Parsing Time

Gabor Angeli
angeli@stanford.edu

with Chris Manning and Dan Jurafsky

Example

Don't worry, June 5 is next week – there's still a few days

Example

Don't worry, **June 5** *is* **next week** – *there's still* **a few days**

Detection Finding temporal *phrases* in a sentence.

Example

Don't worry, **June 5** *is* **next week** – *there's still* **a few days**

Detection Finding temporal *phrases* in a sentence.

Interpretation Finding the grounded *meaning* of a phrase

Example

Don't worry, **June 5** *is* **next week** – *there's still* **a few days**

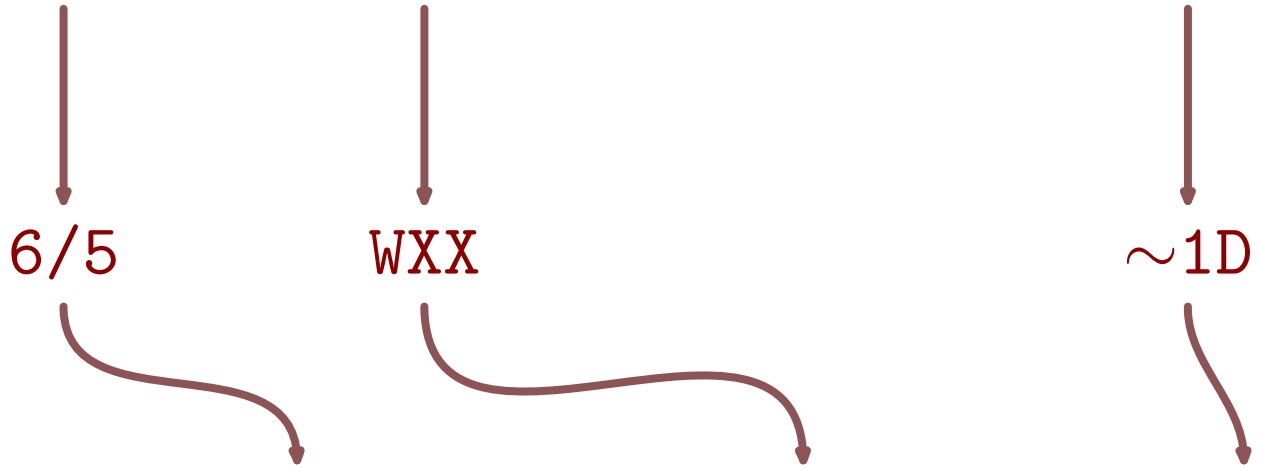
Detection Finding temporal *phrases* in a sentence.

Interpretation Finding the grounded *meaning* of a phrase

But, often incomplete information

Example

Don't worry, **June 5** *is* **next week** – *there's still* **a few days**



[2012-06-02] 6/05/12 6/03/12-6/09/12 ∼1D

Detection Finding temporal *phrases* in a sentence.

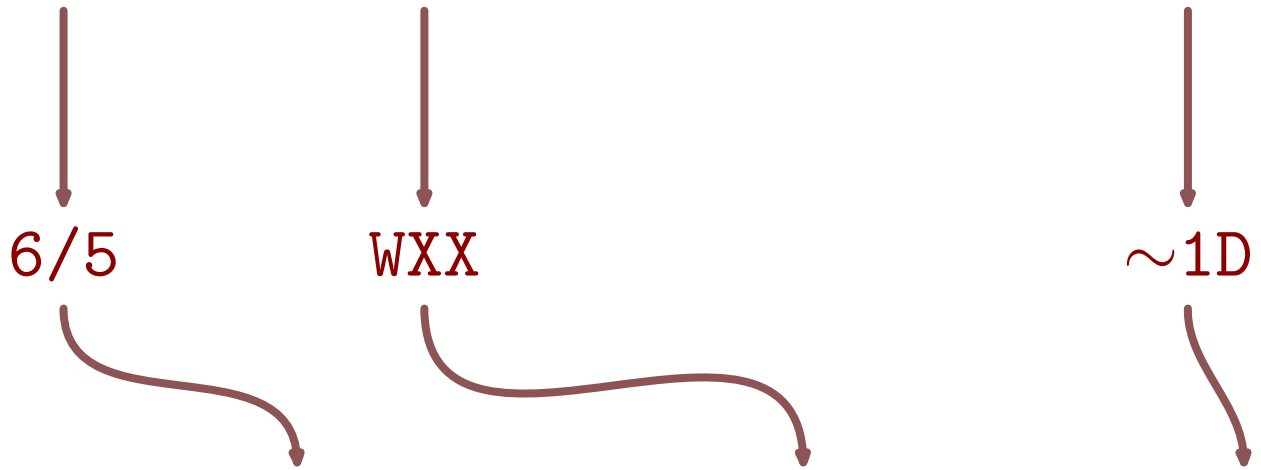
Interpretation Finding the grounded *meaning* of a phrase

But, often incomplete information

Incorporate a `reference time`

Example

Don't worry, **June 5** *is* **next week** – *there's still* **a few days**



[2012-06-02] 6/05/12 6/10/12-6/16/12 ∼1D

Detection Finding temporal *phrases* in a sentence.

Interpretation Finding the grounded *meaning* of a phrase

But, often incomplete information

Incorporate a `reference time`

Sneak Peek: ambiguity in *next week*

Time In Information Extraction

News

Beginning more than **seven hours earlier**, *the space station's robotic arm detached the 14-foot long Dragon [spacecraft]*

Benjamin Franklin Federal Savings and Loan Association said it plans to restructure in the wake of a **third-quarter** *loss*

Time In Information Extraction

News

Beginning more than **seven hours earlier**, *the space station's robotic arm detached the 14-foot long Dragon [spacecraft]*

Benjamin Franklin Federal Savings and Loan Association said it plans to restructure in the wake of a **third-quarter** *loss*

Communication

Actually I am on vacation the **last three weeks of November**

I have some time available at the **end of next week**

Motivation

Current approaches

Collection of hand-coded rules

Mani & Wilson 2000; Strötgen and Gertz 2010; Chang and Manning (2012)

Motivation

Current approaches

Collection of hand-coded rules

Downsides

Rigid

Syntax: *last Friday the 13th*

Mani & Wilson 2000; Strötgen and Gertz 2010; Chang and Manning (2012)

Motivation

Current approaches

Collection of hand-coded rules

Downsides

Rigid

Syntax: *last Friday the 13th*

Pragmatics: when is *last Sunday*?

Mani & Wilson 2000; Strötgen and Gertz 2010; Chang and Manning (2012)

Motivation

Current approaches

Collection of hand-coded rules

Downsides

Rigid

Syntax: *last Friday the 13th*

Pragmatics: when is *last Sunday*?

Rule engineering challenge

Always more rules: *7 days* **prior**, *the* **previous** *7 days*

```
/the/ /past|last/ (?: ($NUM) /to|-/ )? ($NUM)? ($TEUNITS)
/the/ /next|following/ (?: ($NUM) /to|-/ )? ($NUM)? ($TEUNITS)
/another/ (?: ($NUM) /to|-/ )? ($NUM)? ($TEUNITS)
...
```

Mani & Wilson 2000; Strötgen and Gertz 2010; Chang and Manning (2012)

System

Input (w,t) (*phrase* , `reference`)

System

Input (w,t) (*Last Friday the 13th* , 2012-06-05)

System

Input (w,t) (*Last Friday the 13th* , 2012-06-05)

Output τ^* `normalized time`

System

Input (w,t) (*Last Friday the 13th* , 2012-06-05)

Output τ^* 2012-01-13

System

Input (w,t) (*Last Friday the 13th* , 2012-06-05)

Latent parse R

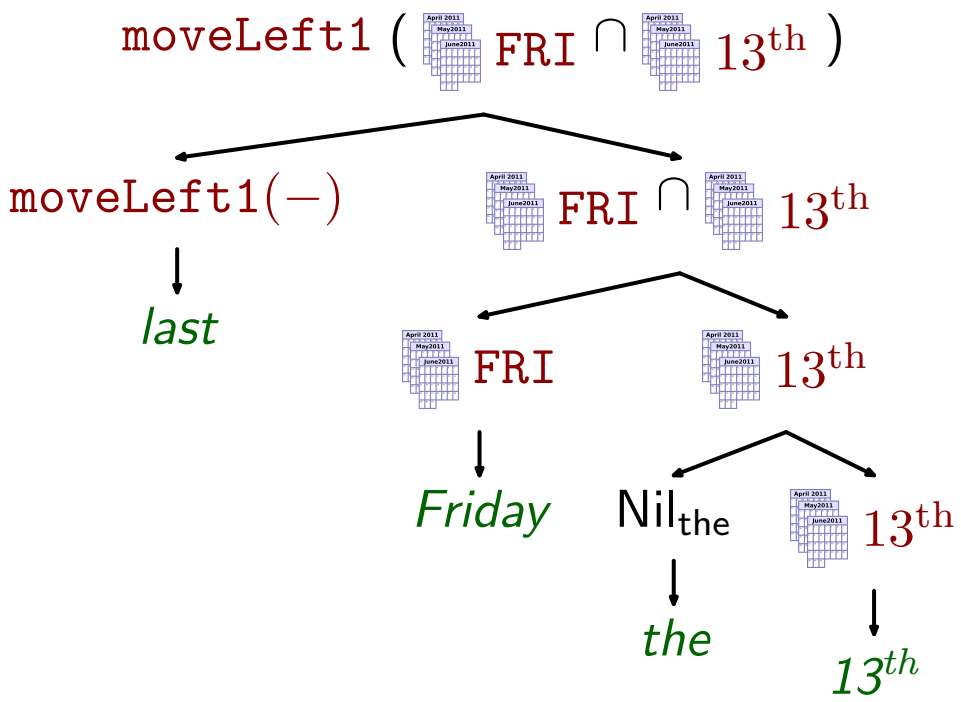
Output τ^* 2012-01-13

System

Input (w,t) (*Last Friday the 13th* , 2012-06-05)

↓

Latent parse R



↓

Output τ^* 2012-01-13

Motivation

Our Model

Gives informative probability

Motivation

Our Model

Gives informative probability

Handle *syntactic* ambiguity

[Last Friday] [the 13th] and *[last] [Friday the 13th]*

Motivation

Our Model

Gives informative probability

Handle *syntactic* ambiguity

[Last Friday] [the 13th] and *[last] [Friday the 13th]*

Handle *pragmatic* ambiguity

Could we meet on Tuesday?

Motivation

Our Model

Gives informative probability

Handle *syntactic* ambiguity

[Last Friday] [the 13th] and *[last] [Friday the 13th]*

Handle *pragmatic* ambiguity

Could we meet on Tuesday?

Plays nice in larger systems

Motivation

Our Model

Gives informative probability

Handle *syntactic* ambiguity

[Last Friday] [the 13th] and *[last] [Friday the 13th]*

Handle *pragmatic* ambiguity

Could we meet on Tuesday?

Plays nice in larger systems

Elegant representation of time

Motivation

Our Model

Gives informative probability

Handle *syntactic* ambiguity

[Last Friday] [the 13th] and *[last] [Friday the 13th]*

Handle *pragmatic* ambiguity

Could we meet on Tuesday?

Plays nice in larger systems

Elegant representation of time

Language/domain flexible

Learn any structure seen in training

Motivation

Our Model

Gives informative probability

Handle *syntactic* ambiguity

[Last Friday] [the 13th] and *[last] [Friday the 13th]*

Handle *pragmatic* ambiguity

Could we meet on Tuesday?

Plays nice in larger systems

Elegant representation of time

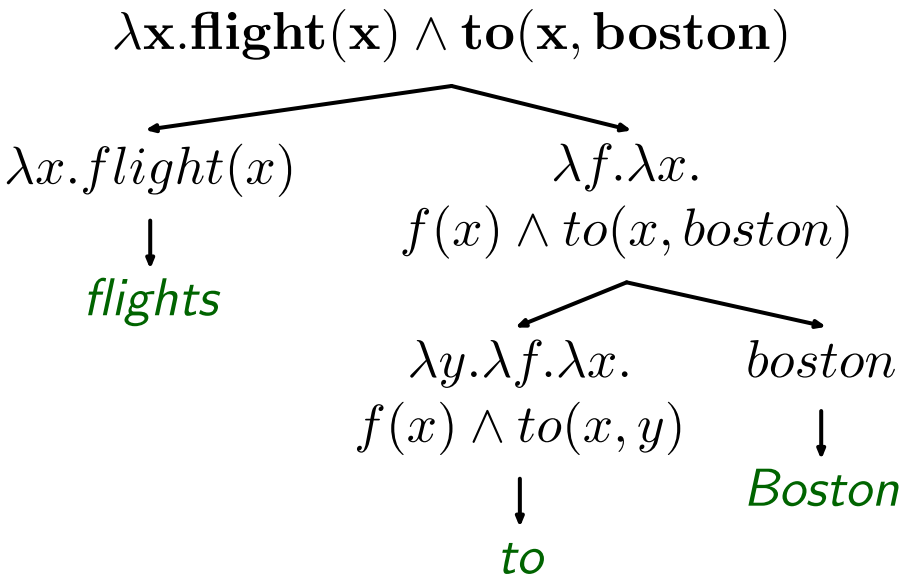
Language/domain flexible

Learn any structure seen in training

Last *Sunday* and *domingo* **pasado**

Comparison To Lambda Calculus

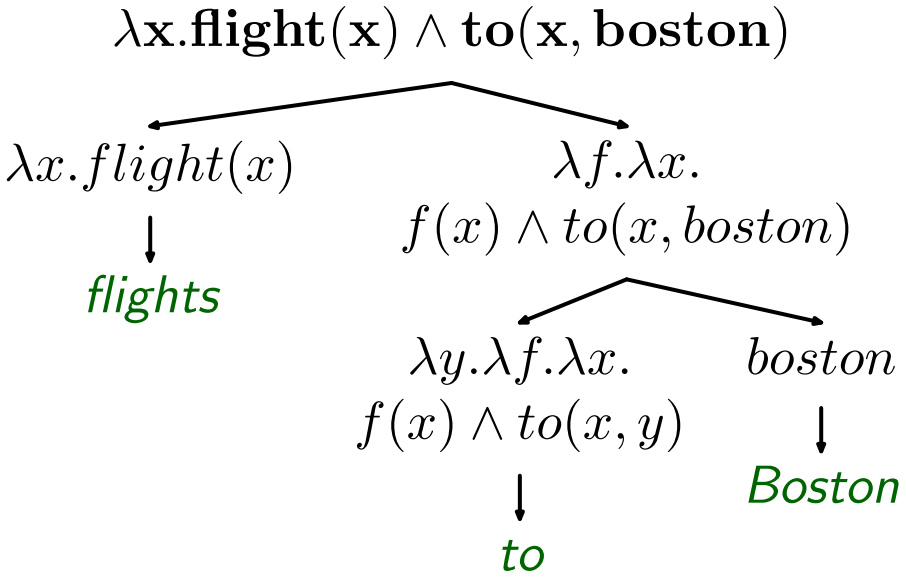
Parallels to semantic parsing

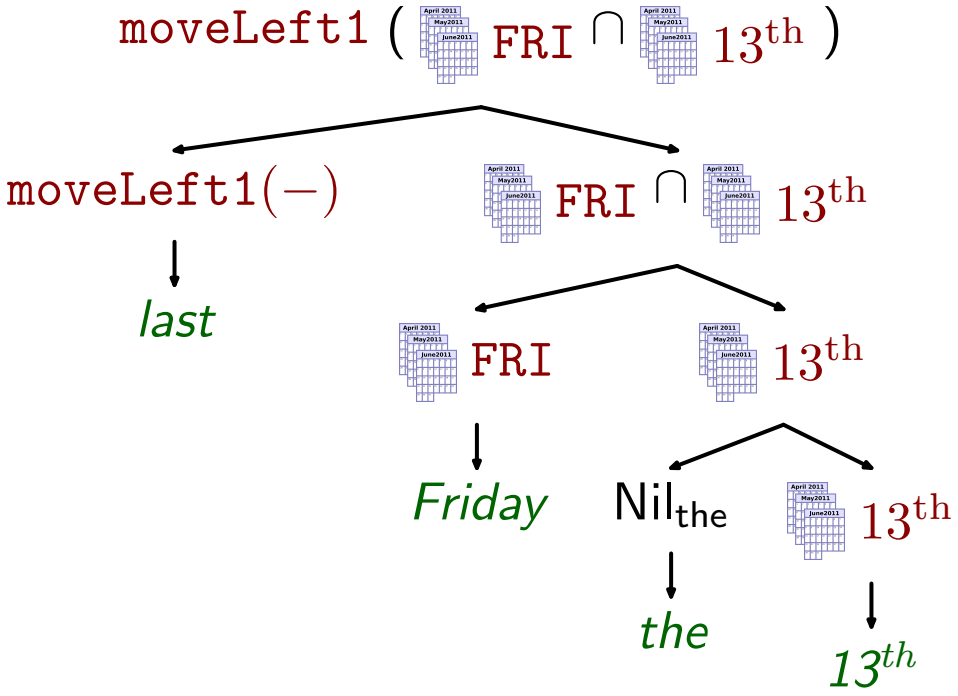


Zettlemoyer & Collins 2005/2007; Liang et al. 2011

Comparison To Lambda Calculus

Parallels to semantic parsing

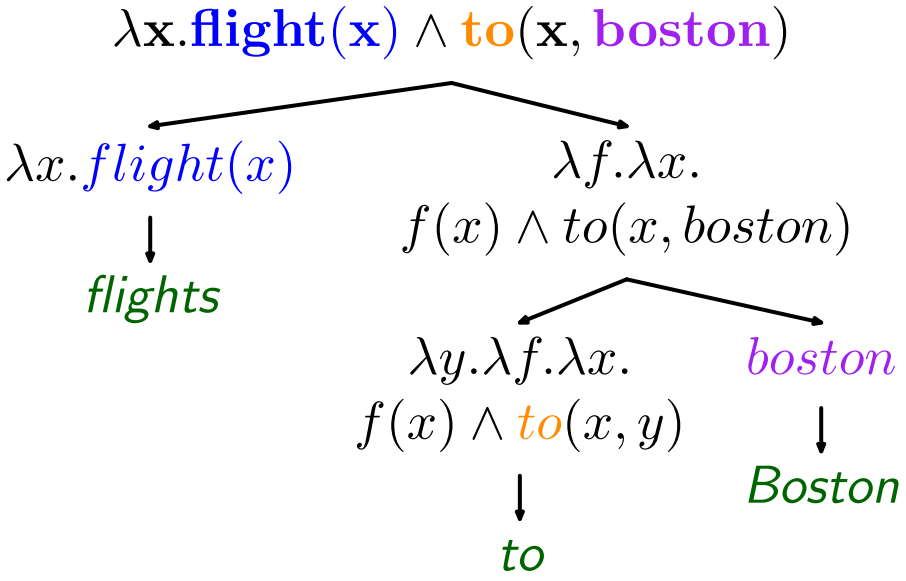


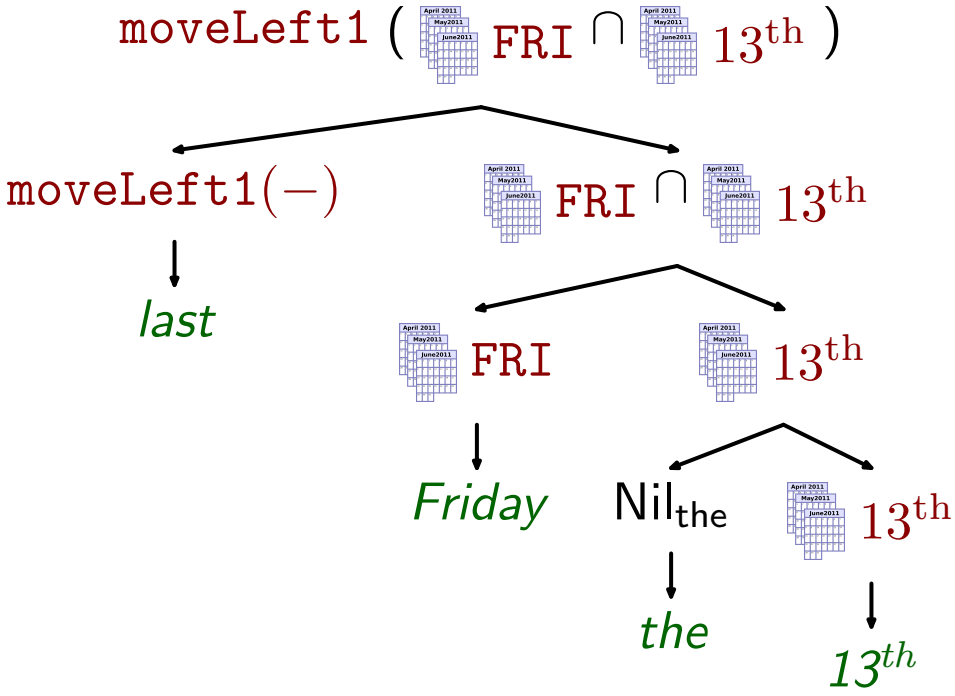


Zettlemoyer & Collins 2005/2007; Liang et al. 2011

Comparison To Lambda Calculus

Option 1: bootstrap from parse

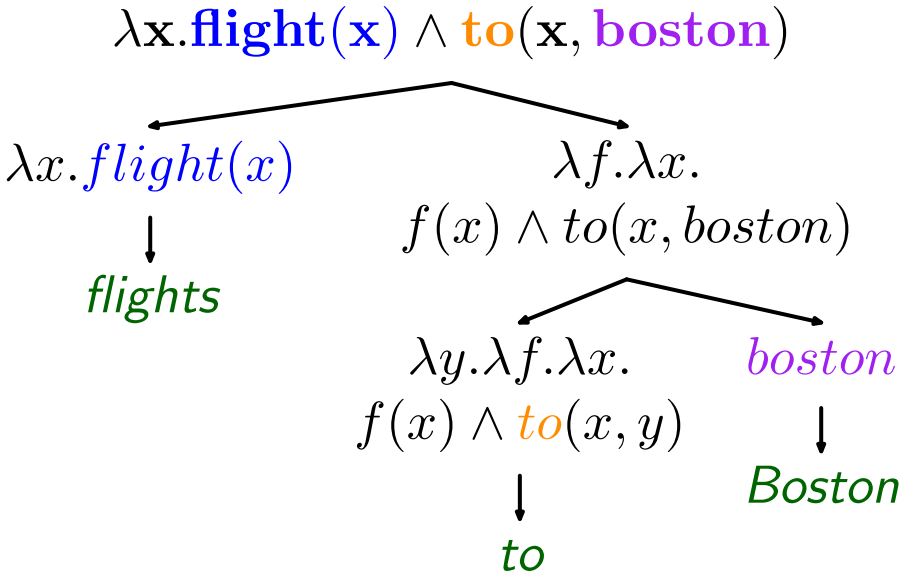


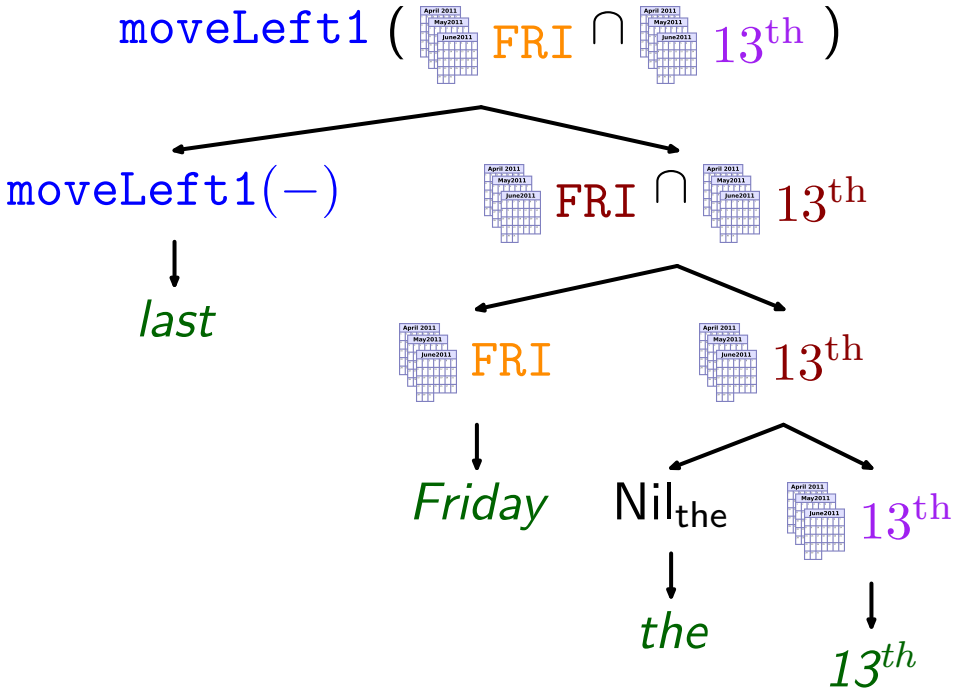


Zettlemoyer & Collins 2005/2007; Liang et al. 2011

Comparison To Lambda Calculus

Option 1: bootstrap from parse

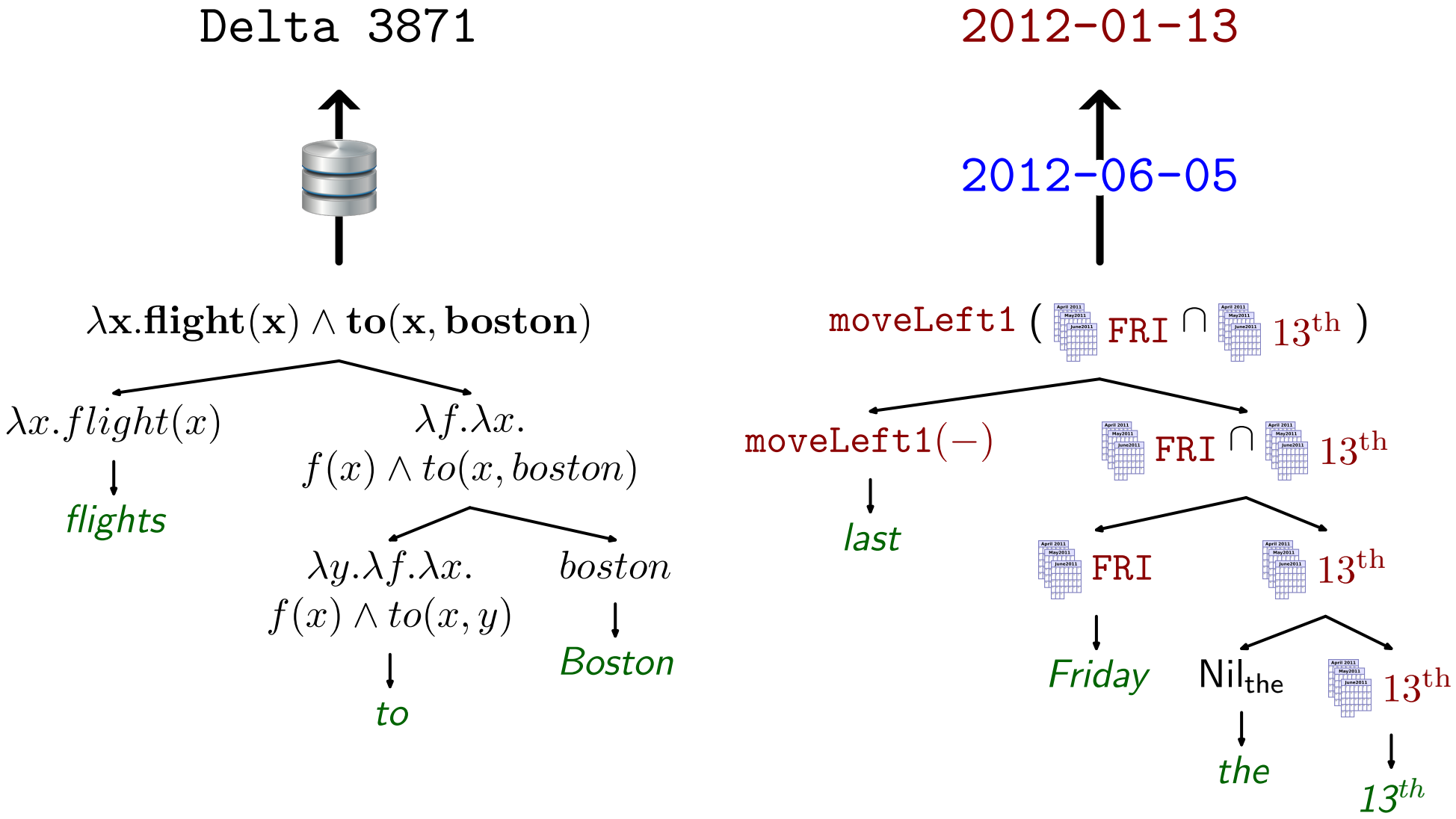




Zettlemoyer & Collins 2005/2007; Liang et al. 2011

Comparison To Lambda Calculus

Option 2: bootstrap from grounded interpretation



Zettlemoyer & Collins 2005/2007; Liang et al. 2011

Latent Parse

Expressions parse compositionally

Latent Parse

Expressions parse compositionally

last 2 days

Latent Parse

Expressions parse compositionally

last 2 days

last

⏳ DAY

2

↓

days

Latent Parse

Expressions parse compositionally

last 2 days

last

2

↓

2

↓

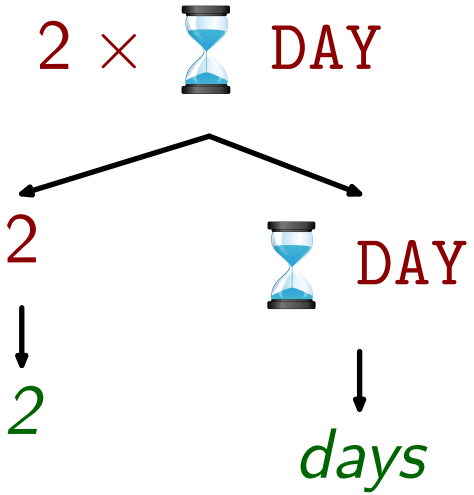
days

Latent Parse

Expressions parse compositionally

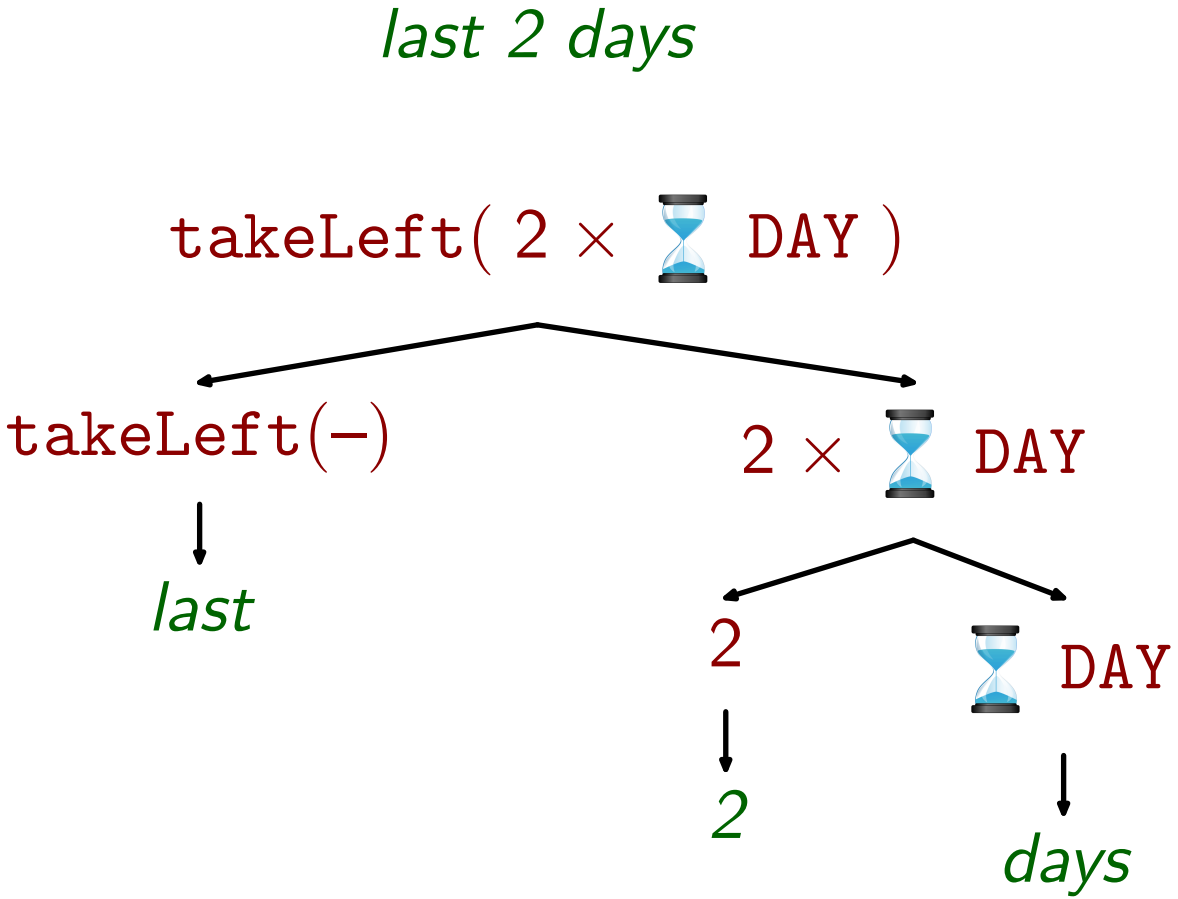
last 2 days

last



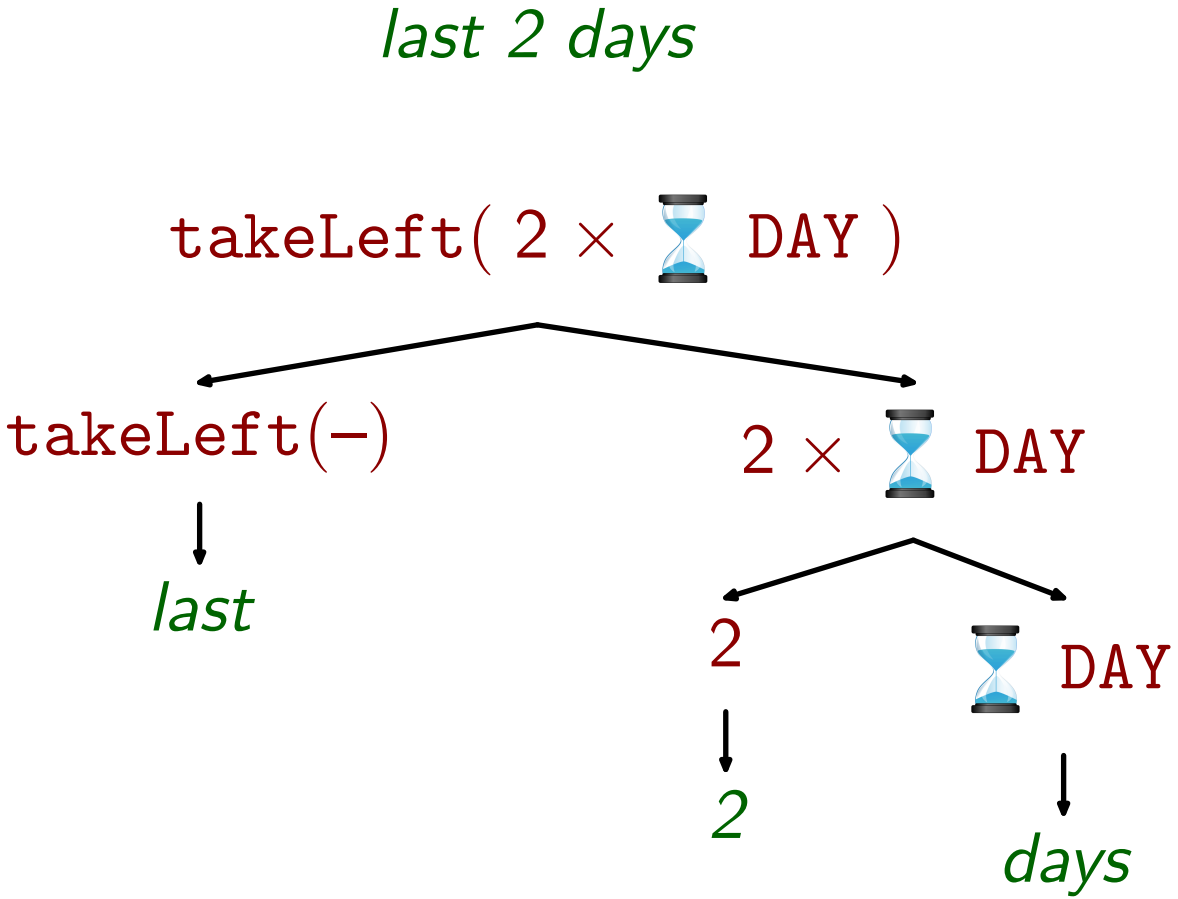
Latent Parse

Expressions parse compositionally



Latent Parse

Expressions parse compositionally



[2012-06-05] → 2012-06-03 – 2012-06-05

Latent Parse

Nonterminals become very sparse

Latent Parse

Nonterminals become very sparse

Consider: *last 7 days* , *last 3 months* , etc.

Latent Parse

Nonterminals become very sparse

Consider: *last 7 days* , *last 3 months* , etc.

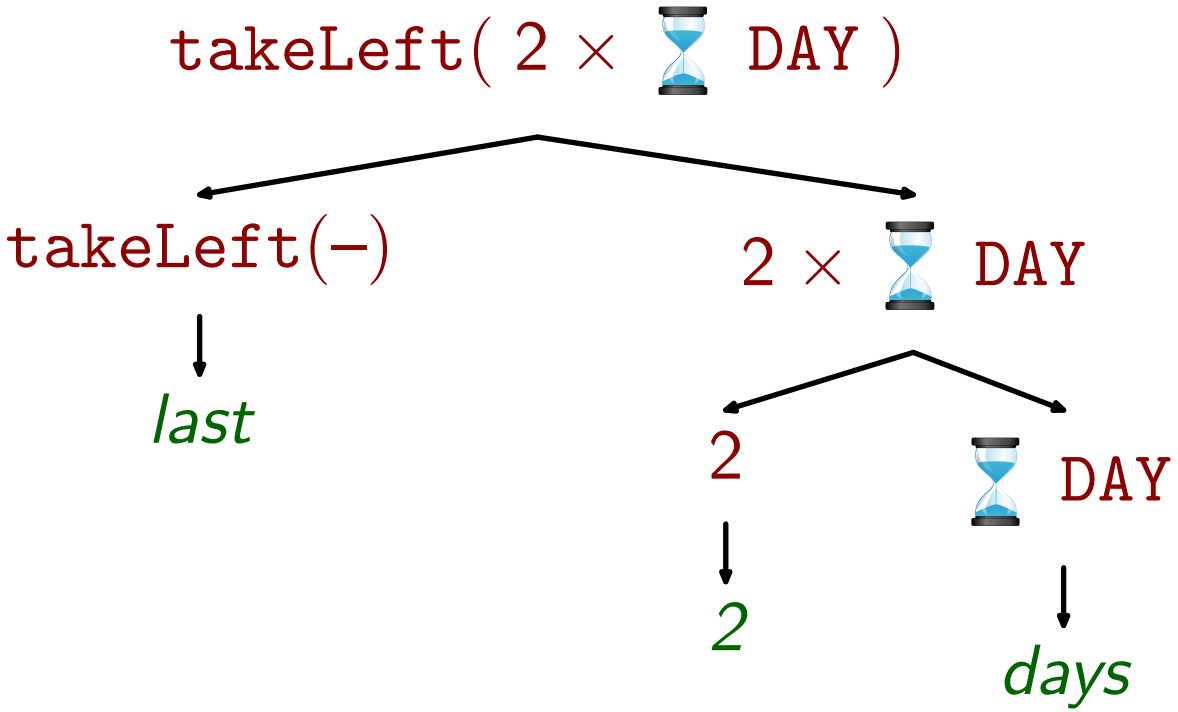
Solution: Group nonterminals based on *types*

Latent Parse

Nonterminals become very sparse

Consider: *last 7 days* , *last 3 months* , etc.

Solution: Group nonterminals based on *types*

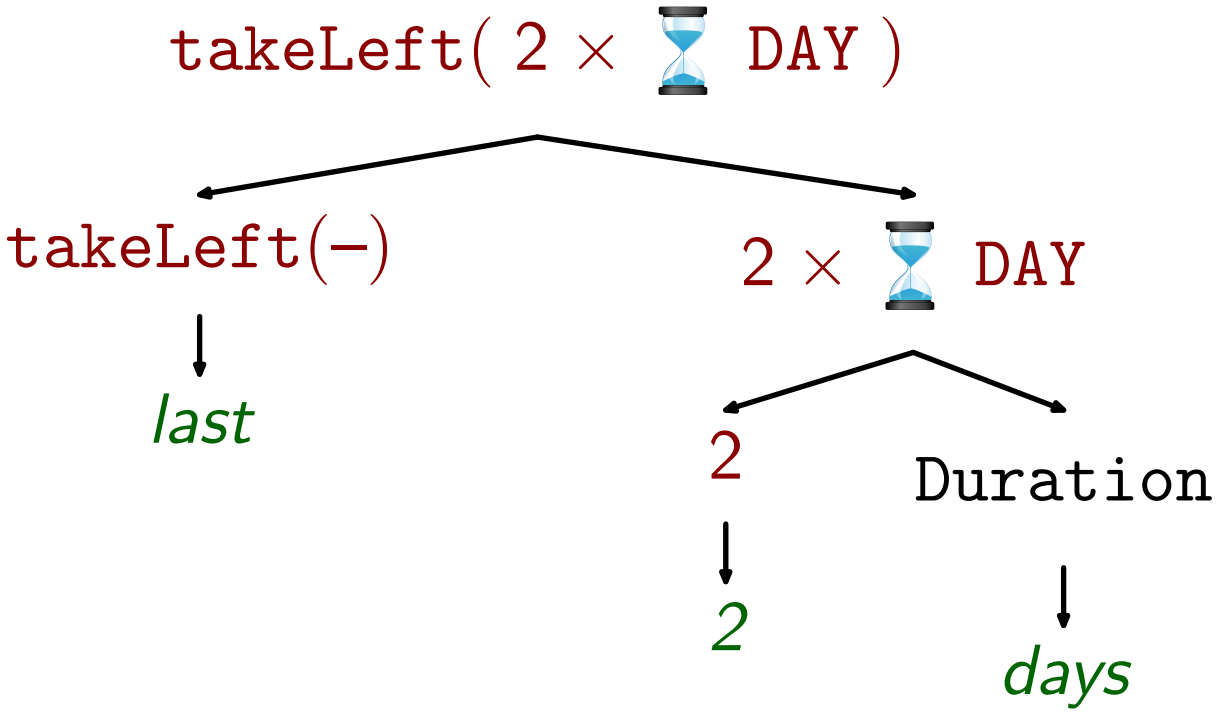


Latent Parse

Nonterminals become very sparse

Consider: *last 7 days* , *last 3 months* , etc.

Solution: Group nonterminals based on *types*

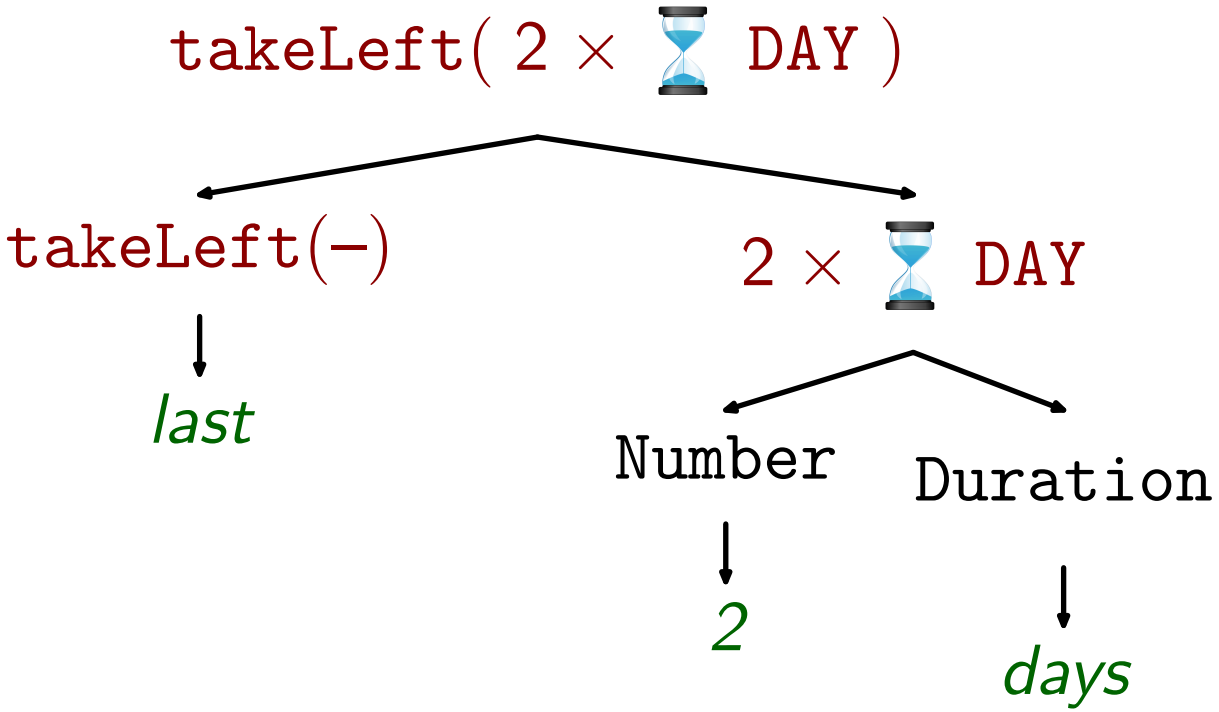


Latent Parse

Nonterminals become very sparse

Consider: *last 7 days* , *last 3 months* , etc.

Solution: Group nonterminals based on *types*

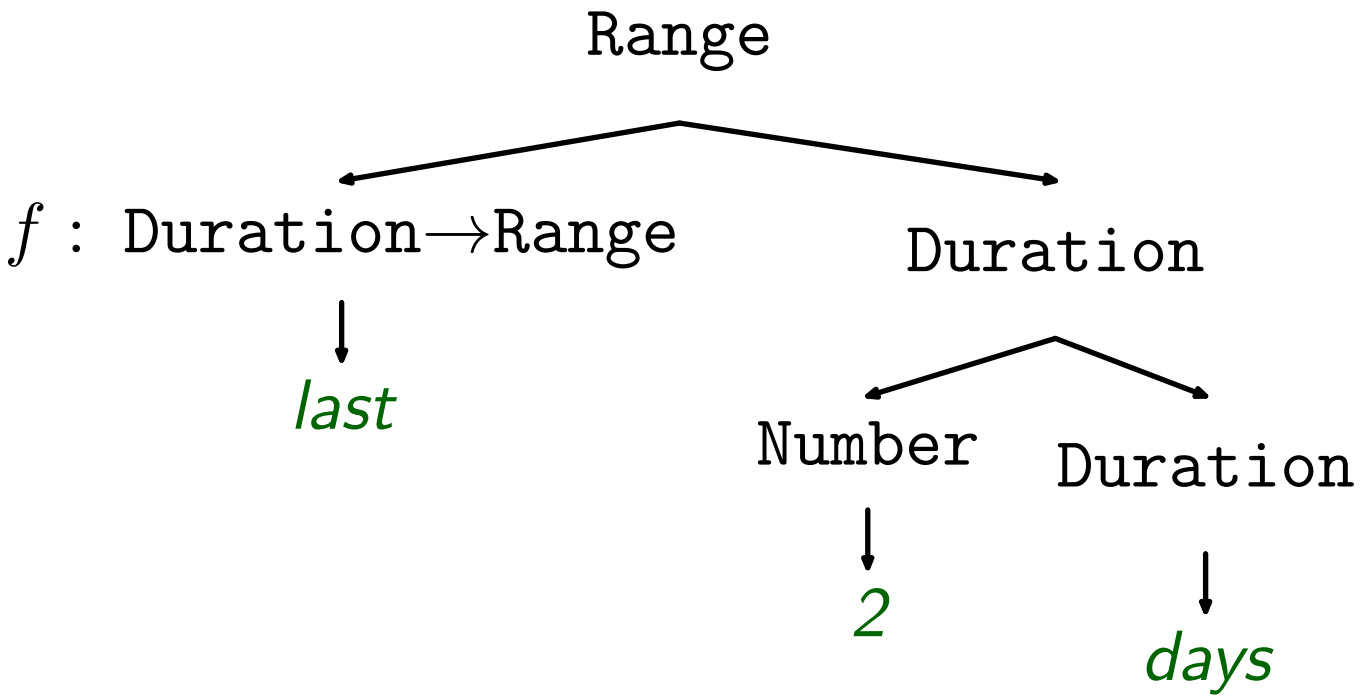


Latent Parse

Nonterminals become very sparse

Consider: *last 7 days* , *last 3 months* , etc.

Solution: Group nonterminals based on *types*

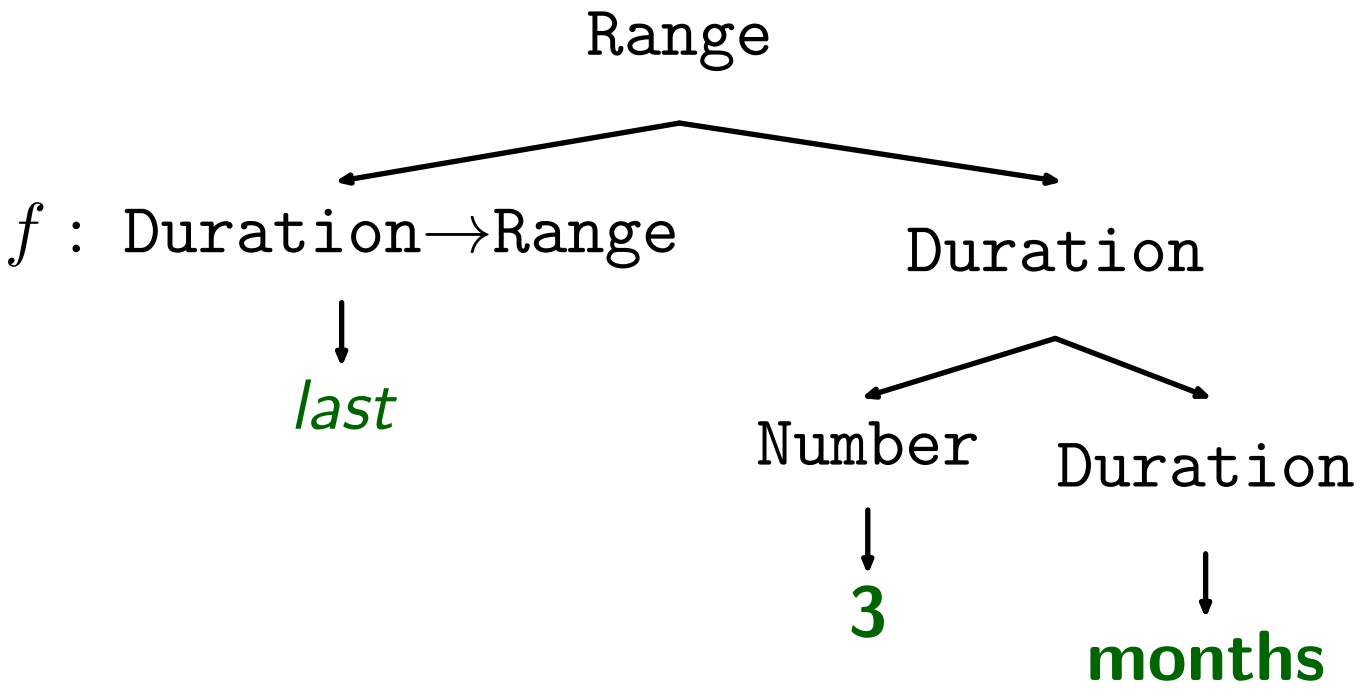


Latent Parse

Nonterminals become very sparse

Consider: *last 7 days* , *last 3 months* , etc.

Solution: Group nonterminals based on *types*

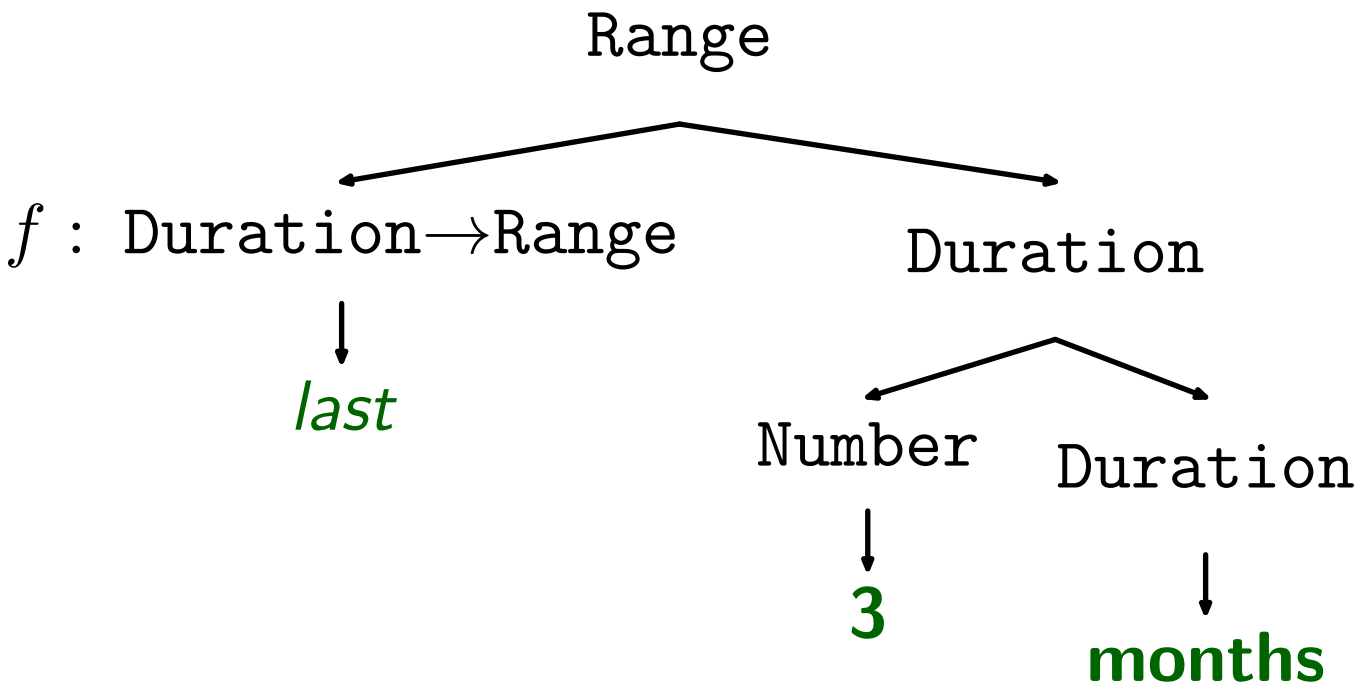


Latent Parse

Nonterminals become very sparse

Consider: *last 7 days* , *last 3 months* , etc.

Solution: Group nonterminals based on *types*



What are these nonterminals, and how do they combine?

Grammar Of Time

Range

Sequence

Duration

Grammar Of Time

Range

Sequence

Duration

Functions

Number

Nil (no temporal meaning)

Grammar Of Time

Range

A period between two dates (or times)

Grammar Of Time

Range

A period between two dates (or times)

Today, *June 5 2012*, *day before yesterday*

Grammar Of Time

Range

A period between two dates (or times)

Today, *June 5 2012*, *day before yesterday*

Interval-based theory of time: instants are ranges

`Today` and `Reference` both ranges, but former has duration

Grammar Of Time

Range

Sequence

A sequence of Ranges (not necessarily at regular intervals)

Grammar Of Time

Range

Sequence

A sequence of Ranges (not necessarily at regular intervals)

June 5, *last Sunday*, *third quarter*

Grammar Of Time

Range

Sequence

A sequence of Ranges (not necessarily at regular intervals)

June 5, *last Sunday*, *third quarter*

Still stuck: which element are we referring to?

Grammar Of Time

Range

Sequence

A sequence of Ranges (not necessarily at regular intervals)

June 5, *last Sunday*, *third quarter*

Still stuck: which element are we referring to?

Answer: We're referring to all of them! (Kind of. . .)

Grammar Of Time

Range

Sequence

Today is 2012-06-05 , what is *last Sunday*?

Grammar Of Time

Range

Sequence

Today is 2012-06-05 , what is *last Sunday*?

June 3, 2012?

Grammar Of Time

Range

Sequence

Today is 2012-06-05 , what is *last Sunday*?

May 27, 2012?
June 3, 2012

Grammar Of Time

Range

Sequence

Today is 2012-06-05 , what is *last Sunday*?

May 20, 2012?
May 27, 2012
June 3, 2012
June 10, 2012?

Grammar Of Time

Range

Sequence

Today is 2012-06-05 , what is *last Sunday*?

0.04	May 20, 2012
0.24	May 27, 2012
0.43	June 3, 2012
0.17	June 10, 2012

Construct a distribution over possible groundings

Grammar Of Time

Range

Sequence

Today is 2012-06-05 , what is *last Sunday*?

0.04	May 20, 2012
0.24	May 27, 2012
0.43	June 3, 2012
0.17	June 10, 2012

Construct a distribution over possible groundings

Appealing to model as a Gaussian

Grammar Of Time

Range

Sequence

Today is 2012-06-05 , what is *last Sunday*?

0.04	May 20, 2012
0.24	May 27, 2012
0.43	June 3, 2012
0.17	June 10, 2012

Construct a distribution over possible groundings

- Appealing to model as a Gaussian
- Learn Gaussian parameters μ, σ

Grammar Of Time

Range

Sequence

Duration

A period of time: *day*, *2 weeks*, *10 years*

Grammar Of Time

Range

Sequence

Duration

Functions

General sequence and interval operations

Grammar Of Time

Range

Sequence

Duration

Functions

General sequence and interval operations

Examples:

- Shift a range/sequence by a duration [*3 days ago*]
- Move the origin of a sequence [*last Sunday*]
- Take the n^{th} element of a sequence [*second week of June*]
- ...

Grammar Of Time

Range

Sequence

Duration

Functions

Number

A number, characterized by its ordinality and magnitude

Grammar Of Time

Range

Sequence

Duration

Functions

Number

Nil

A word without direct temporal meaning

Grammar Of Time

Range

Sequence


Duration

Functions

Number

Nil

A word without direct temporal meaning

Lexicalized: e.g., Nil-the, Nil-a

a week

the week

Grammar Of Time

Range

Sequence


Duration

Functions

Number

Nil

Combination Rules

- Predefined combination rules
- Type checking function application

Training Setup

Given $\{ (x , y) \}$

Training Setup

Given { ((*Phrase* , `Reference`) , `Time`) }

Training Setup

Given { ((*Phrase* , `Reference`) , `Time`) }

Not given latent parse

Not given lexical cues

Training Setup

Given { ((*Phrase* , `Reference`) , `Time`) }

Not given latent parse

Not given lexical cues

Therefore, in general, latent parse is ambiguous

Training Setup

Given { ((*Phrase* , `Reference`) , `Time`) }

Not given latent parse

Not given lexical cues

Therefore, in general, latent parse is ambiguous

((w_1 w_2 , `2012-06-05`) , `2012-06-12`)

Training Setup

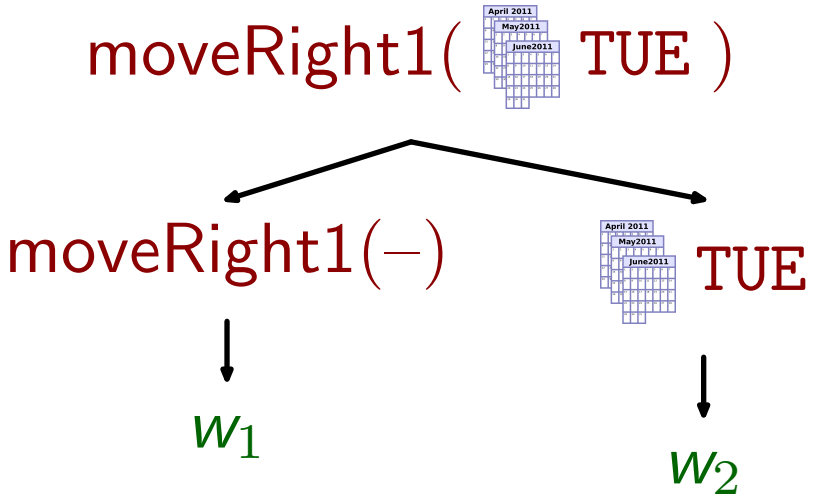
Given { ((*Phrase* , `Reference`) , `Time`) }

Not given latent parse

Not given lexical cues

Therefore, in general, latent parse is ambiguous

((w_1 w_2 , `2012-06-05`) , `2012-06-12`)



e.g., $w_1 = \textit{next}$, $w_2 = \textit{Tuesday}$

Training Setup

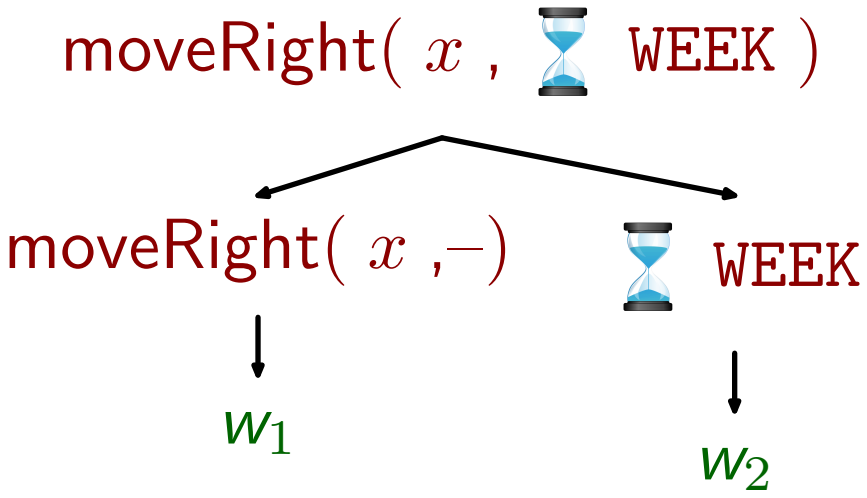
Given { ((*Phrase* , `Reference`) , `Time`) }

Not given latent parse

Not given lexical cues

Therefore, in general, latent parse is ambiguous

((w_1 w_2 , `2012-06-05`) , `2012-06-12`)



e.g., $w_1 = \textit{next}$, $w_2 = \textit{Tuesday}$

e.g., $w_1 = \textit{next}$, $w_2 = \textit{week}$

Training Setup

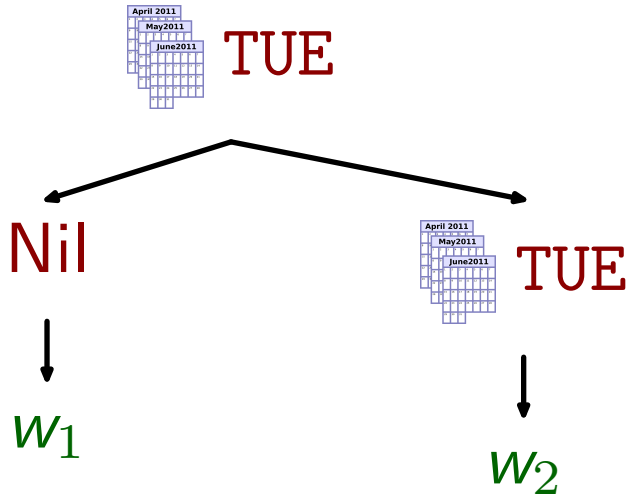
Given { ((*Phrase* , Reference) , Time) }

Not given latent parse

Not given lexical cues

Therefore, in general, latent parse is ambiguous

((w_1 w_2 , 2012-06-05) , 2012-06-12)



e.g., $w_1 = \textit{next}$, $w_2 = \textit{Tuesday}$

e.g., $w_1 = \textit{next}$, $w_2 = \textit{week}$

e.g., $w_1 = \textit{the}$, $w_2 = \textit{Tuesday}$

Training: TimEM

E1 Step: Get k-best parses for phrase

Training: TimEM

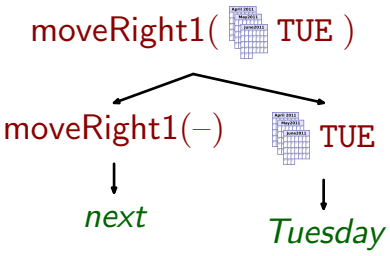
E1 Step: Get k-best parses for phrase

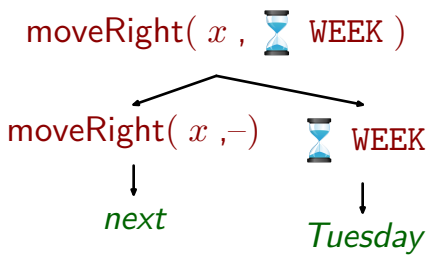
((*next Tuesday* , 2012-06-05) , 2012-06-12)

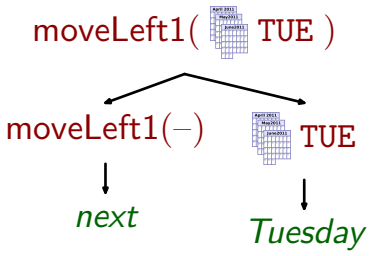
Training: TimEM

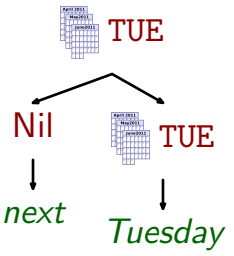
E1 Step: Get k-best parses for phrase

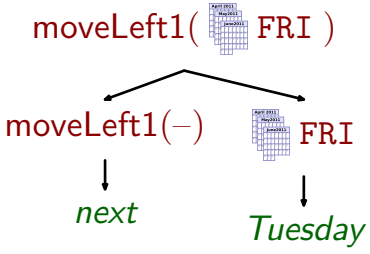
((*next Tuesday* , 2012-06-05) , 2012-06-12)

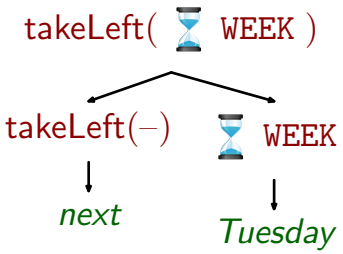


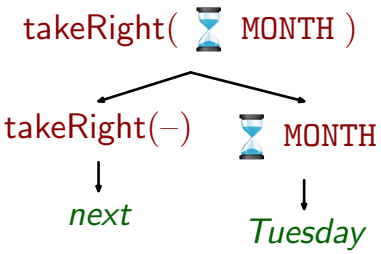


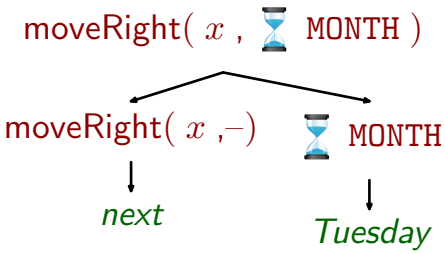








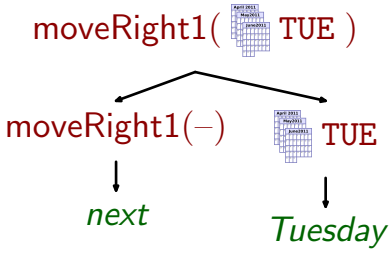




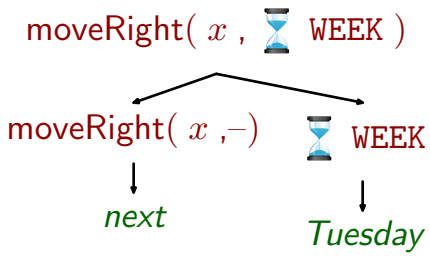
Training: TimEM

E1 Step: Get k-best parses for phrase

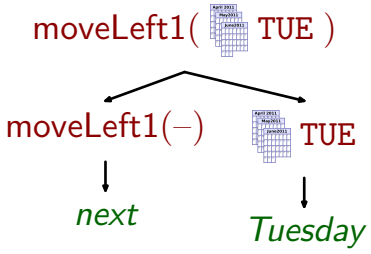
((*next Tuesday* , 2012-06-05) , 2012-06-12)



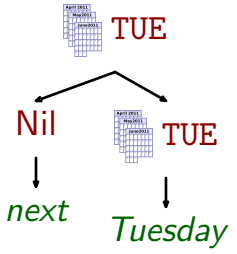
0.008



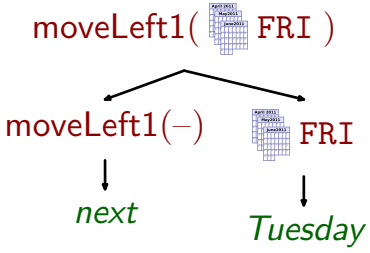
0.005



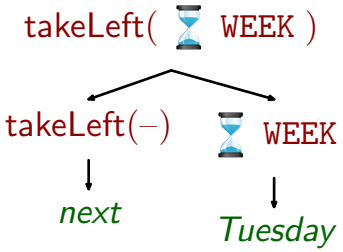
0.004



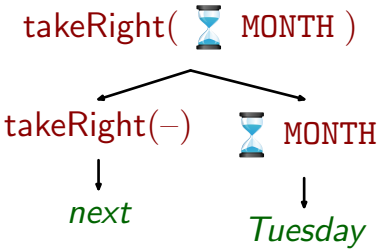
0.001



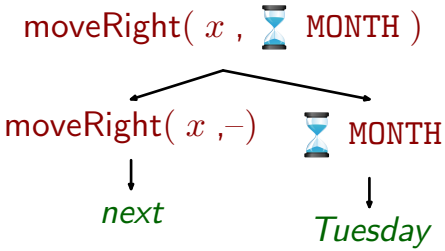
0.0007



0.0004



0.0003



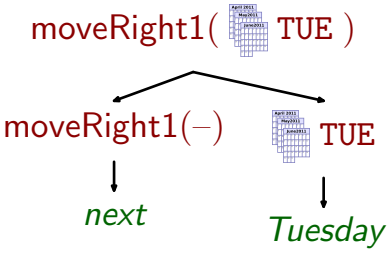
0.0001

Training: TimEM

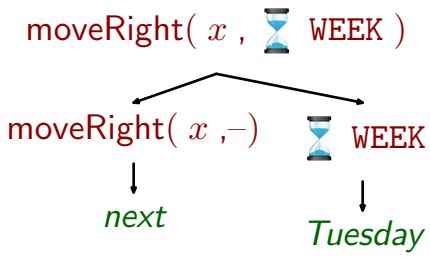
E1 Step: Get k-best parses for phrase

E2 Step: Filter and re-weight correct parses

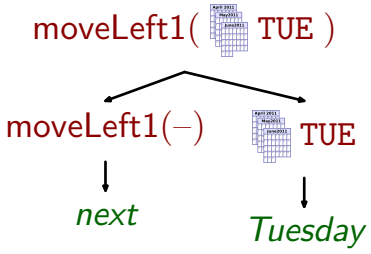
((*next Tuesday* , 2012-06-05) , 2012-06-12)



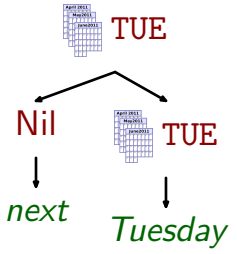
0.008



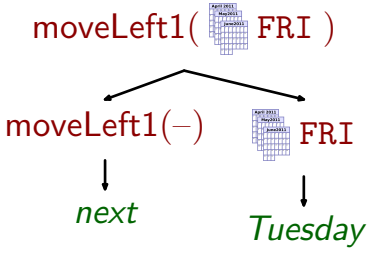
0.005



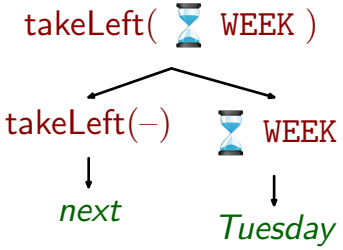
0.004



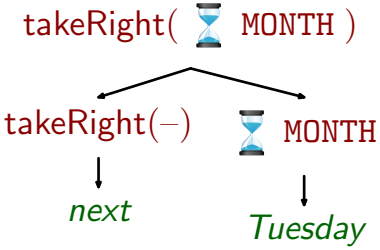
0.001



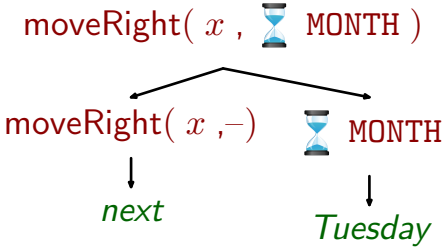
0.0007



0.0004



0.0003



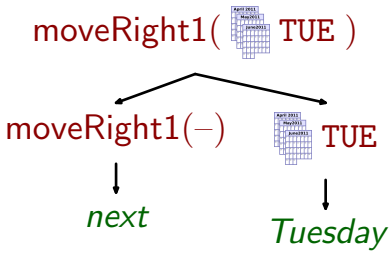
0.0001

Training: TimEM

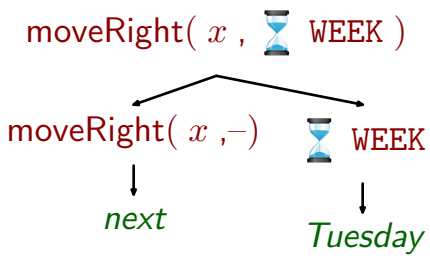
E1 Step: Get k-best parses for phrase

E2 Step: Filter and re-weight correct parses

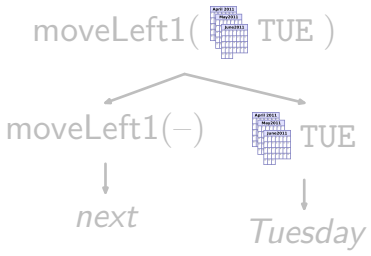
((*next Tuesday* , 2012-06-05) , 2012-06-12)



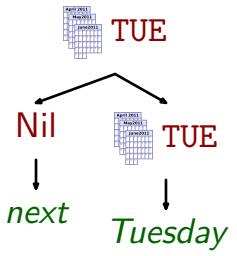
0.008



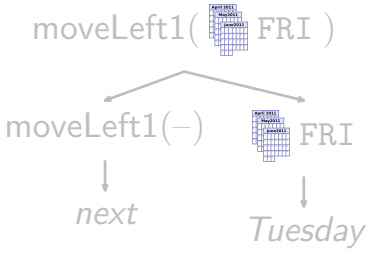
0.005



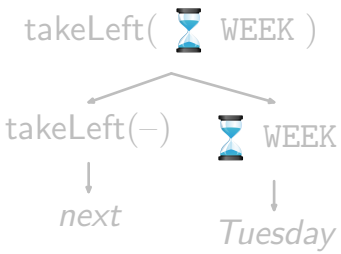
0.004



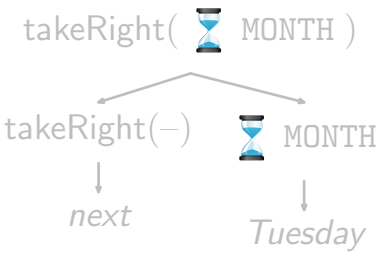
0.001



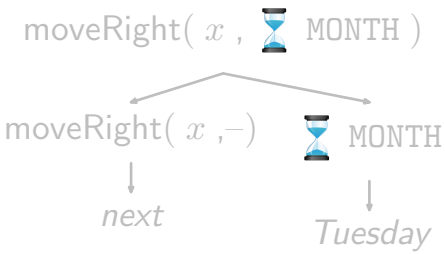
0.0007



0.0004



0.0003



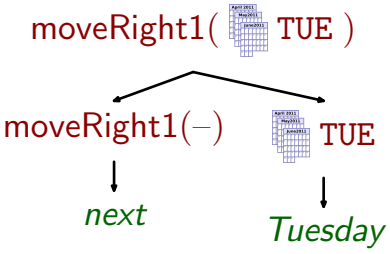
0.0001

Training: TimEM

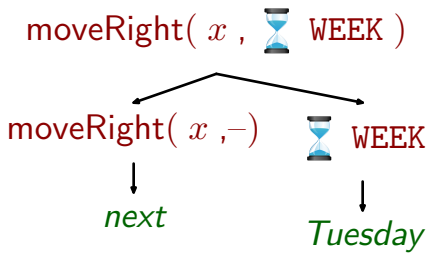
E1 Step: Get k-best parses for phrase

E2 Step: Filter and re-weight correct parses

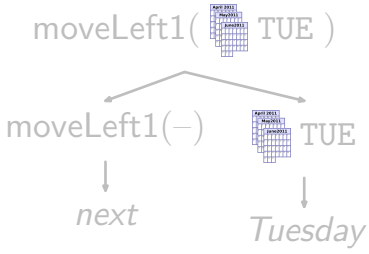
((*next Tuesday* , 2012-06-05) , 2012-06-12)



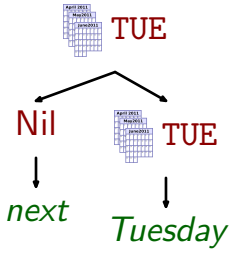
0.57



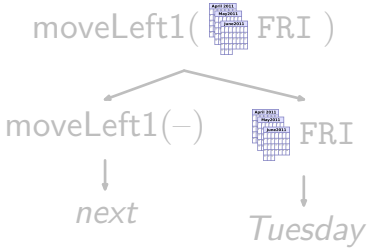
0.36



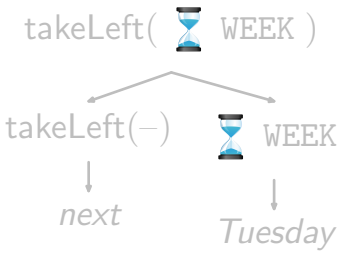
0.00



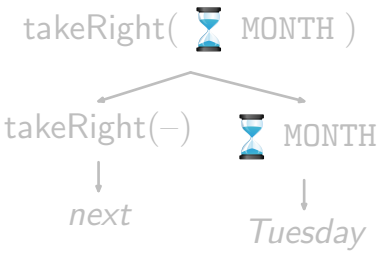
0.07



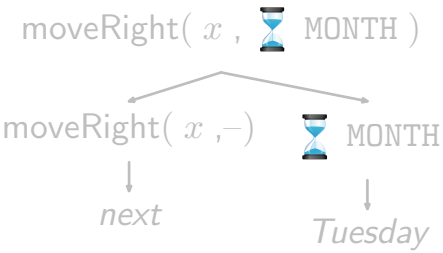
0.00



0.00



0.00



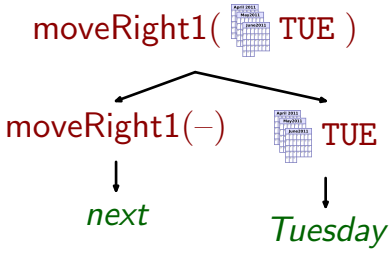
0.00

Training: TimEM

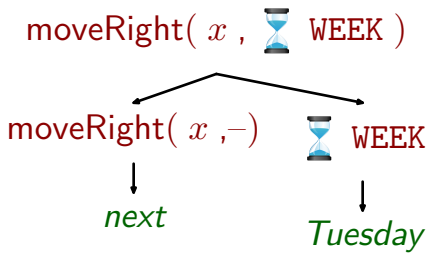
E1 Step: Get k-best parses for phrase

E2 Step: Filter and re-weight correct parses

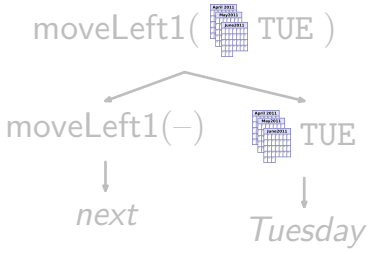
M Step: Update expected sufficient statistics



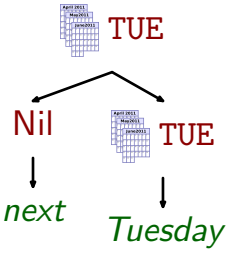
0.57



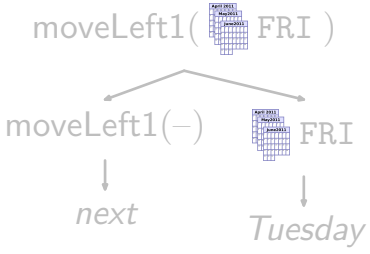
0.36



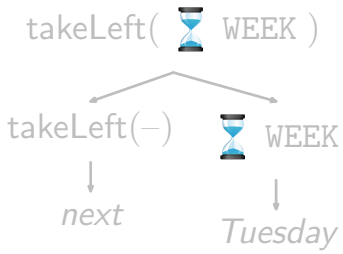
0.00



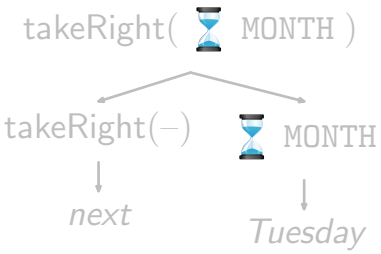
0.07



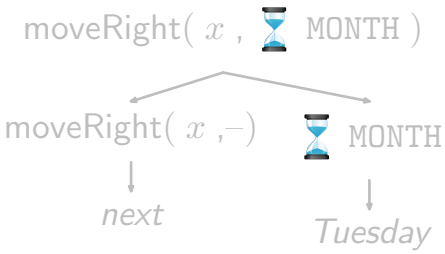
0.00



0.00



0.00



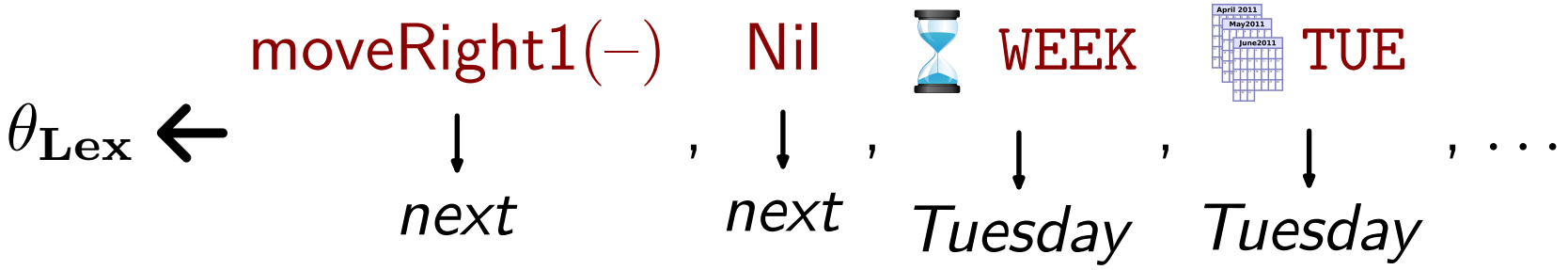
0.00

Training: TimEM

E1 Step: Get k-best parses for phrase

E2 Step: Filter and re-weight correct parses

M Step: Update expected sufficient statistics

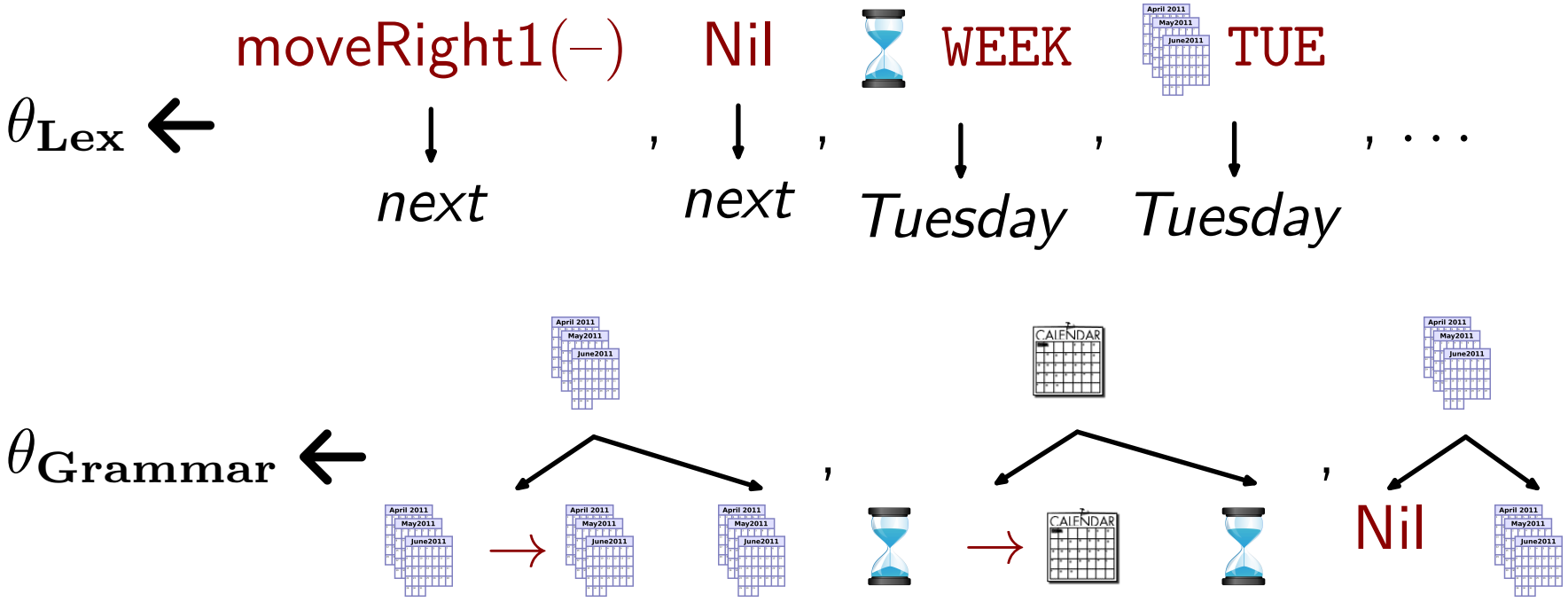


Training: TimEM

E1 Step: Get k-best parses for phrase

E2 Step: Filter and re-weight correct parses

M Step: Update expected sufficient statistics

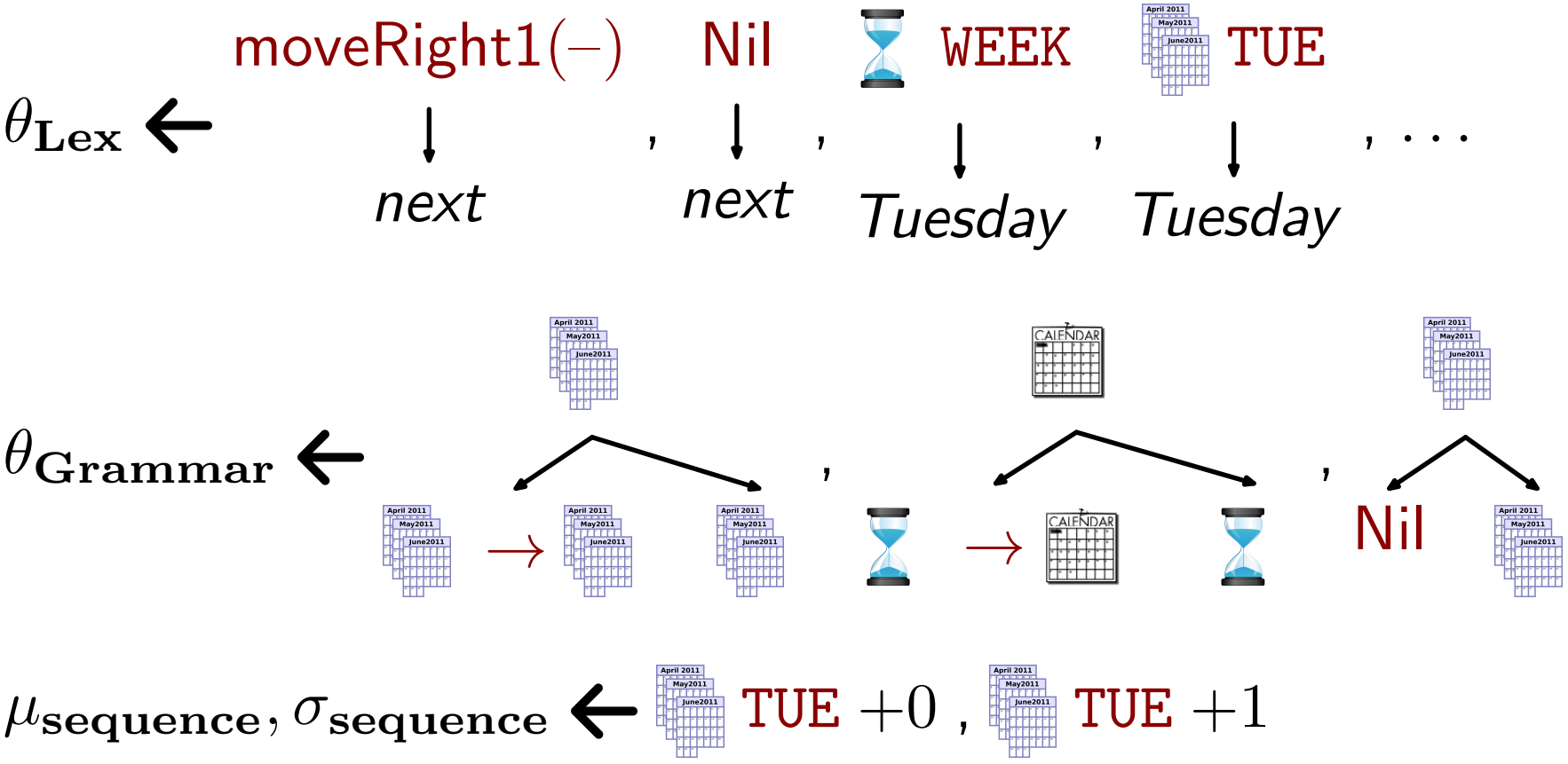


Training: TimEM

E1 Step: Get k-best parses for phrase

E2 Step: Filter and re-weight correct parses

M Step: Update expected sufficient statistics



TimEM Discussion

Intuition

Bootstrap from short examples

TimEM Discussion

Intuition

Bootstrap from short examples

week, then *next week*, . . .

TimEM Discussion

Intuition

Bootstrap from short examples

week, then *next week*, ...

Smoothing

Dirichlet prior on grammar parameters θ

Gaussian prior on μ given MLE σ

TimEM Discussion

Intuition

Bootstrap from short examples

week, then *next week*, ...

Smoothing

Dirichlet prior on grammar parameters θ

Gaussian prior on μ given MLE σ

Uniform initialization $\rightarrow$ deterministic

Dataset

TempEval2

Newswire annotated for temporal expressions

Dataset

TempEval2

Newswire annotated for temporal expressions

1052 training / 156 test expressions

Dataset

TempEval2

Newswire annotated for temporal expressions

1052 training / 156 test expressions

Evaluation

Most likely grounding (e.g., `June 5` $\rightarrow$ `2012-06-05`)

Dataset

TempEval2

Newswire annotated for temporal expressions

1052 training / 156 test expressions

Evaluation

Most likely grounding (e.g., `June 5` $\rightarrow$ `2012-06-05`)

Type: Accuracy over result's temporal type

 `June 5, 2012` =  `June 12, 2012`

Dataset

TempEval2

Newswire annotated for temporal expressions

1052 training / 156 test expressions

Evaluation

Most likely grounding (e.g., `June 5` $\rightarrow$ `2012-06-05`)

Type: Accuracy over result's temporal type

 `June 5, 2012` $=$  `June 12, 2012`

Value: Accuracy over result's value, if types match

 `June 5, 2012` $\neq$  `June 12, 2012`

Dataset

TempEval2

Newswire annotated for temporal expressions

1052 training / 156 test expressions

Evaluation

Most likely grounding (e.g., `June 5` $\rightarrow$ `2012-06-05`)

Type: Accuracy over result's temporal type

 `June 5, 2012` $=$  `June 12, 2012`

Value: Accuracy over result's value, if types match

 `June 5, 2012` $\neq$  `June 12, 2012`

Constrained to guess on each example; no contextual cues

Results

Comparisons

GUTime (Mani and Wilson, 2000)

SUTime (Chang and Manning, 2012)

HeidelTime (Strötgen and Gertz, 2010)

Results

Comparisons

GUTime (Mani and Wilson, 2000)

SUTime (Chang and Manning, 2012)

HeidelTime (Strötgen and Gertz, 2010)

Test (all expressions; gold detection)

System	Type	Value
GUTime	0.80	0.42

Results

Comparisons

GUTime (Mani and Wilson, 2000)

SUTime (Chang and Manning, 2012)

HeidelTime (Strötgen and Gertz, 2010)

Test (all expressions; gold detection)

System	Type	Value
GUTime	0.80	0.42
SUTime	**0.94**	0.71

Results

Comparisons

GUTime (Mani and Wilson, 2000)

SUTime (Chang and Manning, 2012)

HeidelTime (Strötgen and Gertz, 2010)

Test (all expressions; gold detection)

System	Type	Value
GUTime	0.80	0.42
SUTime	**0.94**	0.71
HeidelTime	0.85	0.71

Results

Comparisons

GUTime (Mani and Wilson, 2000)

SUTime (Chang and Manning, 2012)

HeidelTime (Strötgen and Gertz, 2010)

Test (all expressions; gold detection)

System	Type	Value
GUTime	0.80	0.42
SUTime	**0.94**	0.71
HeidelTime	0.85	0.71
ParsingTime	0.88	**0.72**

Conclusion

Probabilistic, compositional temporal parsing

- Elegant compositional representation of time
- EM-like algorithm for learning

Conclusion

Probabilistic, compositional temporal parsing

- Elegant compositional representation of time
- EM-like algorithm for learning
- Results competitive with state-of-the-art

Conclusion

Probabilistic, compositional temporal parsing

- Elegant compositional representation of time
- EM-like algorithm for learning
- Results competitive with state-of-the-art

Takeaway points

- Capture **syntactic** ambiguity: *last Friday the 13th*
- Capture **pragmatic** ambiguity: *last Sunday*

Conclusion

Probabilistic, compositional temporal parsing

- Elegant compositional representation of time
- EM-like algorithm for learning
- Results competitive with state-of-the-art

Takeaway points

- Capture **syntactic** ambiguity: *last Friday the 13th*
- Capture **pragmatic** ambiguity: *last Sunday*
- **Robust** guess and confidence for any expression

Conclusion

Probabilistic, compositional temporal parsing

- Elegant compositional representation of time
- EM-like algorithm for learning
- Results competitive with state-of-the-art

Takeaway points

- Capture **syntactic** ambiguity: *last Friday the 13th*
- Capture **pragmatic** ambiguity: *last Sunday*
- **Robust** guess and confidence for any expression
- **Language/domain independent** approach

Thank You!

(We're out of time)