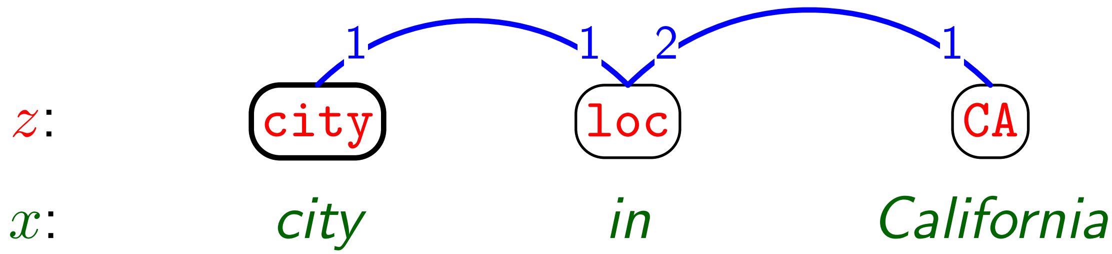
$$\text{features}(x, z) = \begin{pmatrix} \textit{in} \cdots\cdots \texttt{loc} \quad : 1 \\ \\ \\ \end{pmatrix} \in \mathbb{R}^d$$

Log-linear Model



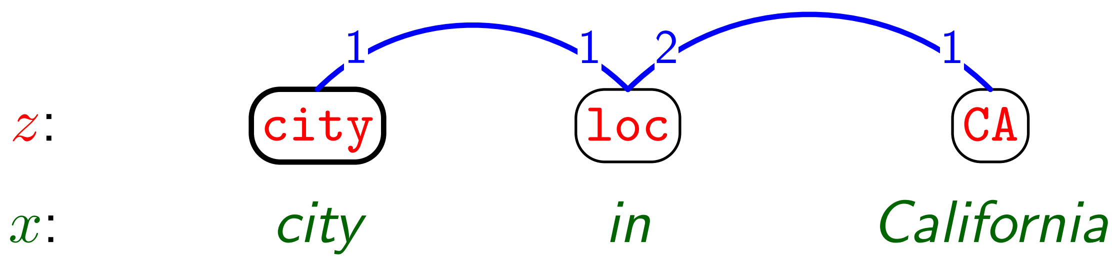
$$\text{features}(x, z) = \begin{pmatrix} \textit{in} \cdots\cdots \texttt{loc} & : 1 \\ \boxed{\texttt{city}}\text{-}1\text{—}1\text{-}\boxed{\texttt{loc}} & : 1 \\ & \end{pmatrix} \in \mathbb{R}^d$$

Log-linear Model



$$\text{features}(x, z) = \begin{pmatrix} \textit{in} \cdots\cdots \texttt{loc} : 1 \\ \boxed{\texttt{city}}\text{-}1\text{—}1\text{-}\boxed{\texttt{loc}} : 1 \\ \cdots \end{pmatrix} \in \mathbb{R}^d$$

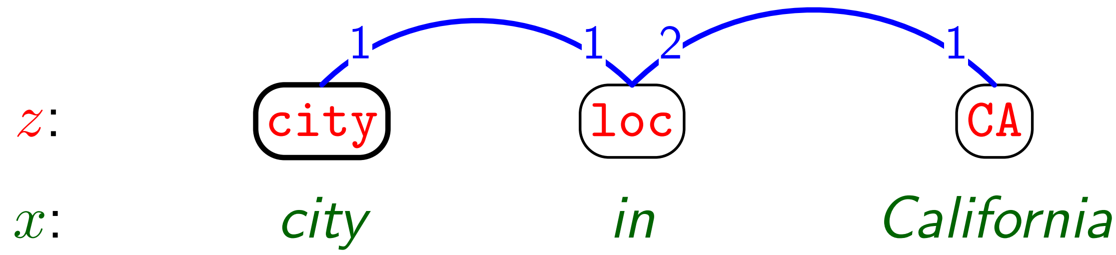
Log-linear Model



$$\text{features}(x, z) = \begin{pmatrix} \textit{in} \cdots\cdots \texttt{loc} & : 1 \\ \boxed{\texttt{city}}\text{-}1\text{—}1\text{-}\boxed{\texttt{loc}} & : 1 \\ \cdots & \end{pmatrix} \in \mathbb{R}^d$$

$$\text{score}(x, z) = \text{features}(x, z) \cdot \theta$$

Log-linear Model

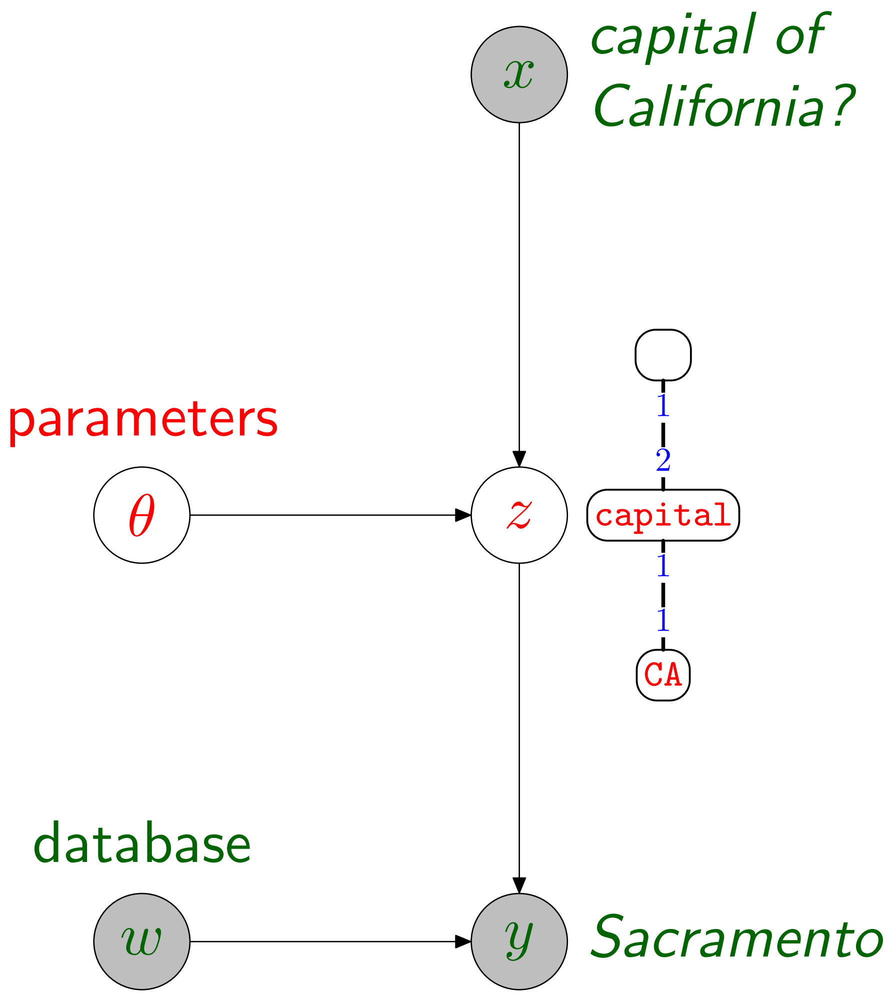


$$\text{features}(x, z) = \begin{pmatrix} \textit{in} \cdots\cdots \texttt{loc} & : 1 \\ \boxed{\texttt{city}} \text{-}1\text{—}1\text{-} \boxed{\texttt{loc}} & : 1 \\ \cdots & \end{pmatrix} \in \mathbb{R}^d$$

$$\text{score}(x, z) = \text{features}(x, z) \cdot \theta$$

$$p(z \mid x, \theta) = \frac{e^{\text{score}(x,z)}}{\sum_{z' \in \mathcal{Z}(x)} e^{\text{score}(x,z')}}$$

Plan



- What's **possible**? $z \in \mathcal{Z}(x)$
- What's **probable**? $p(z \mid x, \theta)$
- **Learning** θ from (x, y) data

Learning

Objective Function:

$$p(y \mid {\color{red}z}, w)\, p({\color{red}z} \mid x, {\color{blue}\theta})$$

Interpretation **Semantic parsing**

Learning

Objective Function:

$$\max_{\theta} \quad \underbrace{p(y \mid z, w)}_{\textbf{Interpretation}} \; \underbrace{p(z \mid x, \theta)}_{\textbf{Semantic parsing}}$$

Learning

Objective Function:

$$\max_{\theta} \sum_{z} p(y \mid z, w)\, p(z \mid x, \theta)$$

Interpretation **Semantic parsing**

Learning

Objective Function:

$$\max_{\theta} \sum_{z} \underbrace{p(y \mid z, w)}_{\textbf{Interpretation}} \underbrace{p(z \mid x, \theta)}_{\textbf{Semantic parsing}}$$

EM-like Algorithm:

parameters θ

$(0, 0, \ldots, 0)$

Learning

Objective Function:

$$\max_{\theta} \sum_{z} p(y \mid z, w)\, p(z \mid x, \theta)$$

Interpretation **Semantic parsing**

EM-like Algorithm:

parameters θ

$(0, 0, \ldots, 0)$ $\xrightarrow{\text{enumerate/score DCS trees}}$

Learning

Objective Function:

$$\max_{\theta} \sum_{z} p(y \mid z, w)\, p(z \mid x, \theta)$$

Interpretation **Semantic parsing**

EM-like Algorithm:

parameters θ

$(0, 0, \ldots, 0)$

$\xrightarrow{\text{enumerate/score DCS trees}}$

k-best list

tree1 ✗
tree2 ✗
tree3 ✓
tree4 ✗
tree5 ✗

Learning

Objective Function:

$$\max_{\theta} \sum_{z} p(y \mid z, w)\, p(z \mid x, \theta)$$

Interpretation **Semantic parsing**

EM-like Algorithm:

parameters θ

$(0.2, -1.3, \ldots, 0.7)$

$\xrightarrow{\text{enumerate/score DCS trees}}$

$\xleftarrow[\text{numerical optimization (L-BFGS)}]{}$

k-best list

- `tree1` ✗
- `tree2` ✗
- `tree3` ✓
- `tree4` ✗
- `tree5` ✗

Learning

Objective Function:

$$\max_{\theta} \sum_{z} p(y \mid z, w)\, p(z \mid x, \theta)$$

Interpretation **Semantic parsing**

EM-like Algorithm:

parameters θ

$(0.2, -1.3, \ldots, 0.7)$

enumerate/score DCS trees →

← numerical optimization (L-BFGS)

k-best list

`tree3` ✓
`tree8` ✓
`tree6` ✗
`tree2` ✗
`tree4` ✗

Learning

Objective Function:

$$\max_{\theta} \sum_{z} p(y \mid z, w)\, p(z \mid x, \theta)$$

Interpretation **Semantic parsing**

EM-like Algorithm:

parameters θ

$(0.3, -1.4, \ldots, 0.6)$

$\xrightarrow{\text{enumerate/score DCS trees}}$

$\xleftarrow[\text{numerical optimization (L-BFGS)}]{}$

k-best list

`tree3` ✓
`tree8` ✓
`tree6` ✗
`tree2` ✗
`tree4` ✗

Learning

Objective Function:

$$\max_{\theta} \sum_{z} \underbrace{p(y \mid z, w)}_{\textbf{Interpretation}} \, \underbrace{p(z \mid x, \theta)}_{\textbf{Semantic parsing}}$$

EM-like Algorithm:

parameters θ

$(0.3, -1.4, \ldots, 0.6)$

enumerate/score DCS trees →

← numerical optimization (L-BFGS)

k-best list

`tree3` ✓

`tree8` ✓

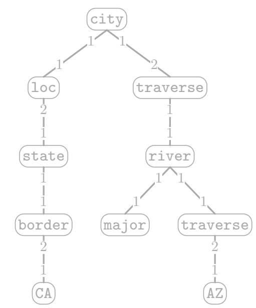
`tree2` ✗

`tree4` ✗

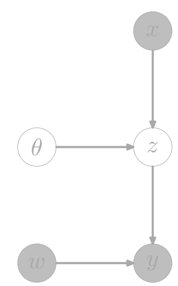
`tree9` ✗

Outline

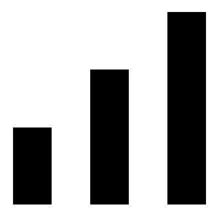
Representation



Learning



Experiments



US Geography Benchmark

Standard semantic parsing benchmark since 1990s

600 training examples, 280 test examples

US Geography Benchmark

Standard semantic parsing benchmark since 1990s

600 training examples, 280 test examples

What is the highest point in Florida?

How many states have a city called Rochester?

What is the longest river that runs through a state that borders Tennessee?

Of the states washed by the Mississippi river which has the lowest point?

...

US Geography Benchmark

Standard semantic parsing benchmark since 1990s

600 training examples, 280 test examples

What is the highest point in Florida?

$\Rightarrow$ `answer(A,highest(A,(place(A),loc(A,B),const(B,stateid(florida)))))`

How many states have a city called Rochester?

$\Rightarrow$ `answer(A,count(B,(state(B),loc(C,B),const(C,cityid(rochester,_))),A))`

What is the longest river that runs through a state that borders Tennessee?

$\Rightarrow$ `answer(A,longest(A,(river(A),traverse(A,B),state(B),next_to(B,C),const(C,stateid(tennessee)))))`

Of the states washed by the Mississippi river which has the lowest point?

$\Rightarrow$ `answer(A,lowest(B,(state(A),traverse(C,A),const(C,riverid(mississippi)),loc(B,A),place(B))))`

...

Supervision in past work: question + program

US Geography Benchmark

Standard semantic parsing benchmark since 1990s

600 training examples, 280 test examples

What is the highest point in Florida?
$\Rightarrow$ *Walton County*

How many states have a city called Rochester?
$\Rightarrow$ *2*

What is the longest river that runs through a state that borders Tennessee?
$\Rightarrow$ *Missouri*

Of the states washed by the Mississippi river which has the lowest point?
$\Rightarrow$ *Louisiana*

...

Supervision in past work: question + program

Supervision in this work: question + answer

Input to Learning Algorithm

Training data (600 examples)

What is the highest point in Florida?	$\Rightarrow$	*Walton County*
How many states have a city called Rochester?	$\Rightarrow$	*2*
What is the longest river that runs through a state that borders Tennessee?	$\Rightarrow$	*Missouri*
Of the states washed by the Mississippi river which has the lowest point?	$\Rightarrow$	*Louisiana*
...		...

Input to Learning Algorithm

Training data (600 examples)

What is the highest point in Florida?	$\Rightarrow$	*Walton County*
How many states have a city called Rochester?	$\Rightarrow$	*2*
What is the longest river that runs through a state that borders Tennessee?	$\Rightarrow$	*Missouri*
Of the states washed by the Mississippi river which has the lowest point?	$\Rightarrow$	*Louisiana*
...		...

Lexicon (75 words)

`city`	$\Rightarrow$	*city*
`state`	$\Rightarrow$	*state*
`mountain`	$\Rightarrow$	*mountain, peak*
...		...

Input to Learning Algorithm

Training data (600 examples)

What is the highest point in Florida?	⇒	*Walton County*
How many states have a city called Rochester?	⇒	*2*
What is the longest river that runs through a state that borders Tennessee?	⇒	*Missouri*
Of the states washed by the Mississippi river which has the lowest point?	⇒	*Louisiana*
...		...

Lexicon (75 words)

`city`	⇒	*city*
`state`	⇒	*state*
`mountain`	⇒	*mountain, peak*
...		...

Database

`city`

San Francisco
Chicago
Boston
...

`state`

Alabama
Alaska
Arizona
...

`loc`

Mount Shasta	California
San Francisco	California
Boston	Massachusetts
...	...

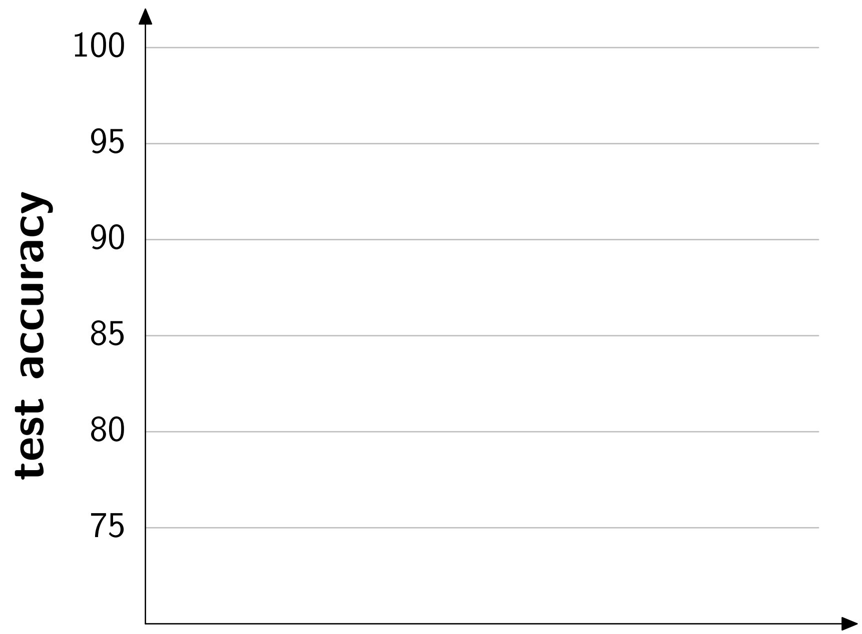
`border`

Washington	Oregon
Washington	Idaho
Oregon	Washington
...	...

... ...

Experiment 1

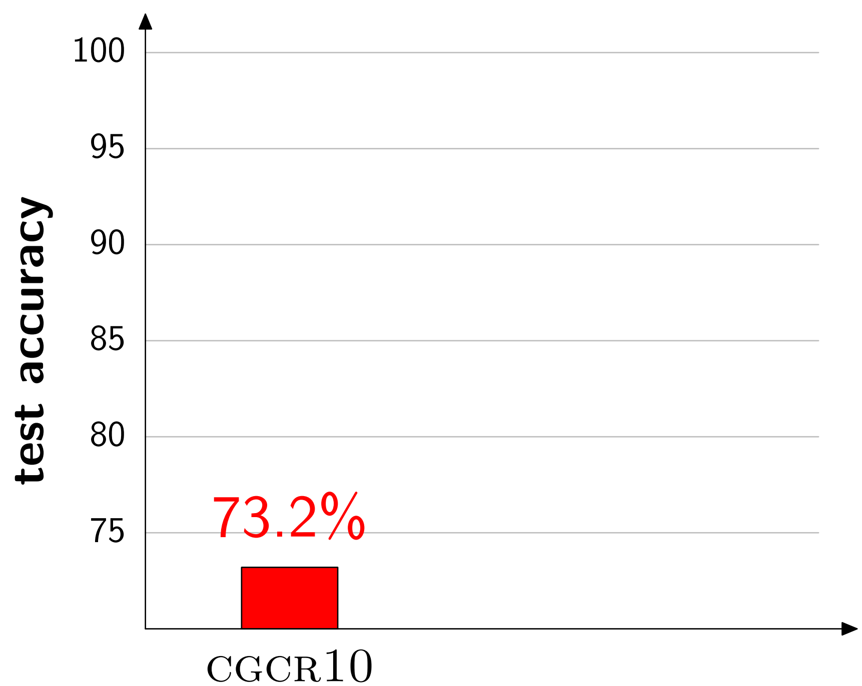
On GEO, 250 training examples, 250 test examples



Experiment 1

On GEO, 250 training examples, 250 test examples

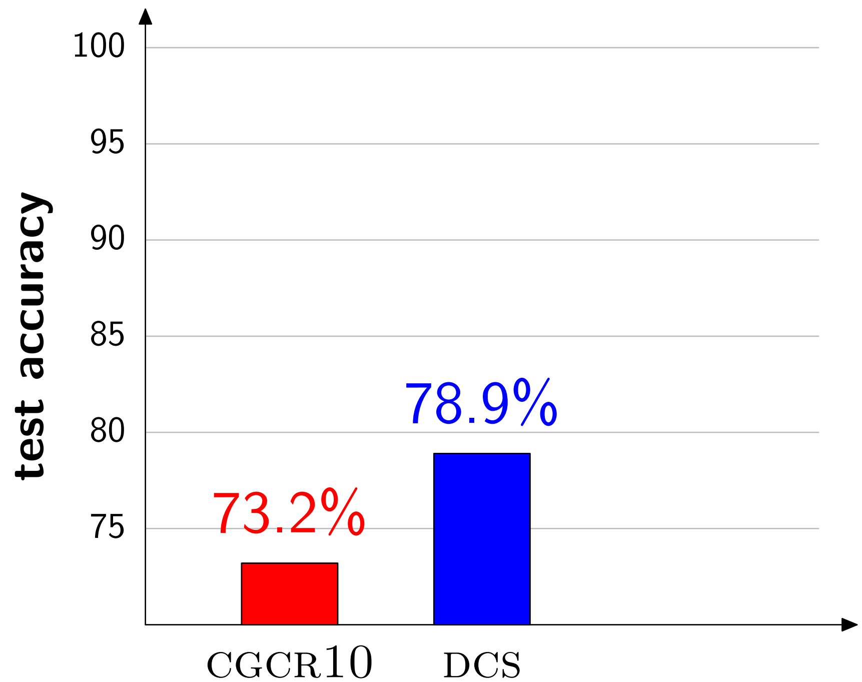
System	Description	Lexicon (gen./spec.)	Logical forms
CGCR10	FunQL [Clarke et al., 2010]	✔ ✔	✗



Experiment 1

On GEO, 250 training examples, 250 test examples

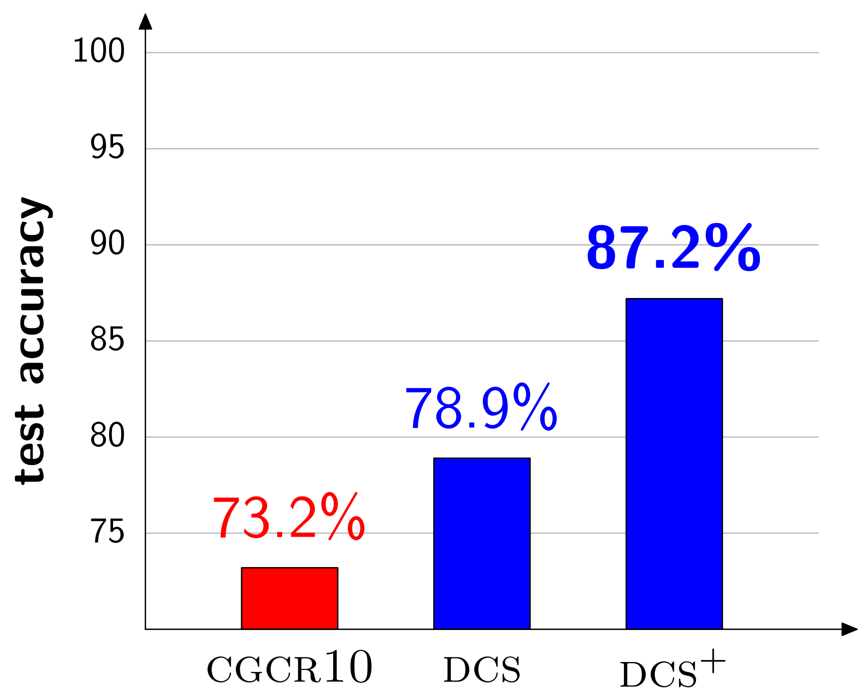
System	Description	Lexicon (gen./spec.)	Logical forms
CGCR10	FunQL [Clarke et al., 2010]	✓ ✓	✗
DCS	our system	✓ ✗	✗



Experiment 1

On GEO, 250 training examples, 250 test examples

System	Description	Lexicon (gen./spec.)	Logical forms
CGCR10	FunQL [Clarke et al., 2010]	✓ ✓	✗
DCS	our system	✓ ✗	✗
DCS$^+$	our system	✓ ✓	✗



Experiment 2

On Geo, 600 training examples, 280 test examples

Experiment 2

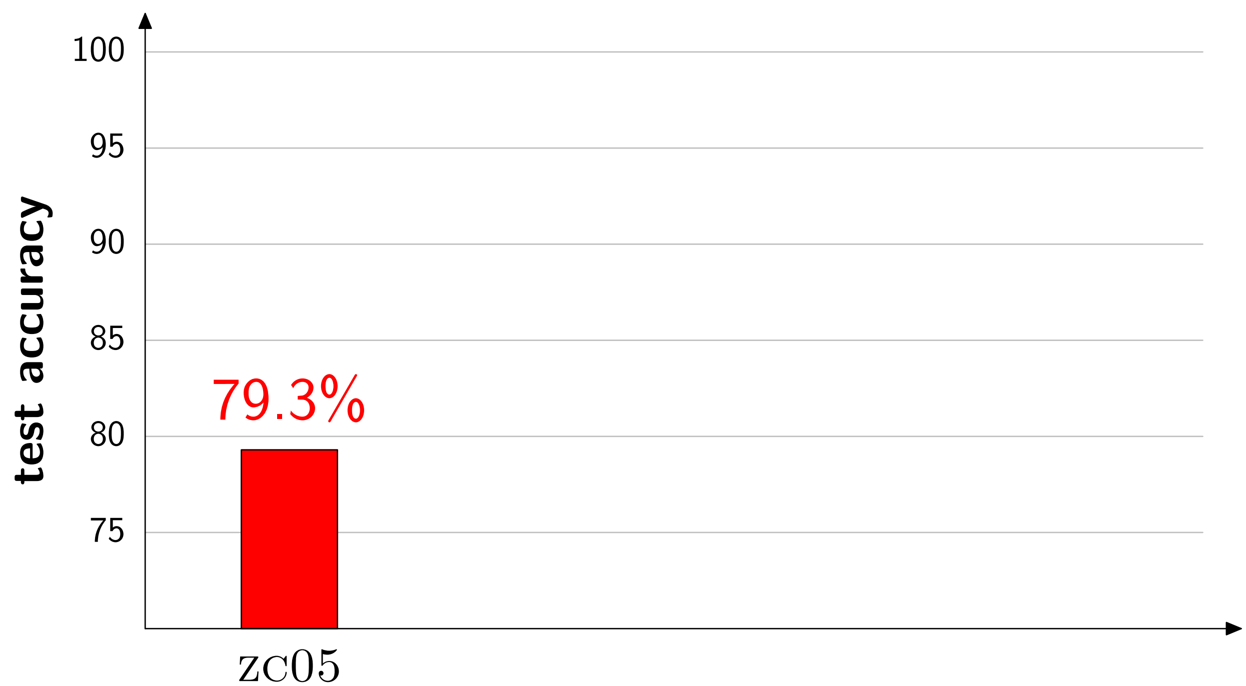
On GEO, 600 training examples, 280 test examples

System	**Description**	**Lexicon**	**Logical forms**

Experiment 2

On GEO, 600 training examples, 280 test examples

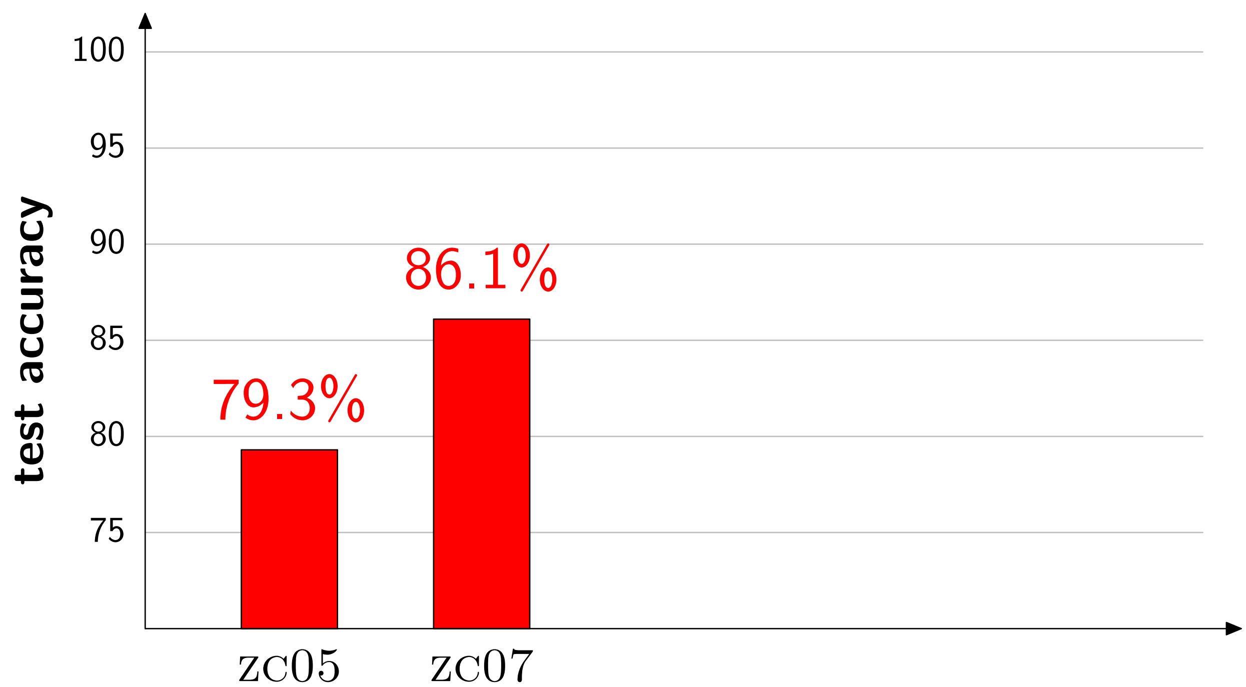
System	Description	Lexicon	Logical forms
ZC05	CCG [Zettlemoyer & Collins, 2005]	✗ ✗	✓



Experiment 2

On GEO, 600 training examples, 280 test examples

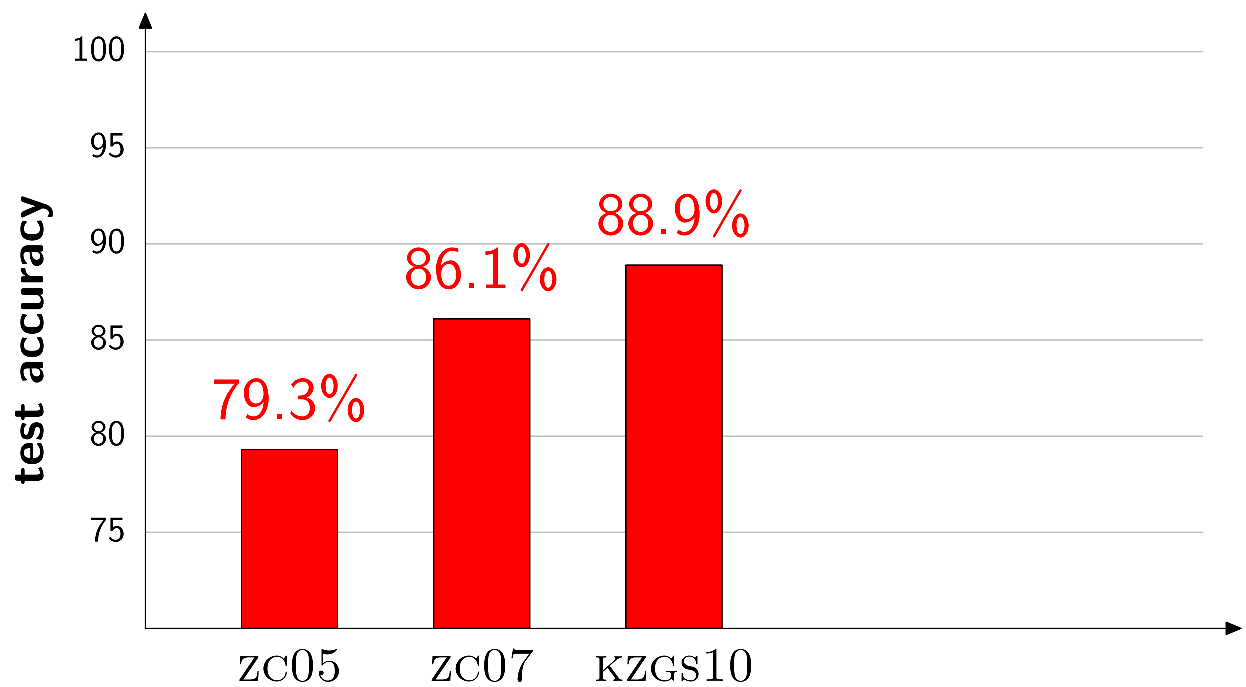
System	**Description**	**Lexicon**	**Logical forms**
ZC05	CCG [Zettlemoyer & Collins, 2005]	✗ ✗	✓
ZC07	relaxed CCG [Zettlemoyer & Collins, 2007]	✗ ✗	✓



Experiment 2

On GEO, 600 training examples, 280 test examples

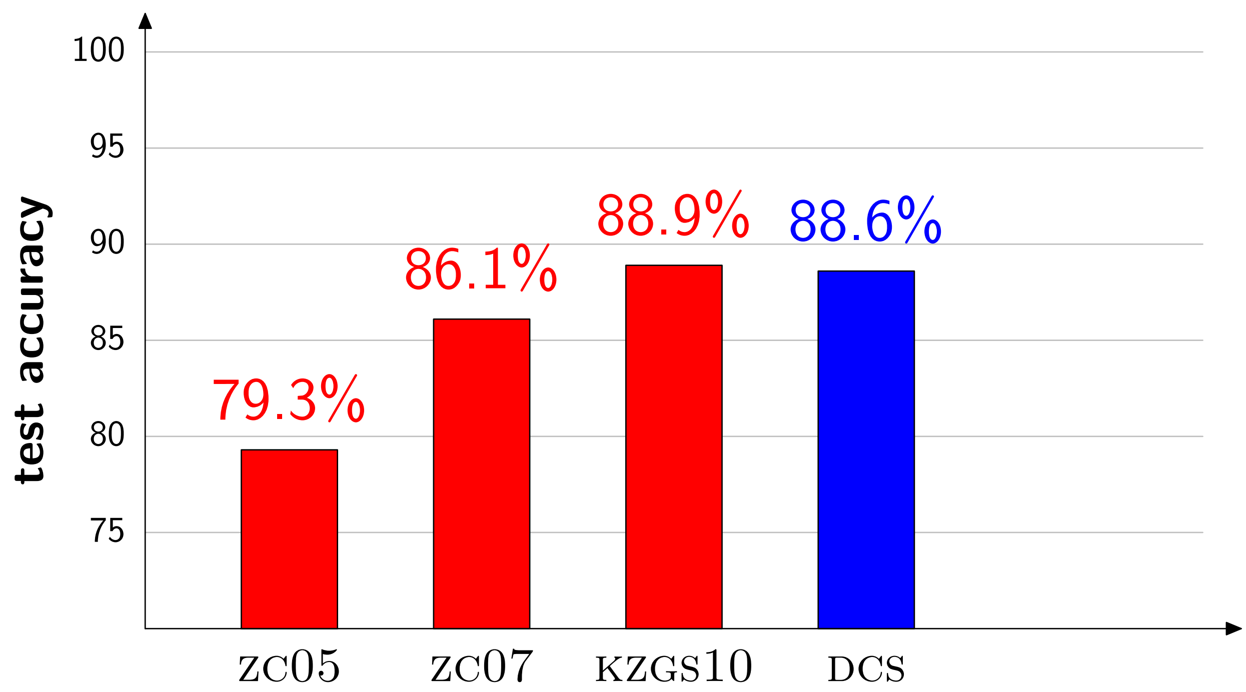
System	Description	Lexicon	Logical forms
ZC05	CCG [Zettlemoyer & Collins, 2005]	✗ ✗	✓
ZC07	relaxed CCG [Zettlemoyer & Collins, 2007]	✗ ✗	✓
KZGS10	CCG w/unification [Kwiatkowski et al., 2010]	✗ ✗	✓



Experiment 2

On GEO, 600 training examples, 280 test examples

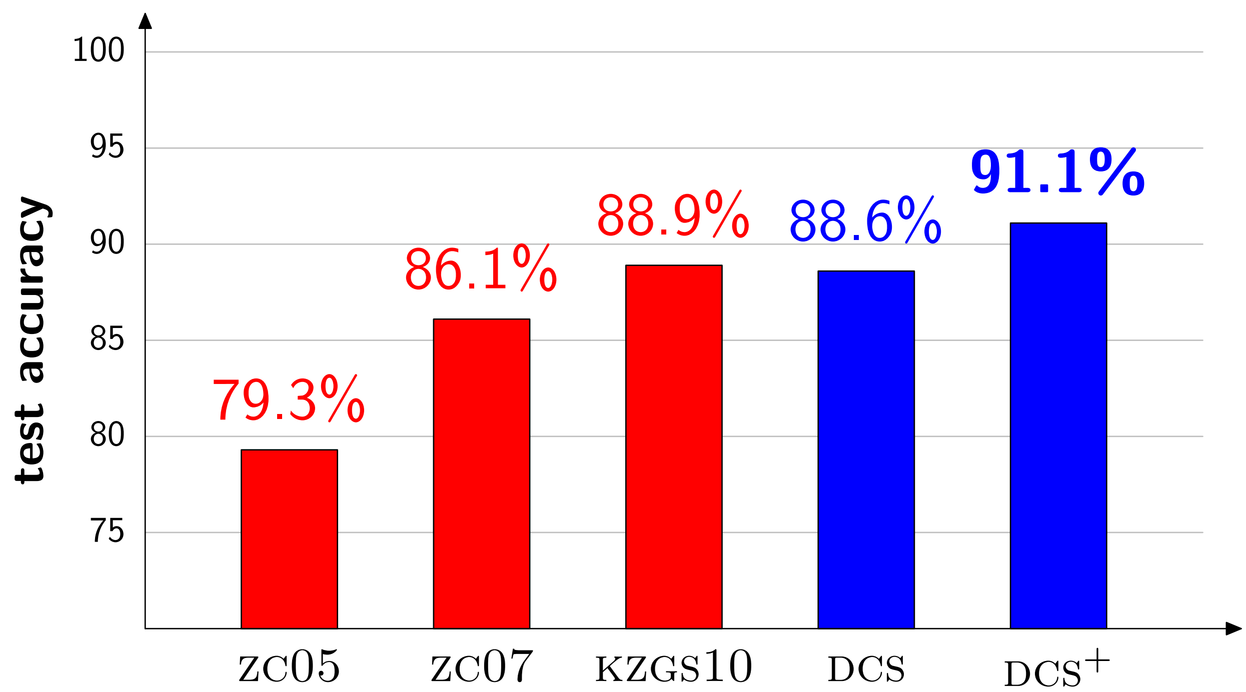
System	Description	Lexicon	Logical forms
ZC05	CCG [Zettlemoyer & Collins, 2005]	✗ ✗	✓
ZC07	relaxed CCG [Zettlemoyer & Collins, 2007]	✗ ✗	✓
KZGS10	CCG w/unification [Kwiatkowski et al., 2010]	✗ ✗	✓
DCS	our system	✓ ✗	✗



Experiment 2

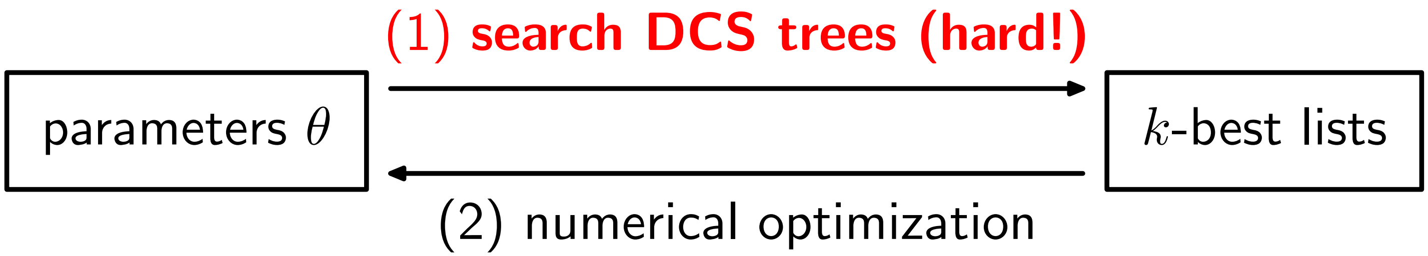
On GEO, 600 training examples, 280 test examples

System	Description	Lexicon	Logical forms
ZC05	CCG [Zettlemoyer & Collins, 2005]	✗ ✗	✓
ZC07	relaxed CCG [Zettlemoyer & Collins, 2007]	✗ ✗	✓
KZGS10	CCG w/unification [Kwiatkowski et al., 2010]	✗ ✗	✓
DCS	our system	✓ ✗	✗
DCS^+	our system	✓ ✓	✗

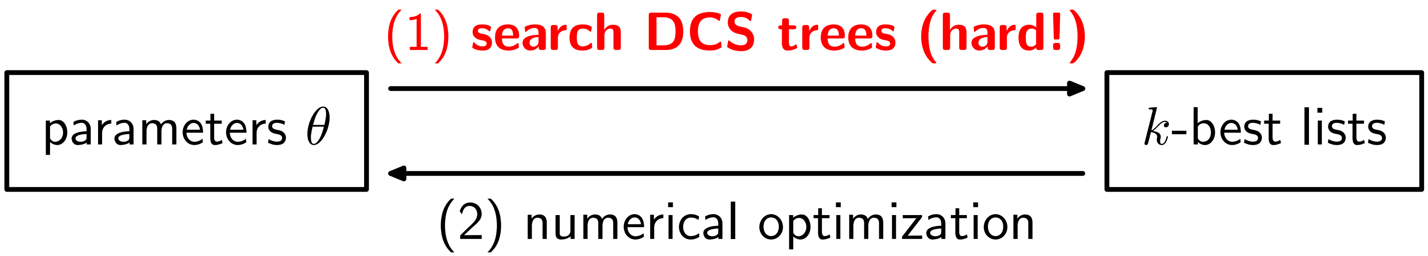


Some Intuition on Learning

Some Intuition on Learning



Some Intuition on Learning



If no DCS tree on k-best list is correct, skip example in (2)

Some Intuition on Learning

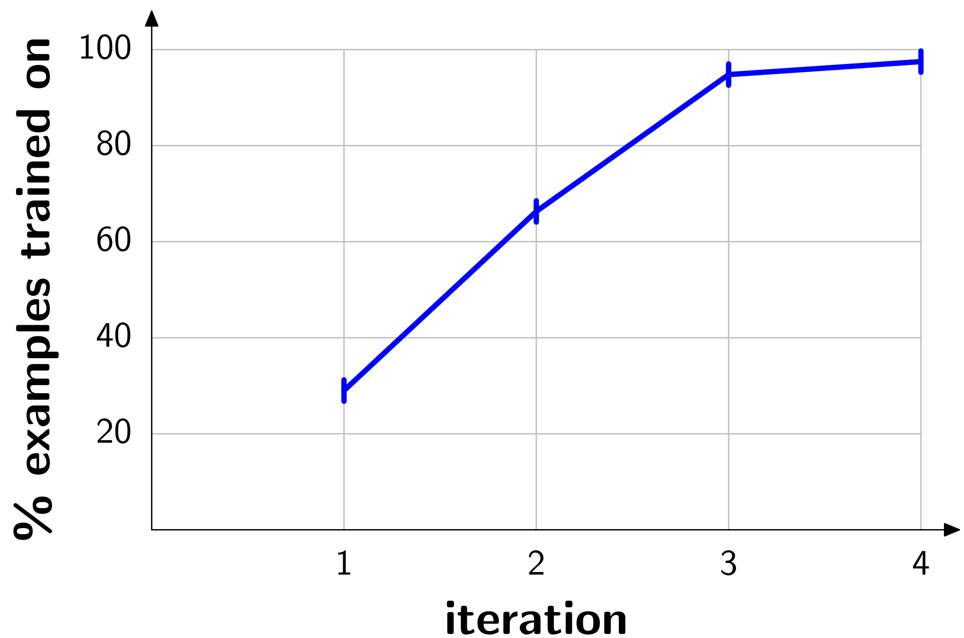
(1) search DCS trees (hard!)

parameters θ → k-best lists

k-best lists → parameters θ

(2) numerical optimization

If no DCS tree on k-best list is correct, skip example in (2)



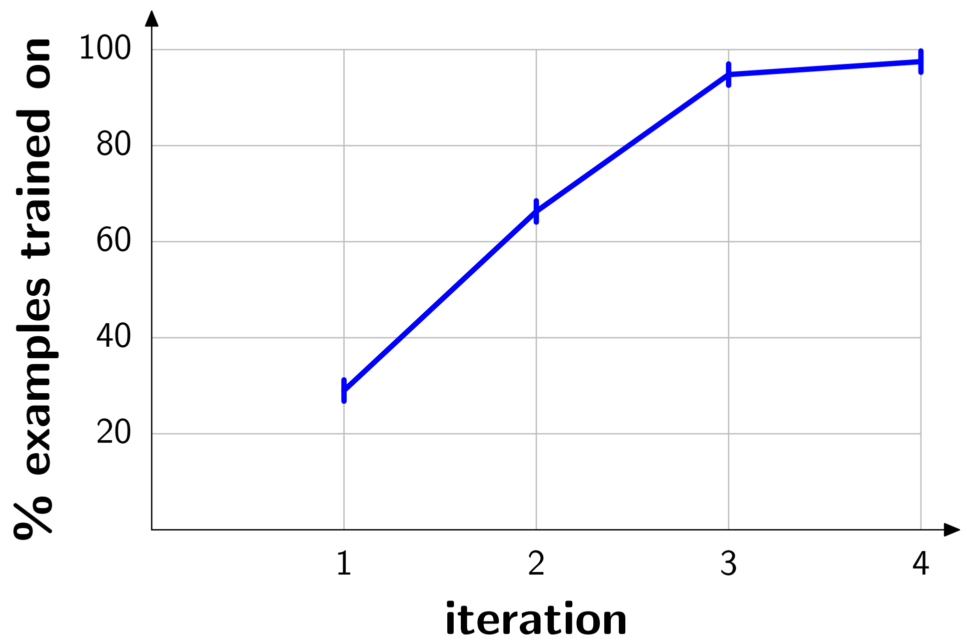
Some Intuition on Learning

(1) **search DCS trees (hard!)**

parameters θ ⇄ k-best lists

(2) numerical optimization

If no DCS tree on k-best list is correct, skip example in (2)



Effect: automatic curriculum learning, learning improves search

Current Limitations

Current Limitations

Only using forward information

Execute program to get answer, but want to invert

Current Limitations

Only using forward information

Execute program to get answer, but want to invert

Non-identifiability of program

If all cities in database are in US, then

can't distinguish $\{c : \texttt{city}(c)\}$ and $\{c : \texttt{city}(c) \wedge \texttt{loc}(c, \texttt{US})\}$

Current Limitations

Only using forward information

Execute program to get answer, but want to invert

Non-identifiability of program

If all cities in database are in US, then

can't distinguish $\{c : \texttt{city}(c)\}$ and $\{c : \texttt{city}(c) \wedge \texttt{loc}(c, \texttt{US})\}$

Unknown facts: *How far is Los Angeles from Boston?*

Database has no distance information

Current Limitations

Only using forward information

Execute program to get answer, but want to invert

Non-identifiability of program

If all cities in database are in US, then

can't distinguish $\{c : \texttt{city}(c)\}$ and $\{c : \texttt{city}(c) \wedge \texttt{loc}(c, \texttt{US})\}$

Unknown facts: *How far is Los Angeles from Boston?*

Database has no distance information

Unknown concepts: *What states are landlocked?*

Need to induce database view for $\texttt{landlocked}(x) = \neg\texttt{border}(x, \texttt{ocean})$

Conclusion

Goal: learn to answer questions from question/answer pairs

Conclusion

Goal: learn to answer questions from question/answer pairs

Empirical result:

DCS (no logical forms) $\approxeq$ existing systems (with logical forms)

Conclusion

Goal: learn to answer questions from question/answer pairs

Empirical result:

DCS (no logical forms) $\approxeq$ existing systems (with logical forms)

Conceptual contribution: DCS trees

- Trees: connects dependency syntax with efficient evaluation

Conclusion

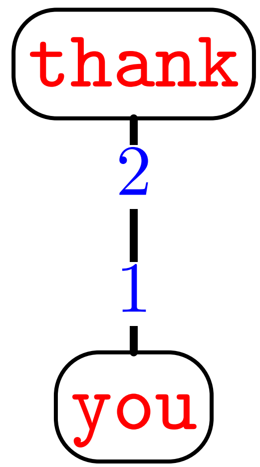
Goal: learn to answer questions from question/answer pairs

Empirical result:

DCS (no logical forms) $\approxeq$ existing systems (with logical forms)

Conceptual contribution: DCS trees

- Trees: connects dependency syntax with efficient evaluation
- Mark-Execute: unifying framework for handling scope


thank
2
1
you