Natural Language Understanding: Foundations and State-of-the-Art

Percy Liang

ICML Tutorial
July 6, 2015

What is **natural language understanding***?*

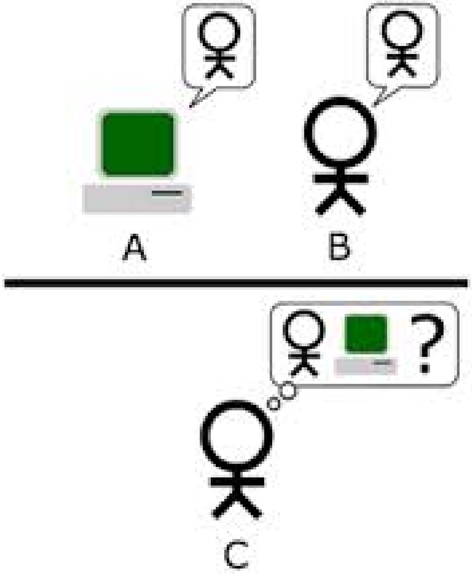
Humans are the only example

The Imitation Game (1950)

"Can machines think?"

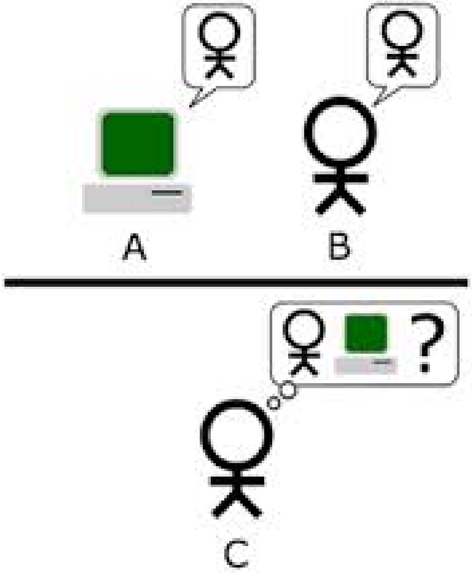
The Imitation Game (1950)

"Can machines think?"



The Imitation Game (1950)

”Can machines think?”



Q: Please write me a sonnet on the subject of the Forth Bridge.

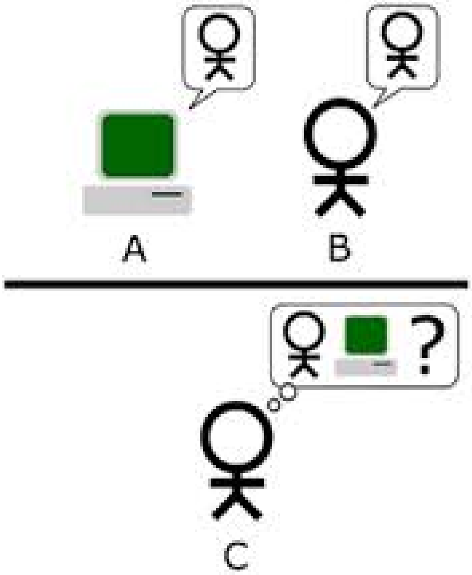
A: Count me out on this one. I never could write poetry.

Q: Add 34957 to 70764.

A: (Pause about 30 seconds and then give as answer) 105621.

The Imitation Game (1950)

"Can machines think?"



Q: Please write me a sonnet on the subject of the Forth Bridge.

A: Count me out on this one. I never could write poetry.

Q: Add 34957 to 70764.

A: (Pause about 30 seconds and then give as answer) 105621.

- **Behavioral** test
- ...of **intelligence**, not just natural language understanding

IBM Watson

William Wilkinson's "An Account of the Principalities of Wallachia and Moldavia" inspired this author's most famous novel.

Siri


7:34 AM
What can I help you with?
“What's the best Japanese restaurant in San Francisco”
I found a number of Japanese restaurants… 24 of them are in San Francisco. I've sorted them by rating:
Market Street
Sushi Zone
690 reviews

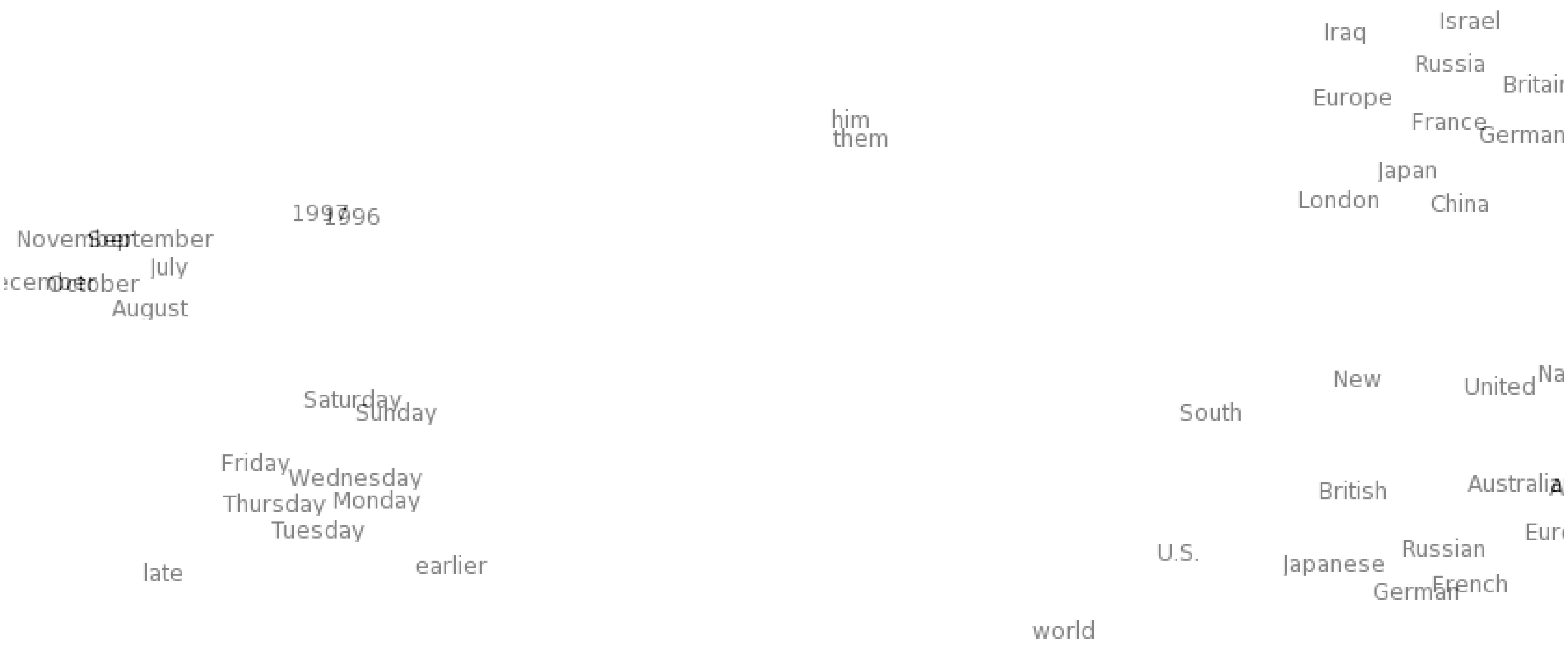
Google


Google
how many people live in lille
Web
Maps
News
Shopping
Images
More
Search tools
About 14,200,000 results (0.54 seconds)
227,560 (2010)
Lille, Population

Representations for **natural language understanding***?*

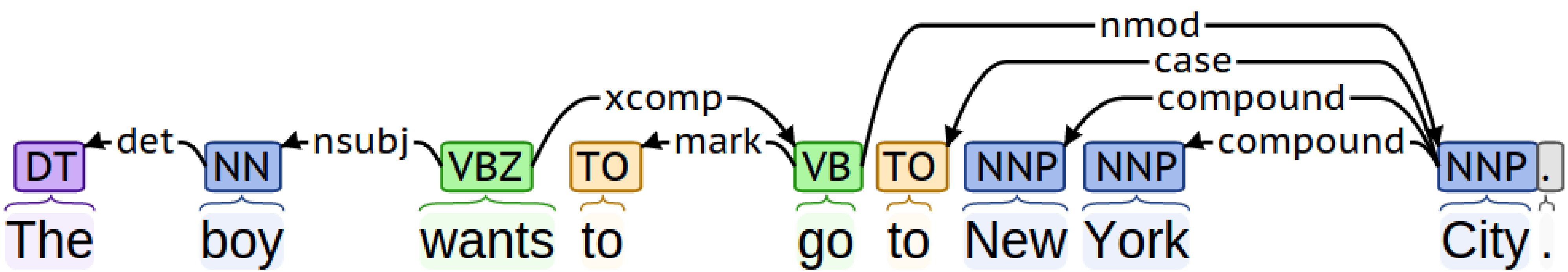
Word vectors?

Word vectors?



Dependency parse trees?

The boy wants to go to New York City.



Frames?

Cynthia	*sold*	*the bike*	*to*	*Bob*	*for*	*$200*
SELLER	**PREDICATE**	GOODS		BUYER		PRICE

Logical forms?

What is the largest city in California?

$$\downarrow$$

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

Why ICML?

Opportunity for transfer of ideas between ML and NLP

Why ICML?

Opportunity for transfer of ideas between ML and NLP

- mid-1970s: **HMMs** for speech recognition $\Rightarrow$ probabilistic models

Why ICML?

Opportunity for transfer of ideas between ML and NLP

- mid-1970s: **HMMs** for speech recognition $\Rightarrow$ probabilistic models
- early 2000s: **conditional random fields** for part-of-speech tagging $\Rightarrow$ structured prediction

Why ICML?

Opportunity for transfer of ideas between ML and NLP

- mid-1970s: **HMMs** for speech recognition $\Rightarrow$ probabilistic models
- early 2000s: **conditional random fields** for part-of-speech tagging $\Rightarrow$ structured prediction
- early 2000s: **Latent Dirichlet Allocation** for modeling text documents $\Rightarrow$ topic modeling

Why ICML?

Opportunity for transfer of ideas between ML and NLP

- mid-1970s: **HMMs** for speech recognition $\Rightarrow$ probabilistic models
- early 2000s: **conditional random fields** for part-of-speech tagging $\Rightarrow$ structured prediction
- early 2000s: **Latent Dirichlet Allocation** for modeling text documents $\Rightarrow$ topic modeling
- mid 2010s: **sequence-to-sequence models** for machine translation $\Rightarrow$ neural networks with memory/state

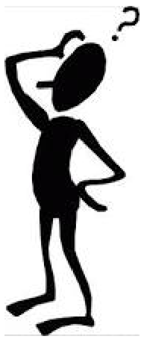
Why ICML?

Opportunity for transfer of ideas between ML and NLP

- mid-1970s: **HMMs** for speech recognition $\Rightarrow$ probabilistic models
- early 2000s: **conditional random fields** for part-of-speech tagging $\Rightarrow$ structured prediction
- early 2000s: **Latent Dirichlet Allocation** for modeling text documents $\Rightarrow$ topic modeling
- mid 2010s: **sequence-to-sequence models** for machine translation $\Rightarrow$ neural networks with memory/state
- now: **???** for natural language understanding

Goals of this tutorial

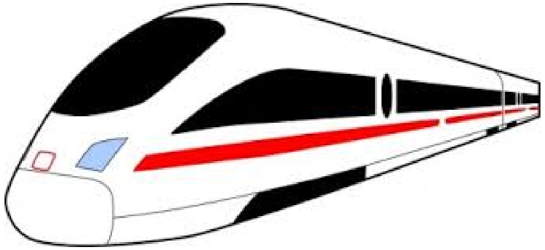
- Provide **intuitions** about natural language



Goals of this tutorial

- Provide **intuitions** about natural language

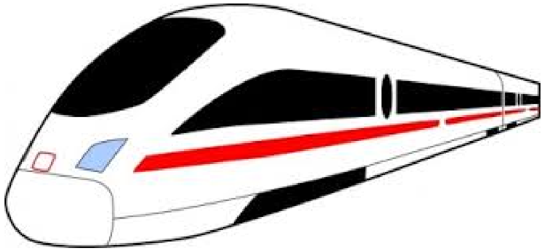
- Describe current **state-of-the-art** methods



Goals of this tutorial

- Provide **intuitions** about natural language

- Describe current **state-of-the-art** methods



- Propose **challenges** / opportunities

Tips

What to expect:

- A lot of tutorial is about thinking about the phenomena in language
- Minimal details on methods and empirical results

Tips

What to expect:

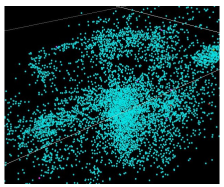
- A lot of tutorial is about thinking about the phenomena in language
- Minimal details on methods and empirical results

What to look for:

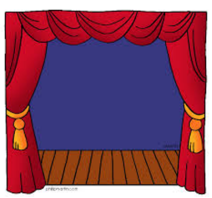
- Challenging machine learning problems: representation learning, structured prediction
- Think about the **end-to-end** problem and decide what phenomena to focus on, which ones to punt on, which ones are bulldozed by ML

Outline

Properties of language



Distributional semantics



Frame semantics

Model-theoretic semantics

Reflections

Levels of linguistic analyses

natural language utterance

Levels of linguistic analyses

Syntax: what is grammatical?

natural language utterance

Levels of linguistic analyses

$\boxed{\textbf{Semantics}\text{: what does it mean?}}$

$\boxed{\textbf{Syntax}\text{: what is grammatical?}}$

natural language utterance

Levels of linguistic analyses

Pragmatics: what does it do?

Semantics: what does it mean?

Syntax: what is grammatical?

natural language utterance

Analogy with programming languages

Syntax: no compiler errors

Semantics: no implementation bugs

Pragmatics: implemented the right algorithm

Analogy with programming languages

Syntax: no compiler errors

Semantics: no implementation bugs

Pragmatics: implemented the right algorithm

Different syntax, same semantics (5):

$$2 + 3 \Leftrightarrow 3 + 2$$

Analogy with programming languages

Syntax: no compiler errors

Semantics: no implementation bugs

Pragmatics: implemented the right algorithm

Different syntax, same semantics (5):

$$2 + 3 \quad \Leftrightarrow \quad 3 + 2$$

Same syntax, different semantics (1 and 1.5):

$$3 \ / \ 2 \text{ (Python 2.7)} \quad \not\Leftrightarrow \quad 3 \ / \ 2 \text{ (Python 3)}$$

Analogy with programming languages

Syntax: no compiler errors

Semantics: no implementation bugs

Pragmatics: implemented the right algorithm

Different syntax, same semantics (5):

$$2 + 3 \Leftrightarrow 3 + 2$$

Same syntax, different semantics (1 and 1.5):

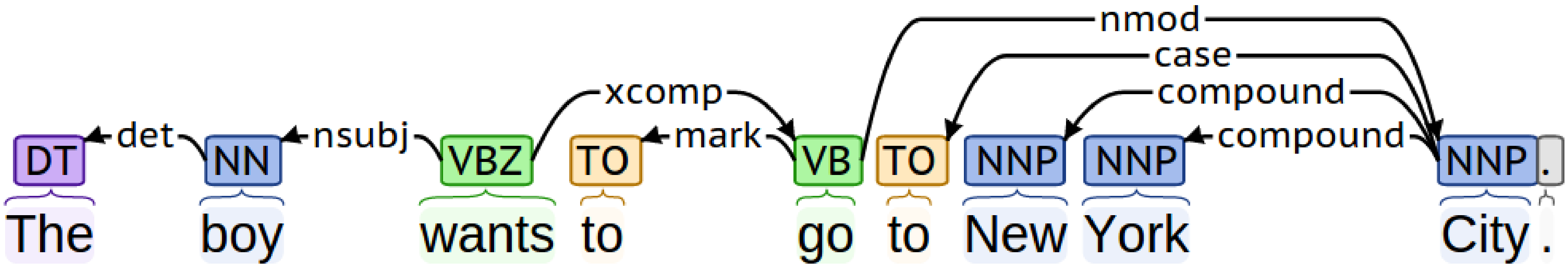
$$3 \;/\; 2 \text{ (Python 2.7)} \not\Leftrightarrow 3 \;/\; 2 \text{ (Python 3)}$$

Good semantics, bad pragmatics:

correct implementation of deep neural network
for estimating coin flip prob.

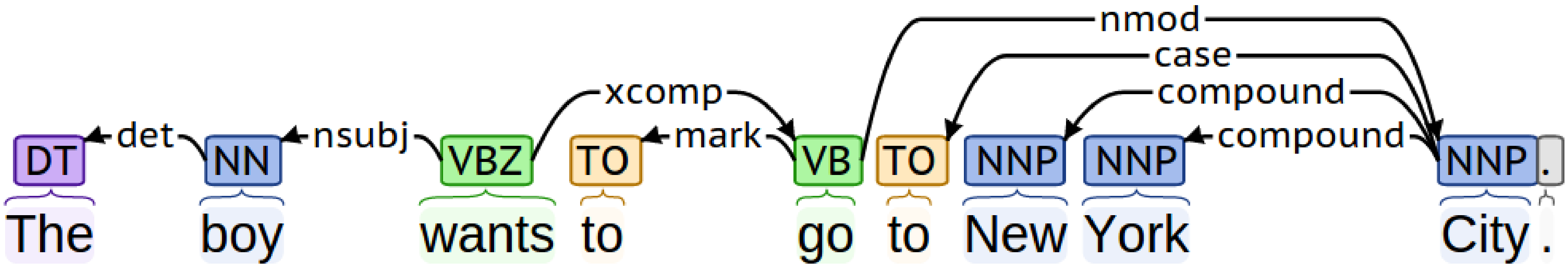
Syntax

Dependency parse tree:



Syntax

Dependency parse tree:

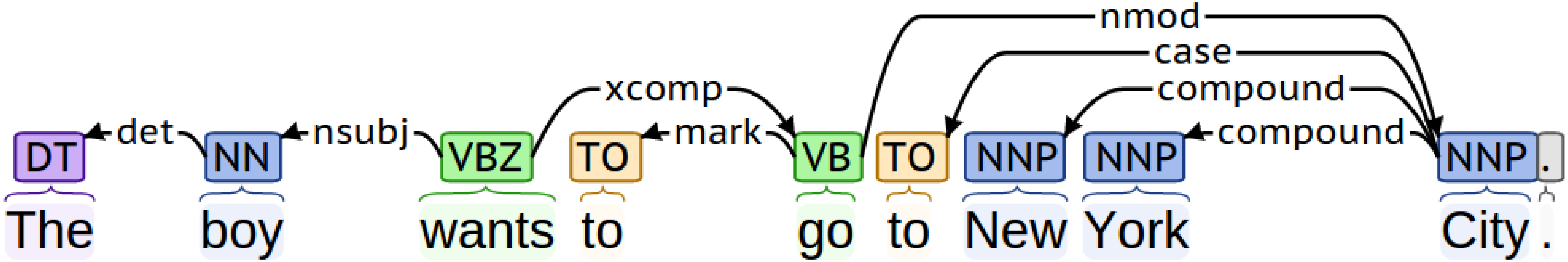


Parts of speech:

- NN: common noun
- NNP: proper noun
- VBZ: verb, 3rd person singular

Syntax

Dependency parse tree:



Parts of speech:

- NN: common noun
- NNP: proper noun
- VBZ: verb, 3rd person singular

Dependency relations:

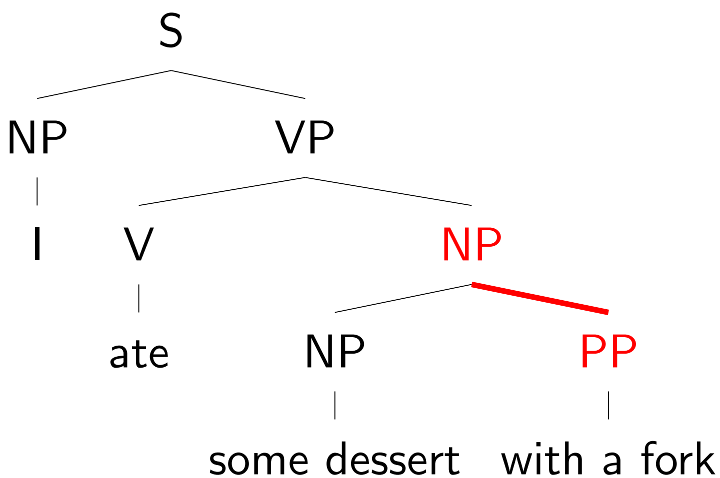
- nsubj: subject (nominal)
- nmod: modifier (nominal)

Prepositional attachment ambiguity

I ate some dessert with a fork.

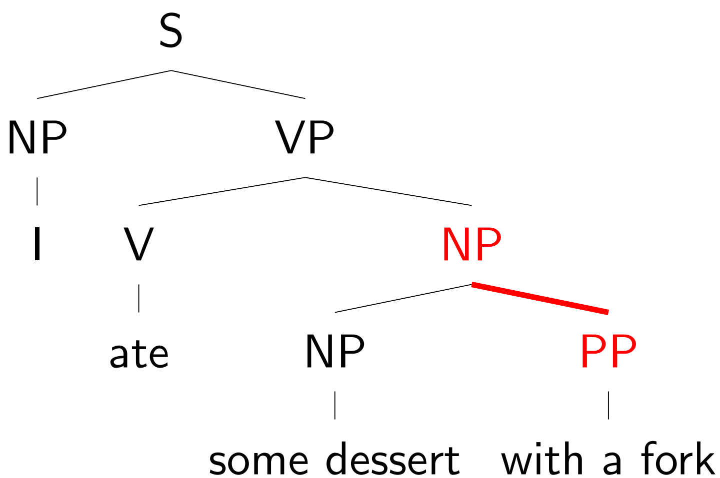
Prepositional attachment ambiguity

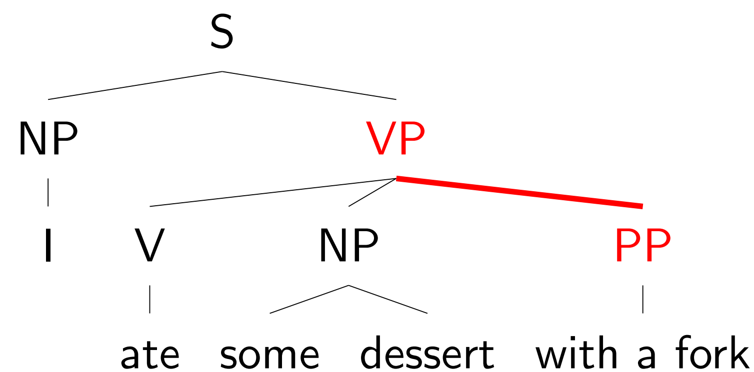
I ate some dessert with a fork.



Prepositional attachment ambiguity

I ate some dessert with a fork.





Prepositional attachment ambiguity

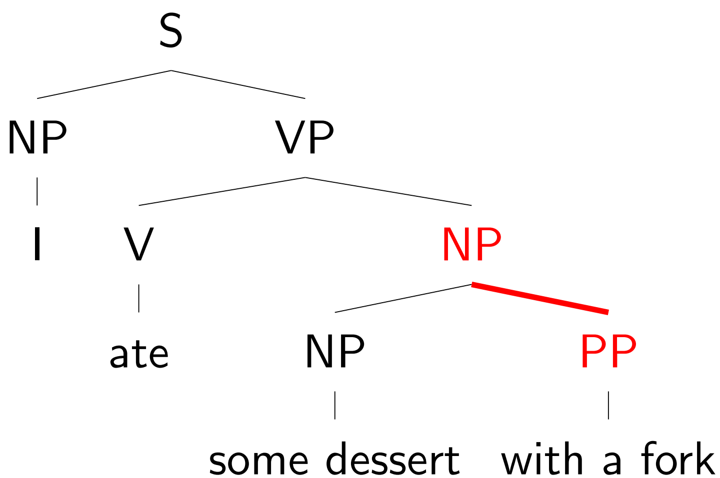
I ate some dessert with a fork.

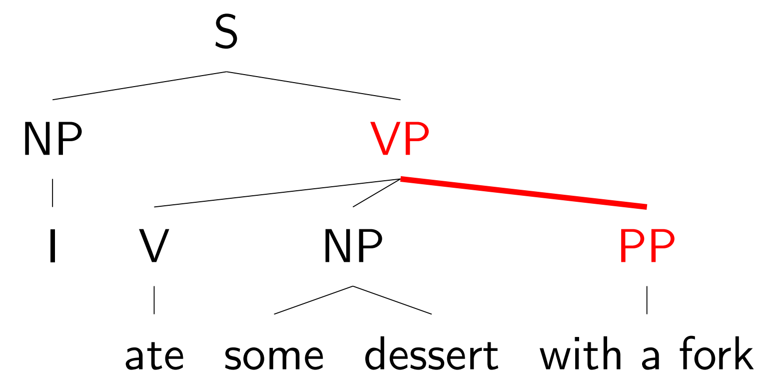
```
S
NP VP
I V NP
ate NP PP
some dessert with a fork
```

```
S
NP VP
I V NP PP
ate some dessert with a fork
```

Prepositional attachment ambiguity

I ate some dessert with a fork.





Both are grammatical; is syntax enough to disambiguate?

Semantics

Meaning

Semantics

Meaning

This is the tree of life.

Lexical semantics: what words mean

Compositional semantics: how meaning gets combined

What's a word?

light

What's a word?

light

Multi-word expressions: meaning unit beyond a word

light bulb

What's a word?

light

Multi-word expressions: meaning unit beyond a word

light bulb

Morphology: meaning unit within a word

light *lighten* *lightening* *relight*

What's a word?

light

Multi-word expressions: meaning unit beyond a word

light bulb

Morphology: meaning unit within a word

light *lighten* *lightening* *relight*

Polysemy: one word has multiple meanings (**word senses**)

- *The* ***light*** *was filtered through a soft glass window.*
- *He stepped into the* ***light****.*
- *This lamp* ***lights*** *up the room.*
- *The load is not* ***light****.*

Synonymy

Words:

confusing

Synonymy

Words:

confusing *unclear* *perplexing* *mystifying*

Synonymy

Words:

confusing   *unclear*   *perplexing*   *mystifying*

Sentences:

I have fond memories of my childhood.

I reflect on my childhood with a certain fondness.

I enjoy thinking back to when I was a kid.

Synonymy

Words:

confusing *unclear* *perplexing* *mystifying*

Sentences:

I have fond memories of my childhood.

I reflect on my childhood with a certain fondness.

I enjoy thinking back to when I was a kid.

Beware: no true equivalence due to subtle diferences in meaning; think **distance metric**

Synonymy

Words:

confusing *unclear* *perplexing* *mystifying*

Sentences:

I have fond memories of my childhood.

I reflect on my childhood with a certain fondness.

I enjoy thinking back to when I was a kid.

Beware: no true equivalence due to subtle diferences in meaning; think **distance metric**

But there's more to meaning than similarity...

Other lexical relations

Hyponymy (is-a):

a **cat** is a **mammal**

Other lexical relations

Hyponymy (is-a):

a **cat** is a **mammal**

Meronomy (has-a):

a **cat** has a **tail**

Other lexical relations

Hyponymy (is-a):

a **cat** is a **mammal**

Meronomy (has-a):

a **cat** has a **tail**

Useful for **entailment**:

I am giving an NLP tutorial at ICML.

$\Rightarrow$

I am speaking at a conference.

Compositional semantics

Two ideas: **model theory** and **compositionality**

Model theory: sentences refer to the world

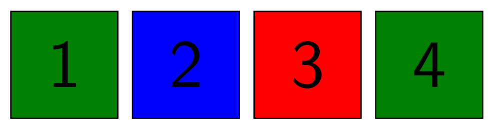
Block 2 is blue.

Compositional semantics

Two ideas: **model theory** and **compositionality**

Model theory: sentences refer to the world

Block 2 is blue.

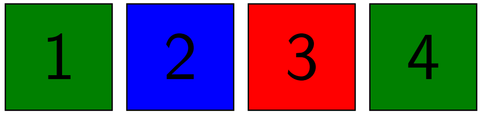


Compositional semantics

Two ideas: **model theory** and **compositionality**

Model theory: sentences refer to the world

Block 2 is blue.



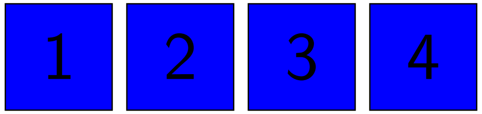
Compositionality: meaning of whole is meaning of parts

The [block left of the red block] is blue.

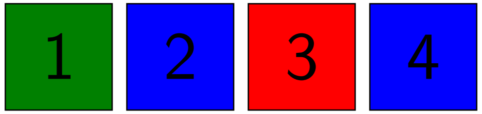
Quantifiers

Universal and existential quantification:

Every *block is blue.*



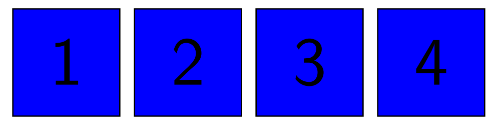
Some *block is blue.*



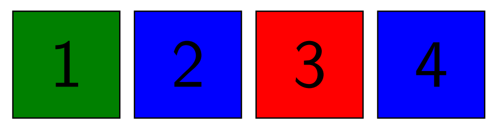
Quantifiers

Universal and existential quantification:

Every *block is blue.*



Some *block is blue.*



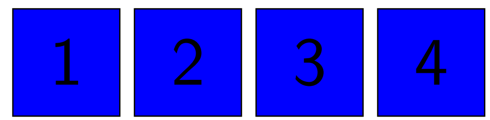
Quantifier scope ambiguity:

Every *non-blue block is next to* **some** *blue block.*

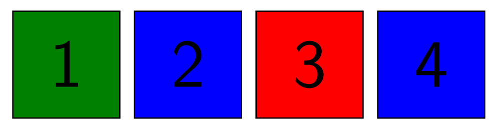
Quantifiers

Universal and existential quantification:

Every *block is blue.*



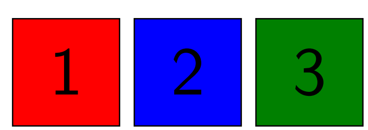
Some *block is blue.*



Quantifier scope ambiguity:

Every *non-blue block is next to* **some** *blue block.*

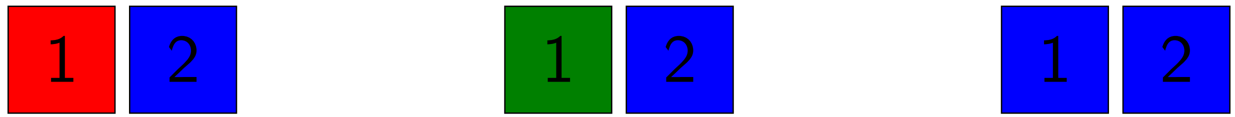
Every *non-blue block is next to* **some** *blue block.*



Multiple possible worlds

Modality:

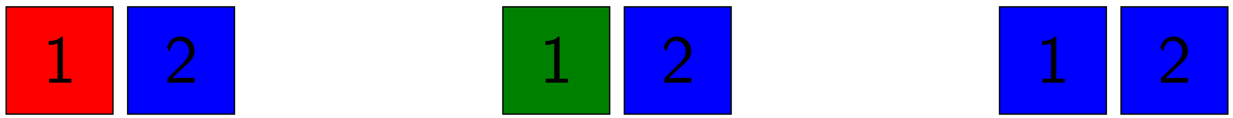
Block 2 **must** *be blue. Block 1* **can** *be red.*



Multiple possible worlds

Modality:

Block 2 **must** *be blue. Block 1* **can** *be red.*



Beliefs:

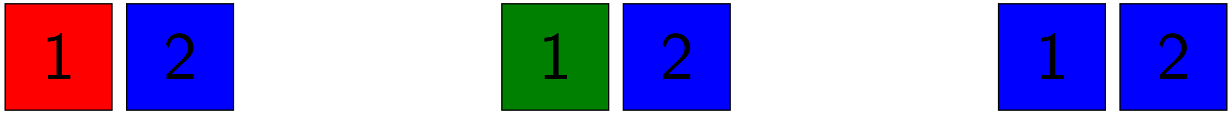
Clark Kent

Superman

Multiple possible worlds

Modality:

Block 2 **must** *be blue. Block 1* **can** *be red.*



Beliefs:

Clark Kent

Superman

Lois **believes** *Superman is a hero.*

$\neq$

Lois **believes** *Clark Kent is a hero.*

Anaphora

The **dog** *chased the* **cat***, which ran up a tree.* **It** *waited at the top.*

Anaphora

The **dog** *chased the* **cat***, which ran up a tree.* **It** *waited at the top.*

The **dog** *chased the* **cat***, which ran up a tree.* **It** *waited at the bottom.*

Anaphora

The **dog** *chased the* **cat***, which ran up a tree.* **It** *waited at the top.*

The **dog** *chased the* **cat***, which ran up a tree.* **It** *waited at the bottom.*

"The Winograd Schema Challenge" (Levesque, 2011)

- Easy for humans, can't use surface-level patterns

Pragmatics

Conversational implicature: new material **suggested** (not logically implied) by sentence

- *A: What on earth has happened to the roast beef?*

  B: The dog is looking very happy.

Pragmatics

Conversational implicature: new material **suggested** (not logically implied) by sentence

- *A: What on earth has happened to the roast beef?*

  B: The dog is looking very happy.

- Implicature: *The dog at the roast beef.*

Pragmatics

Conversational implicature: new material **suggested** (not logically implied) by sentence

- *A: What on earth has happened to the roast beef?*

  B: The dog is looking very happy.

- Implicature: *The dog at the roast beef.*

Presupposition: background **assumption** independent of truth of sentence

- *I have stopped eating meat.*

Pragmatics

Conversational implicature: new material **suggested** (not logically implied) by sentence

- *A: What on earth has happened to the roast beef?*

  B: The dog is looking very happy.

- Implicature: *The dog at the roast beef.*

Presupposition: background **assumption** independent of truth of sentence

- *I have stopped eating meat.*
- Presupposition: *I once was eating meat.*

Pragmatics

Semantics: what does it mean **literally**?

Pragmatics: what is the speaker really conveying?

Pragmatics

Semantics: what does it mean **literally**?

Pragmatics: what is the speaker really conveying?

- Underlying principle (Grice, 1975): language is cooperative game between speaker and listener

- Implicatures and presuppositions depend on people and context and involves soft inference (machine learning opportunities here!)

Vagueness, ambiguity, uncertainty

Vagueness: does not specify full information

I had a ***late*** *lunch.*

Vagueness, ambiguity, uncertainty

Vagueness: does not specify full information

I had a ***late*** *lunch.*

Ambiguity: more than one possible (precise) interpretations

One morning I shot an elephant ***in*** *my pajamas.*

Vagueness, ambiguity, uncertainty

Vagueness: does not specify full information

I had a **late** *lunch.*

Ambiguity: more than one possible (precise) interpretations

One morning I shot an elephant **in** *my pajamas.*
How he got in my pajamas, I don't know. — Groucho Marx

Vagueness, ambiguity, uncertainty

Vagueness: does not specify full information

I had a ***late*** *lunch.*

Ambiguity: more than one possible (precise) interpretations

One morning I shot an elephant ***in*** *my pajamas.*
How he got in my pajamas, I don't know. — Groucho Marx

Uncertainty: due to an imperfect statistical model

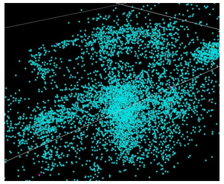
The witness was being ***contumacious****.*

Summary so far

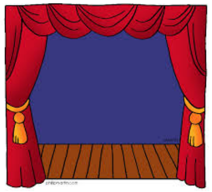
- Analyses: syntax, semantics, pragmatics
- Lexical semantics: synonymy, hyponymy/meronymy
- Compositional semantics: model theory, compositionality
- Challenges: polysemy, vagueness, ambiguity, uncertainty

Outline

Properties of language



Distributional semantics



Frame semantics

Model-theoretic semantics

Reflections

Distributional semantics: warmup

The new design has _____ *lines.*

Let's try to keep the kitchen _____.

I forgot to _____ *out the cabinet.*

Distributional semantics: warmup

The new design has _____ *lines.*

Let's try to keep the kitchen _____.

I forgot to _____ *out the cabinet.*

What does _____ mean?

Distributional semantics

The new design has _____ *lines.*

Observation: **context** can tell us a lot about word meaning

Context: local window around a word occurrence (for now)

Distributional semantics

The new design has _____ lines.

Observation: **context** can tell us a lot about word meaning

Context: local window around a word occurrence (for now)

Roots in linguistics:

- **Distributional hypothesis**: Semantically similar words occur in similar contexts [Harris, 1954]
- "You shall know a word by the company it keeps." [Firth, 1957]

Distributional semantics

The new design has _____ lines.

Observation: **context** can tell us a lot about word meaning

Context: local window around a word occurrence (for now)

Roots in linguistics:

- **Distributional hypothesis**: Semantically similar words occur in similar contexts [Harris, 1954]
- "You shall know a word by the company it keeps." [Firth, 1957]
- Contrast: Chomsky's generative grammar (lots of hidden prior structure, no data)

Distributional semantics

The new design has _____ lines.

Observation: **context** can tell us a lot about word meaning

Context: local window around a word occurrence (for now)

Roots in linguistics:

- **Distributional hypothesis**: Semantically similar words occur in similar contexts [Harris, 1954]
- "You shall know a word by the company it keeps." [Firth, 1957]
- Contrast: Chomsky's generative grammar (lots of hidden prior structure, no data)

Upshot: **data-driven**!

General recipe

1. Form a **word-context matrix** of counts (data)

context c

word w $\quad N$

General recipe

1. Form a **word-context matrix** of counts (data)

context c

word w N

2. Perform **dimensionality reduction** (generalize)

word w Θ $\Rightarrow$ word vectors $\theta_w \in \mathbb{R}^d$

[Deerwater/Dumais/Furnas/Landauer/Harshman, 1990]

Latent semantic analysis

Data:

Doc1: *Cats have tails.*

Doc2: *Dogs have tails.*

[Deerwater/Dumais/Furnas/Landauer/Harshman, 1990]

Latent semantic analysis

Data:

Doc1: *Cats have tails.*

Doc2: *Dogs have tails.*

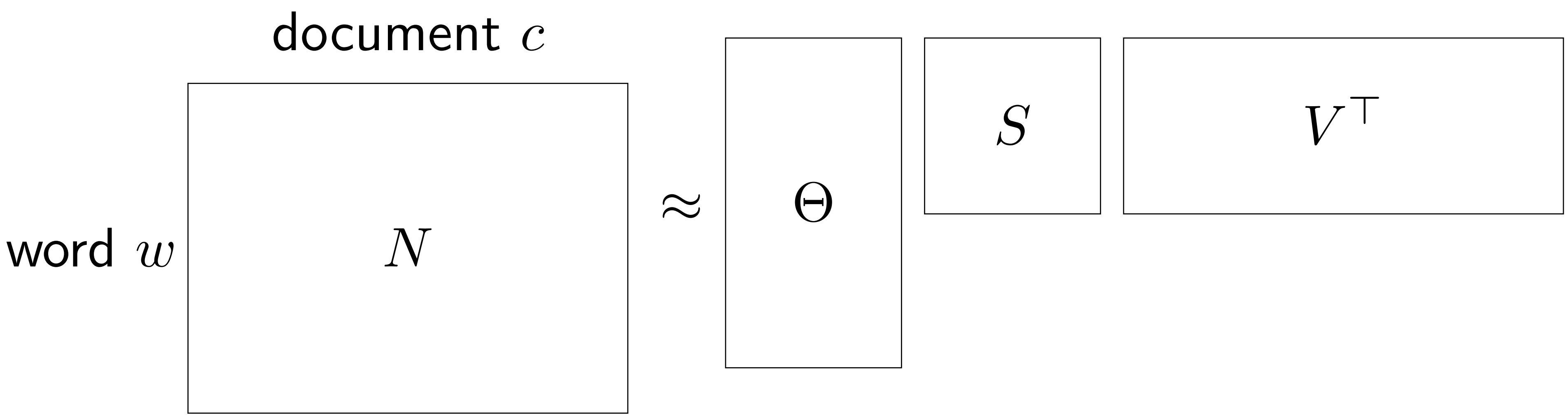
Matrix: contexts = **documents** that word appear in

	Doc1	Doc2
cats	1	0
dogs	0	1
have	1	1
tails	1	1

[Deerwater/Dumais/Furnas/Landauer/Harshman, 1990]

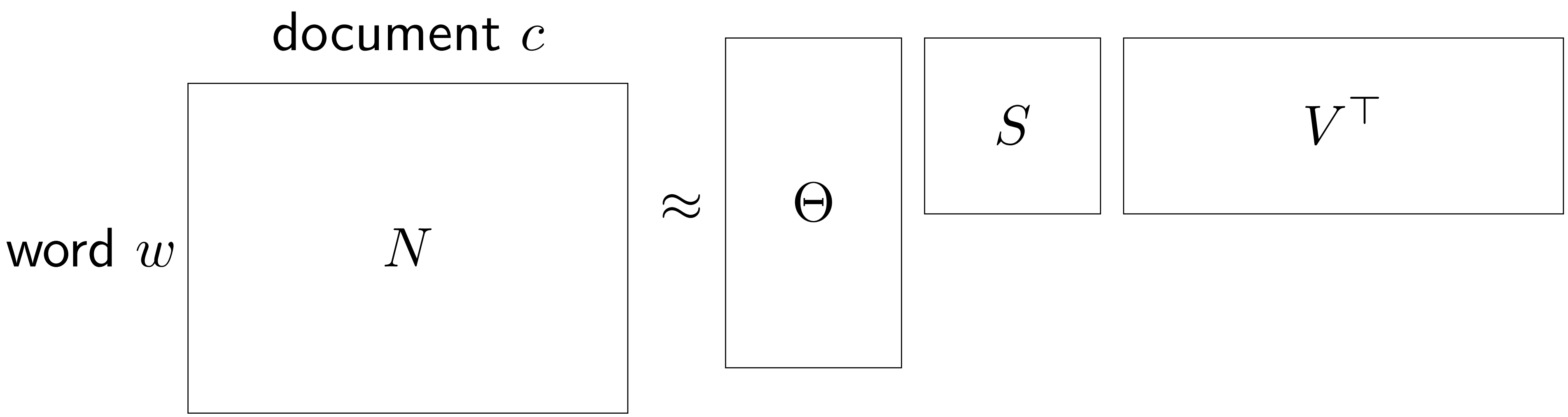
Latent semantic analysis

Dimensionality reduction: **SVD**



Latent semantic analysis

Dimensionality reduction: **SVD**



- Used for information retrieval

- Match query to documents in latent space rather than on keywords

Unsupervised part-of-speech induction

Data:

Cats have tails.

Dogs have tails.

[Schuetze, 1995]

Unsupervised part-of-speech induction

Data:

Cats have tails.

Dogs have tails.

Matrix: contexts = words on left, words on right

	cats_L	dogs_L	tails_R	have_L	have_R
cats	0	0	0	0	1
dogs	0	0	0	0	1
have	1	1	1	0	0
tails	0	0	0	1	0

Dimensionality reduction: **SVD**

Effect of context

Suppose *Barack Obama* always appear together (a **collocation**).

Effect of context

Suppose *Barack Obama* always appear together (a **collocation**).

Global context (document):

- same context $\Rightarrow \theta_{\text{Barack}}$ close to θ_{Obama}
- more "semantic"

Effect of context

Suppose *Barack Obama* always appear together (a **collocation**).

Global context (document):

- same context $\Rightarrow \theta_{\text{Barack}}$ close to θ_{Obama}
- more "semantic"

Local context (neighbors):

- different context $\Rightarrow \theta_{\text{Barack}}$ far from θ_{Obama}
- more "syntactic"

[Mikolov/Sutskever/Chen/Corrado/Dean, 2013 (word2vec)]

Skip-gram model with negative sampling

Data:

Cats and dogs have tails.

[Mikolov/Sutskever/Chen/Corrado/Dean, 2013 (word2vec)]

Skip-gram model with negative sampling

Data:

Cats and dogs have tails.

Form matrix: contexts = words in a window

	cats	and	dogs	have	tails
cats	0	1	1	0	0
and	1	0	1	1	0
dogs	1	1	0	1	1
have	0	1	1	0	1
tails	0	0	1	1	0

Skip-gram model with negative sampling

Dimensionality reduction: **logistic regression with SGD**

[Mikolov/Sutskever/Chen/Corrado/Dean, 2013 (word2vec)]

Skip-gram model with negative sampling

Dimensionality reduction: **logistic regression with SGD**

Model: predict good (w, c) using logistic regression

$$p_\theta(g = 1 \mid w, c) = (1 + \exp(\theta_w \cdot \beta_c))^{-1}$$

[Mikolov/Sutskever/Chen/Corrado/Dean, 2013 (word2vec)]

Skip-gram model with negative sampling

Dimensionality reduction: **logistic regression with SGD**

Model: predict good (w, c) using logistic regression

$$p_\theta(g = 1 \mid w, c) = (1 + \exp(\theta_w \cdot \beta_c))^{-1}$$

Positives: (w, c) from data

Negatives: (w, c') for irrelevant c' (k times more)

+(cats, AI) −(cats, linguistics) −(cats, statistics)

Skip-gram model with negative sampling

Data distribution:

$$\hat{p}(w, c) \propto N(w, c)$$

Objective:

$$\max_{\theta, \beta} \sum_{w,c} \hat{p}(w, c) \log p(g = 1 \mid w, c) +$$

$$k \sum_{w,c'} \hat{p}(w)\hat{p}(c') \log p(g = 0 \mid w, c')$$

Skip-gram model with negative sampling

Data distribution:

$$\hat{p}(w,c) \propto N(w,c)$$

Objective:

$$\max_{\theta,\beta} \sum_{w,c} \hat{p}(w,c) \log p(g=1 \mid w,c) +$$

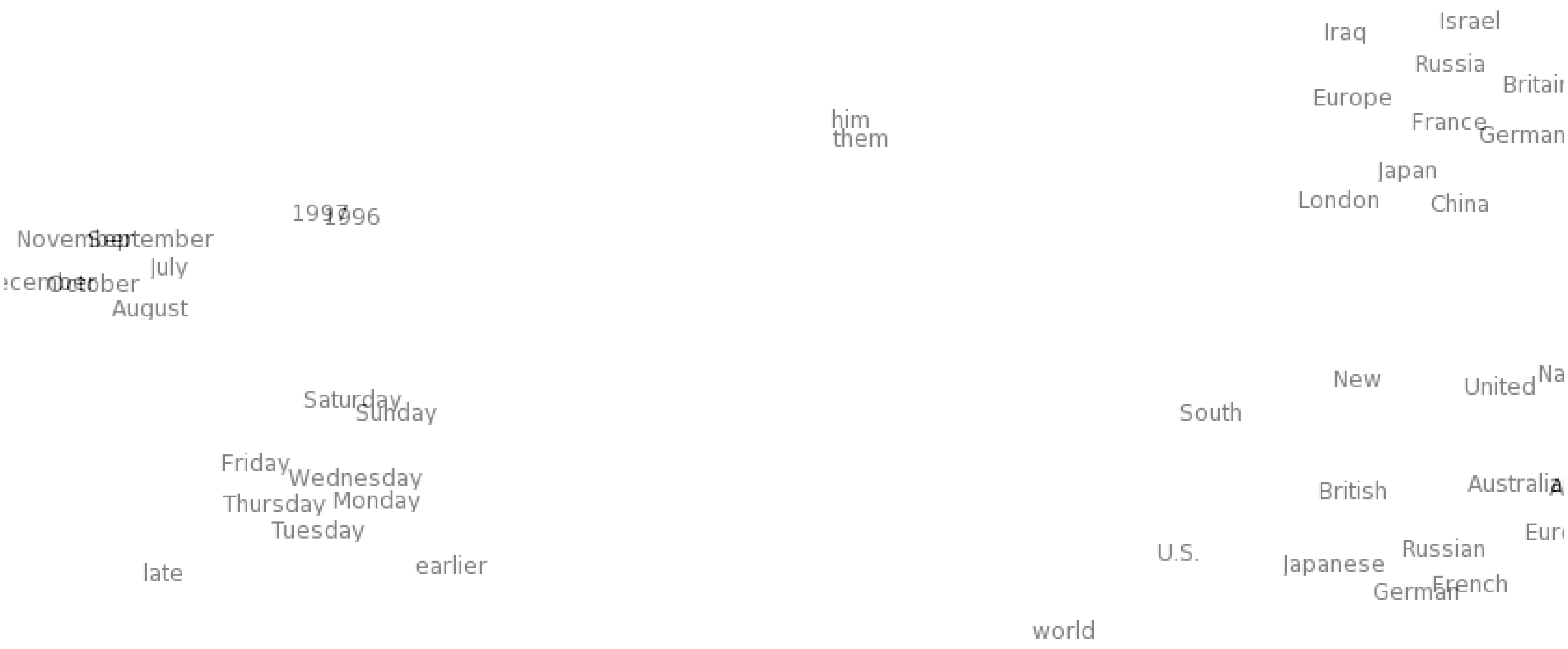
$$k \sum_{w,c'} \hat{p}(w)\hat{p}(c') \log p(g=0 \mid w,c')$$

If no dimensionality reduction:

$$\theta_w \cdot \beta_c = \log\left(\frac{\hat{p}(w,c)}{\hat{p}(w)\hat{p}(c)}\right) = \mathsf{PMI}(w,c)$$

2D visualization of word vectors

2D visualization of word vectors



Nearest neighbors

cherish

(words)

adore
love
admire
embrace
rejoice

(contexts)

cherish
both
love
pride
thy

quasi-synonyms

Nearest neighbors

cherish	**tiger**
(words)	(words)
adore	*leopard*
love	*dhole*
admire	*warthog*
embrace	*rhinoceros*
rejoice	*lion*
(contexts)	(contexts)
cherish	*tiger*
both	*leopard*
love	*panthera*
pride	*woods*
thy	*puma*
quasi-synonyms	co-hyponyms

Nearest neighbors

cherish	**tiger**	**good**
(words)	(words)	(words)
adore	*leopard*	*bad*
love	*dhole*	*decent*
admire	*warthog*	*excellent*
embrace	*rhinoceros*	*lousy*
rejoice	*lion*	*nice*
(contexts)	(contexts)	(contexts)
cherish	*tiger*	*faith*
both	*leopard*	*natured*
love	*panthera*	*luck*
pride	*woods*	*riddance*
thy	*puma*	*both*
quasi-synonyms	co-hyponyms	includes antonyms

Nearest neighbors

cherish	**tiger**	**good**
(words)	(words)	(words)
adore	*leopard*	*bad*
love	*dhole*	*decent*
admire	*warthog*	*excellent*
embrace	*rhinoceros*	*lousy*
rejoice	*lion*	*nice*
(contexts)	(contexts)	(contexts)
cherish	*tiger*	*faith*
both	*leopard*	*natured*
love	*panthera*	*luck*
pride	*woods*	*riddance*
thy	*puma*	*both*
quasi-synonyms	co-hyponyms	includes antonyms

Many things under **semantic similarity**!

[Mikolov/Yih/Zweig, 2013; Levy/Goldberg, 2014]

Analogies

Differences in context vectors capture relations:

$$\theta_{\text{king}} - \theta_{\text{man}} \approx \theta_{\text{queen}} - \theta_{\text{woman}} \text{ (gender)}$$

Analogies

Differences in context vectors capture relations:

$$\theta_{\text{king}} - \theta_{\text{man}} \approx \theta_{\text{queen}} - \theta_{\text{woman}} \text{ (gender)}$$

$$\theta_{\text{france}} - \theta_{\text{french}} \approx \theta_{\text{mexico}} - \theta_{\text{spanish}} \text{ (language)}$$

$$\theta_{\text{car}} - \theta_{\text{cars}} \approx \theta_{\text{apple}} - \theta_{\text{apples}} \text{ (plural)}$$

[Mikolov/Yih/Zweig, 2013; Levy/Goldberg, 2014]

Analogies

Differences in context vectors capture relations:

$$\theta_{\text{king}} - \theta_{\text{man}} \approx \theta_{\text{queen}} - \theta_{\text{woman}} \text{ (gender)}$$

$$\theta_{\text{france}} - \theta_{\text{french}} \approx \theta_{\text{mexico}} - \theta_{\text{spanish}} \text{ (language)}$$

$$\theta_{\text{car}} - \theta_{\text{cars}} \approx \theta_{\text{apple}} - \theta_{\text{apples}} \text{ (plural)}$$

Intuition:

$$\underbrace{\theta_{\text{king}}}_{\text{[crown,he]}} - \underbrace{\theta_{\text{man}}}_{\text{[he]}} \approx \underbrace{\theta_{\text{queen}}}_{\text{[crown,she]}} - \underbrace{\theta_{\text{woman}}}_{\text{[she]}}$$

Don't need dimensionality reduction for this to work!

Other models

Multinomial models:

- HMM word clustering [Brown et al., 1992]
- Latent Dirichlet Allocation [Blei et al., 2003]

Other models

Multinomial models:

- HMM word clustering [Brown et al., 1992]
- Latent Dirichlet Allocation [Blei et al., 2003]

Neural network models:

- Multi-tasking neural network [Weston/Collobert, 2008]

Other models

Multinomial models:

- HMM word clustering [Brown et al., 1992]
- Latent Dirichlet Allocation [Blei et al., 2003]

Neural network models:

- Multi-tasking neural network [Weston/Collobert, 2008]

Recurrent/recursive models: (can embed phrases too)

- Neural language models [Bengio et al., 2003]
- Neural machine translation [Sutskever/Vinyals/Le, 2014, Cho/Merrienboer/Bahdanau/Bengio, 2014]
- Recursive neural networks [Socher/Lin/Ng/Manning, 2011]

Hearst patterns for hyponyms

The bow lute, such as the **Bambara ndang**, *is plucked...*

[Hearst, 1992]

Hearst patterns for hyponyms

The bow lute, such as the **Bambara ndang**, *is plucked...*

$\Downarrow$

Bambara ndang hyponym-of *bow lute*

Hearst patterns for hyponyms

The bow lute, such as the **Bambara ndang**, *is plucked...*

$$\Downarrow$$

Bambara ndang hyponym-of *bow lute*

General rules:

C such as $X \Rightarrow [X$ hyponym-of $C]$

X and other $C \Rightarrow [X$ hyponym-of $C]$

C including $X \Rightarrow [X$ hyponym-of $C]$

Hearst patterns for hyponyms

The bow lute, such as the **Bambara ndang***, is plucked...*

$$\Downarrow$$

Bambara ndang hyponym-of *bow lute*

General rules:

C such as $X \Rightarrow [X$ hyponym-of $C]$
X and other $C \Rightarrow [X$ hyponym-of $C]$
C including $X \Rightarrow [X$ hyponym-of $C]$

- Thrust: apply simple patterns to large web corpora
- Again, context reveals information about semantics

[Hearst, 1992]

Hearst patterns for hyponyms

The bow lute, such as the **Bambara ndang**, *is plucked...*

$$\Downarrow$$

Bambara ndang hyponym-of *bow lute*

General rules:

C such as $X \Rightarrow [X$ hyponym-of $C]$

X and other $C \Rightarrow [X$ hyponym-of $C]$

C including $X \Rightarrow [X$ hyponym-of $C]$

- Thrust: apply simple patterns to large web corpora
- Again, context reveals information about semantics
- Can learn patterns via bootstrapping (semi-supervised learning)

Summary so far

- Premise: semantics = context of word/phrase

Summary so far

- Premise: semantics = context of word/phrase
- Recipe: form word-context matrix + dimensionality reduction

context c

word w $\boxed{N}$

Summary so far

- Premise: semantics = context of word/phrase
- Recipe: form word-context matrix + dimensionality reduction

context c

word w $\boxed{\quad N \quad}$

Pros:

- Simple models, leverage tons of raw text
- Context captures nuanced information about usage
- Word vectors useful in downstream tasks

Food for thought

What **contexts**?

- No such thing as pure unsupervised learning, representation depends on choice of context (e.g., global/local/task-specific)
- Language is not just text in isolation, context should include world/environment

Food for thought

What **contexts**?

- No such thing as pure unsupervised learning, representation depends on choice of context (e.g., global/local/task-specific)
- Language is not just text in isolation, context should include world/environment

What **models**?

- Currently very fine-grained (non-parametric idiot savants)
- Language is about speaker's **intention**, not words

Food for thought

What **contexts**?

- No such thing as pure unsupervised learning, representation depends on choice of context (e.g., global/local/task-specific)
- Language is not just text in isolation, context should include world/environment

What **models**?

- Currently very fine-grained (non-parametric idiot savants)
- Language is about speaker's **intention**, not words

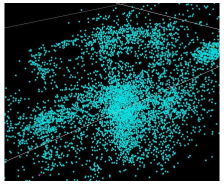
Examples to ponder:

Cynthia sold the bike for $200.

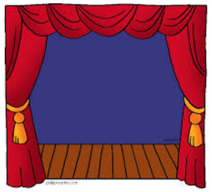
The bike sold for $200.

Outline

Properties of language



Distributional semantics



Frame semantics

Model-theoretic semantics

Reflections

Word meaning revisited

sold

Word meaning revisited

sold

Distributional semantics: all the contexts in which *sold* occurs

...was sold by... *...sold me that piece of...*

- Can find similar words/contexts and generalize (dimensionality reduction), but monolithic (no internal structure on word vectors)

Word meaning revisited

sold

Distributional semantics: all the contexts in which *sold* occurs

...was sold by... *...sold me that piece of...*

- Can find similar words/contexts and generalize (dimensionality reduction), but monolithic (no internal structure on word vectors)

Frame semantics: meaning given by a **frame**, a stereotypical situation

Commercial transaction

SELLER : ?
BUYER : ?
GOODS : ?
PRICE : ?

More subtle frames

I spent three hours on **land** *this afternoon.*

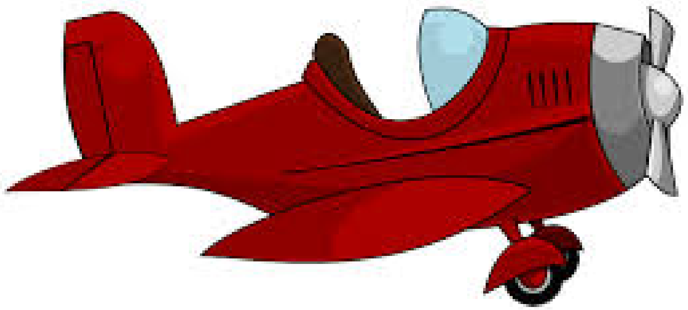
I spent three hours on the **ground** *this afternoon.*

[Fillmore, 1977]

More subtle frames

I spent three hours on ***land*** *this afternoon.*

I spent three hours on the ***ground*** *this afternoon.*



[Fillmore, 1977; Langacker, 1987]

Two properties of frames

Prototypical: don't need to handle all the cases

widow

Two properties of frames

Prototypical: don't need to handle all the cases

widow

- Frame: woman marries one man, man dies

[Fillmore, 1977; Langacker, 1987]

Two properties of frames

Prototypical: don't need to handle all the cases

widow

- Frame: woman marries one man, man dies
- What if a woman has 3 husbands, 2 of which died?

[Fillmore, 1977; Langacker, 1987]

Two properties of frames

Prototypical: don't need to handle all the cases

widow

- Frame: woman marries one man, man dies
- What if a woman has 3 husbands, 2 of which died?

Profiling: highlight one aspect

- *sell* is seller-centric, *buy* is buyer-centric

Cynthia sold the bike (to Bob).
Bob bought the bike (from Cynthia).

[Fillmore, 1977; Langacker, 1987]

Two properties of frames

Prototypical: don't need to handle all the cases

widow

- Frame: woman marries one man, man dies
- What if a woman has 3 husbands, 2 of which died?

Profiling: highlight one aspect

- *sell* is seller-centric, *buy* is buyer-centric

Cynthia sold the bike (to Bob).

Bob bought the bike (from Cynthia).

- *rob* highlights person, *steal* highlights goods

Cynthia robbed Bob (of the bike).

Cynthia stole the bike (from Bob).

A story

Joe went to a restaurant. Joe ordered a hamburger. When the hamburger came, it was burnt to a crisp. Joe stormed out without paying.

A story

Joe went to a restaurant. Joe ordered a hamburger. When the hamburger came, it was burnt to a crisp. Joe stormed out without paying.

- Need background knowledge to really **understand**
- Schank and Abelson developed notion of a **script** which captures this knowledge
- Same idea as frame, but tailored for event sequences

[Schank/Abelson, 1977]

A story

Joe went to a restaurant. Joe ordered a hamburger. When the hamburger came, it was burnt to a crisp. Joe stormed out without paying.

- Need background knowledge to really **understand**
- Schank and Abelson developed notion of a **script** which captures this knowledge
- Same idea as frame, but tailored for event sequences

Restaurant script (simplified):

Entering: S PTRANS S into restaurant, S PTRANS S to table

Ordering: S PTRANS$<$ menu to S, waiter PTRANS to table, S MTRANS$<$ 'I want food' to waiter

Eating: waiter PTRANS food to S, S INGEST food

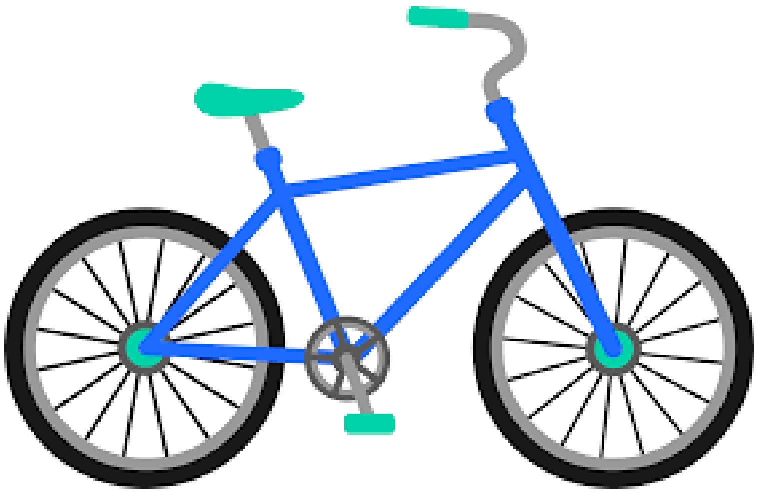
Exiting: waiter PTRANS to S, waiter ATRANS check to S, S ATRANS money to waiter, S PTRANS out of restaurant

Back to language

Cynthia sold the bike for \$200.

Back to language

Cynthia sold the bike for \$200.

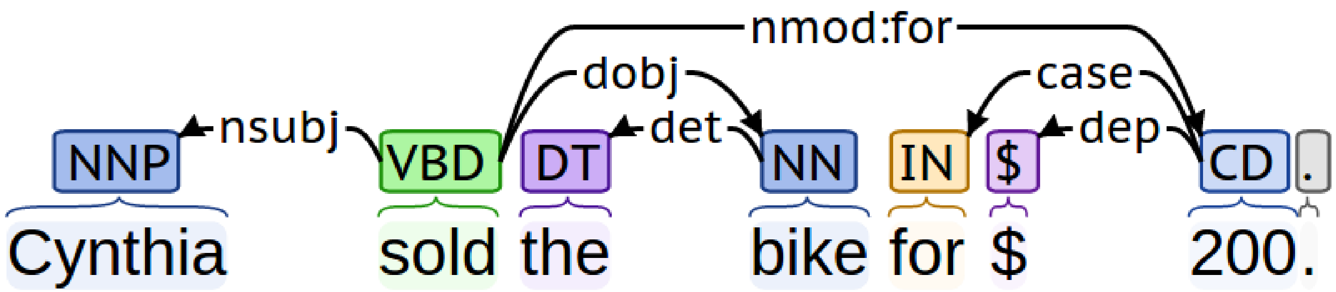


Commercial transaction

SELLER : *Cynthia*
GOODS : *the bike*
PRICE : *\$200*

From syntax to semantics

Dependency parse tree:



From syntax to semantics

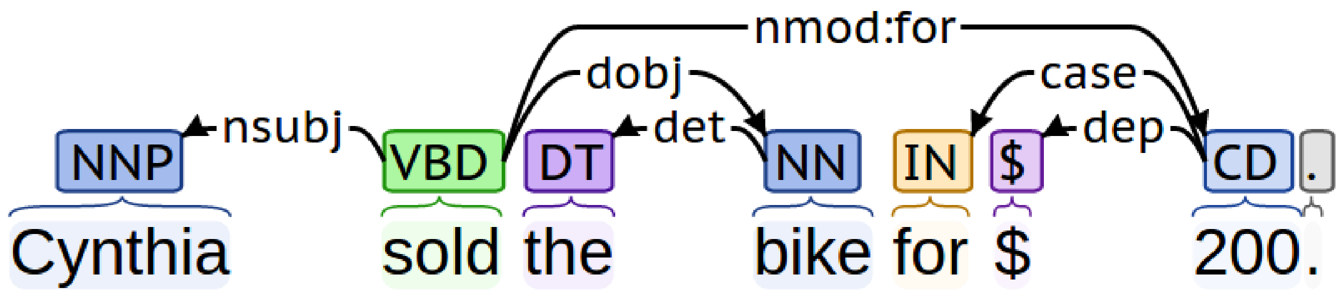
Extraction rules:

sold nsubj $X \Rightarrow$ SELLER:X

sold dobj $X \Rightarrow$ GOODS:X

sold nmod:for $X \Rightarrow$ PRICE:X

Dependency parse tree:



From syntax to semantics

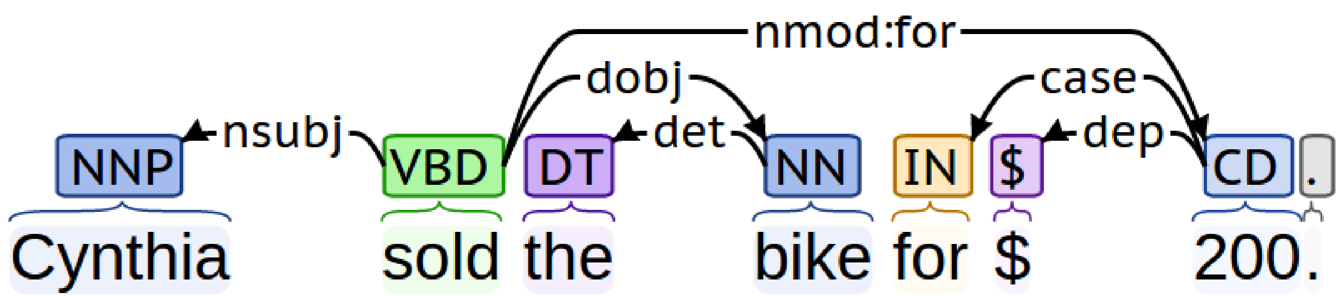
Extraction rules:

sold nsubj $X \Rightarrow$ SELLER:X

sold dobj $X \Rightarrow$ GOODS:X

sold nmod:for $X \Rightarrow$ PRICE:X

Dependency parse tree:



Commercial transaction

SELLER : *Cynthia*

GOODS : *the bike*

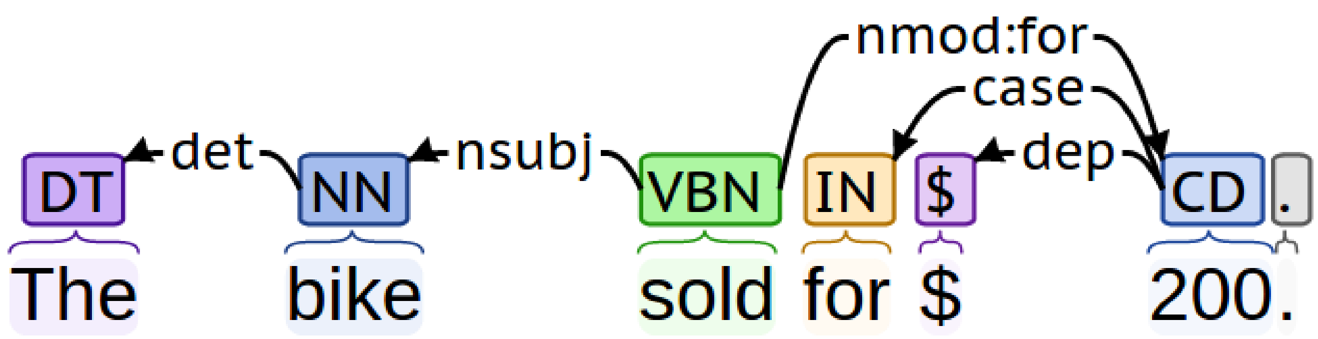
PRICE : *$200*

From syntax to semantics

Extraction rules:

> *sold* nsubj $X \Rightarrow$ SELLER:X
>
> *sold* dobj $X \Rightarrow$ GOODS:X
>
> *sold* nmod:for $X \Rightarrow$ GOODS:X

Dependency structure:



From syntax to semantics

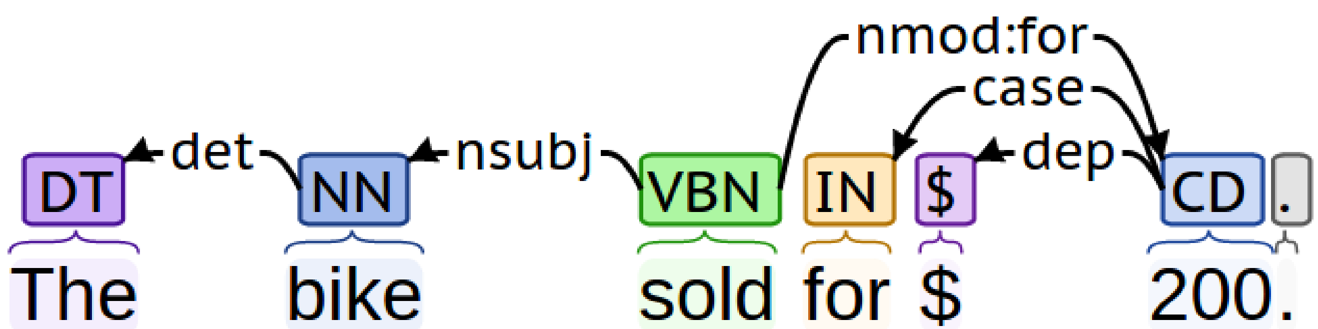
Extraction rules:

sold nsubj $X \Rightarrow$ SELLER:X

sold dobj $X \Rightarrow$ GOODS:X

sold nmod:for $X \Rightarrow$ GOODS:X

Dependency structure:



Commercial transaction

SELLER : *the bike*???

PRICE : *$200*

From syntax to semantics

Commercial transaction

SELLER : *Cynthia*
BUYER : *Bob*
GOODS : *the bike*
PRICE : *$200*

From syntax to semantics

Commercial transaction

SELLER : *Cynthia*
BUYER : *Bob*
GOODS : *the bike*
PRICE : *$200*

Many **syntactic alternations** with different arguments/verbs:

Cynthia sold the bike to Bob for $200.

The bike sold for $200.

From syntax to semantics

Commercial transaction

SELLER : *Cynthia*
BUYER : *Bob*
GOODS : *the bike*
PRICE : *$200*

Many **syntactic alternations** with different arguments/verbs:

Cynthia sold the bike to Bob for $200.

The bike sold for $200.

Bob bought the bike from Cynthia.

The bike was bought by Bob.

The bike was bought for $200.

The bike was bought for $200 by Bob.

From syntax to semantics

Commercial transaction

SELLER : *Cynthia*
BUYER : *Bob*
GOODS : *the bike*
PRICE : *$200*

Many **syntactic alternations** with different arguments/verbs:

Cynthia sold the bike to Bob for $200.

The bike sold for $200.

Bob bought the bike from Cynthia.

The bike was bought by Bob.

The bike was bought for $200.

The bike was bought for $200 by Bob.

Goal: **syntactic positions** $\Rightarrow$ **semantic roles**

Historical developments

Linguistics:

- Case grammar [Fillmore, 1968]: introduced idea of deep semantic roles (agents, themes, patients) which are tied to surface syntax (subjects, objects)

Historical developments

Linguistics:

- Case grammar [Fillmore, 1968]: introduced idea of deep semantic roles (agents, themes, patients) which are tied to surface syntax (subjects, objects)

AI / cognitive science:

- Frames [Minsky, 1975]: "a data-structure for representing a stereotyped situation, like...a child's birthday party"

Historical developments

Linguistics:

- Case grammar [Fillmore, 1968]: introduced idea of deep semantic roles (agents, themes, patients) which are tied to surface syntax (subjects, objects)

AI / cognitive science:

- Frames [Minsky, 1975]: "a data-structure for representing a stereotyped situation, like...a child's birthday party"
- Scripts [Schank & Abelson, 1977]: represent procedural knowledge (going to a restaurant)

Historical developments

Linguistics:

- Case grammar [Fillmore, 1968]: introduced idea of deep semantic roles (agents, themes, patients) which are tied to surface syntax (subjects, objects)

AI / cognitive science:

- Frames [Minsky, 1975]: "a data-structure for representing a stereotyped situation, like...a child's birthday party"
- Scripts [Schank & Abelson, 1977]: represent procedural knowledge (going to a restaurant)
- Frames [Fillmore, 1977]: coherent individuatable perception, memory, experience, action, or object

Historical developments

Linguistics:

- Case grammar [Fillmore, 1968]: introduced idea of deep semantic roles (agents, themes, patients) which are tied to surface syntax (subjects, objects)

AI / cognitive science:

- Frames [Minsky, 1975]: "a data-structure for representing a stereotyped situation, like...a child's birthday party"
- Scripts [Schank & Abelson, 1977]: represent procedural knowledge (going to a restaurant)
- Frames [Fillmore, 1977]: coherent individuatable perception, memory, experience, action, or object

NLP:

- FrameNet (1998) and PropBank (2002)

Concrete realization: FrameNet

FrameNet [Baker/Fillmore/Lowe, 1998]:

- Centered around frames, argument labels are shared across frames

Commerce (sell)

SELLER : ?
BUYER : ?
GOODS : ?
PRICE : ?

Concrete realization: FrameNet

FrameNet [Baker/Fillmore/Lowe, 1998]:

- Centered around frames, argument labels are shared across frames

Commerce (sell)
SELLER : ?
BUYER : ?
GOODS : ?
PRICE : ?

Lexical units that trigger frame:

auction.n, auction.v

retail.v, retailer.n

sale.n, sell.v, seller.n

vend.v, vendor.n

Concrete realization: FrameNet

FrameNet [Baker/Fillmore/Lowe, 1998]:

- Centered around frames, argument labels are shared across frames

Commerce (sell)

SELLER : ?
BUYER : ?
GOODS : ?
PRICE : ?

Lexical units that trigger frame:

auction.n, auction.v
retail.v, retailer.n
sale.n, sell.v, seller.n
vend.v, vendor.n

- Abstract away from the syntax by normalizing across different lexical units
- 4K predicates

Concrete realization: PropBank

PropBank [Palmer/Gildea/Kingsbury, 2002]:

- Centered around verbs and syntax, argument labels are verb-specific

sell.01

Concrete realization: PropBank

PropBank [Palmer/Gildea/Kingsbury, 2002]:

- Centered around verbs and syntax, argument labels are verb-specific

sell.01

Commerce (sell)

sell.01.A0 (seller)	: ?
sell.01.A1 (goods)	: ?
sell.01.A2 (buyer)	: ?
sell.01.A3 (price)	: ?
sell.01.A4 (beneficiary)	: ?

Concrete realization: PropBank

PropBank [Palmer/Gildea/Kingsbury, 2002]:

- Centered around verbs and syntax, argument labels are verb-specific

sell.01

Commerce (sell)

sell.01.A0 (seller)	: ?
sell.01.A1 (goods)	: ?
sell.01.A2 (buyer)	: ?
sell.01.A3 (price)	: ?
sell.01.A4 (beneficiary)	: ?

- Word senses tied to WordNet
- Created based on a corpus, so more popular

Semantic role labeling

Task:

Input: *Cynthia sold the bike to Bob for $200*

Semantic role labeling

Task:

Input:	*Cynthia*	*sold*	*the bike*	*to*	*Bob*	*for*	*\$200*
Output:	SELLER	**PREDICATE**	GOODS		BUYER		PRICE

Semantic role labeling

Task:

Input:	*Cynthia*	*sold*	*the bike*	*to*	*Bob*	*for*	*$200*
Output:	SELLER	**PREDICATE**	GOODS		BUYER		PRICE

Subtasks:

1. Frame identification (PREDICATE)
2. Argument identification (SELLER, GOODS, etc.)

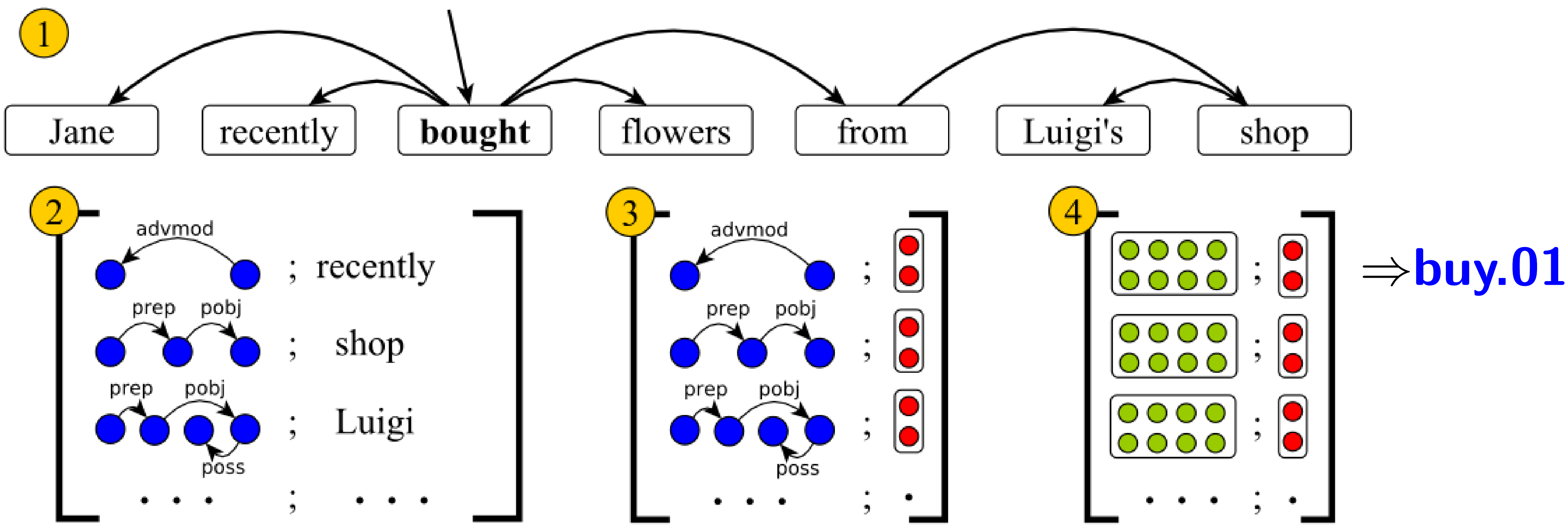
[Hermann/Das/Weston/Ganchev, 2014]

Frame identification

Jane recently bought flowers from Luigi's shop.

$\Rightarrow$**buy.01**

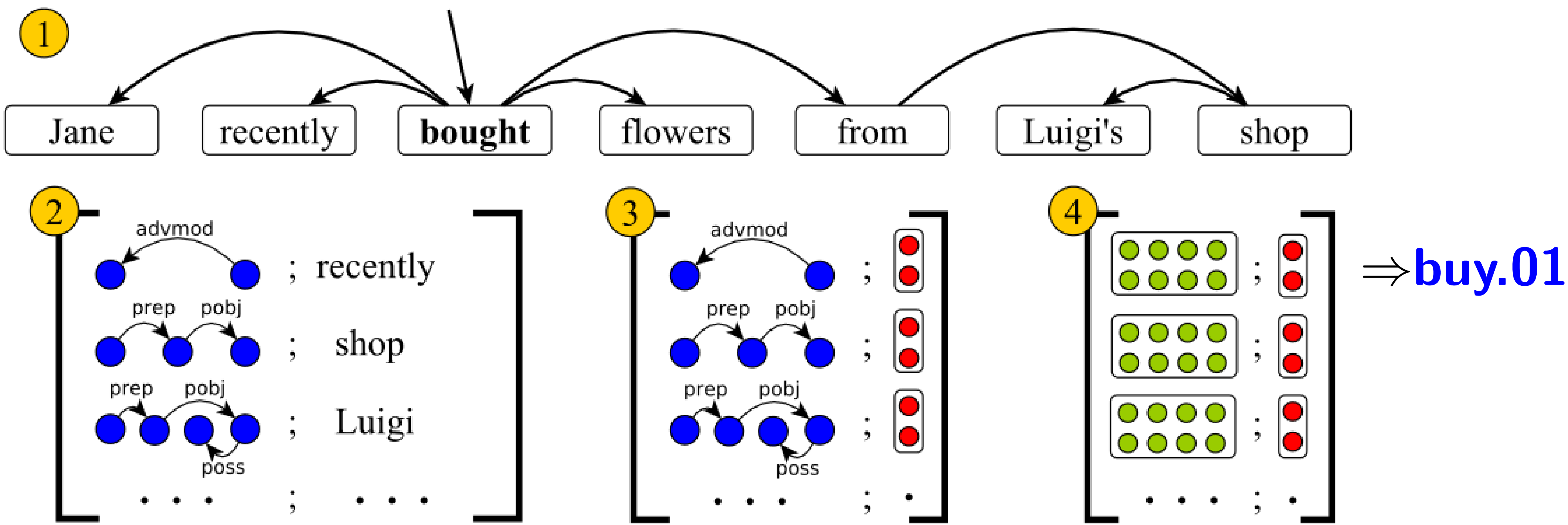
Frame identification



1. Construct dependency parse, choose predicate p (*bought*)

[Hermann/Das/Weston/Ganchev, 2014]

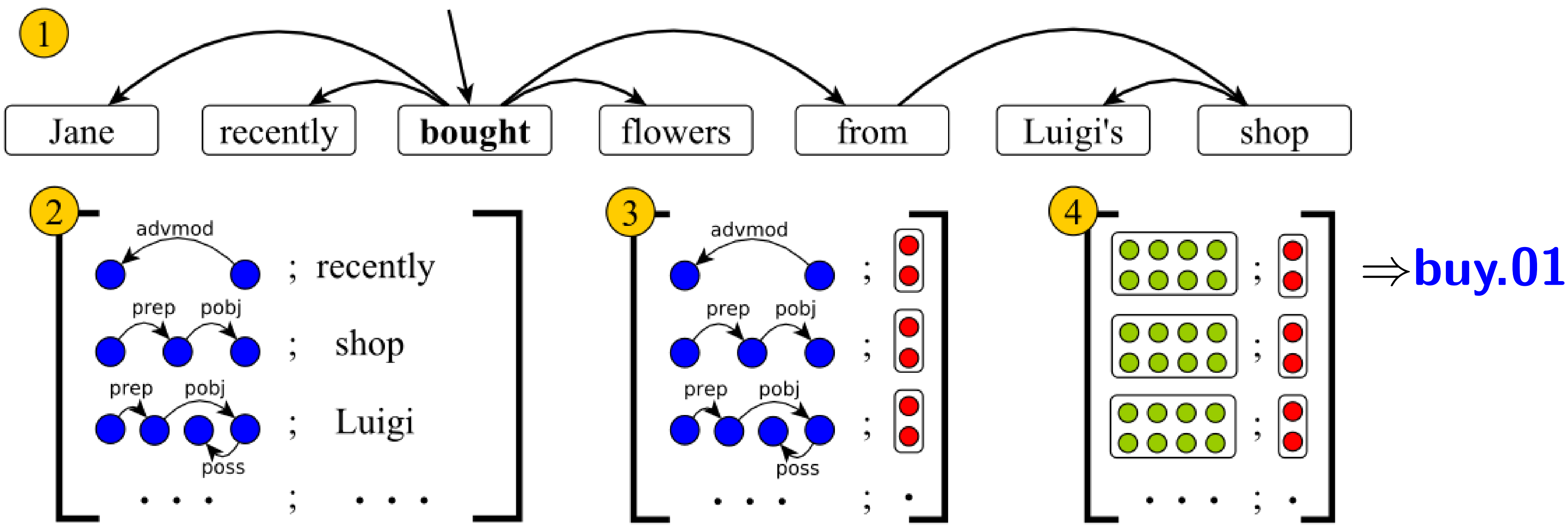
Frame identification



1. Construct dependency parse, choose predicate p (*bought*)
2. Extract paths from p to dependents a

[Hermann/Das/Weston/Ganchev, 2014]

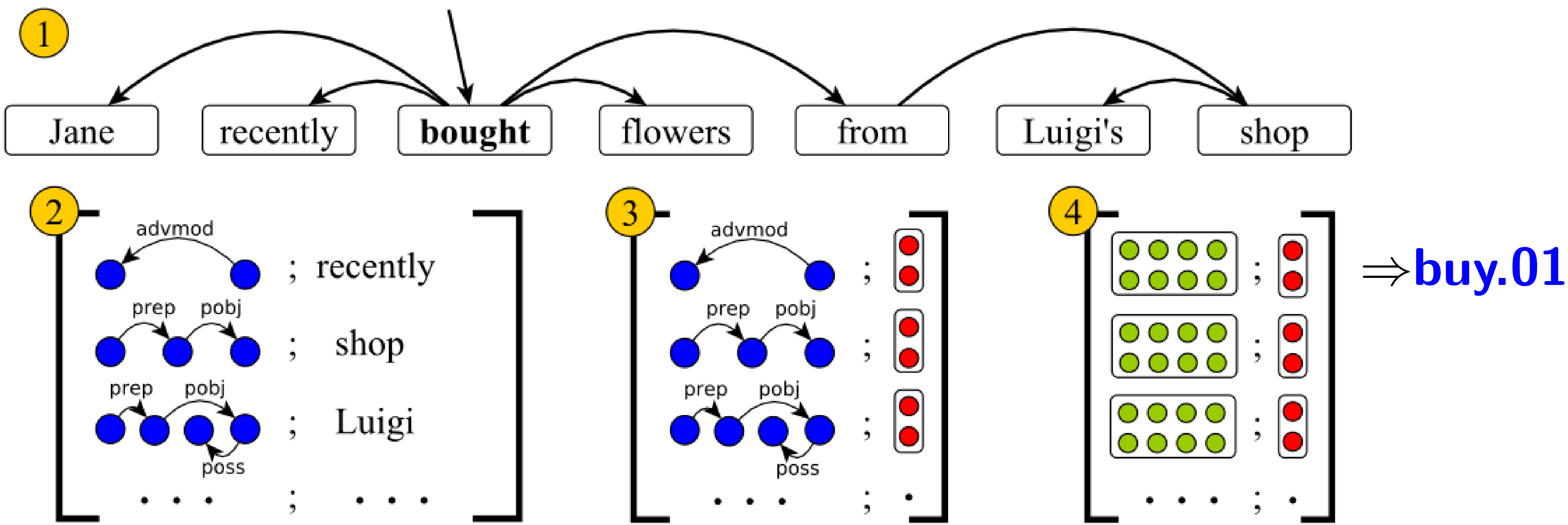
Frame identification



1. Construct dependency parse, choose predicate p (*bought*)
2. Extract paths from p to dependents a
3. Map each dependent a to vector v_a (word vectors)

[Hermann/Das/Weston/Ganchev, 2014]

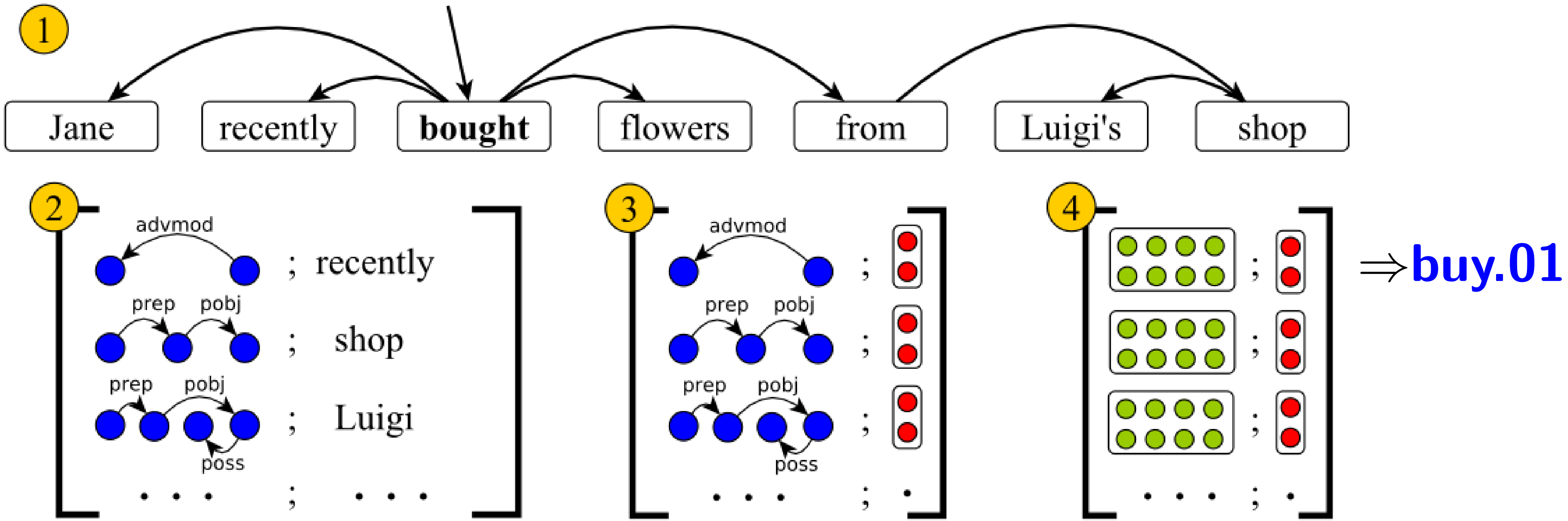
Frame identification



1. Construct dependency parse, choose predicate p (*bought*)
2. Extract paths from p to dependents a
3. Map each dependent a to vector v_a (word vectors)
4. Compute low. dim. representation $\phi = M[v_{a_1}, \dots, v_{a_n}]$

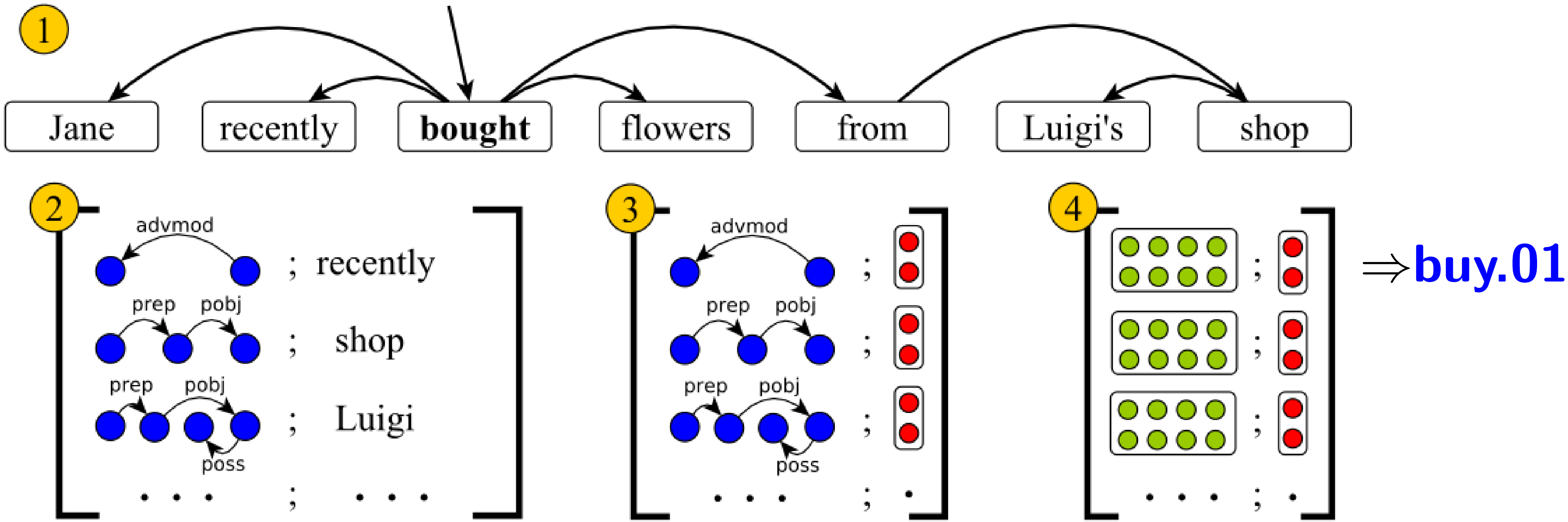
[Hermann/Das/Weston/Ganchev, 2014]

Frame identification



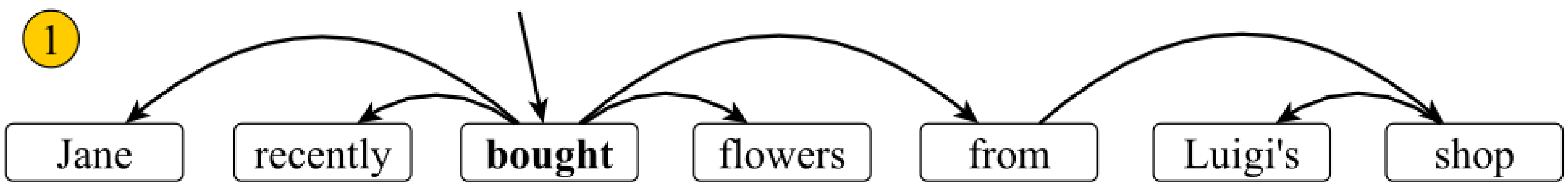
1. Construct dependency parse, choose predicate p (*bought*)
2. Extract paths from p to dependents a
3. Map each dependent a to vector v_a (word vectors)
4. Compute low. dim. representation $\phi = M[v_{a_1}, \dots, v_{a_n}]$
5. Predict score $\phi \cdot \theta_y$ for label y (e.g., **buy.01**)

Frame identification



- Learn parameters $\{v_w\}, M, \{\theta_y\}$ from full supervision
- Vectors allow generalization across verbs and arguments

Argument identification

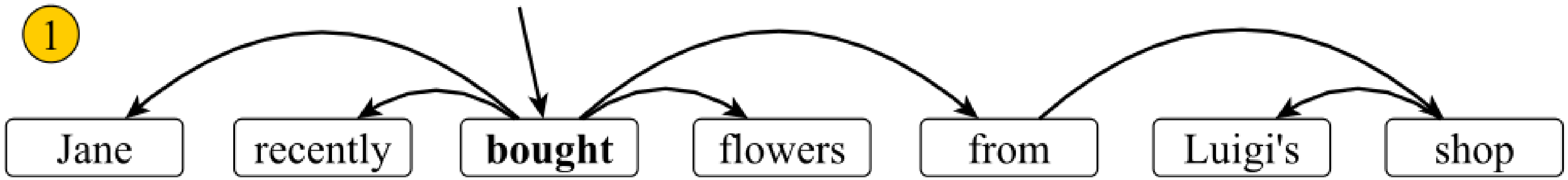


1. Extract candidate argument spans $\{a\}$ (using rules)

Jane *Luigi's shop* *flowers* *flowers from Luigi's shop*

[Punyakanok/Roth/Yih, 2008; Tackstrom/Ganchev/Das, 2015]

Argument identification



1. Extract candidate argument spans $\{a\}$ (using rules)

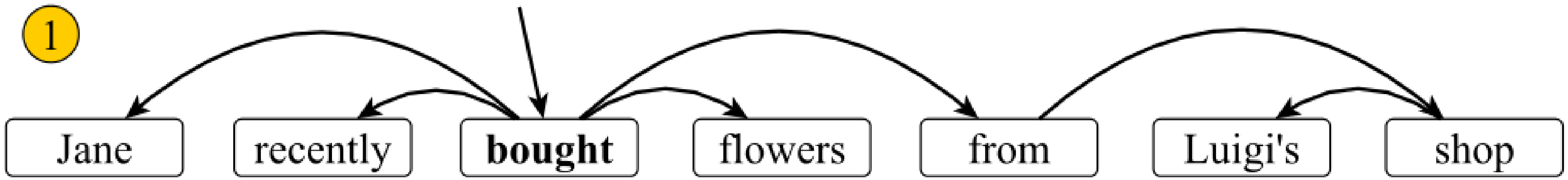
   Jane *Luigi's shop* *flowers* *flowers from Luigi's shop*

2. Predict argument label y_a for each candidate a

   A0, A1, A2, A3, A4, A5, AA, AA-TMP, AA-LOC, $\emptyset$

[Punyakanok/Roth/Yih, 2008; Tackstrom/Ganchev/Das, 2015]

Argument identification



1. Extract candidate argument spans $\{a\}$ (using rules)

   Jane   *Luigi's shop*   *flowers*   *flowers from Luigi's shop*

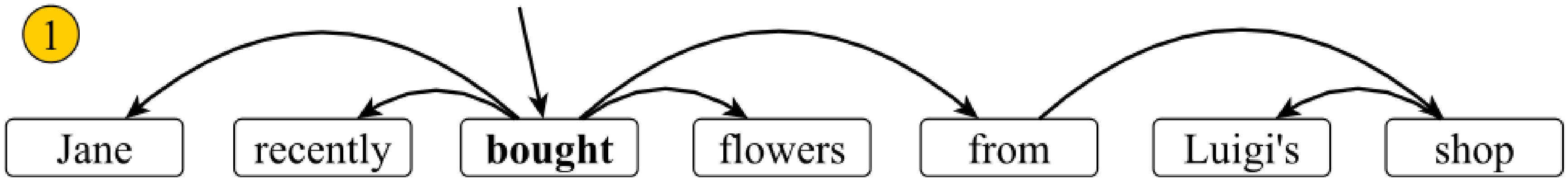
2. Predict argument label y_a for each candidate a

   A0, A1, A2, A3, A4, A5, AA, AA-TMP, AA-LOC, $\emptyset$

Constraints include:

- Assigned spans cannot overlap
- Each core role can be used at most once

Argument identification



1. Extract candidate argument spans $\{a\}$ (using rules)

Jane	*Luigi's shop*	*flowers*	*flowers from Luigi's shop*
A0	A2	A1	$\emptyset$

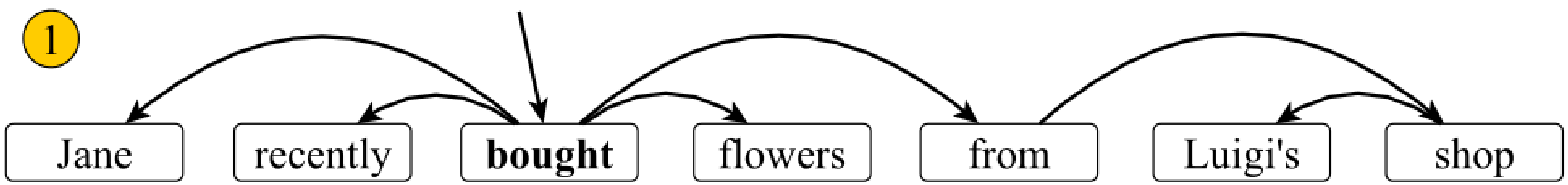
2. Predict argument label y_a for each candidate a

   A0, A1, A2, A3, A4, A5, AA, AA-TMP, AA-LOC, $\emptyset$

Constraints include:

- Assigned spans cannot overlap
- Each core role can be used at most once

Argument identification



1. Extract candidate argument spans $\{a\}$ (using rules)

Jane	*Luigi's shop*	*flowers*	*flowers from Luigi's shop*
A0	A2	A1	$\emptyset$

2. Predict argument label y_a for each candidate a

   A0, A1, A2, A3, A4, A5, AA, AA-TMP, AA-LOC, $\emptyset$

Constraints include:

- Assigned spans cannot overlap
- Each core role can be used at most once

Structured prediction: ILP or dynamic programming

A brief history

- First system (on FrameNet) [Gildea/Jurafsky, 2002]
- CoNLL shared tasks [2004, 2005]
- Use ILP to enforce constraints on arguments [Punyakanok/Roth/Yih, 2008]
- No feature engineering or parse trees [Collobert/Weston, 2008]
- Semi-supervised frame identification [Das/Smith, 2011]
- Embeddings for frame identification [Hermann/Das/Weston/Ganchev, 2014]
- Dynamic programming for some argument constraints [Tackstrom/Ganchev/Das, 2015]

[Banarescu et al., 2013]

Abstract meaning representation (AMR)

Semantic role labeling:

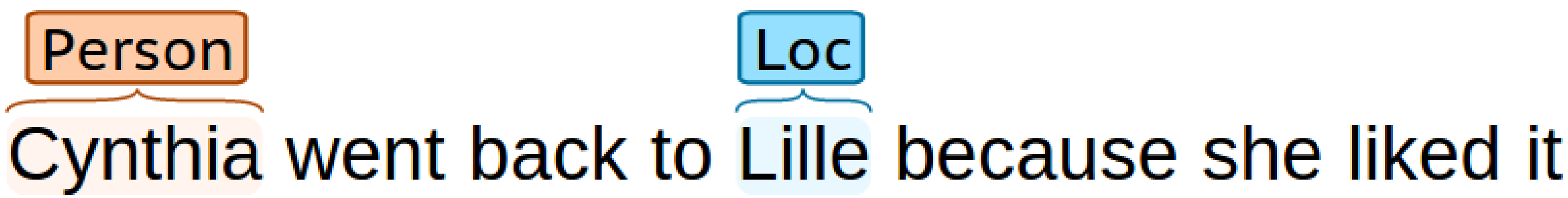
- predicate + semantic roles

Abstract meaning representation (AMR)

Semantic role labeling:

- predicate + semantic roles

Named-entity recognition:



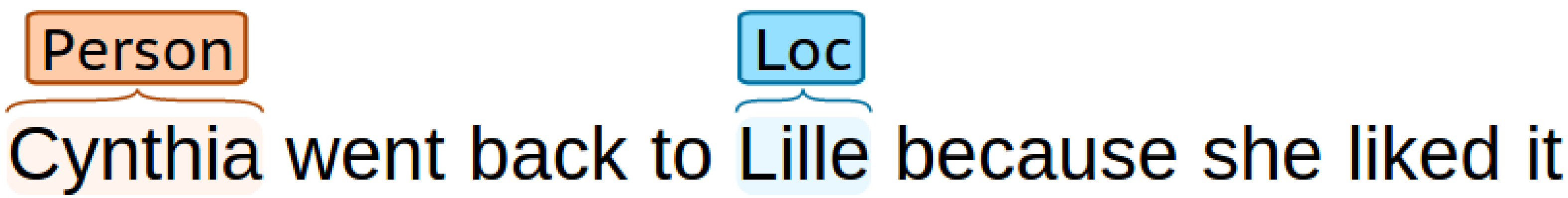
[Banarescu et al., 2013]

Abstract meaning representation (AMR)

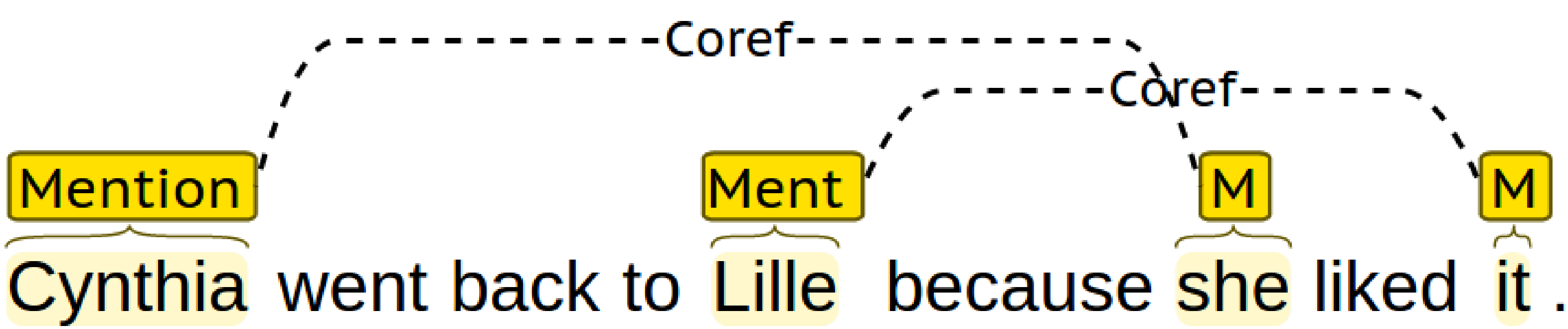
Semantic role labeling:

- predicate + semantic roles

Named-entity recognition:



Coreference resolution:



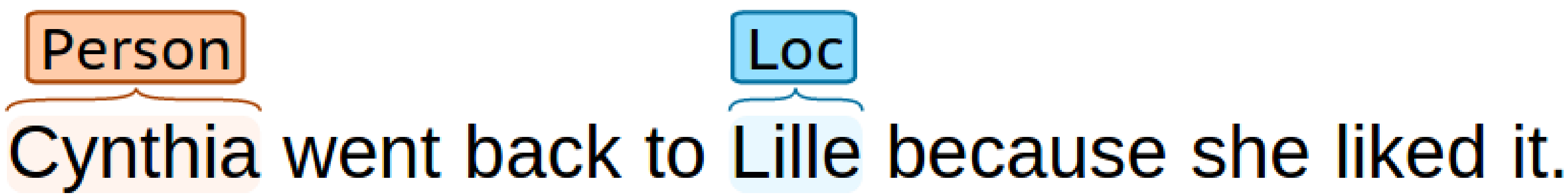
[Banarescu et al., 2013]

Abstract meaning representation (AMR)

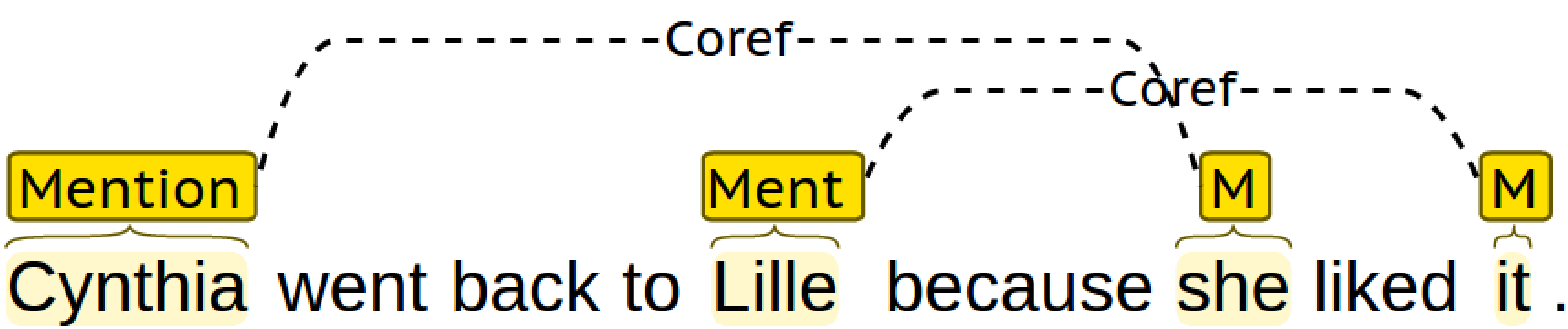
Semantic role labeling:

- predicate + semantic roles

Named-entity recognition:



Coreference resolution:



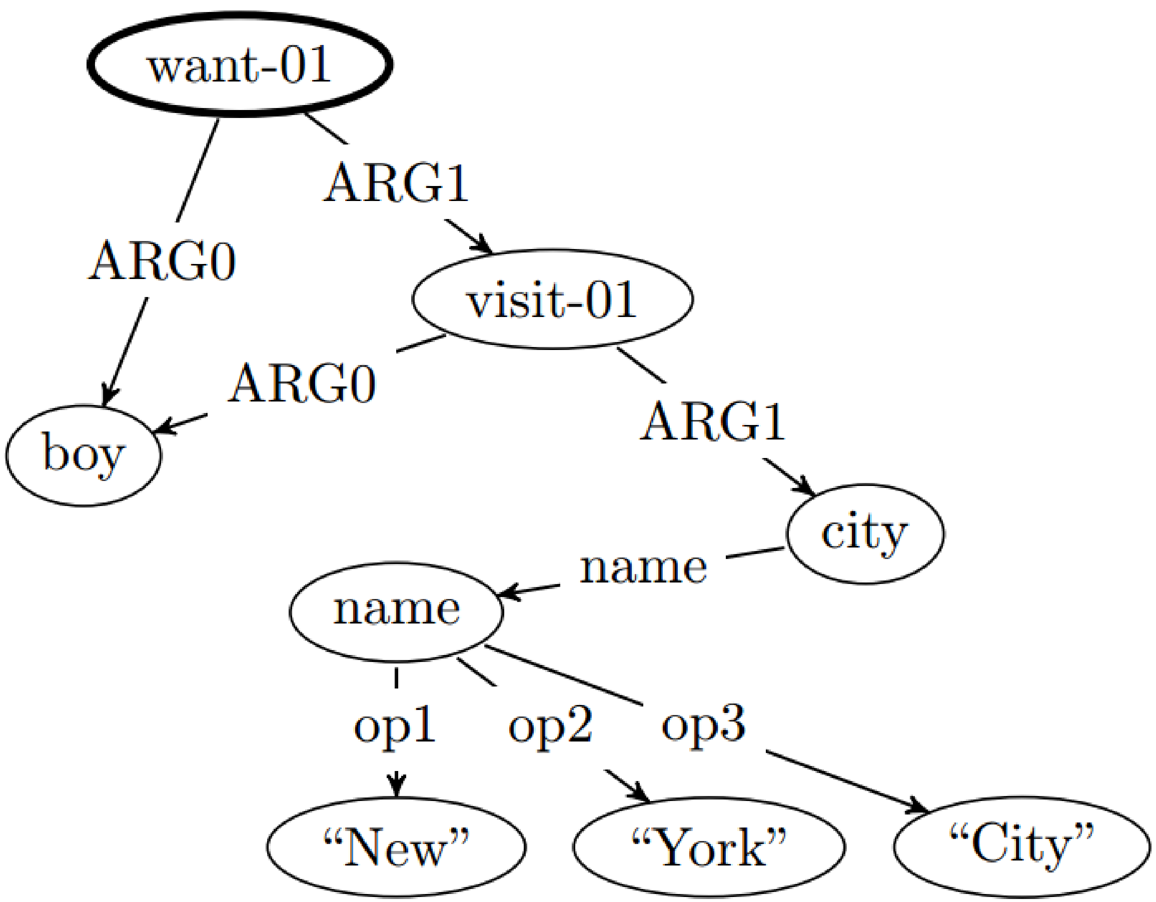
Motivation of AMR: **unify all semantic annotation**

AMR parsing task

Input: sentence

The boy wants to go to New York City.

Output: graph



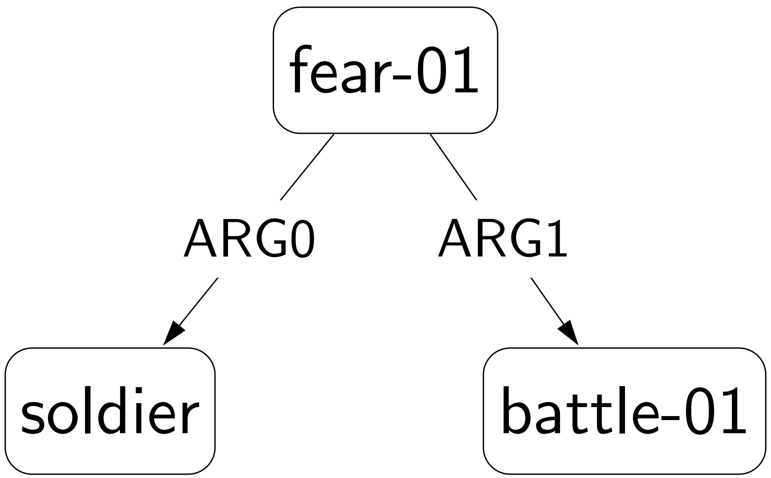
[Banarescu et al., 2013]

AMR: normalize aggressively

The soldier feared battle.

AMR: normalize aggressively

The soldier feared battle.



AMR: normalize aggressively

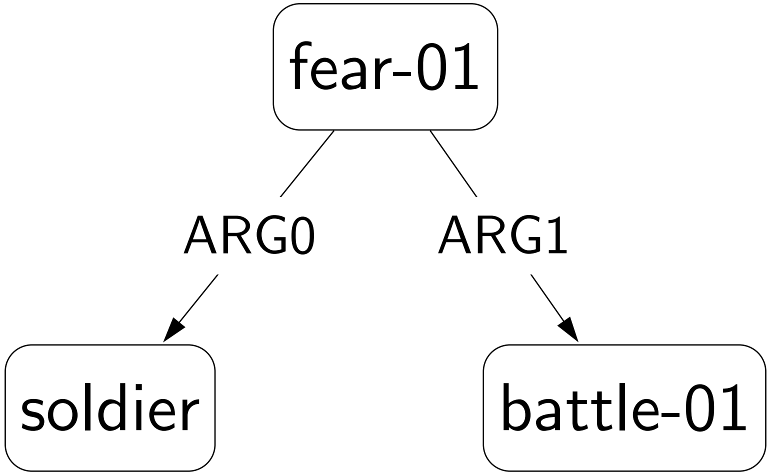
The soldier feared battle.

The soldier was afraid of battle.

The soldier had a fear of battle.

Battle was feared by the soldier.

Battle was what the soldier was afraid of.



AMR: normalize aggressively

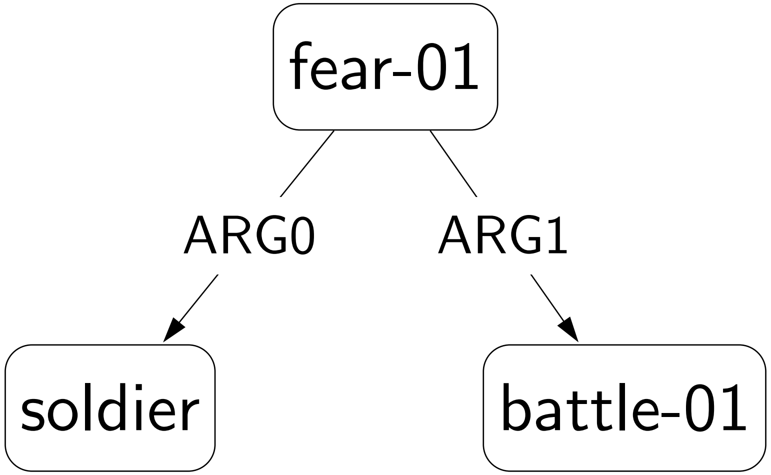
The soldier feared battle.

The soldier was afraid of battle.

The soldier had a fear of battle.

Battle was feared by the soldier.

Battle was what the soldier was afraid of.

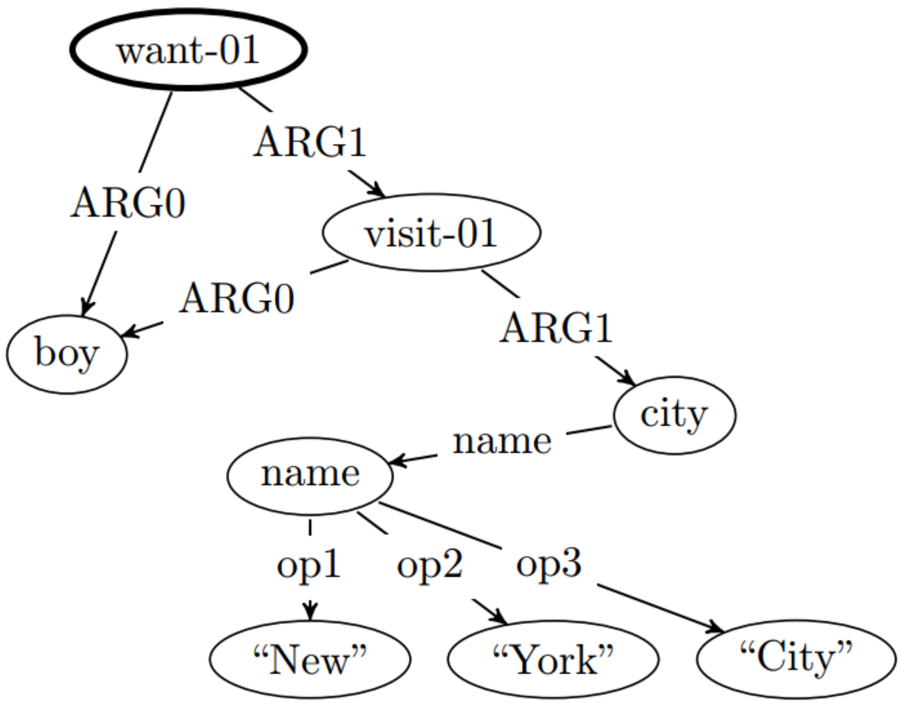


- **Sentence-level annotation** (unlike semantic role labeling)
- Challenge: must learn an (implicit) **alignment**!

AMR parsing: extract lexicon (step 1)

- Goal: given sentence-graph training examples, extract mapping from phrases to graph fragments

The boy wants to go to New York City.

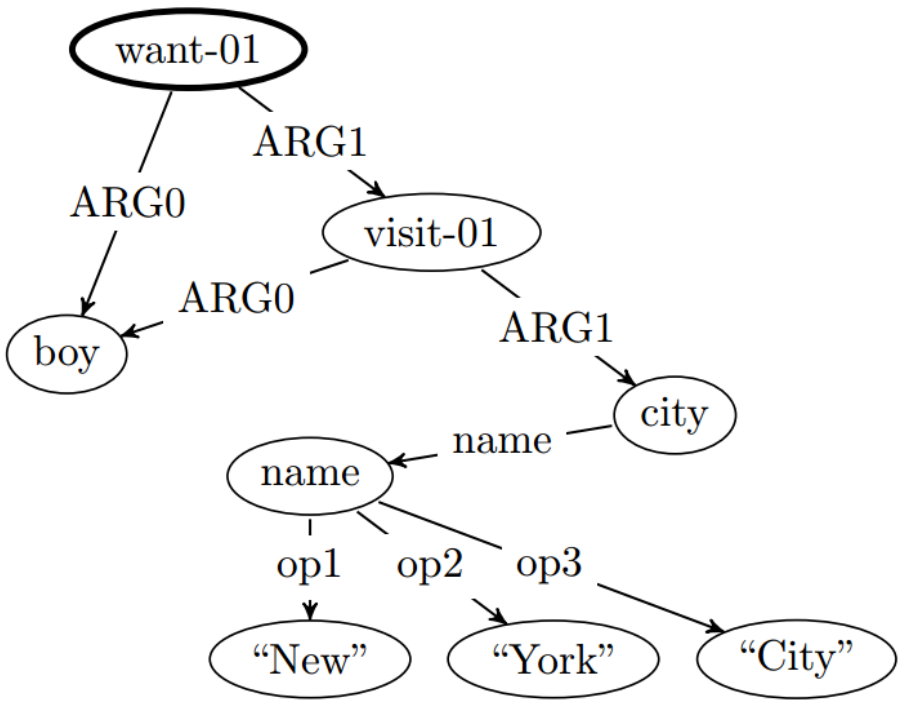


[Flanigan/Thomson/Carbonell/Dyer/Smith, 2014]

AMR parsing: extract lexicon (step 1)

- Goal: given sentence-graph training examples, extract mapping from phrases to graph fragments

The boy wants to go to New York City.



...

wants $\Rightarrow$ want-01

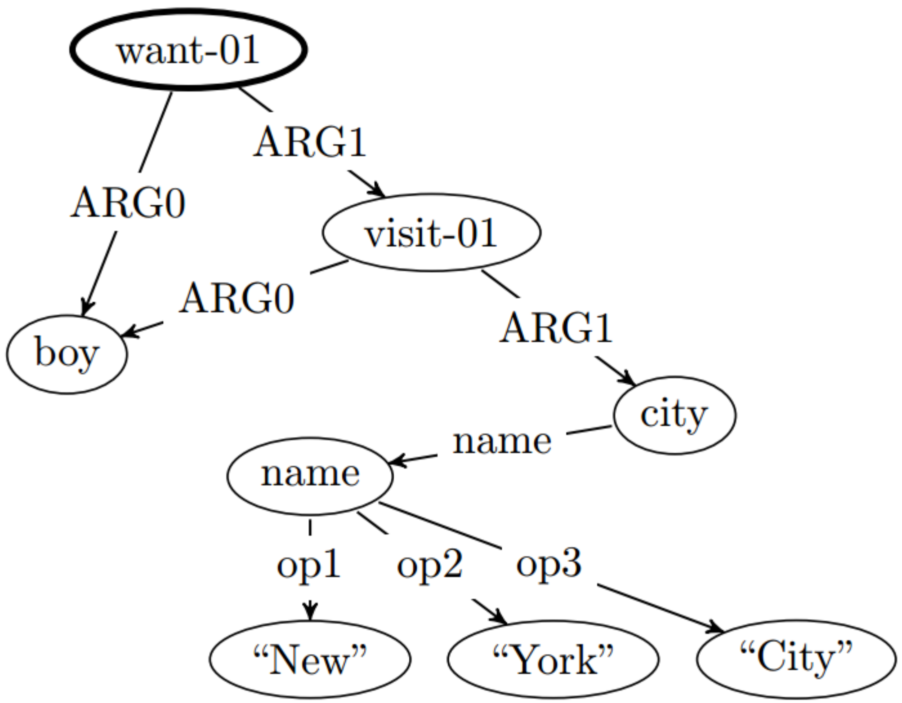
...

[Flanigan/Thomson/Carbonell/Dyer/Smith, 2014]

AMR parsing: extract lexicon (step 1)

- Goal: given sentence-graph training examples, extract mapping from phrases to graph fragments

The boy wants to go to New York City.



...

wants $\Rightarrow$ want-01

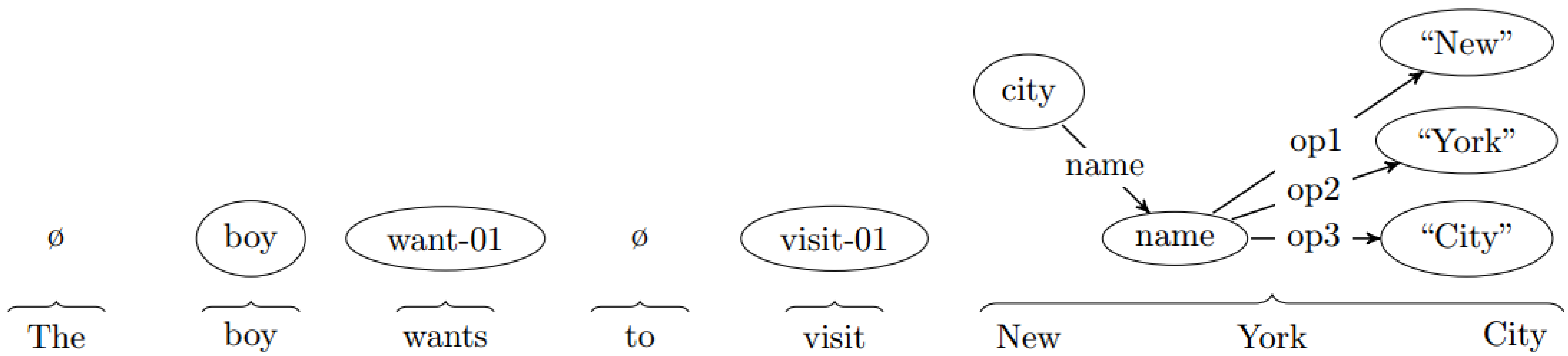
...

- Rule-based system (14 rules)

[Flanigan/Thomson/Carbonell/Dyer/Smith, 2014]

AMR parsing: concept labeling (step 2)

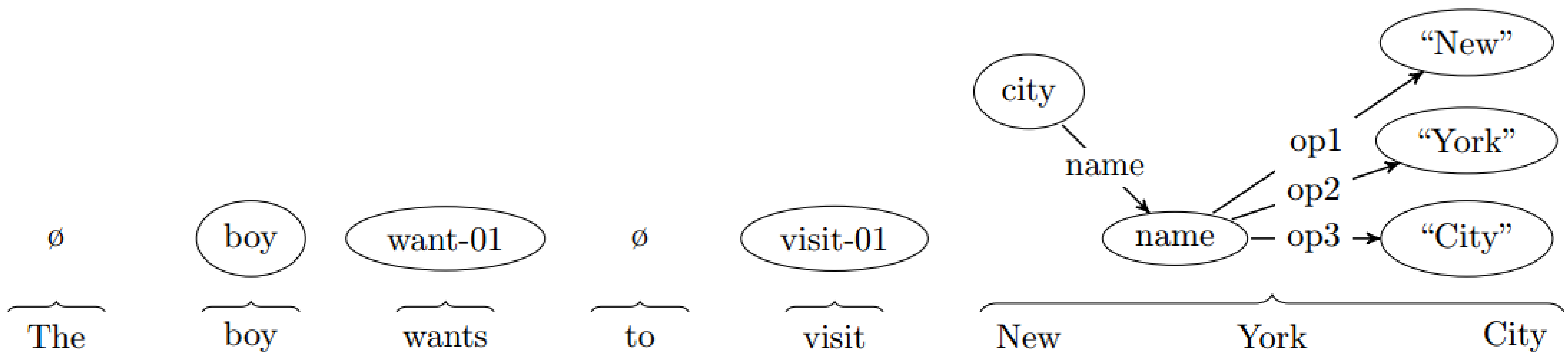
- Semi-Markov model: segment new sentence into phrases and label each with at most one **concept graph**



[Flanigan/Thomson/Carbonell/Dyer/Smith, 2014]

AMR parsing: concept labeling (step 2)

- Semi-Markov model: segment new sentence into phrases and label each with at most one **concept graph**



- Dynamic programming for computing best labeling

[Flanigan/Thomson/Carbonell/Dyer/Smith, 2014]

AMR parsing: connect concepts (step 3)

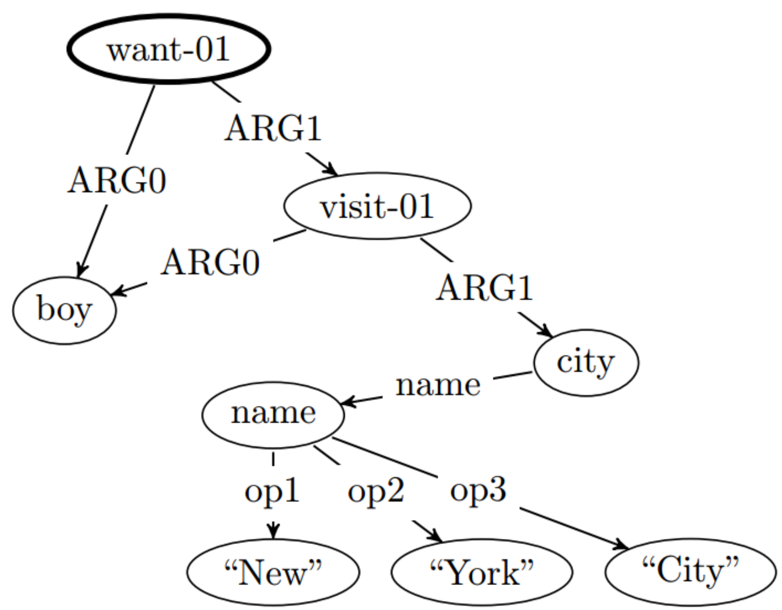
- Build a graph over concepts satisfying constraints

  All concept graphs produced by labeling are used

  At most 1 edge between two nodes

  For each node, at most one instance of label

  Weakly connected



AMR parsing: connect concepts (step 3)

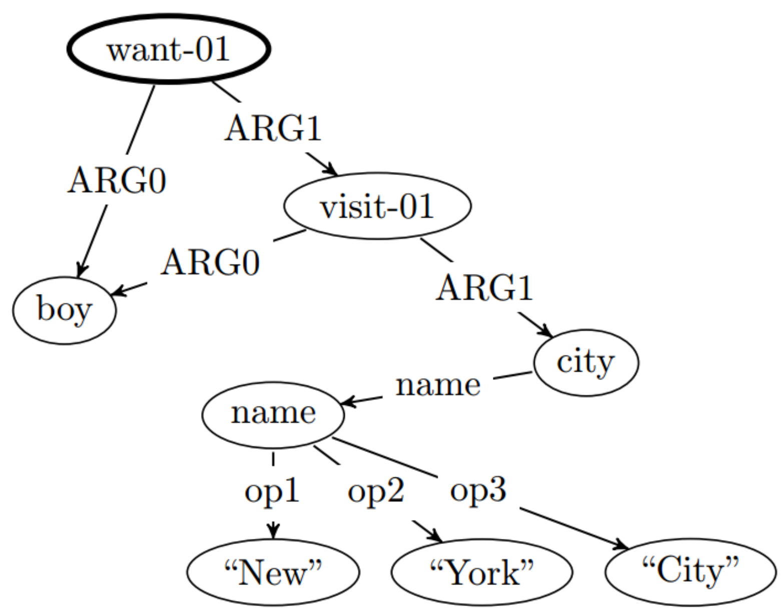
- Build a graph over concepts satisfying constraints

  All concept graphs produced by labeling are used

  At most 1 edge between two nodes

  For each node, at most one instance of label

  Weakly connected



- Algorithm: adaptation of maximum spanning tree

Summary so far

- Frames: stereotypical situations that provide rich structure for understanding

Summary so far

- Frames: stereotypical situations that provide rich structure for understanding

- Semantic role labeling (FrameNet, PropBank): resource and task that operationalize frames

- AMR graphs: unified broad-coverage semantic annotation

Summary so far

- Frames: stereotypical situations that provide rich structure for understanding

- Semantic role labeling (FrameNet, PropBank): resource and task that operationalize frames

- AMR graphs: unified broad-coverage semantic annotation

- Methods: classification (featurize a structured object), structured prediction (not a tractable structure)

Food for thought

- Both distributional semantics (DS) and frame semantics (FS) involve compression/abstraction

- Frame semantics exposes more structure, more tied to an external world, but requires more supervision

Food for thought

- Both distributional semantics (DS) and frame semantics (FS) involve compression/abstraction
- Frame semantics exposes more structure, more tied to an external world, but requires more supervision

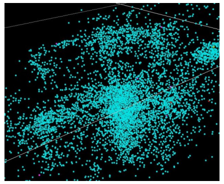
Examples to ponder:

Cynthia went to the bike shop ***yesterday****.*

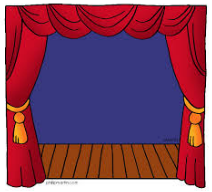
Cynthia bought the ***cheapest*** *bike.*

Outline

Properties of language



Distributional semantics



Frame semantics

Model-theoretic semantics

Reflections

Types of semantics

Every *non-blue block is next to* **some** *blue block.*

Types of semantics

Every *non-blue block is next to* **some** *blue block.*

Distributional semantics: *block* is like *brick*, *some* is like *every*

Types of semantics

Every *non-blue block is next to* **some** *blue block.*

Distributional semantics: *block* is like *brick*, *some* is like *every*

Frame semantics: *is next to* has two arguments, *block* and *block*

Types of semantics

***Every** non-blue block is next to **some** blue block.*

Distributional semantics: *block* is like *brick*, *some* is like *every*

Frame semantics: *is next to* has two arguments, *block* and *block*

Model-theoretic semantics: tell the difference between

and

[Montague, 1973]

Model-theoretic/compositional semantics

Two ideas: **model theory** and **compositionality**

Model theory: interpretation depends on the world state

Block 2 is blue.

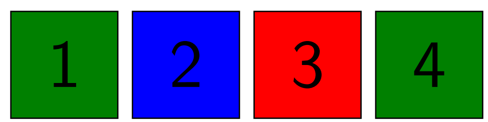
[Montague, 1973]

Model-theoretic/compositional semantics

Two ideas: **model theory** and **compositionality**

Model theory: interpretation depends on the world state

Block 2 is blue.

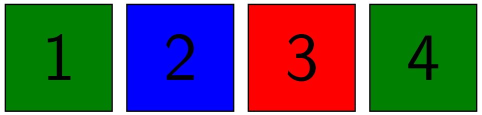


Model-theoretic/compositional semantics

Two ideas: **model theory** and **compositionality**

Model theory: interpretation depends on the world state

Block 2 is blue.



Compositionality: meaning of whole is meaning of parts

The [block left of the red block] is blue.

Model-theoretic semantics

Framework: map natural language into **logical forms**

Model-theoretic semantics

Framework: map natural language into **logical forms**

Factorization: **understanding** and **knowing**

What is the largest city in California?

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

Model-theoretic semantics

Framework: map natural language into **logical forms**

Factorization: **understanding** and **knowing**

What is the largest city in California?

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

`Los Angeles`

Systems

Rule-based systems:

- STUDENT for solving algebra word problems [Bobrow et al., 1968]
- LUNAR question answering system about moon rocks [Woods et al., 1972]

Systems

Rule-based systems:

- STUDENT for solving algebra word problems [Bobrow et al., 1968]
- LUNAR question answering system about moon rocks [Woods et al., 1972]

Statistical semantic parsers:

- Learn from logical forms [Zelle/Mooney, 1996; Zettlemoyer/Collins, 2005, 2007, 2009; Wong/Mooney, 2006; Kwiatkowski et al. 2010]
- Learn from denotations [Clarke et. al, 2010; Liang et al. 2011]

Systems

Rule-based systems:

- STUDENT for solving algebra word problems [Bobrow et al., 1968]
- LUNAR question answering system about moon rocks [Woods et al., 1972]

Statistical semantic parsers:

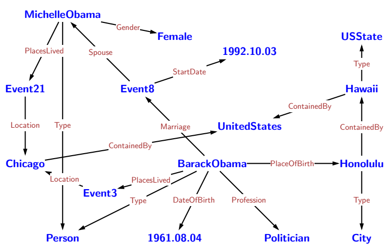
- Learn from logical forms [Zelle/Mooney, 1996; Zettlemoyer/Collins, 2005, 2007, 2009; Wong/Mooney, 2006; Kwiatkowski et al. 2010]
- Learn from denotations [Clarke et. al, 2010; Liang et al. 2011]

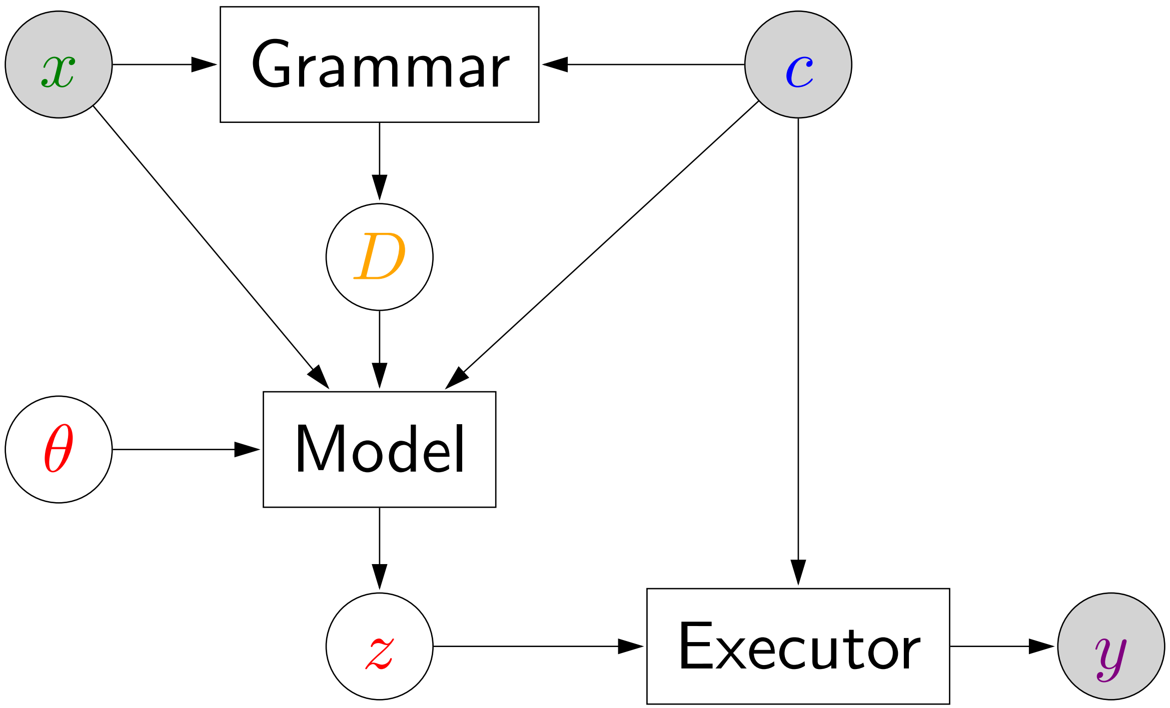
Applications of semantic parsing:

- Question answering on knowledge bases [Berant et al., 2013, 2014; Kwiatkowski et al., 2013; Pasupat et al., 2015]
- Robot control [Tellex et. al, 2011; Artzi/Zettlemoyer, 2013; Misra et al. 2014, 2015]
- Identifying objects in a scene [Matuszek et. al, 2012]
- Solving algebra word problems [Kushman et. al, 2014; Hosseini et al., 2014]

Components of a semantic parser

people who have lived in Chicago





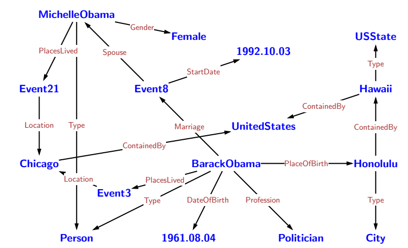
$\texttt{Type.Person} \sqcap \texttt{PlacesLived.Location.Chicago}$

$\{\texttt{BarackObama}, \ldots\}$

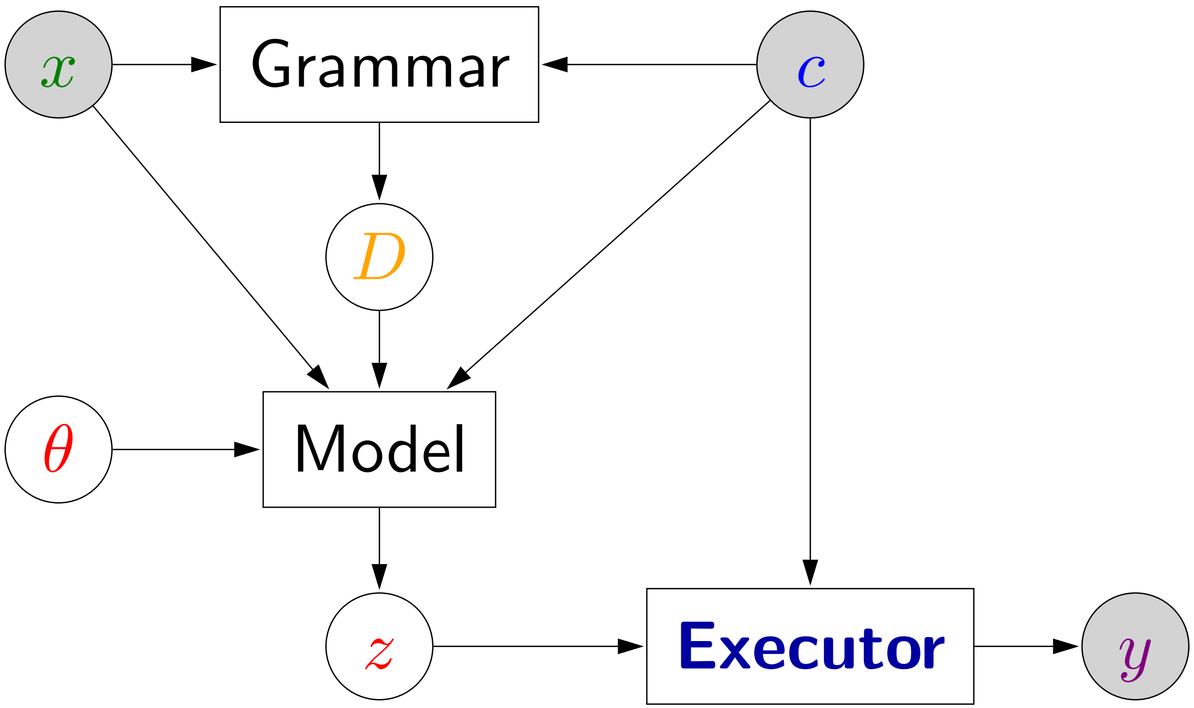
Parser

Learner

Components of a semantic parser



people who have lived in Chicago



$\texttt{Type.Person} \sqcap \texttt{PlacesLived.Location.Chicago}$

$\{\texttt{BarackObama}, \ldots\}$

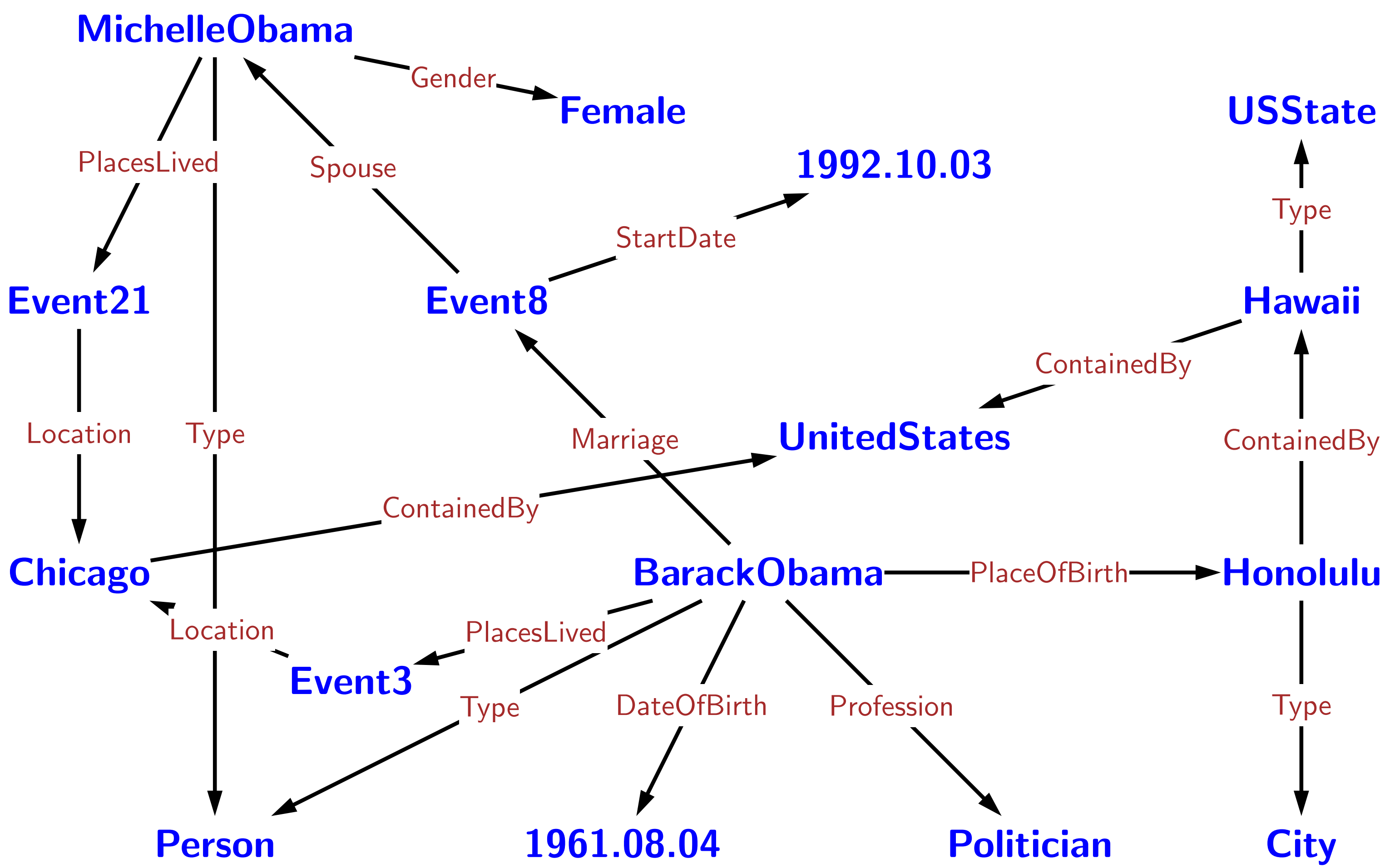
Parser

Learner

[Bollacker, 2008; Google, 2013]

Freebase

100M entities (nodes)   1B assertions (edges)



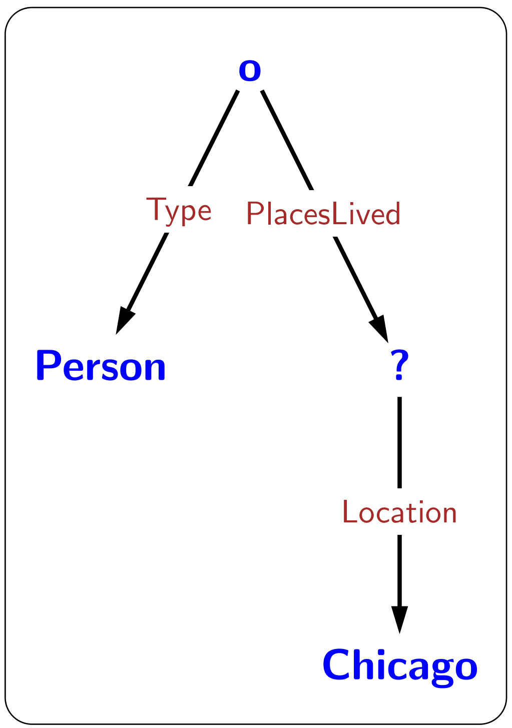
Logical forms: lambda DCS

$\texttt{Type.Person} \sqcap \texttt{PlacesLived.Location.Chicago}$

[Liang, 2013]

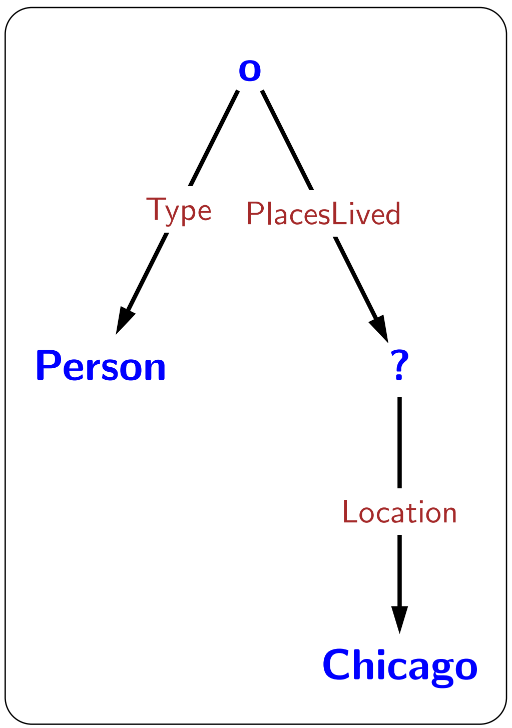
Logical forms: lambda DCS

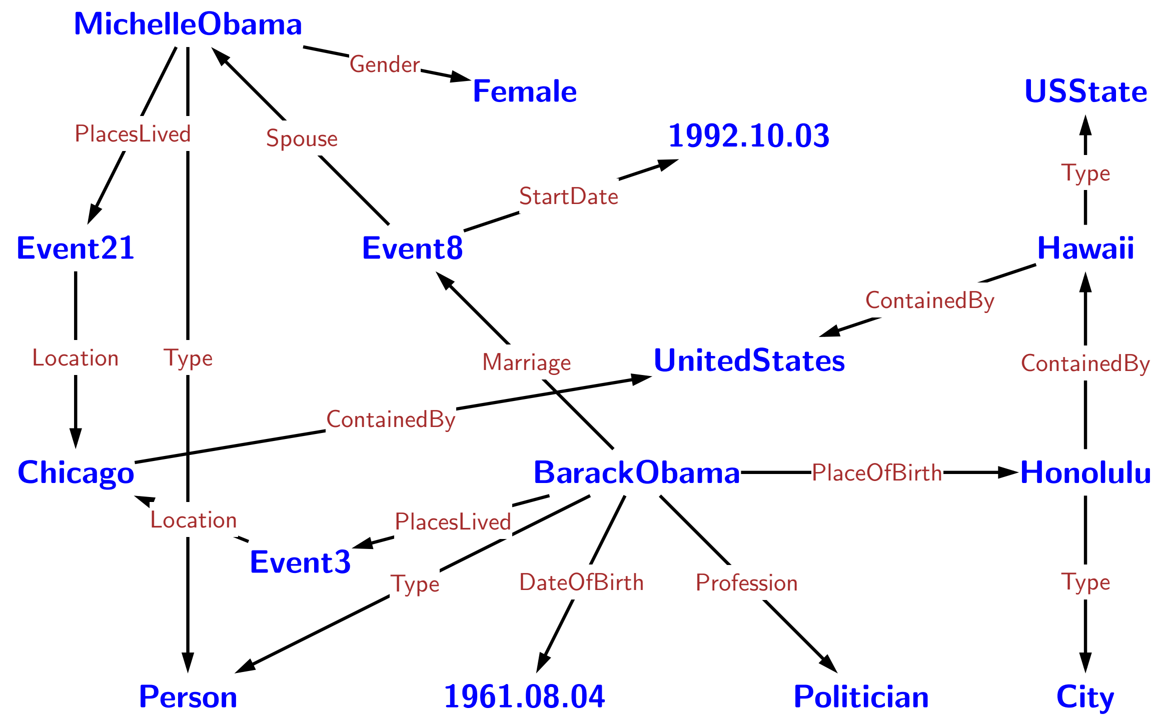
$\texttt{Type.Person} \sqcap \texttt{PlacesLived.Location.Chicago}$



Logical forms: lambda DCS

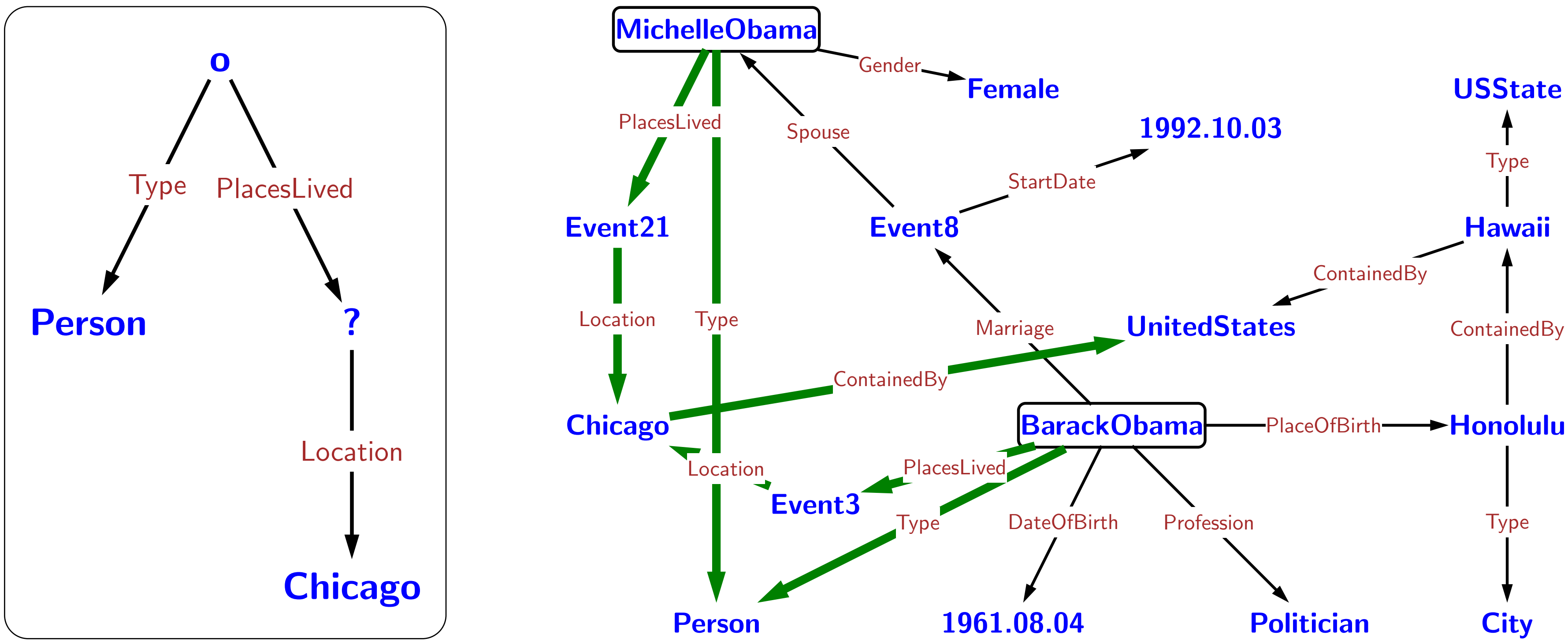
$\texttt{Type.Person} \sqcap \texttt{PlacesLived.Location.Chicago}$





Logical forms: lambda DCS

$\texttt{Type.Person} \sqcap \texttt{PlacesLived.Location.Chicago}$



Lambda DCS

Entity

Chicago

Lambda DCS

Entity

Chicago

Join

PlaceOfBirth.Chicago

Lambda DCS

Entity

Chicago

Join

PlaceOfBirth.Chicago

Intersect

Type.Person $\sqcap$ PlaceOfBirth.Chicago

Lambda DCS

Entity

Chicago

Join

PlaceOfBirth.Chicago

Intersect

Type.Person $\sqcap$ PlaceOfBirth.Chicago

Aggregation

count(Type.Person $\sqcap$ PlaceOfBirth.Chicago)

Lambda DCS

Entity

Chicago

Join

PlaceOfBirth.Chicago

Intersect

Type.Person $\sqcap$ PlaceOfBirth.Chicago

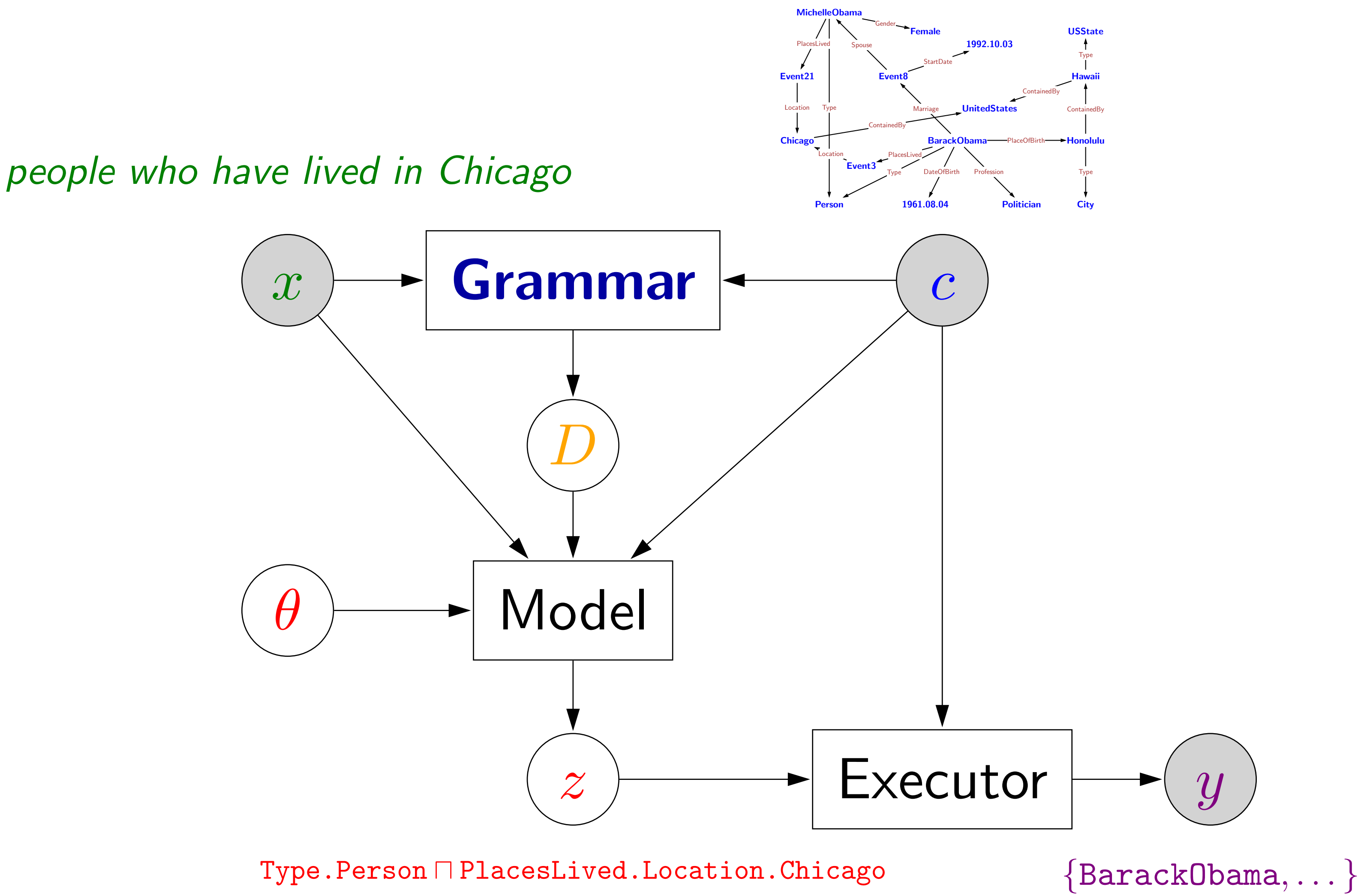
Aggregation

count(Type.Person $\sqcap$ PlaceOfBirth.Chicago)

Superlative

argmin(Type.Person $\sqcap$ PlaceOfBirth.Chicago, DateOfBirth)

Components of a semantic parser



Parser

Learner

Generating candidate derivations

utterance → Grammar → *derivation 1*

derivation 2

...

Generating candidate derivations

utterance → Grammar → *derivation 1*
derivation 2
...

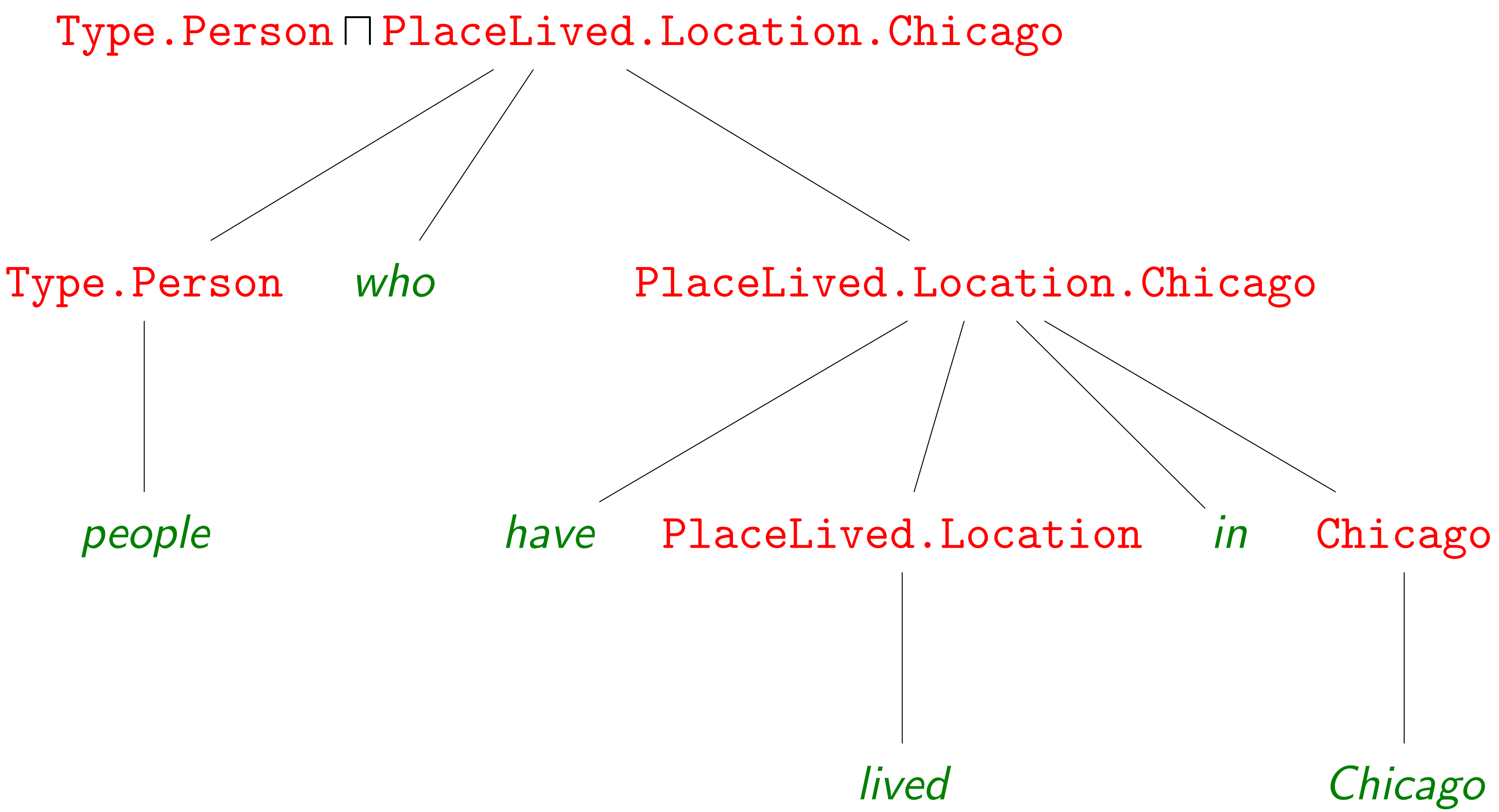
A Simple Grammar

(lexicon)	*Chicago*		$\Rightarrow$	N : `Chicago`
(lexicon)	*people*		$\Rightarrow$	N : `Type.Person`
(lexicon)	*lived*		$\Rightarrow$	N—N : `PlacesLived.Location`
(join)	N—N : r	N : z	$\Rightarrow$	N : $r.z$
(intersect)	N : z_1	N : z_2	$\Rightarrow$	N : $z_1 \sqcap z_2$

Derivations

A Simple Grammar

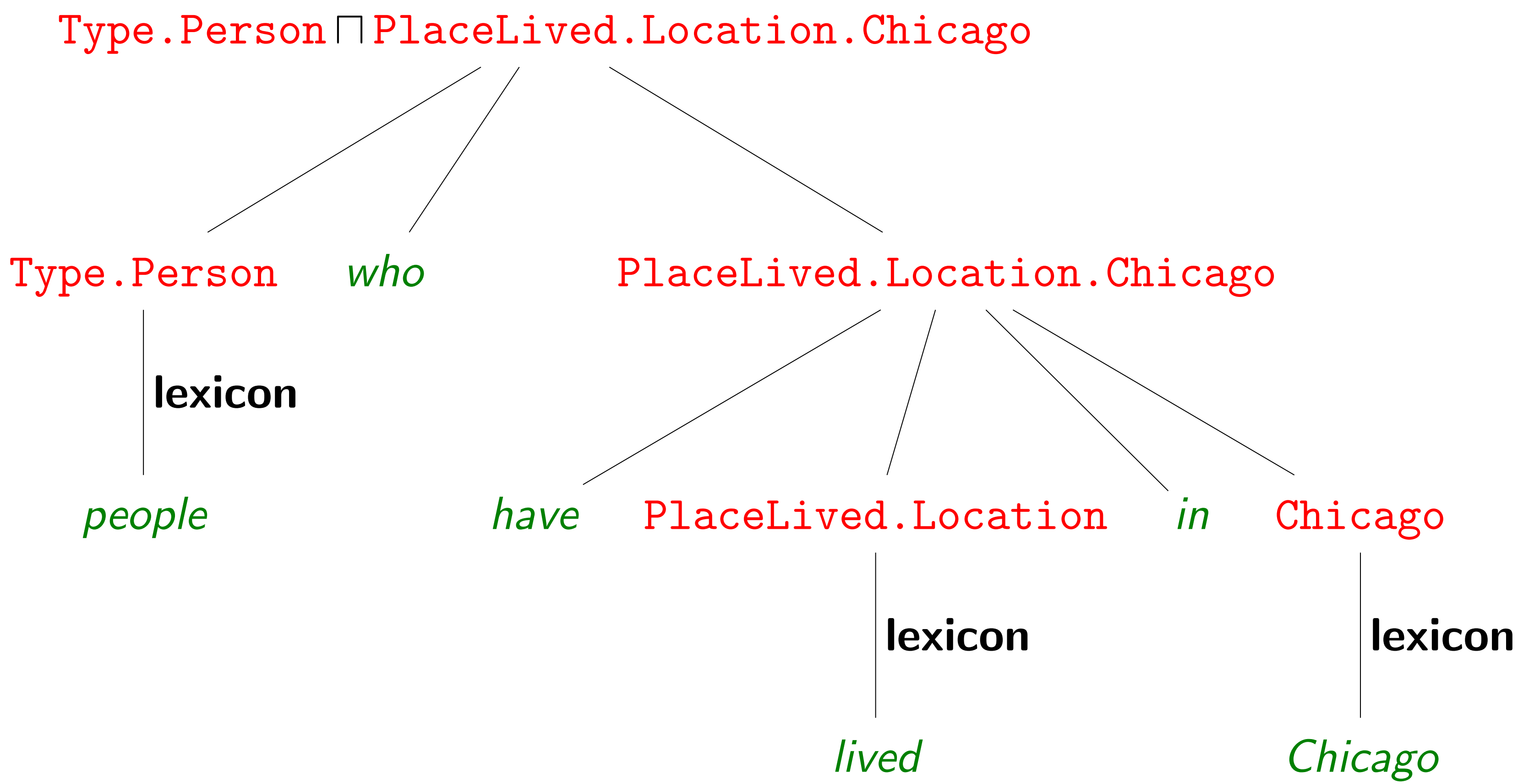
(lexicon)	*Chicago*		$\Rightarrow$	N : `Chicago`
(lexicon)	*people*		$\Rightarrow$	N : `Type.Person`
(lexicon)	*lived*		$\Rightarrow$	N—N : `PlacesLived.Location`
(join)	N—N : r	N : z	$\Rightarrow$	N : $r.z$
(intersect)	N : z_1	N : z_2	$\Rightarrow$	N : $z_1 \sqcap z_2$



Derivations

A Simple Grammar

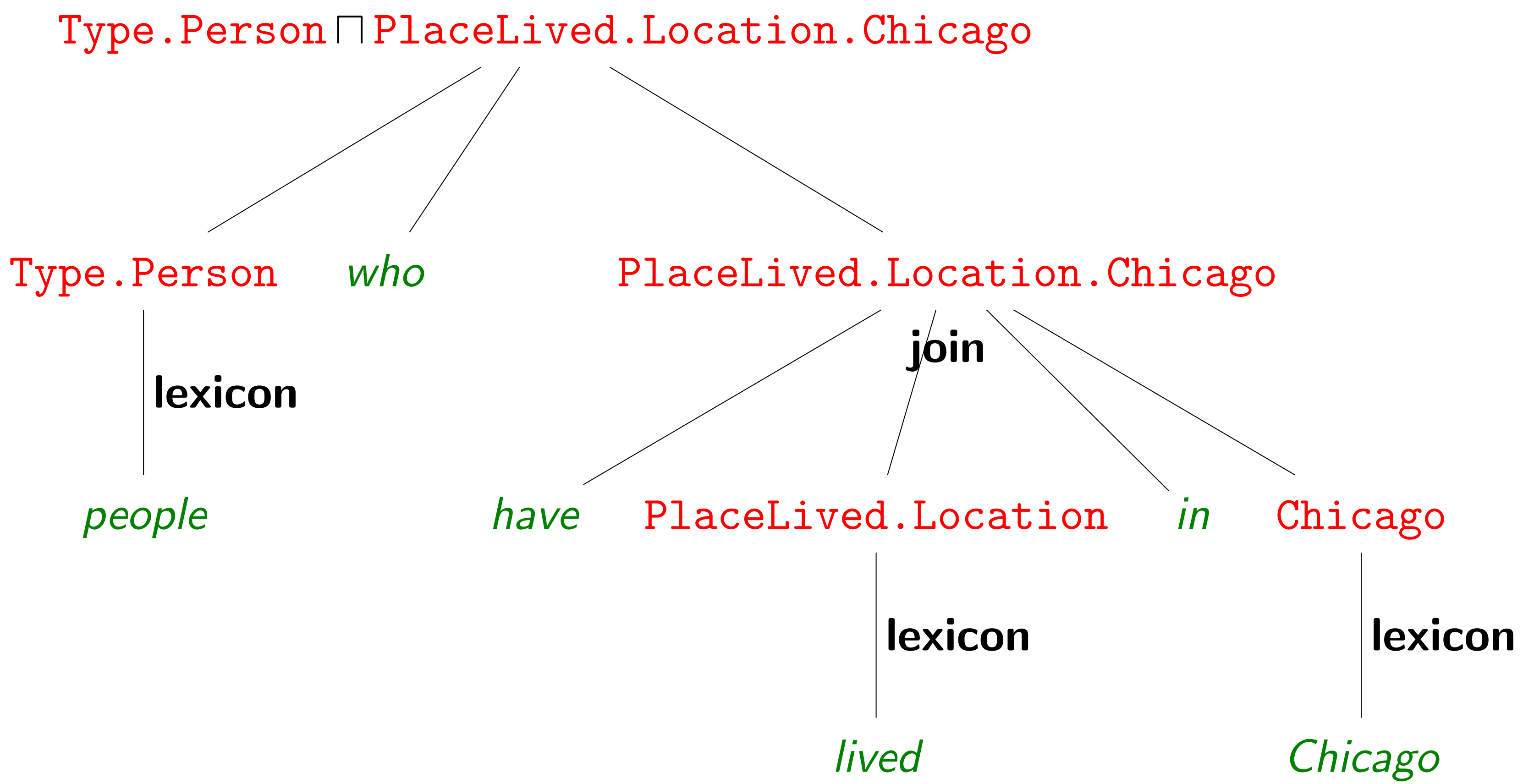
(lexicon)	*Chicago*		$\Rightarrow$	N : `Chicago`
(lexicon)	*people*		$\Rightarrow$	N : `Type.Person`
(lexicon)	*lived*		$\Rightarrow$	N—N : `PlacesLived.Location`
(join)	N—N : r	N : z	$\Rightarrow$	N : $r.z$
(intersect)	N : z_1	N : z_2	$\Rightarrow$	N : $z_1 \sqcap z_2$



Derivations

A Simple Grammar

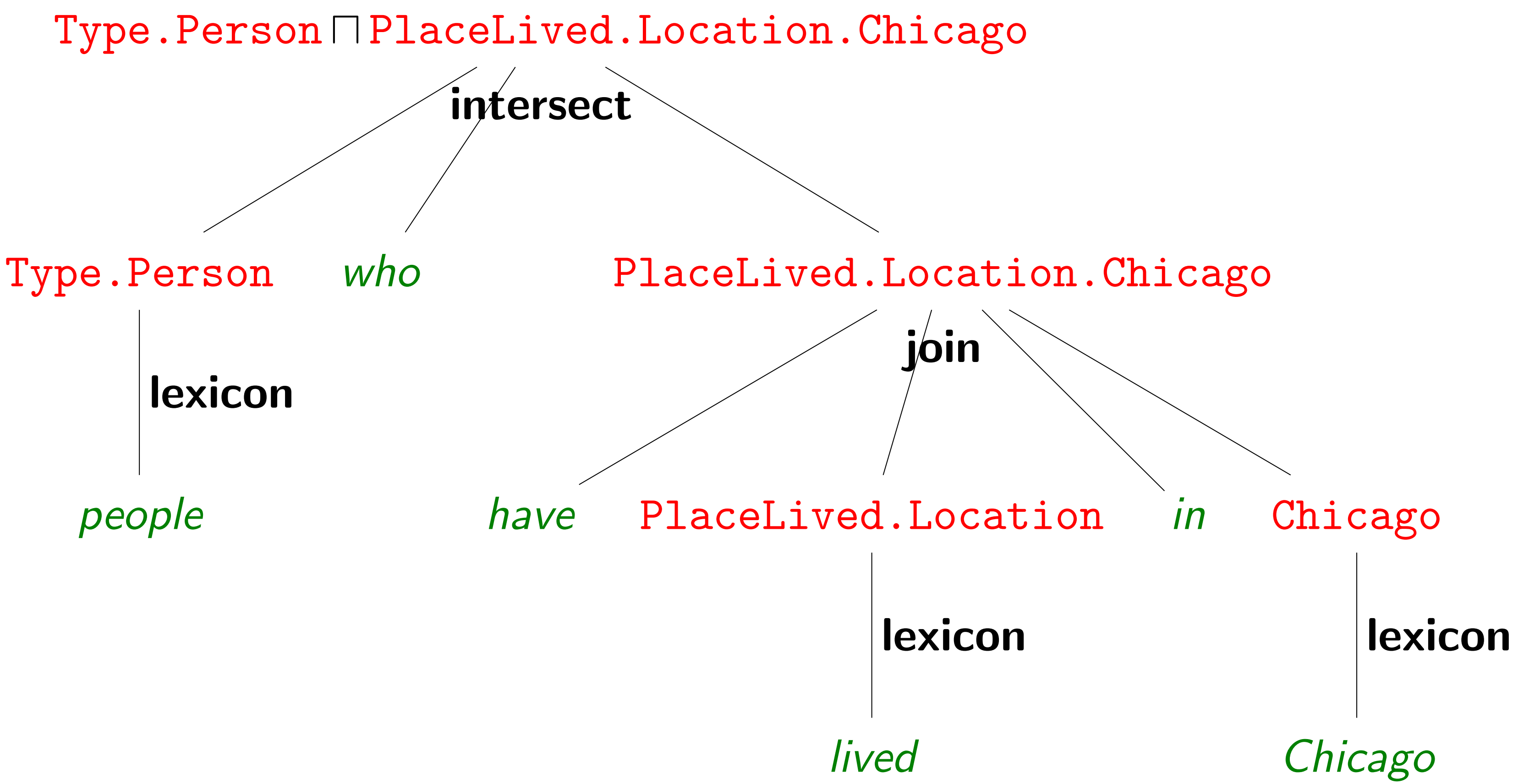
(lexicon)	*Chicago*		$\Rightarrow$	N : `Chicago`
(lexicon)	*people*		$\Rightarrow$	N : `Type.Person`
(lexicon)	*lived*		$\Rightarrow$	N—N : `PlacesLived.Location`
(join)	N—N : r	N : z	$\Rightarrow$	N : $r.z$
(intersect)	N : z_1	N : z_2	$\Rightarrow$	N : $z_1 \sqcap z_2$



Derivations

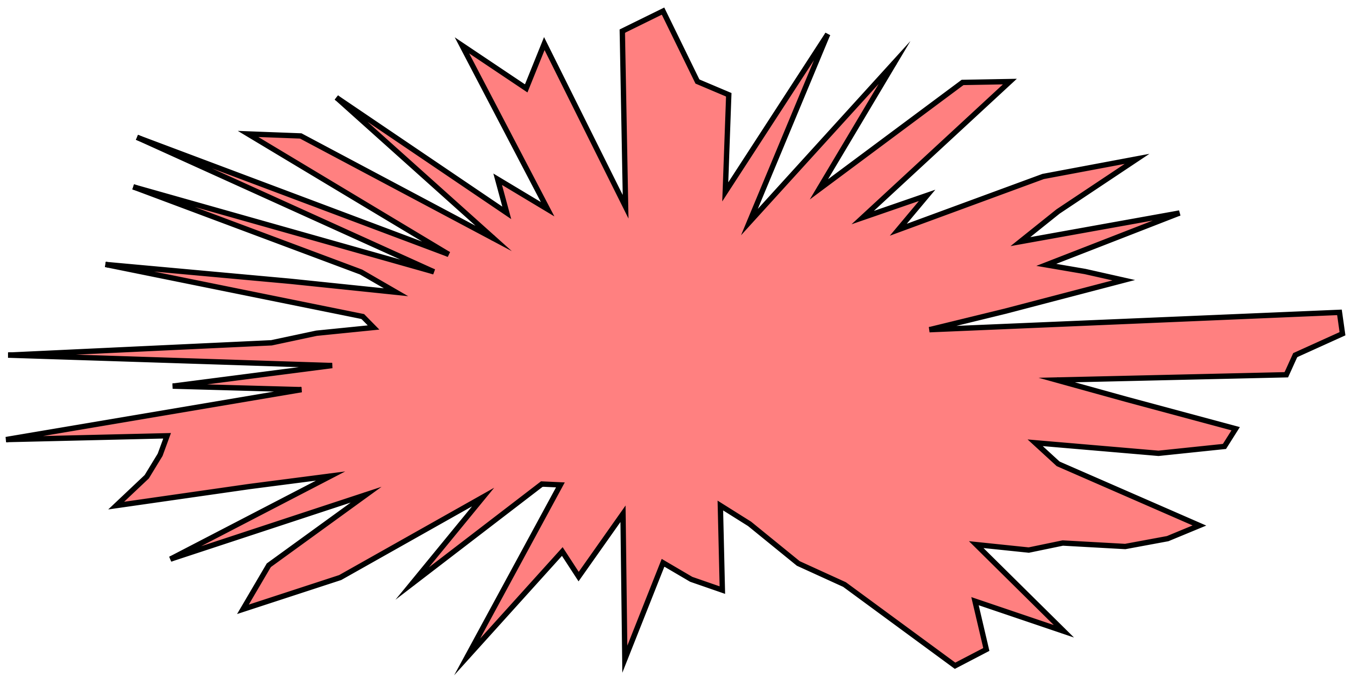
A Simple Grammar

(lexicon)	*Chicago*		$\Rightarrow$	N : `Chicago`
(lexicon)	*people*		$\Rightarrow$	N : `Type.Person`
(lexicon)	*lived*		$\Rightarrow$	N—N : `PlacesLived.Location`
(join)	N—N : r	N : z	$\Rightarrow$	N : $r.z$
(intersect)	N : z_1	N : z_2	$\Rightarrow$	N : $z_1 \sqcap z_2$



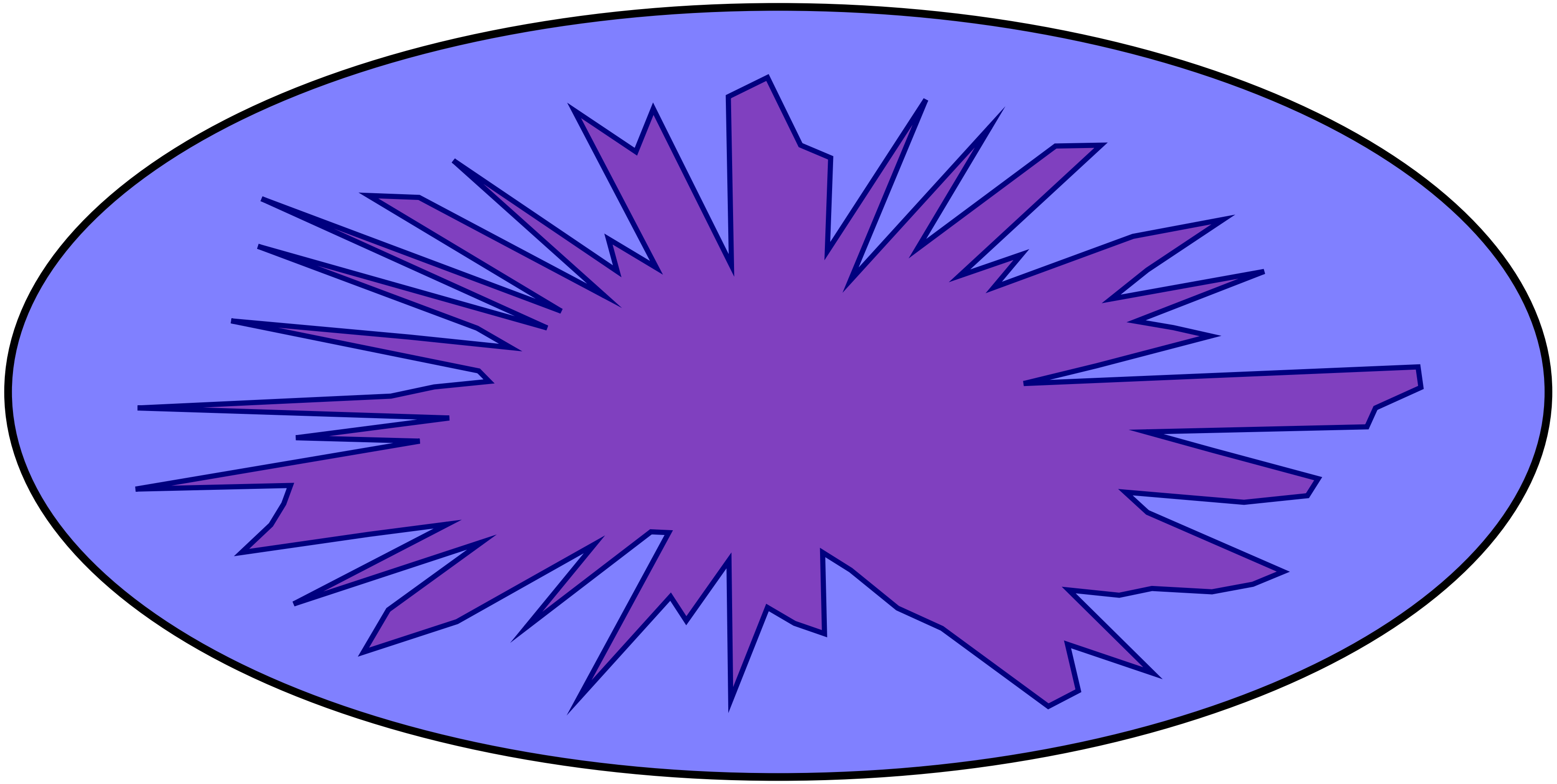
Overapproximation via simple grammars

- Modeling correct derivations requires complex rules



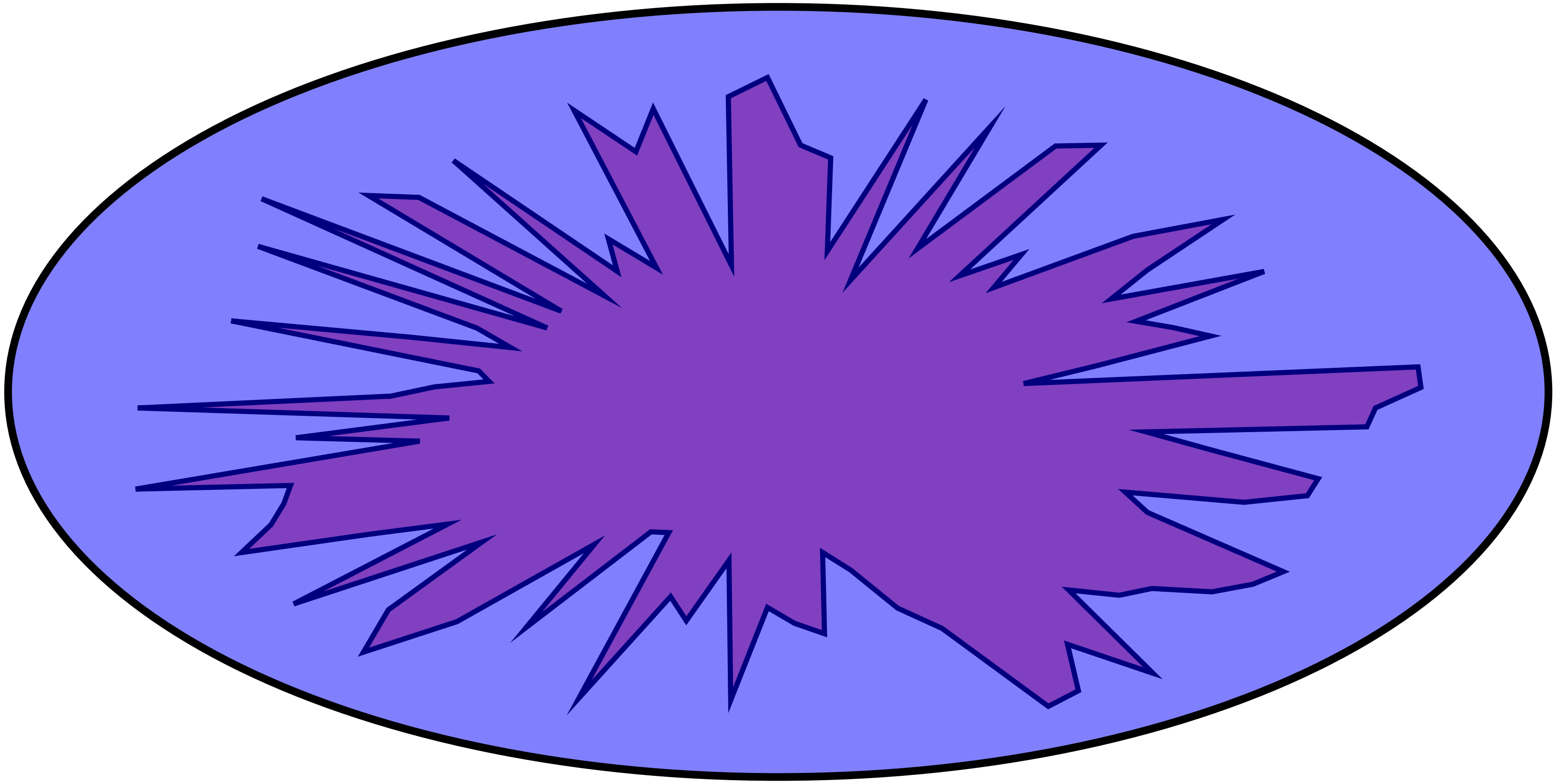
Overapproximation via simple grammars

- Modeling correct derivations requires complex rules
- Simple rules generate overapproximation of good derivations



Overapproximation via simple grammars

- Modeling correct derivations requires complex rules
- Simple rules generate overapproximation of good derivations



- Hard grammar rules $\Rightarrow$ soft/overlapping features

Many possible derivations!

$x =$ *people who have lived in Chicago*

Many possible derivations!

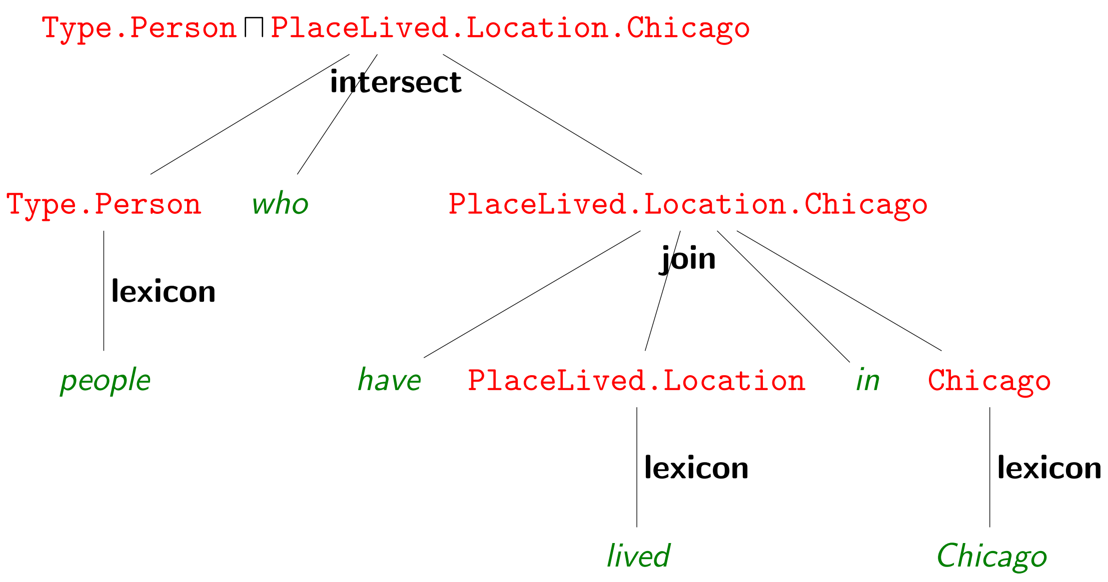
$x =$ *people who have lived in Chicago*

$\Downarrow$?

Many possible derivations!

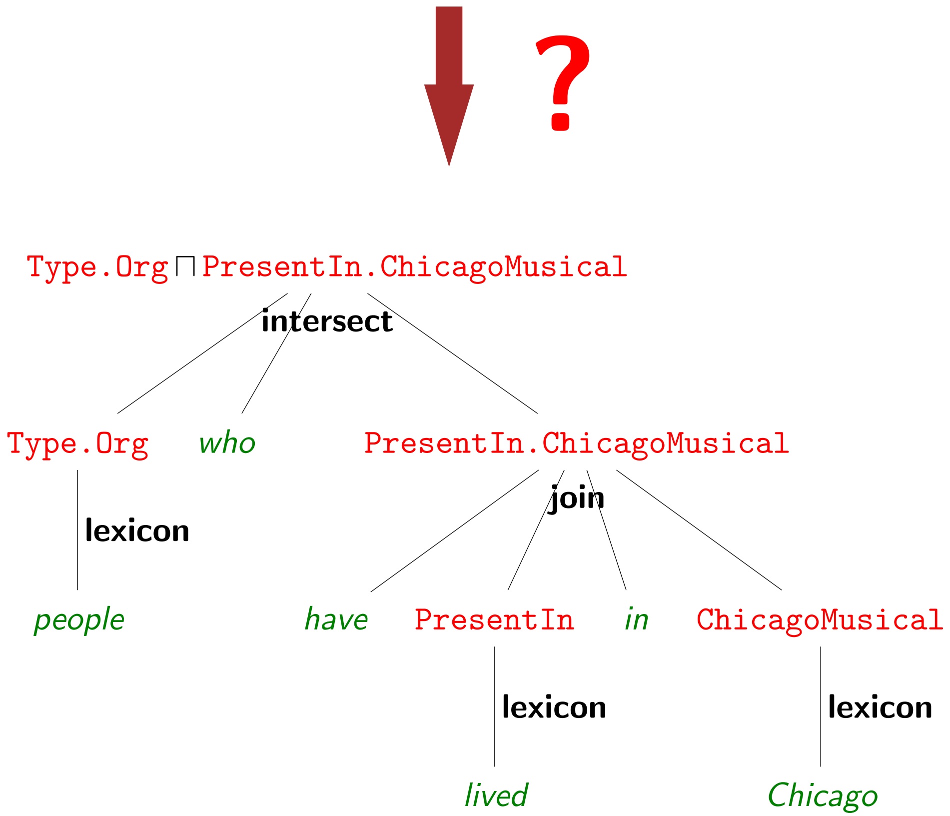
$x =$ *people who have lived in Chicago*

?

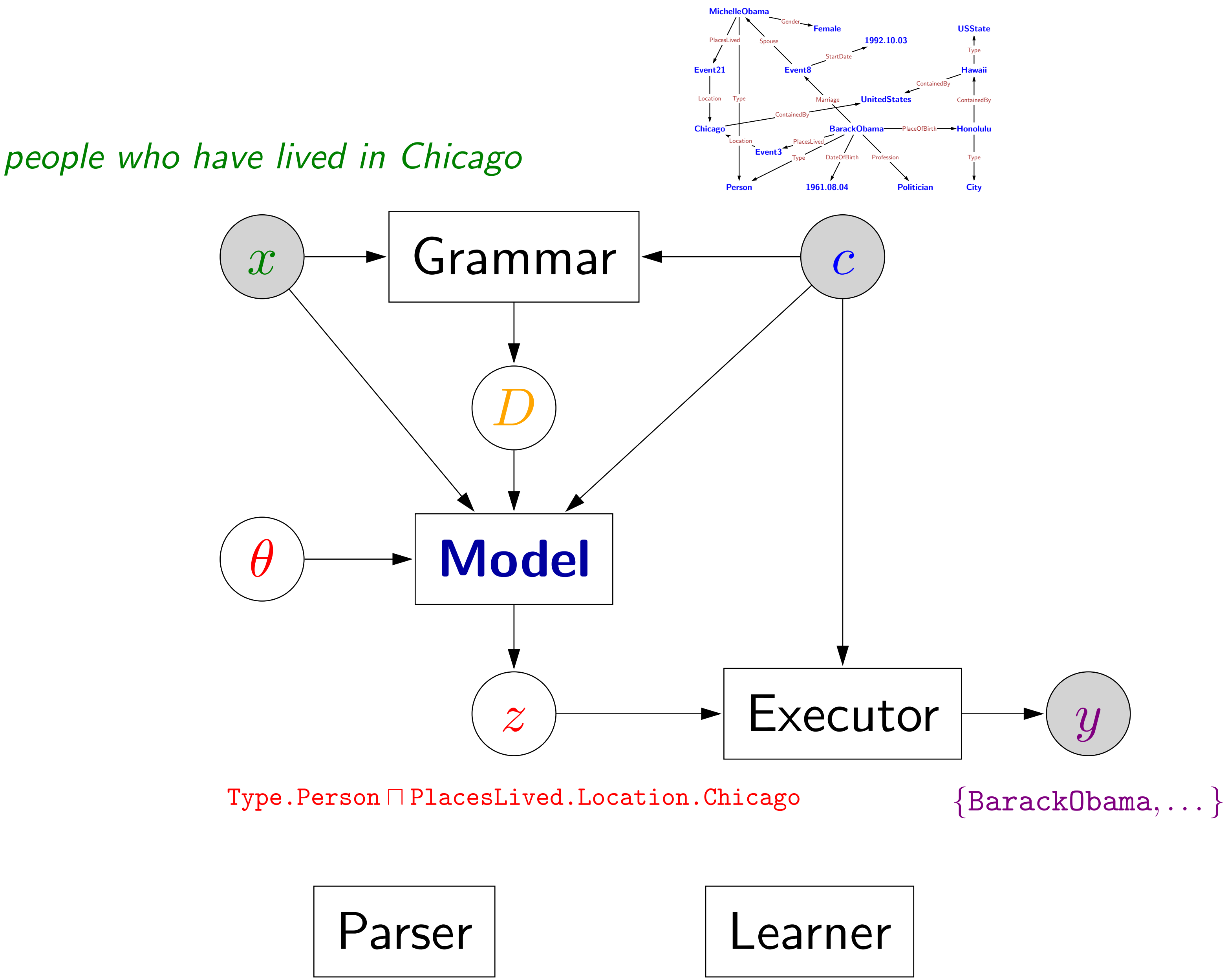


Many possible derivations!

$x =$ *people who have lived in Chicago*

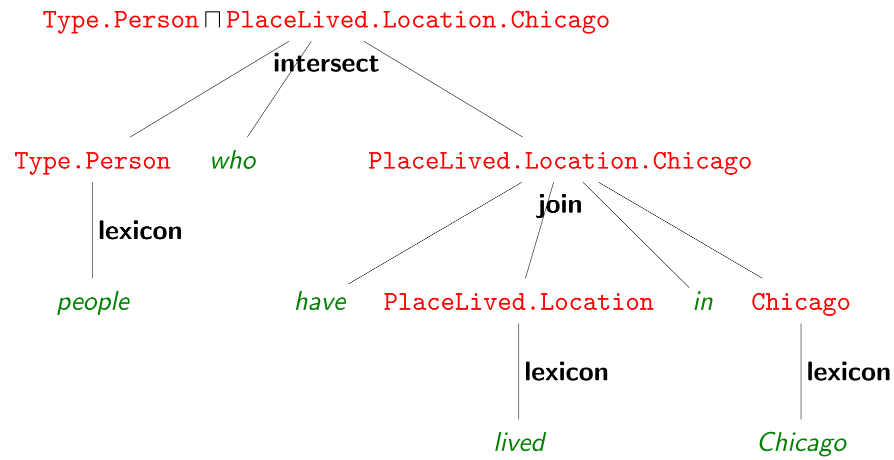


Components of a semantic parser

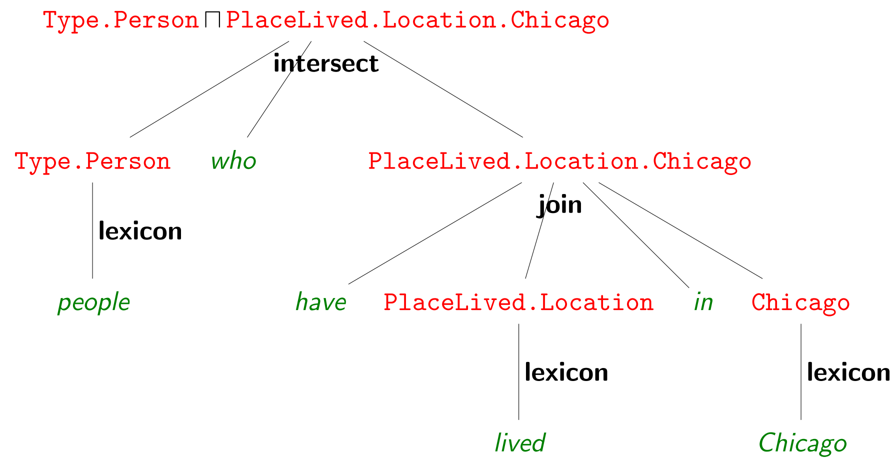


x: utterance

d: derivation



Feature vector $\phi(x, d) \in \mathbb{R}^F$:



x: utterance

d: derivation

Feature vector $\phi(x, d) \in \mathbb{R}^F$:

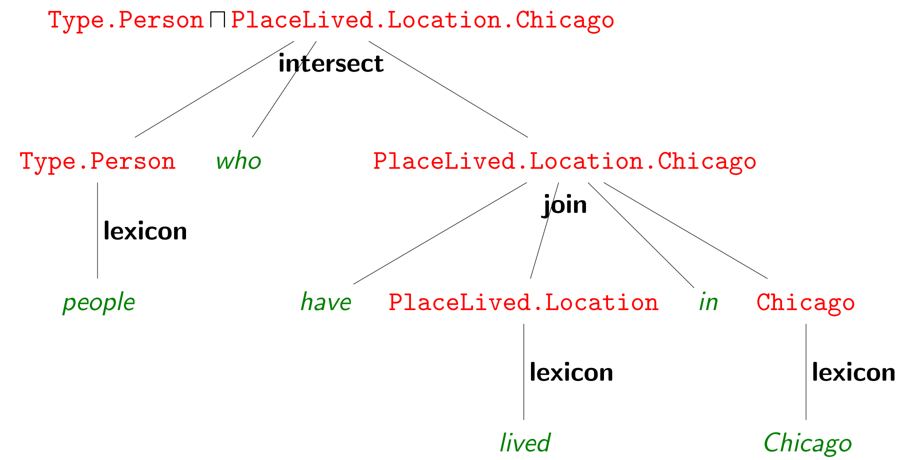
apply join	1
skipped IN	1
lived maps to `PlacesLived.Location`	1
...	...

Scoring function:

$$\text{Score}_\theta(x, d) = \phi(x, d) \cdot \theta$$

x: utterance

d: derivation



Feature vector $\phi(x, d) \in \mathbb{R}^F$:

apply join	1
skipped IN	1
lived maps to PlacesLived.Location	1
...	...

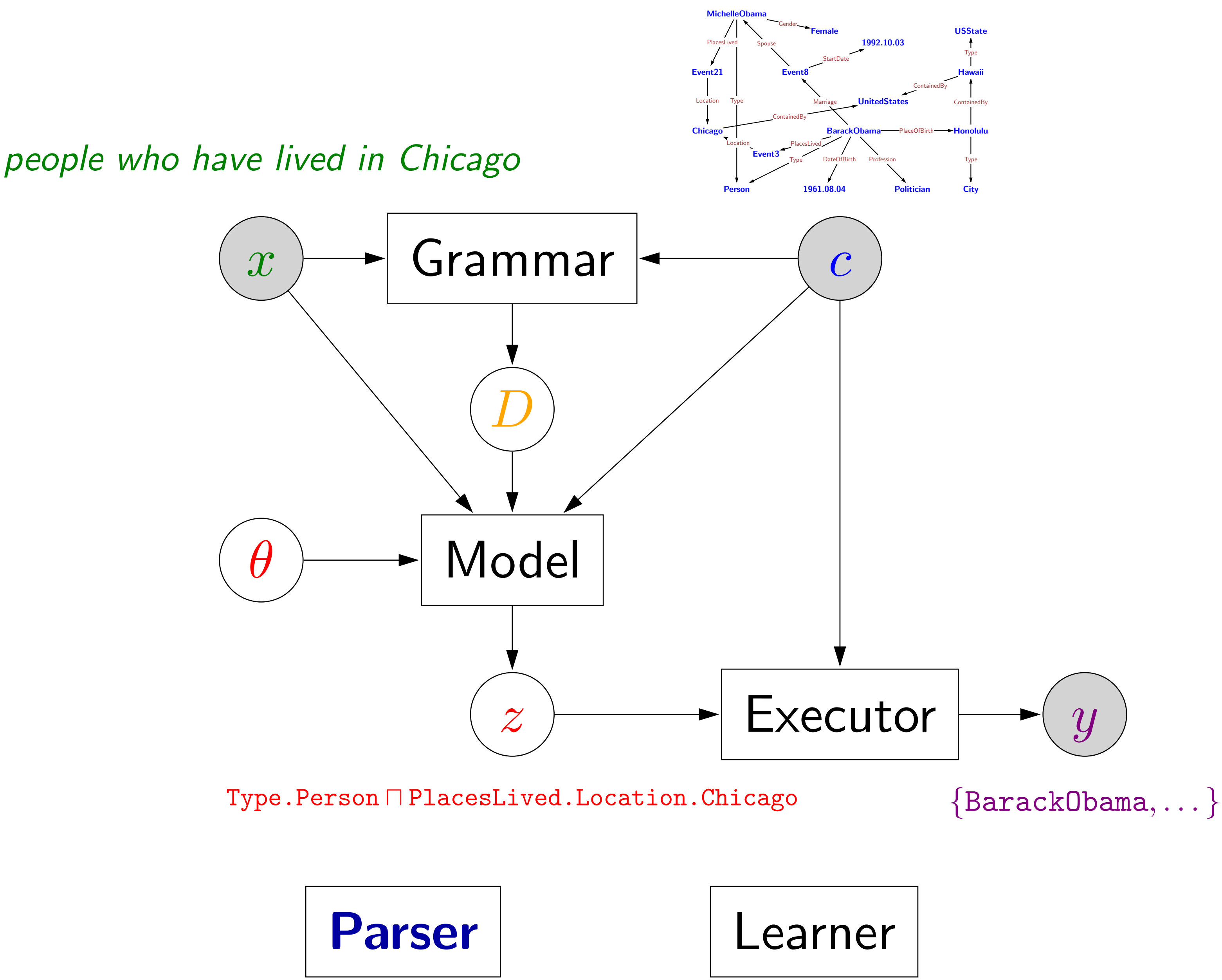
Scoring function:

$$\text{Score}_\theta(x, d) = \phi(x, d) \cdot \theta$$

Model:

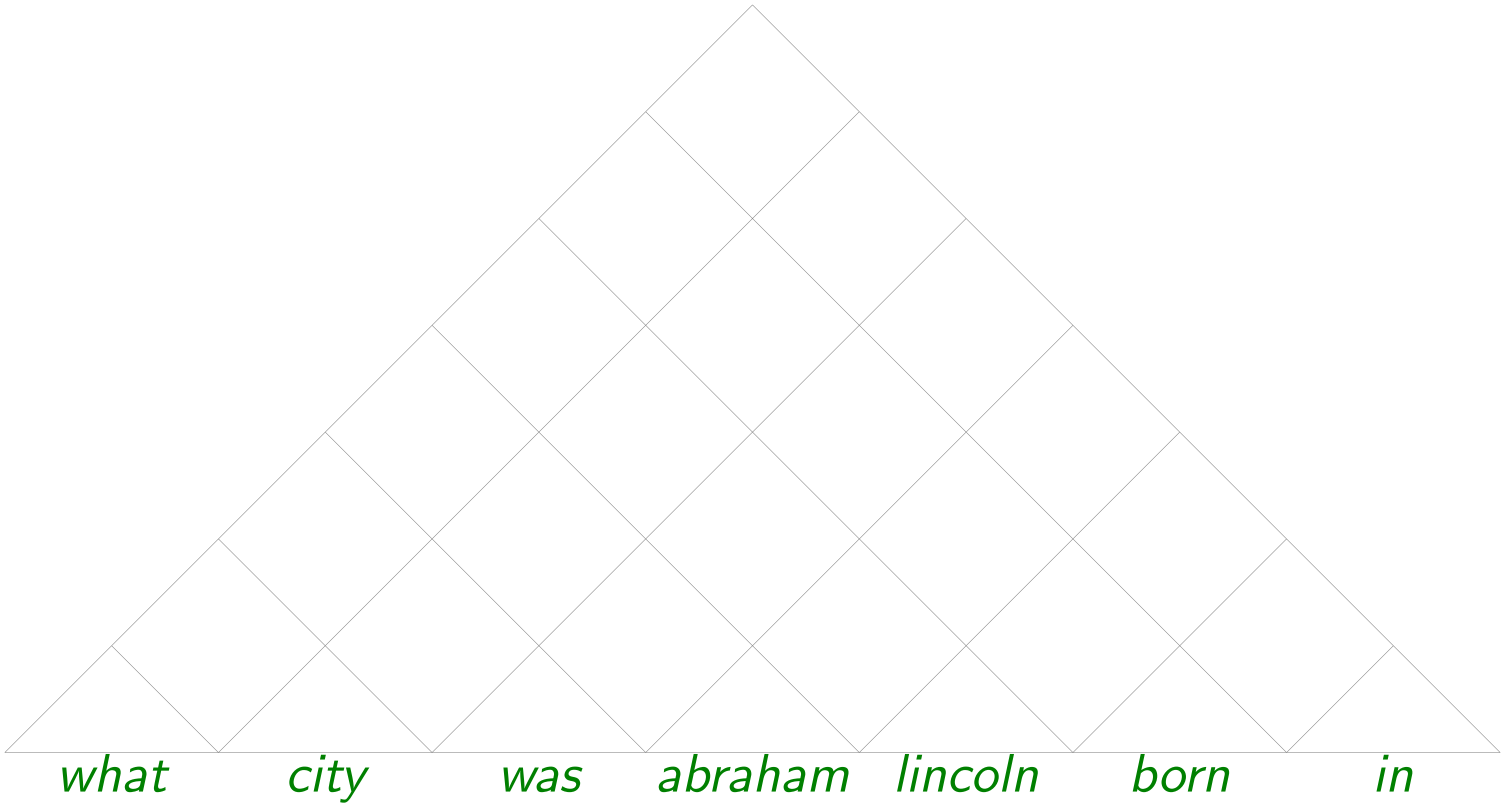
$$p(d \mid x, D, \theta) = \frac{\exp(\text{Score}_\theta(x,d))}{\sum_{d' \in D} \exp(\text{Score}_\theta(x,d'))}$$

Components of a semantic parser



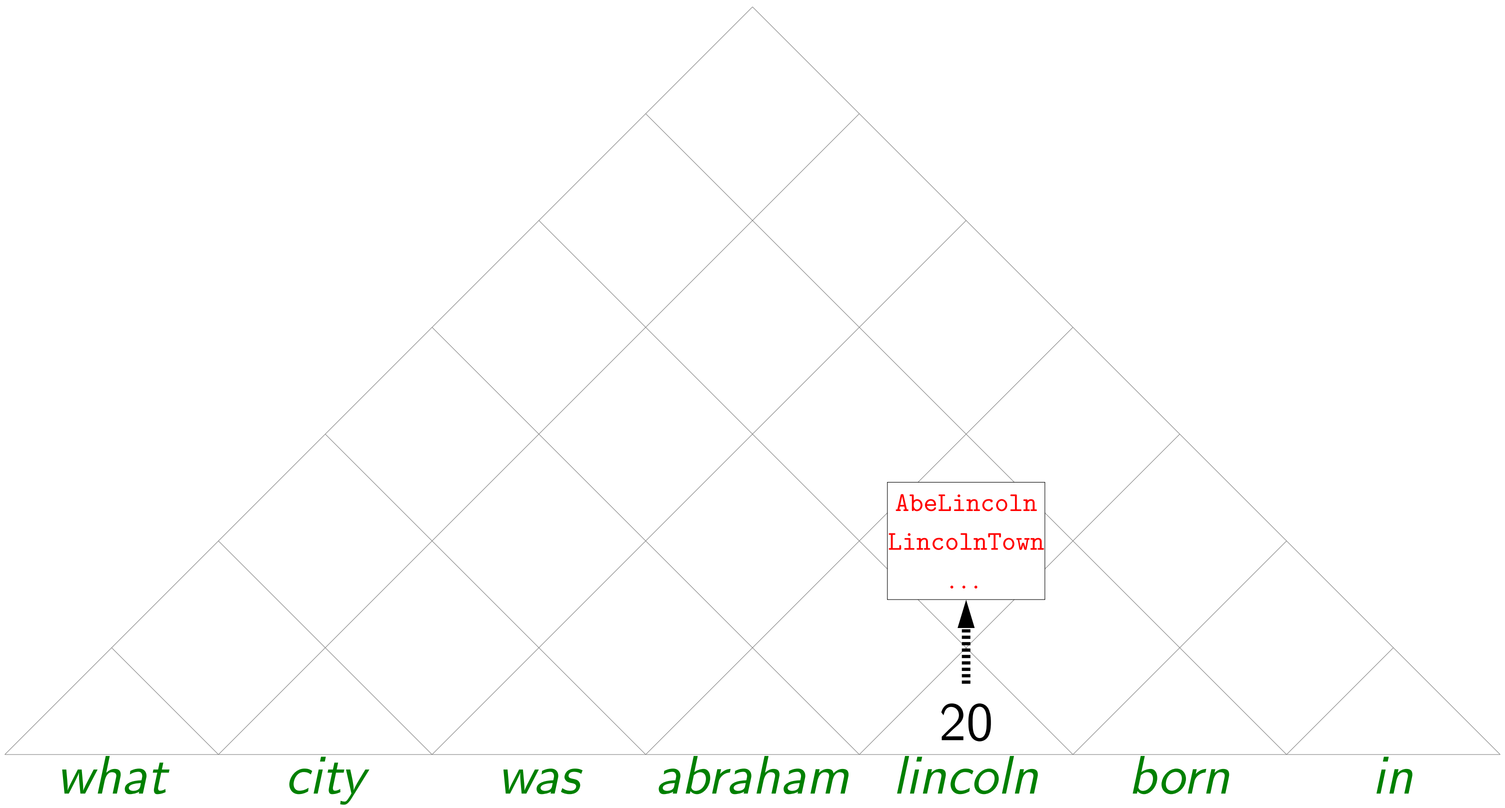
Parser

Goal: given grammar and model, enumerate derivations with high score



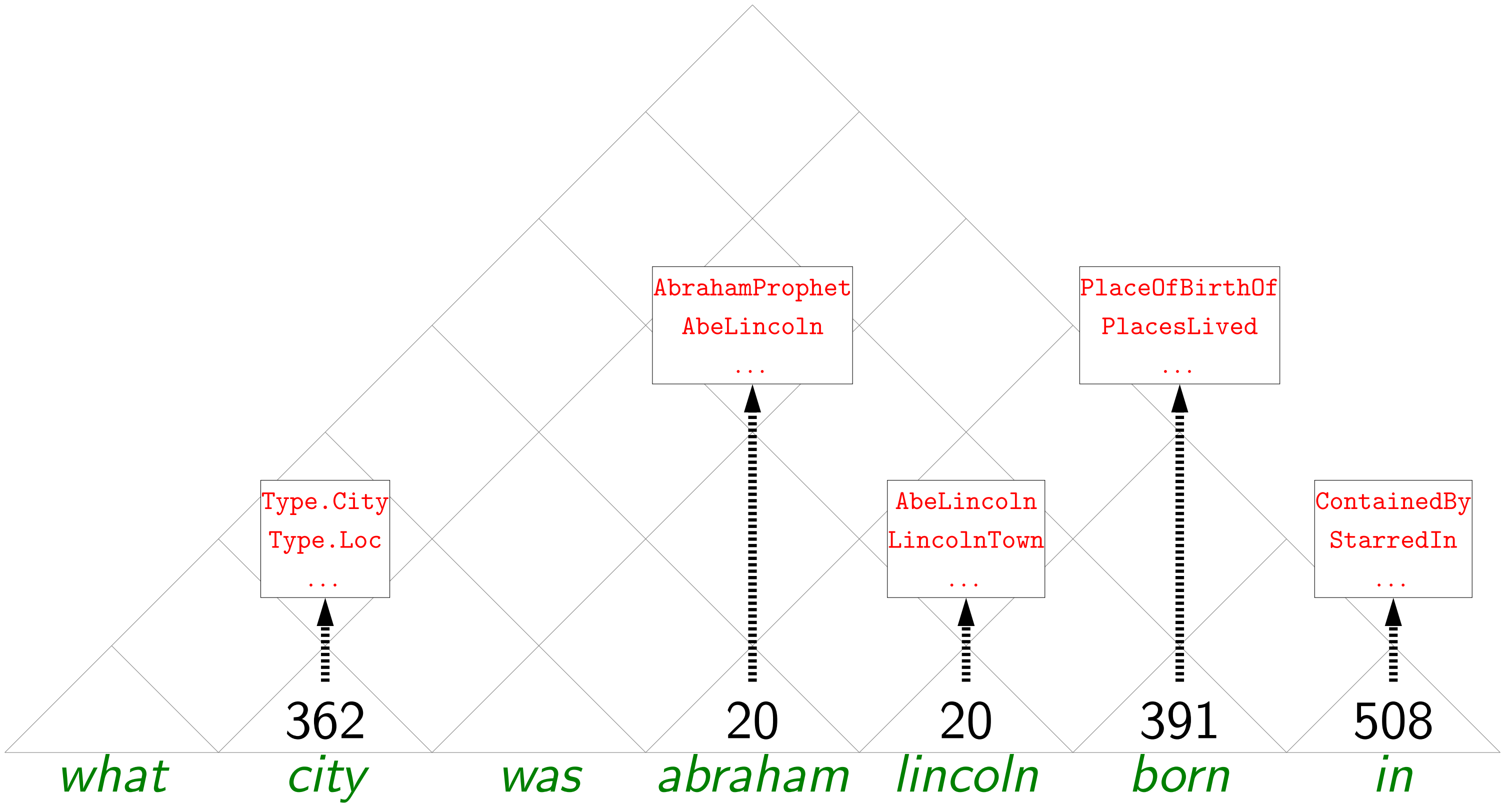
Parser

Goal: given grammar and model, enumerate derivations with high score



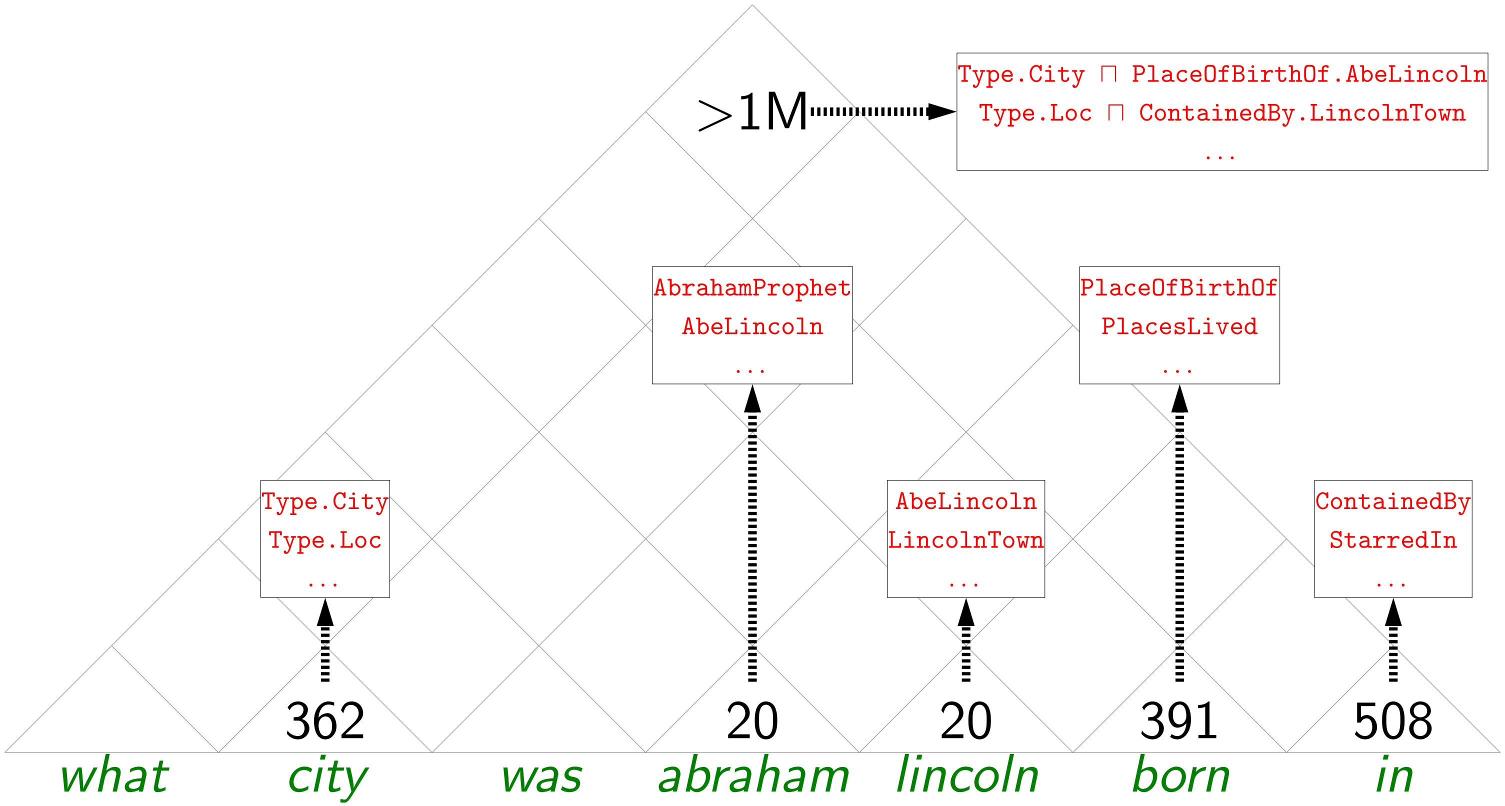
Parser

Goal: given grammar and model, enumerate derivations with high score



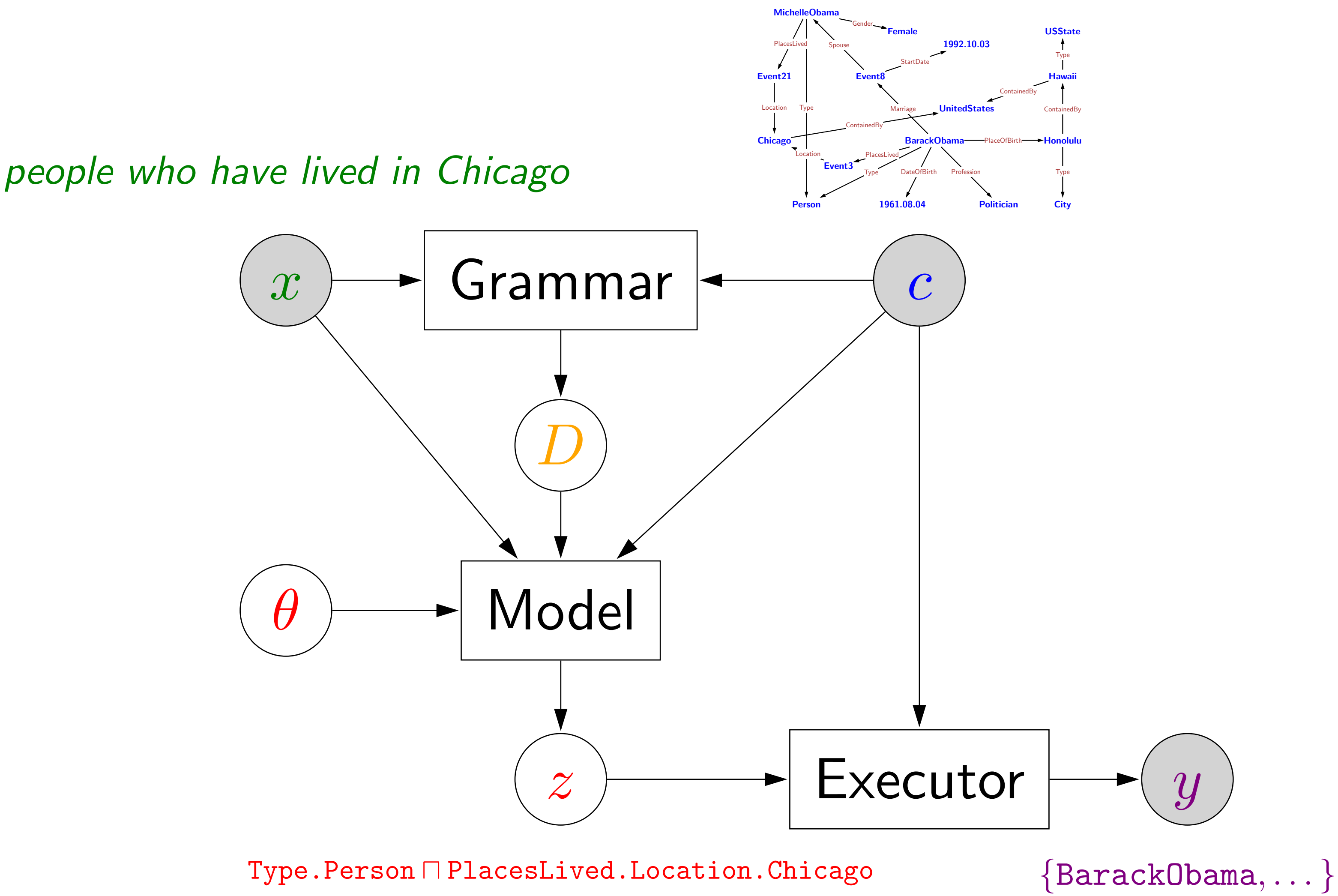
Parser

Goal: given grammar and model, enumerate derivations with high score



Use beam search: keep K derivations for each cell

Components of a semantic parser



Parser

Learner

[Zelle & Mooney, 1996; Zettlemoyer & Collins, 2005; Clarke et al. 2010; Liang et al., 2011]

Training data for semantic parsing

Heavy supervision

What's Bulgaria's capital?

`Capital.Bulgaria`

When was Walmart started?

`DateFounded.Walmart`

What movies has Tom Cruise been in?

`Type.Movie` $\sqcap$ `Starring.TomCruise`

...

[Zelle & Mooney, 1996; Zettlemoyer & Collins, 2005; Clarke et al. 2010; Liang et al., 2011]

Training data for semantic parsing

Heavy supervision

What's Bulgaria's capital?

`Capital.Bulgaria`

When was Walmart started?

`DateFounded.Walmart`

What movies has Tom Cruise been in?

`Type.Movie` $\sqcap$ `Starring.TomCruise`

...

Light supervision

What's Bulgaria's capital?

`Sofia`

When was Walmart started?

`1962`

What movies has Tom Cruise been in?

`TopGun,VanillaSky,...`

...

Training intuition

Where did Mozart tupress?

Vienna

Training intuition

Where did Mozart tupress?

`PlaceOfBirth.WolfgangMozart`

`PlaceOfDeath.WolfgangMozart`

`PlaceOfMarriage.WolfgangMozart`

Vienna

Training intuition

Where did Mozart tupress?

`PlaceOfBirth.WolfgangMozart`	$\Rightarrow$ Salzburg
`PlaceOfDeath.WolfgangMozart`	$\Rightarrow$ Vienna
`PlaceOfMarriage.WolfgangMozart`	$\Rightarrow$ Vienna

Vienna

Training intuition

Where did Mozart tupress?

~~`PlaceOfBirth.WolfgangMozart` $\Rightarrow$ Salzburg~~

`PlaceOfDeath.WolfgangMozart` $\Rightarrow$ Vienna

`PlaceOfMarriage.WolfgangMozart` $\Rightarrow$ Vienna

Vienna

Training intuition

Where did Mozart tupress?

~~`PlaceOfBirth.WolfgangMozart` ⇒ Salzburg~~

`PlaceOfDeath.WolfgangMozart` ⇒ Vienna

`PlaceOfMarriage.WolfgangMozart` ⇒ Vienna

Vienna

Where did Hogarth tupress?

Training intuition

Where did Mozart tupress?

~~`PlaceOfBirth.WolfgangMozart` $\Rightarrow$ Salzburg~~

`PlaceOfDeath.WolfgangMozart` $\Rightarrow$ Vienna

`PlaceOfMarriage.WolfgangMozart` $\Rightarrow$ Vienna

Vienna

Where did Hogarth tupress?

`PlaceOfBirth.WilliamHogarth`

`PlaceOfDeath.WilliamHogarth`

`PlaceOfMarriage.WilliamHogarth`

London

Training intuition

Where did Mozart tupress?

~~`PlaceOfBirth.WolfgangMozart`~~	~~$\Rightarrow$~~	~~Salzburg~~
`PlaceOfDeath.WolfgangMozart`	$\Rightarrow$	Vienna
`PlaceOfMarriage.WolfgangMozart`	$\Rightarrow$	Vienna

Vienna

Where did Hogarth tupress?

`PlaceOfBirth.WilliamHogarth`	$\Rightarrow$	London
`PlaceOfDeath.WilliamHogarth`	$\Rightarrow$	London
`PlaceOfMarriage.WilliamHogarth`	$\Rightarrow$	Paddington

London

Training intuition

Where did Mozart tupress?

$\cancel{\texttt{PlaceOfBirth.WolfgangMozart} \Rightarrow \text{Salzburg}}$

$\texttt{PlaceOfDeath.WolfgangMozart} \Rightarrow \text{Vienna}$

$\texttt{PlaceOfMarriage.WolfgangMozart} \Rightarrow \text{Vienna}$

Vienna

Where did Hogarth tupress?

$\texttt{PlaceOfBirth.WilliamHogarth} \Rightarrow \text{London}$

$\texttt{PlaceOfDeath.WilliamHogarth} \Rightarrow \text{London}$

$\cancel{\texttt{PlaceOfMarriage.WilliamHogarth} \Rightarrow \text{Paddington}}$

London

Training intuition

Where did Mozart tupress?

~~PlaceOfBirth.WolfgangMozart $\Rightarrow$ Salzburg~~

$\boxed{\texttt{PlaceOfDeath.WolfgangMozart} \Rightarrow \text{Vienna}}$

PlaceOfMarriage.WolfgangMozart $\Rightarrow$ Vienna

Vienna

Where did Hogarth tupress?

PlaceOfBirth.WilliamHogarth $\Rightarrow$ London

$\boxed{\texttt{PlaceOfDeath.WilliamHogarth} \Rightarrow \text{London}}$

~~PlaceOfMarriage.WilliamHogarth $\Rightarrow$ Paddington~~

London

Summary so far

- Two ideas: model theory and compositionality, both about factorization / **generalization**

- Modular framework: executor, grammar, model, parser, learner

- Applications: question answering, natural language interfaces to robots, programming by natural language

Food for thought

- Learning from denotations is hard; interaction between search (parsing) and learning: one improves the other — bootstrapping; don't have good formalism yet

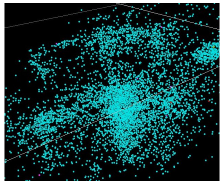
- Semantic parsing works on short sentences (user to computer); distributional/frame semantics has broader coverage; how to bridge the gap?

Food for thought

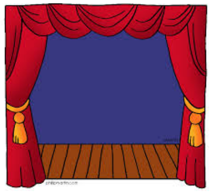
- Learning from denotations is hard; interaction between search (parsing) and learning: one improves the other — bootstrapping; don't have good formalism yet
- Semantic parsing works on short sentences (user to computer); distributional/frame semantics has broader coverage; how to bridge the gap?
- Really about end-to-end training (logical forms are means to an end), captures pragmatics
- What is the best way to produce answer (blur lines between parser and executor)?

Outline

Properties of language



Distributional semantics



Frame semantics

Model-theoretic semantics

Reflections

Three types of semantics

1. Distributional semantics:
    - Pro: Most broadly applicable, ML-friendly
    - Con: Monolithic representations

Three types of semantics

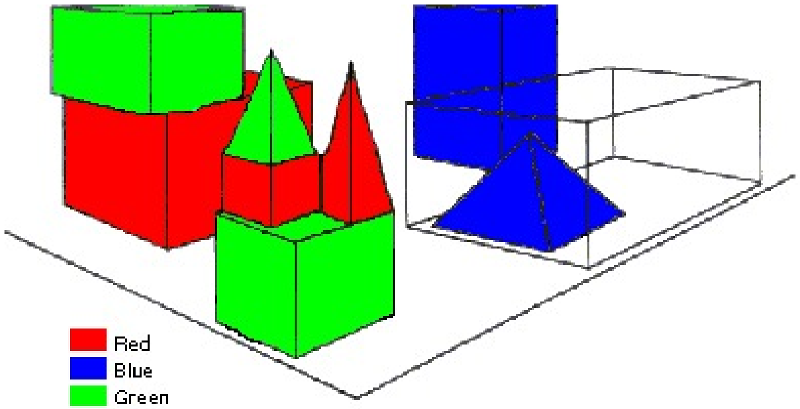
1. Distributional semantics:
   - Pro: Most broadly applicable, ML-friendly
   - Con: Monolithic representations
2. Frame semantics:
   - Pro: More structured representations
   - Con: Not full representation of world

Three types of semantics

1. Distributional semantics:
   - Pro: Most broadly applicable, ML-friendly
   - Con: Monolithic representations
2. Frame semantics:
   - Pro: More structured representations
   - Con: Not full representation of world
3. Model-theoretic semantics:
   - Pro: Full world representation, rich semantics, end-to-end
   - Con: Narrower in scope

many opportunities for synthesis

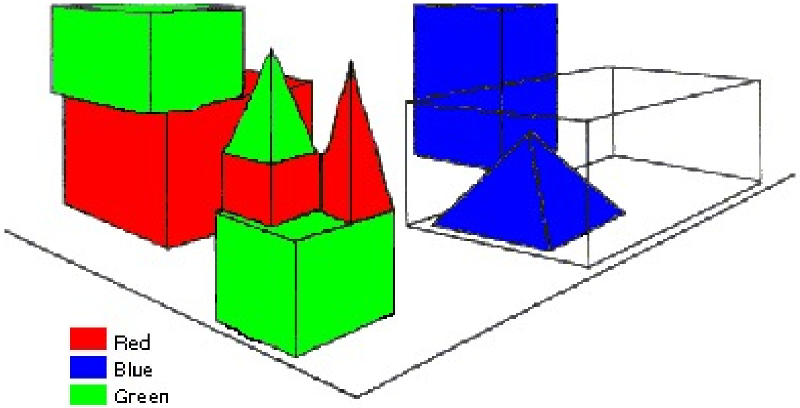
SHRDLU [1971]



SHRDLU [1971]

Person: Pick up a big red block.

Computer: OK.



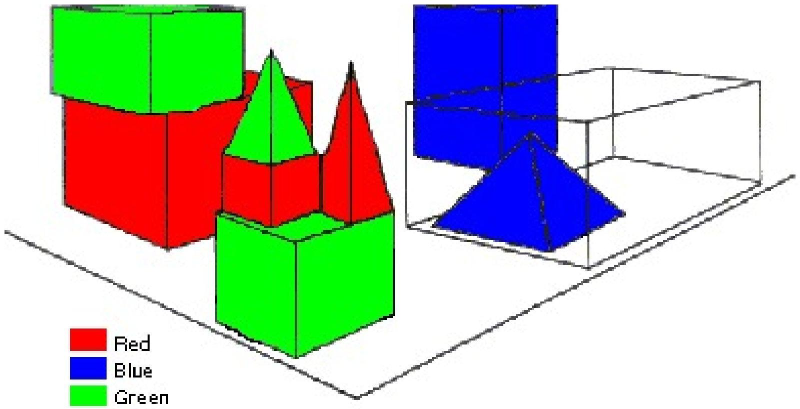
SHRDLU [1971]

Person: Pick up a big red block.

Computer: OK.

Person: Grasp the pyramid.

Computer: I don't understand which pyramid you mean.



SHRDLU [1971]

Person: Pick up a big red block.

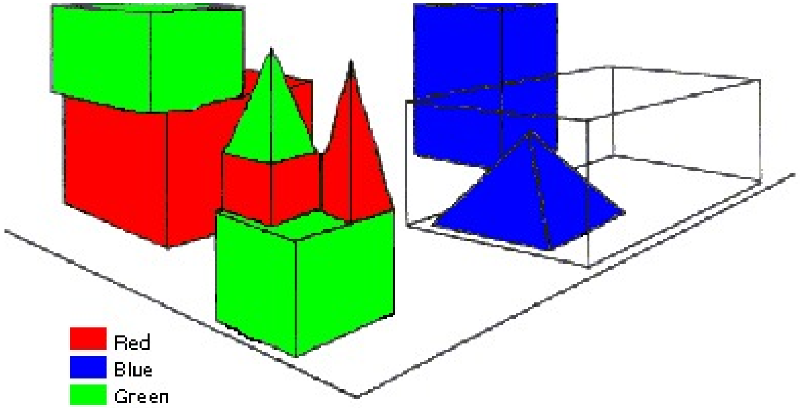
Computer: OK.

Person: Grasp the pyramid.

Computer: I don't understand which pyramid you mean.

Person (changing their mind): Find a block which is taller than the one you are holding and put it into the box.

Computer: By "it", I assume you mean the block which is taller than the one I am holding.



SHRDLU [1971]

Person: Pick up a big red block.

Computer: OK.

Person: Grasp the pyramid.

Computer: I don't understand which pyramid you mean.

Person (changing their mind): Find a block which is taller than the one you are holding and put it into the box.

Computer: By "it", I assume you mean the block which is taller than the one I am holding.

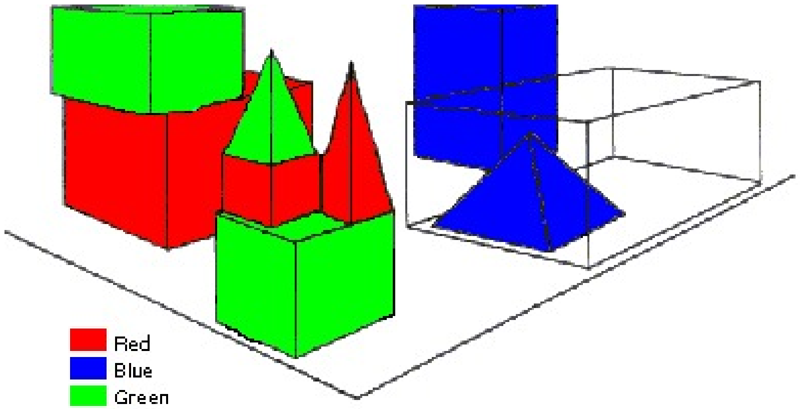
Computer: OK.

Person: What does the box contain?

Computer: The blue pyramid and the blue block.

Person: What is the pyramid supported by?

Computer: The box.



SHRDLU [1971]

Person: Pick up a big red block.

Computer: OK.

Person: Grasp the pyramid.

Computer: I don't understand which pyramid you mean.

Person (changing their mind): Find a block which is taller than the one you are holding and put it into the box.

Computer: By "it", I assume you mean the block which is taller than the one I am holding.

Computer: OK.

Person: What does the box contain?

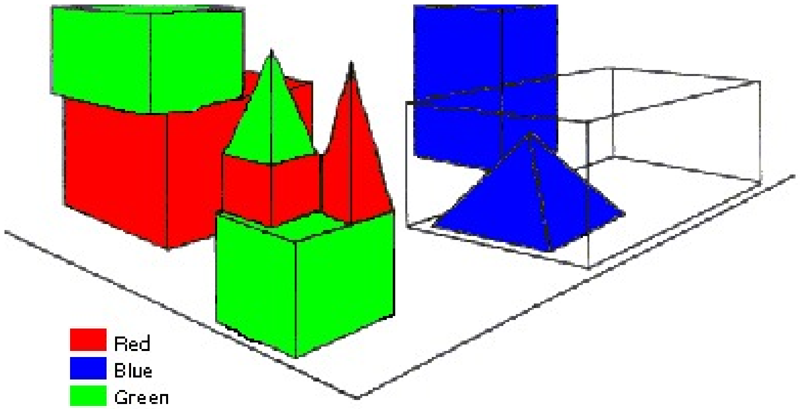
Computer: The blue pyramid and the blue block.

Person: What is the pyramid supported by?

Computer: The box.

- **End-to-end**

  (syntax, semantics, dialogue, planning)



The Complexity Barrier

A number of people have suggested to me that large programs like the SHRDLU program for understanding natural language represent a kind of dead end in AI programming. Complex interactions between its components give the program much of its power, but at the same time they present a formidable obstacle to understanding and extending it. In order to grasp any part, it is necessary to understand how it fits with other parts, presents a dense mass, with no easy footholds. Even having written the program, I find it near the limit of what I can keep in mind at once.

— Terry Winograd (1972)

Memory networks [2014]

Goal: learn to do reasoning tasks **end-to-end** from scratch

John is in the playground.

Bob is in the office.

John picked up the football.

Bob went to the kitchen.

Where is the football? A:playground

[Weston/Chopra/Bordes, 2014]

Memory networks [2014]

Goal: learn to do reasoning tasks **end-to-end** from scratch

John is in the playground.
Bob is in the office.
John picked up the football.
Bob went to the kitchen.
Where is the football? A:playground

- Pure learning based, so much simpler than SHRDLU (+)
- Currently using artificial data, simpler than SHRDLU (-)

Memory networks [2014]

Goal: learn to do reasoning tasks **end-to-end** from scratch

John is in the playground.

Bob is in the office.

John picked up the football.

Bob went to the kitchen.

Where is the football? A:playground

- Pure learning based, so much simpler than SHRDLU (+)
- Currently using artificial data, simpler than SHRDLU (-)
- How to get **real data** and how much do we need to get to SHRDLU level?
- Can the model incorporate some **structure** without getting too complex?

The future

Instead of trying to produce a programme to simulate the adult mind, why not rather try to produce one which simulates the child's?

The future

Instead of trying to produce a programme to simulate the adult mind, why not rather try to produce one which simulates the child's?

It can also be maintained that it is best to provide the machine with the best sense organs that money can buy, and then teach it to understand and speak English. This process could follow the normal teaching of a child. Things would be pointed out and named, etc.

The future

Instead of trying to produce a programme to simulate the adult mind, why not rather try to produce one which simulates the child's?

It can also be maintained that it is best to provide the machine with the best sense organs that money can buy, and then teach it to understand and speak English. This process could follow the normal teaching of a child. Things would be pointed out and named, etc.

— Alan Turing (1950)

Questions?