A Type-coherent, Expressive Representation as an Initial Step to Language Understanding

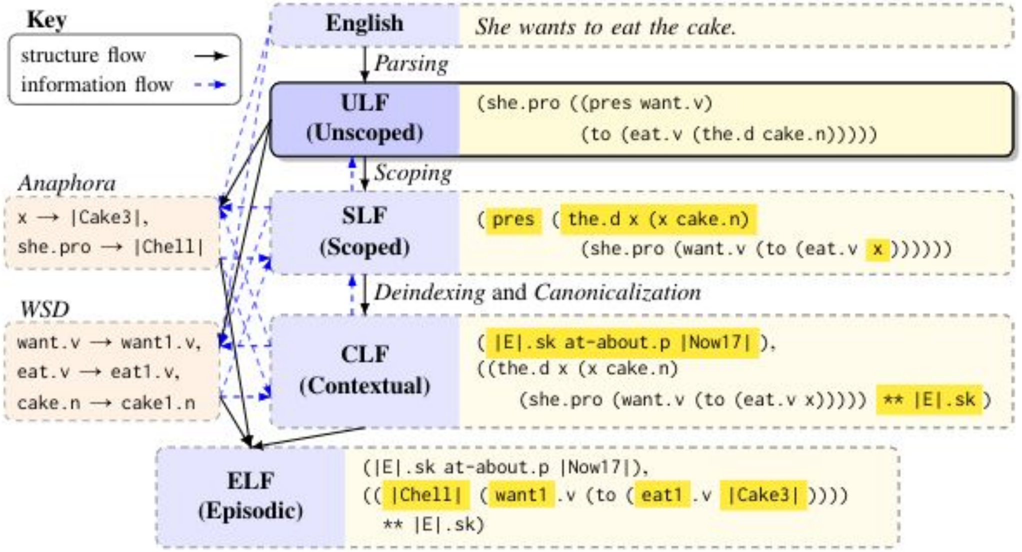
Gene Louis Kim and Lenhart Schubert

Presented by: Gene Louis Kim
May 2019

Introduction

Unscoped {Episodic} Logical Form (ULF)

- An underspecified Episodic Logic (EL)
- Starting point for EL parsing
- Enables situated inferences



Motivation

Semantic representation desiderata

1. Adequately models the complexity of language semantics
2. Enables the production of general inferences
3. Can be recovered accurately

Motivation

Semantic representation desiderata

1. Adequately models the complexity of language semantics
2. Enables the production of general inferences
3. Can be recovered accurately

Episodic Logic

- Extended FOL
- Closely matches expressivity of natural languages
  - Predicates, connectives, quantifiers, equality → FOL
  - Predicate and sentence modification ***(e.g. very, gracefully, nearly, possibly)***
  - Predicate and sentence reification ***(e.g. <u>Beauty</u> is subjective, <u>That exoplanets exist</u> is now certain)***
  - Generalized quantifiers ***(e.g. most men who smoke)***
  - Intensional predicates ***(e.g. believe, intend, resemble)***
  - Reference to events and situations ***(Many children had not been vaccinated against measles; <u>this situation</u> caused sporadic outbreaks of the disease)***
- Suitable for deductive, uncertain, and Natural-Logic-like inference
- A fast and comprehensive theorem prover, EPILOG, is already available.

Motivation

Language understanding is a growing area of interest in NLP

- **Question Answering**: AI2 Reasoning challenge, RACE, SQuAD, TriviaQA, NarrativeQA…
- **Dialogue**: Amazon Alexa Challenge, Google Home, Microsoft Cortana...
- **Inferring from Language**: JOCI, SNLI, MultiNLI...
- **Semantic Parsing**: AMR, DRS Parsing (IWCS-2019 Shared Task), Cross-lingual Semantic Parsing

Motivation

Language understanding is a growing area of interest in NLP

Question Answering: AI2 Reasoning challenge, RACE, SQuAD, TriviaQA, NarrativeQA…

Dialogue: Amazon Alexa Challenge, Google Home, Microsoft Cortana...

Inferring from Language: JOCI, SNLI, MultiNLI...

Semantic Parsing: AMR, DRS Parsing (IWCS-2019 Shared Task), Cross-lingual Semantic Parsing

Current state-of-the-art systems often end up modeling artifacts

SQuAD question answering and reading comprehension *(Jia & Liang 2017)*

80.0% → ***Unrelated Information*** → **34.2%**

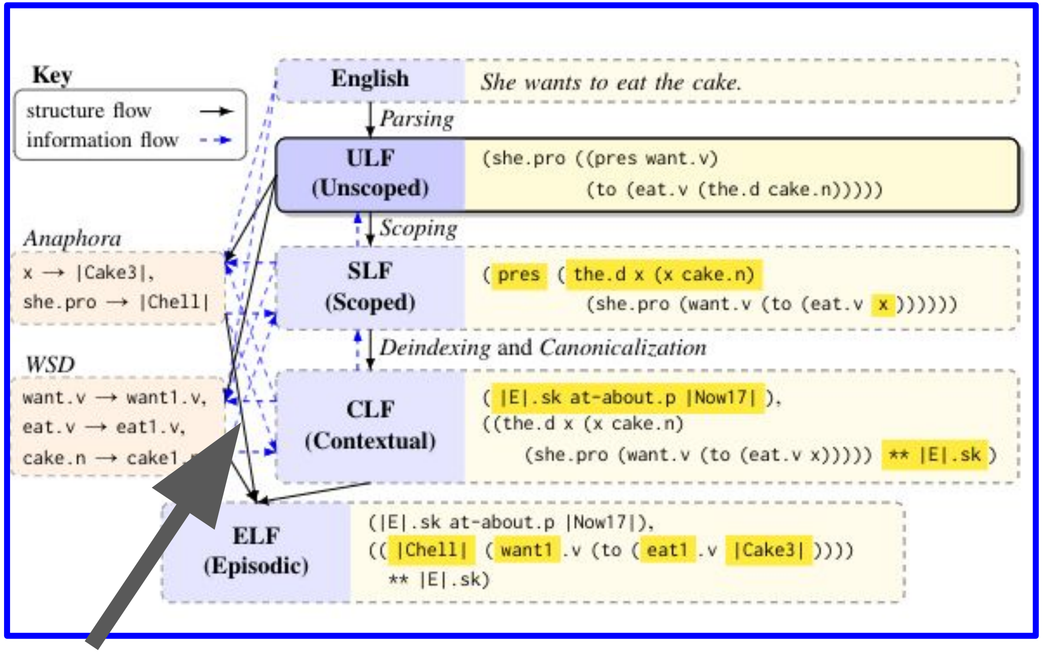
Inferring from language *(Gururangan et al., 2018; Poliak et al., 2018)*

SNLI - majority class baseline: **34.3%**

→ ***Hypothesis Only*** → **69.0%**

Our Driving Hypotheses

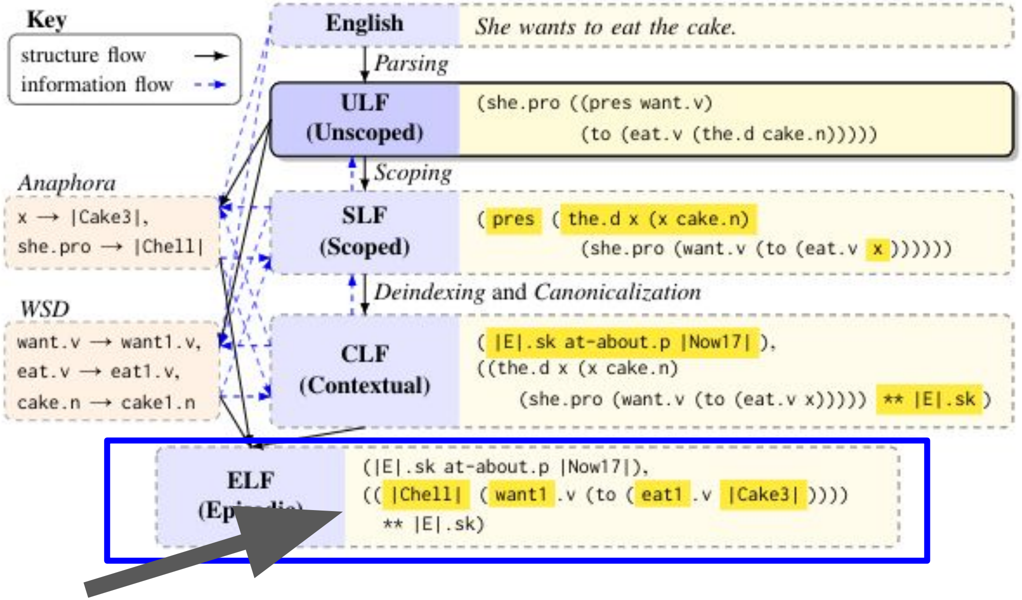
1. A divide-and-conquer approach to semantic parsing will ultimately lead to more precise and useful representations for reasoning over language.



Hypothesis 1: Divide-and-conquer

Our Driving Hypotheses

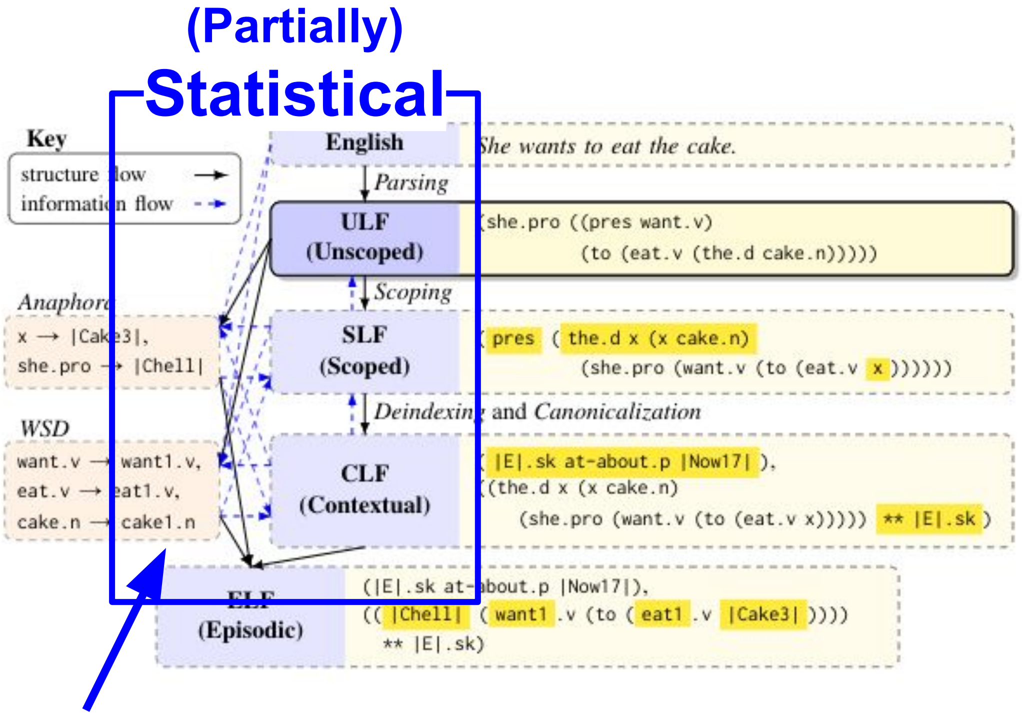
1. A divide-and-conquer approach to semantic parsing will ultimately lead to more precise and useful representations for reasoning over language.
2. An expressive logical representation with model-theoretic backing will enable reasoning capabilities that are not offered by other semantic representations available today.



Hypothesis 2: Expressive Model-theoretic Logic

Our Driving Hypotheses

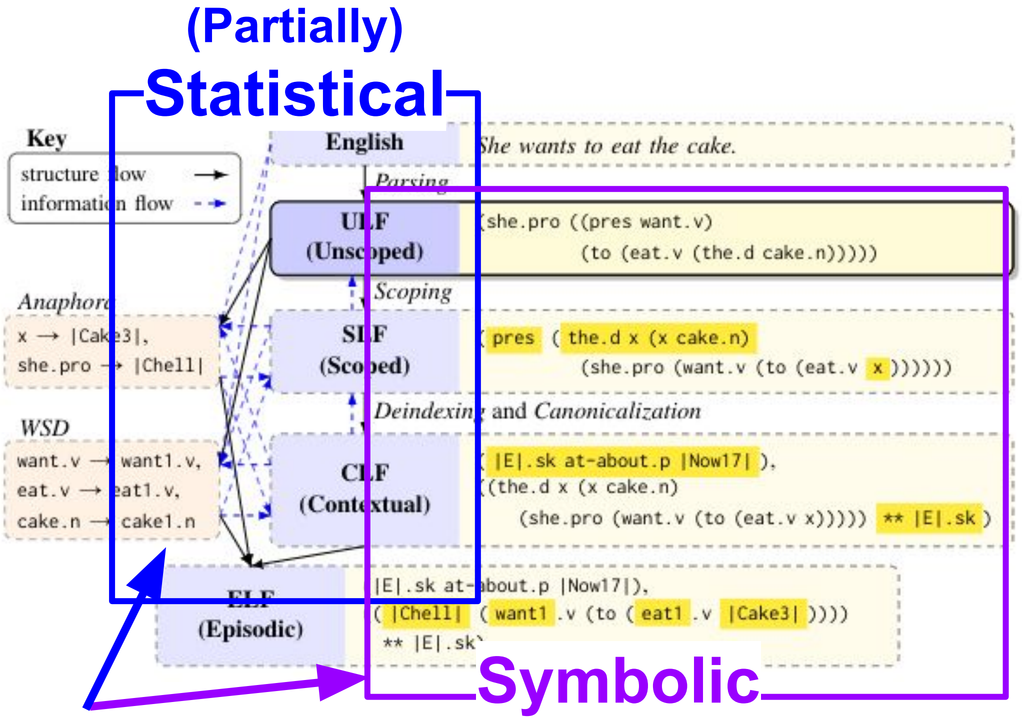
1. A divide-and-conquer approach to semantic parsing will ultimately lead to more precise and useful representations for reasoning over language.
2. An expressive logical representation with model-theoretic backing will enable reasoning capabilities that are not offered by other semantic representations available today.
3. Better language understanding and reasoning systems can be built by combining the strengths of statistical systems in converting raw signals to structured representations and symbolic systems in performing precise and flexible manipulations over complex structures.



Hypothesis 3: Combine Statistical and Symbolic Methods

Our Driving Hypotheses

1. A divide-and-conquer approach to semantic parsing will ultimately lead to more precise and useful representations for reasoning over language.
2. An expressive logical representation with model-theoretic backing will enable reasoning capabilities that are not offered by other semantic representations available today.
3. Better language understanding and reasoning systems can be built by combining the strengths of statistical systems in converting raw signals to structured representations and symbolic systems in performing precise and flexible manipulations over complex structures.



Hypothesis 3: Combine Statistical and Symbolic Methods

What is ULF?

A minimal step across from syntax to semantics in Episodic Logic

“Alice thinks that John nearly fell”

ULF

```
(|Alice| (((pres think.v)
           (that (|John| (nearly.adv-a (past fall.v)))))))
```

Syntax (simplified)

```
(S (NP Alice.nnp) (VP thinks.vbz
                  (SBAR that.rb (S (NP John.nnp) (ADVP nearly.rb) (VP fell.vbd)))))
```

What is ULF?

A minimal step across from syntax to semantics in Episodic Logic

“Alice thinks that John nearly fell”

ULF

```
(|Alice| (((pres think.v)
          (that (|John| (nearly.adv-a (past fall.v)))))))
```

Syntax (simplified)

```
(S (NP Alice.nnp) (VP thinks.vbz
                  (SBAR that.rb (S (NP John.nnp) (ADVP nearly.rb) (VP fell.vbd)))))
```

Proper Nouns

What is ULF?

A minimal step across from syntax to semantics in Episodic Logic

“Alice thinks that John nearly fell”

ULF

```
(|Alice| (((pres think.v)
           (that (|John| (nearly.adv-a (past fall.v)))))))
```

Syntax (simplified)

```
(S (NP Alice.nnp) (VP thinks.vbz
                  (SBAR that.rb (S (NP John.nnp) (ADVP nearly.rb) (VP fell.vbd)))))
```

Verbs

What is ULF?

A minimal step across from syntax to semantics in Episodic Logic

"Alice thinks that John nearly fell"

ULF

```
(|Alice| (((pres think.v)
           (that (|John| (nearly.adv-a (past fall.v)))))))
```

Syntax (simplified)

```
(S (NP Alice.nnp) (VP thinks.vbz
                  (SBAR that.rb (S (NP John.nnp) (ADVP nearly.rb) (VP fell.vbd)))))
```

Adverbs

What is ULF?

A minimal step across from syntax to semantics in Episodic Logic

“Alice thinks that John nearly fell”

ULF

```
(|Alice| (((pres think.v)
           (that (|John| (nearly.adv-a (past fall.v)))))))
```

Syntax (simplified)

```
(S (NP Alice.nnp) (VP thinks.vbz
                 (SBAR that.rb (S (NP John.nnp) (ADVP nearly.rb) (VP fell.vbd)))))
```

Not just syntax!

What is ULF?

A minimal step across from syntax to semantics in Episodic Logic

"Alice thinks that John nearly fell", "Could you dial for me?"

ULFs

```
(|Alice| (((pres think.v)
            (that (|John| (nearly.adv-a (past fall.v)))))))

(((pres could.aux-v) you.pro
  (dial.v {ref1}.pro (adv-a (for.p me.pro)))) ?)
```

Basic Ontological Types

$\mathcal{D}$	Domain
$\mathcal{S}$	Situations
$\mathbf{2}$	Truth-value

Monadic Predicate $\mathcal{N} : \mathcal{D} \rightarrow (\mathcal{S} \rightarrow \mathbf{2})$

What is ULF?

A minimal step across from syntax to semantics in Episodic Logic

“Alice thinks that John nearly fell”, “Could you dial for me?”

ULFs

```
(|Alice| (((pres think.v)
          (that (|John| (nearly.adv-a (past fall.v)))))))

(((pres could.aux-v) you.pro
  (dial.v {ref1}.pro (adv-a (for.p me.pro)))) ?)
```

Entity($\mathcal{D}$): `|Alice|, |John|, you.pro, {ref1}.pro, me.pro`

Basic Ontological Types

$\mathcal{D}$	Domain
$\mathcal{S}$	Situations
$\mathbf{2}$	Truth-value

Monadic Predicate $\mathcal{N} : \mathcal{D} \rightarrow (\mathcal{S} \rightarrow \mathbf{2})$

What is ULF?

A minimal step across from syntax to semantics in Episodic Logic

“Alice thinks that John nearly fell”, “Could you dial for me?”

ULFs

```
(|Alice| (((pres think.v)
            (that (|John| (nearly.adv-a (past fall.v)))))))

(((pres could.aux-v) you.pro
  (dial.v {ref1}.pro (adv-a (for.p me.pro)))) ?)
```

Entity($\mathcal{D}$): `|Alice|, |John|, you.pro, {ref1}.pro, me.pro`
n-ary predicate($\mathcal{D}^n \rightarrow (\mathcal{S} \rightarrow \mathbf{2})$): `think.v, fall.v, dial.v, for.p`

Basic Ontological Types

$\mathcal{D}$	Domain
$\mathcal{S}$	Situations
$\mathbf{2}$	Truth-value

Monadic Predicate $\mathcal{N} : \mathcal{D} \rightarrow (\mathcal{S} \rightarrow \mathbf{2})$

What is ULF?

A minimal step across from syntax to semantics in Episodic Logic

“Alice thinks that John nearly fell”, “Could you dial for me?”

ULFs

```
(|Alice| (((pres think.v)
            (that (|John| (nearly.adv-a (past fall.v)))))))

(((pres could.aux-v) you.pro
  (dial.v {ref1}.pro (adv-a (for.p me.pro)))) ?)
```

Entity($\mathcal{D}$): `|Alice|, |John|, you.pro, {ref1}.pro, me.pro`
n-ary predicate($\mathcal{D}^n \rightarrow (\mathcal{S} \rightarrow \mathbf{2})$): `think.v, fall.v, dial.v, for.p`
Predicate modifier($\mathcal{N} \rightarrow \mathcal{N}$): `nearly.adv-a, (adv-a (for.p me.pro))`

Basic Ontological Types

$\mathcal{D}$	Domain
$\mathcal{S}$	Situations
$\mathbf{2}$	Truth-value

Monadic Predicate $\mathcal{N} : \mathcal{D} \rightarrow (\mathcal{S} \rightarrow \mathbf{2})$

What is ULF?

A minimal step across from syntax to semantics in Episodic Logic

"Alice thinks that John nearly fell", "Could you dial for me?"

ULFs

```
(|Alice| (((pres think.v)
            (that (|John| (nearly.adv-a (past fall.v)))))))

(((pres could.aux-v) you.pro
  (dial.v {ref1}.pro (adv-a (for.p me.pro)))) ?)
```

Entity($\mathcal{D}$): `|Alice|, |John|, you.pro, {ref1}.pro, me.pro`
n-ary predicate($\mathcal{D}^n \rightarrow (\mathcal{S} \rightarrow \mathbf{2})$): `think.v, fall.v, dial.v, for.p`
Predicate modifier($\mathcal{N} \rightarrow \mathcal{N}$): `nearly.adv-a, (adv-a (for.p me.pro))`
Sentence reifier($(\mathcal{S} \rightarrow \mathbf{2}) \rightarrow \mathcal{D}$): `that`

Basic Ontological Types

$\mathcal{D}$	Domain
$\mathcal{S}$	Situations
$\mathbf{2}$	Truth-value

Monadic Predicate $\mathcal{N} : \mathcal{D} \rightarrow (\mathcal{S} \rightarrow \mathbf{2})$

What is ULF?

A minimal step across from syntax to semantics in Episodic Logic

"Alice thinks that John nearly fell", "Could you dial for me?"

ULFs

```
(|Alice| (((pres think.v)
            (that (|John| (nearly.adv-a (past fall.v)))))))

(((pres could.aux-v) you.pro
  (dial.v {ref1}.pro (adv-a (for.p me.pro)))) ?)
```

Entity($\mathcal{D}$): `|Alice|, |John|, you.pro, {ref1}.pro, me.pro`
n-ary predicate($\mathcal{D}^n \to (\mathcal{S} \to \mathbf{2})$): `think.v, fall.v, dial.v, for.p`
Predicate modifier($\mathcal{N} \to \mathcal{N}$): `nearly.adv-a, (adv-a (for.p me.pro))`
Sentence reifier($(\mathcal{S} \to \mathbf{2}) \to \mathcal{D}$): `that`
Tense($(\mathcal{S} \to \mathbf{2}) \to (\mathcal{S} \to \mathbf{2})$): `pres, past`

Basic Ontological Types

$\mathcal{D}$	Domain
$\mathcal{S}$	Situations
$\mathbf{2}$	Truth-value

Monadic Predicate $\mathcal{N} : \mathcal{D} \to (\mathcal{S} \to \mathbf{2})$

What is ULF?

A minimal step across from syntax to semantics in Episodic Logic

"Alice thinks that John nearly fell", "Could you dial for me?"

ULFs

```
(|Alice| (((pres think.v)
            (that (|John| (nearly.adv-a (past fall.v)))))))

(((pres could.aux-v) you.pro
  (dial.v {ref1}.pro (adv-a (for.p me.pro)))) ?)
```

Entity($\mathcal{D}$): `|Alice|, |John|, you.pro, {ref1}.pro, me.pro`
n-ary predicate($\mathcal{D}^n \to (\mathcal{S} \to \mathbf{2})$): `think.v, fall.v, dial.v, for.p`
Predicate modifier($\mathcal{N} \to \mathcal{N}$): `nearly.adv-a, (adv-a (for.p me.pro))`
Sentence reifier($(\mathcal{S} \to \mathbf{2}) \to \mathcal{D}$): `that`
Tense($(\mathcal{S} \to \mathbf{2}) \to (\mathcal{S} \to \mathbf{2})$): `pres, past`
Modifier constructor($\mathcal{N} \to (\mathcal{N} \to \mathcal{N})$): `adv-a`

Basic Ontological Types

$\mathcal{D}$	Domain
$\mathcal{S}$	Situations
$\mathbf{2}$	Truth-value

Monadic Predicate $\mathcal{N} : \mathcal{D} \to (\mathcal{S} \to \mathbf{2})$

What is ULF?

A minimal step across from syntax to semantics in Episodic Logic

“Alice thinks that John nearly fell”, “Could you dial for me?”

ULFs

```
(|Alice| (((pres think.v)
            (that (|John| (nearly.adv-a (past fall.v)))))))

(((pres could.aux-v) you.pro
  (dial.v {ref1}.pro (adv-a (for.p me.pro)))) ?)
```

Basic Ontological Types

$\mathcal{D}$	Domain
$\mathcal{S}$	Situations
$\mathbf{2}$	Truth-value

Monadic Predicate $\mathcal{N} : \mathcal{D} \to (\mathcal{S} \to \mathbf{2})$

Entity($\mathcal{D}$): `|Alice|, |John|, you.pro, {ref1}.pro, me.pro`
n-ary predicate($\mathcal{D}^n \to (\mathcal{S} \to \mathbf{2})$): `think.v, fall.v, dial.v, for.p`
Predicate modifier($\mathcal{N} \to \mathcal{N}$): `nearly.adv-a, (adv-a (for.p me.pro))`
Sentence reifier($(\mathcal{S} \to \mathbf{2}) \to \mathcal{D}$): `that`
Tense($(\mathcal{S} \to \mathbf{2}) \to (\mathcal{S} \to \mathbf{2})$): `pres, past`
Modifier constructor($\mathcal{N} \to (\mathcal{N} \to \mathcal{N})$): `adv-a`

Also... *determiner, sentence modifier, connective, lambda abstract, predicate reifier*

What is ULF?

A minimal step across from syntax to semantics in Episodic Logic

"Alice thinks that John nearly fell", "Could you dial for me?"

ULFs

```
(|Alice| (((pres think.v)
           (that (|John| (nearly.adv-a (past fall.v)))))))

(((pres could.aux-v) you.pro
  (dial.v {ref1}.pro (adv-a (for.p me.pro)))) ?)
```

Basic Ontological Types

$\mathcal{D}$ Domain
$\mathcal{S}$ Situations
$\mathbf{2}$ Truth-value

Monadic Predicate $\mathcal{N} : \mathcal{D} \rightarrow (\mathcal{S} \rightarrow \mathbf{2})$

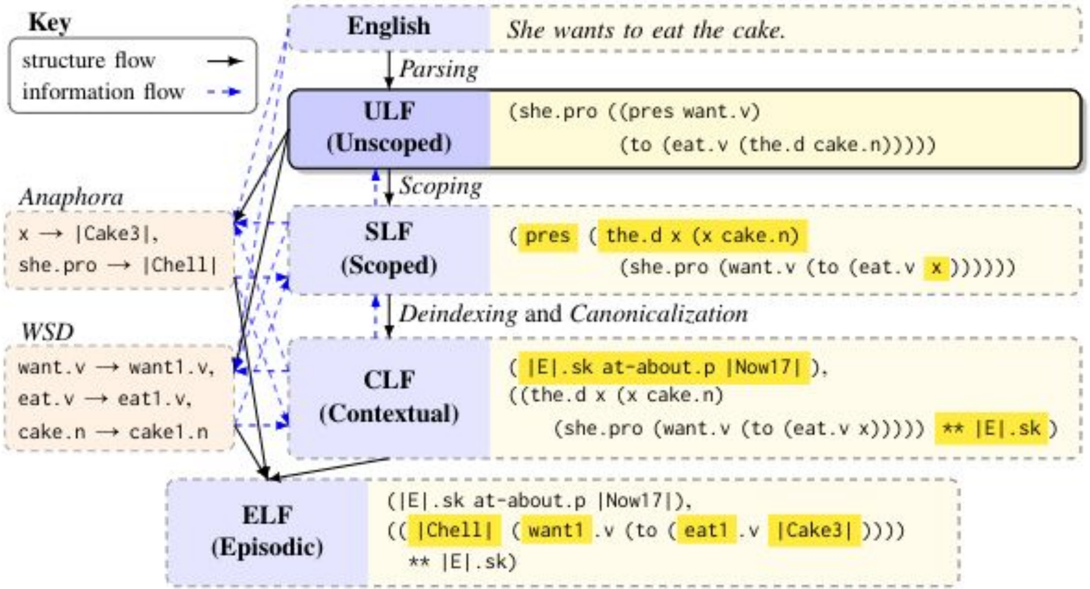
Entity($\mathcal{D}$): `|Alice|, |John|, you.pro, {ref1}.pr`
n-ary predicate($\mathcal{D}^n \rightarrow (\mathcal{S} \rightarrow \mathbf{2})$): `think.v, fall.v,`
Predicate modifier($\mathcal{N} \rightarrow \mathcal{N}$): `nearly.adv-a, (adv-a`
Sentence reifier($(\mathcal{S} \rightarrow \mathbf{2}) \rightarrow \mathcal{D}$): `that`
Tense($(\mathcal{S} \rightarrow \mathbf{2}) \rightarrow (\mathcal{S} \rightarrow \mathbf{2})$): `pres, past`
Modifier constructor($\mathcal{N} \rightarrow (\mathcal{N} \rightarrow \mathcal{N})$): `adv-a`

Captures the full predicate argument structure!

Also... *determiner, sentence modifier, connective, lambda abstract, predicate reifier*

How does ULF fit into EL interpretation?

ULF sets the foundation, but there's a lot left!



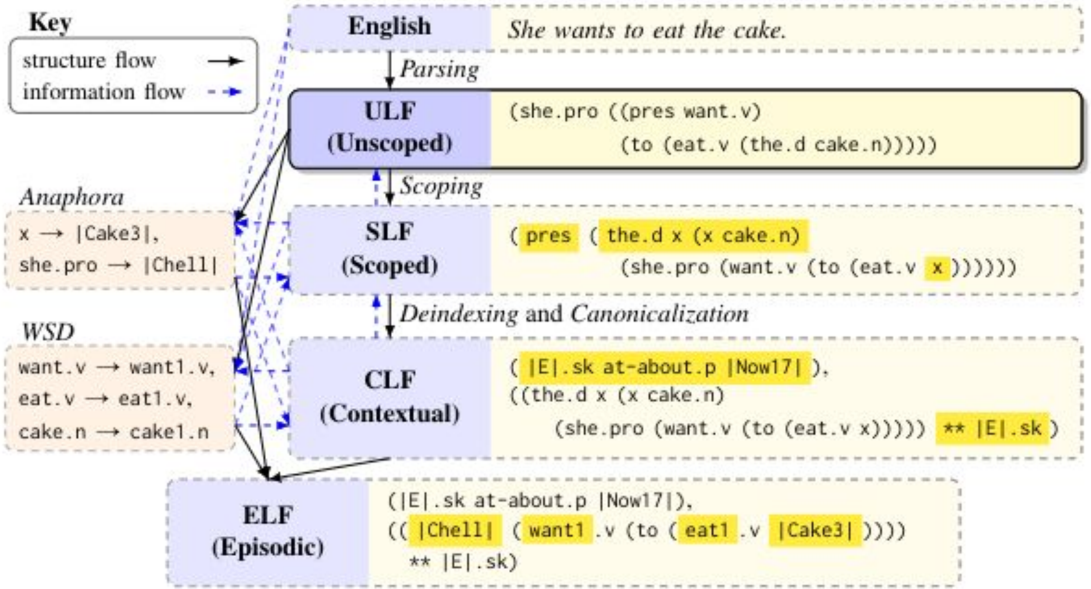
How does ULF fit into EL interpretation?

ULF sets the foundation, but there's a lot left!

We still have

1. Word sense disambiguation

"Chell eats a cake" vs
"This situation is eating at me"



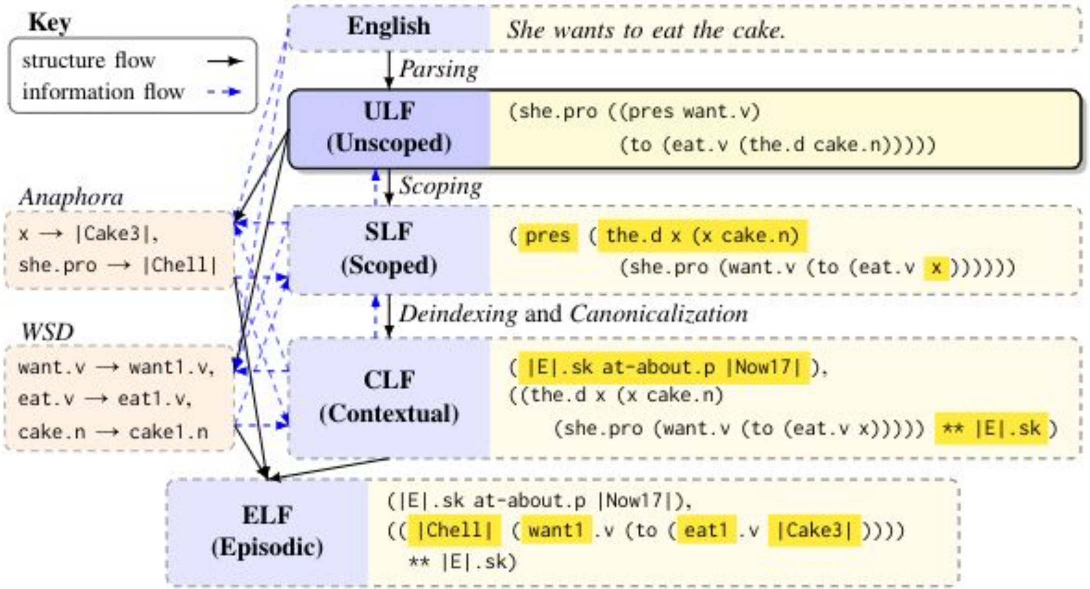
How does ULF fit into EL interpretation?

ULF sets the foundation, but there's a lot left!

We still have

2. Anaphora

Who does "she" refer to?



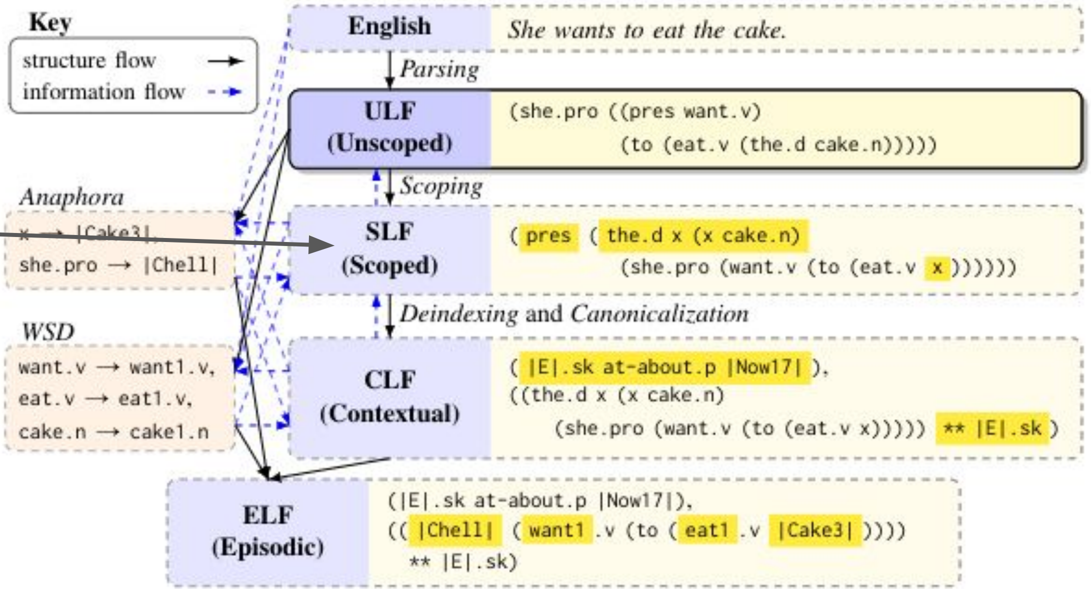
How does ULF fit into EL interpretation?

ULF sets the foundation, but there's a lot left!

We still have

3. Scoping

"Every child loves a dog"
Is there a single dog?
Or a different dog for each child?



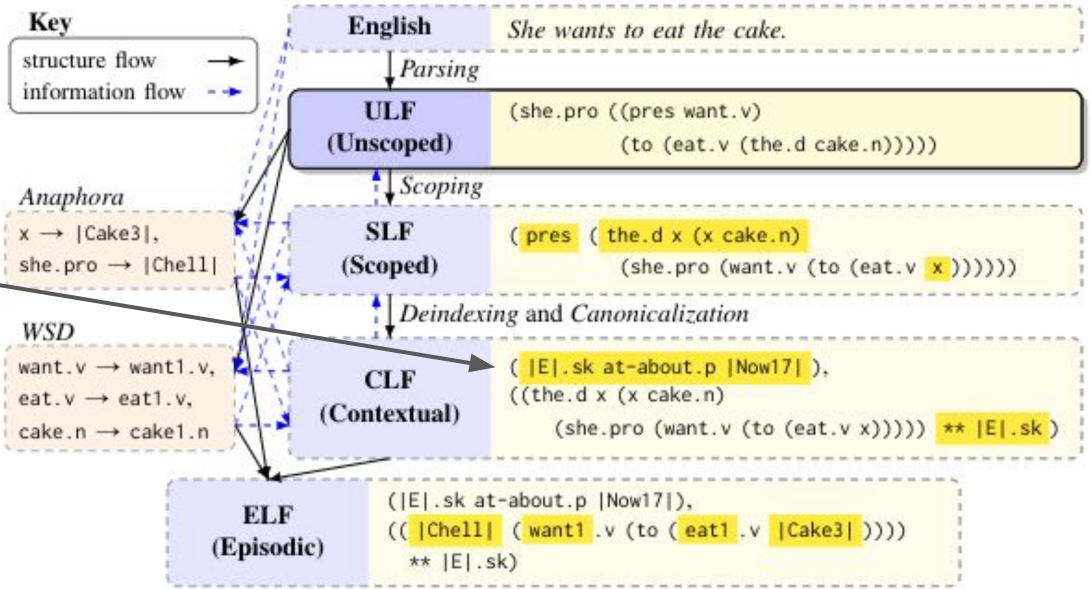
How does ULF fit into EL interpretation?

ULF sets the foundation, but there's a lot left!

We still have

4. Event structure

What are the events and how are they related causally and temporally?



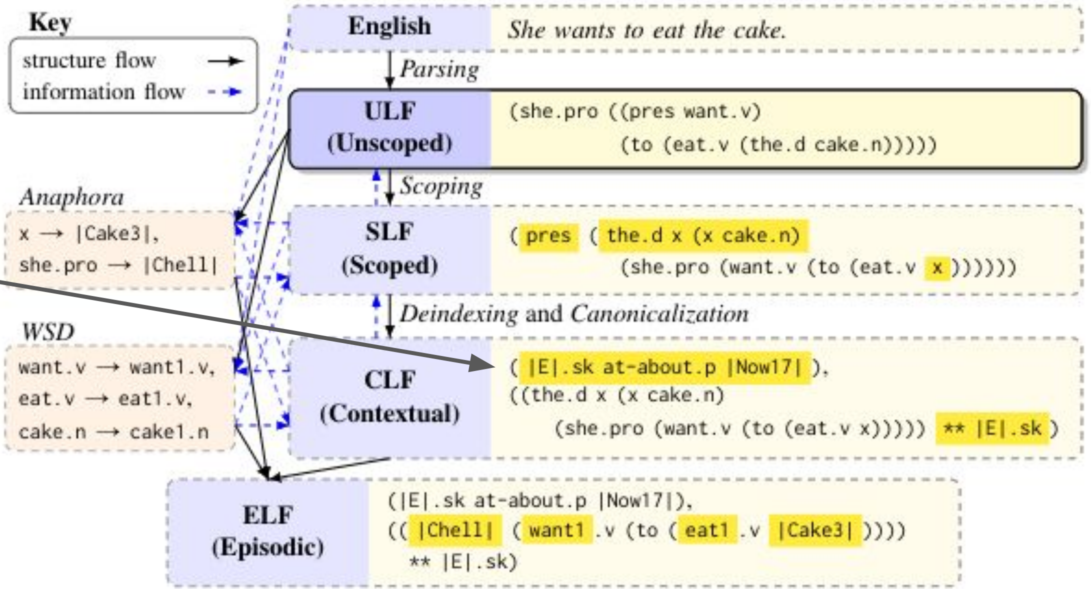
How does ULF fit into EL interpretation?

ULF sets the foundation, but there's a lot left!

We still have

5. Canonicalization

Reduce formulas to minimal propositions for inferential flexibility



Using ULF Directly for Inference

Wh-questions (presuppose that something happened)

"Who did you see yesterday?" > > > presupposes > > > *You saw someone yesterday*.

Using ULF Directly for Inference

Wh-questions (presuppose that something happened)

"Who did you see yesterday?" > > > presupposes > > > *You saw someone yesterday*.

Inference

"If a wh-question is uttered, the some-version of that sentence is true"

```
(all_wfulf w
  (((w ?) and (wh-sent? w))
    => (uninvert-sent! (wh-sent-to-some-sent! w))))
```

Starting with *"Who did you see yesterday?"* - `((sub who.pro ((past do.aux-s) you.pro (see.v *h yesterday.adv-e))) ?)`

We conclude *"You saw someone yesterday"* - `(you.pro ((past see.v) someone.pro yesterday.adv-e))`

Using ULF Directly for Inference

Wh-questions (presuppose that something happened)

"Who did you see yesterday?" > > > presupposes > > > *You saw someone yesterday.*

Inference

"If a wh-question is uttered, the some-version of that sentence is true"

```
(all_wfulf w
  (((w ?) and (wh-sent? w))
    => (uninvert-sent! (wh-sent-to-some-sent! w))))
```

Starting with *"Who did you see yesterday?"* - `((sub who.pro ((past do.aux-s) you.pro (see.v *h yesterday.adv-e))) ?)`

We conclude *"You saw someone yesterday"* - `(you.pro ((past see.v) someone.pro yesterday.adv-e))`

Also can do **counterfactuals** *"If I were rich …"* means that *I am not rich*

and **clause-taking verbs** *"I denouce x as y"* means that *I probably believe that x is y* and

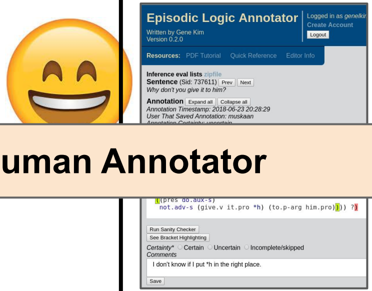
I want my listener to believe that x is y

and more!

Annotation and Parsing

Human ULF annotations...

“She wants to eat the cake”



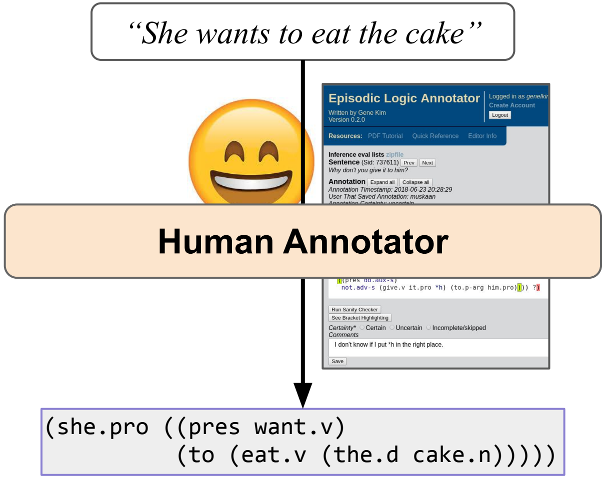
Human Annotator

```
(she.pro ((pres want.v)
          (to (eat.v (the.d cake.n)))))
```

Annotation and Parsing

Human ULF annotations...

- are fast (~8 min/sent)



Annotation and Parsing

Human ULF annotations...

- are fast (~8 min/sent)
- are consistent (up to 0.88 IAA)

"She wants to eat the cake"

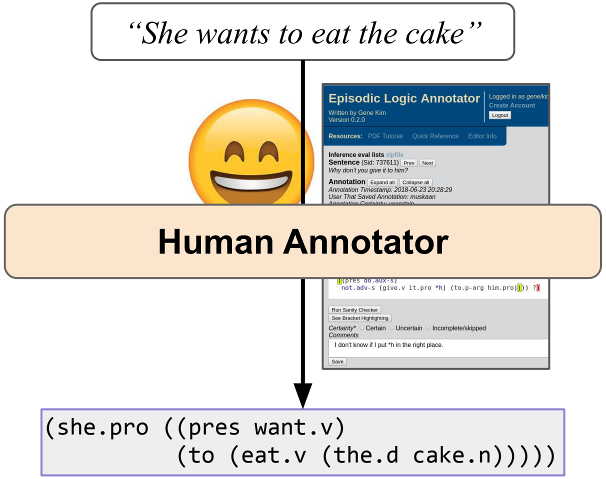
Human Annotator

```
(she.pro ((pres want.v)
          (to (eat.v (the.d cake.n)))))
```

Annotation and Parsing

Human ULF annotations...

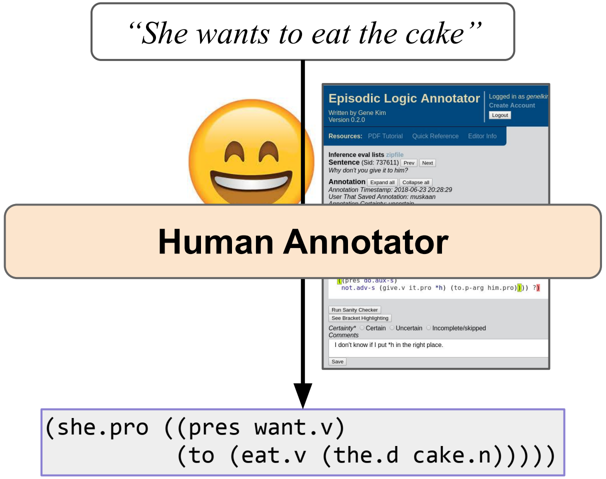
- are fast (~8 min/sent)
- are consistent (up to 0.88 IAA)
- number over 2000 sentences



Annotation and Parsing

Human ULF annotations...

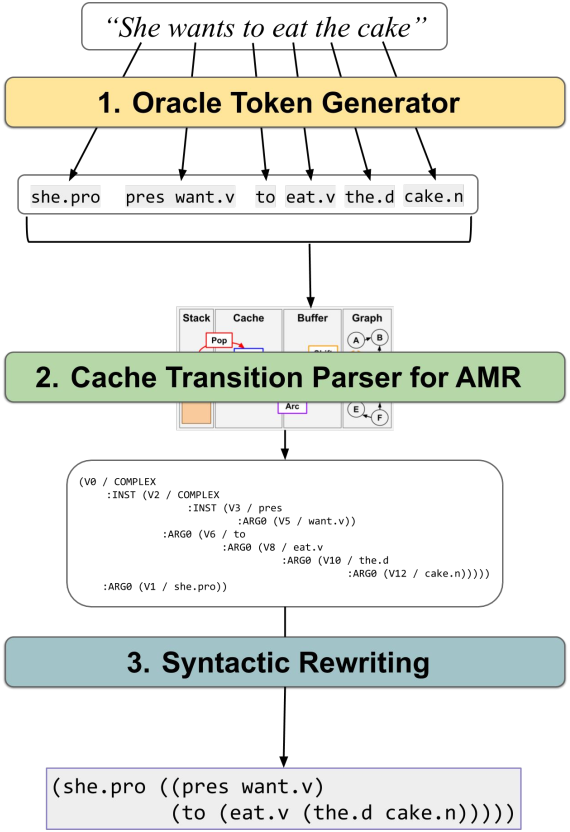
- are fast (~8 min/sent)
- are consistent (up to 0.88 IAA)
- number over 2000 sentences
- preliminary trained parsing results are promising



System Outline

900 sentence dataset

No ULF-specific features



Conclusions

- We presented an underspecified variant of Episodic Logic, ULF
- ULF is an intermediary representation to EL capturing predicate-argument structure while retaining some syntax
- ULF forms the foundation for further EL resolution, which can be done in context
- Annotating ULF is fast and reliable and automatic parsing seems feasible

Acknowledgements

We would like to thank Burkay Donderici, Benjamin Kane, Lane Lawley, Tianyi Ma, Graeme McGuire, Muskaan Mendiratta, Akihiro Minami, Georgiy Platonov, Sophie Sackstein, and Siddharth Vashishtha for raising thoughtful questions about prior iterations of this work. This work was supported by DARPA CwC subcontract W911NF-15-1-0542.

Introduction

Language understanding is a growing area of interest in NLP

Introduction

Language understanding is a growing area of interest in NLP

Question Answering: AI2 Reasoning challenge, RACE, bAbI, SQuAD, TriviaQA, NarrativeQA, FreebaseQA, WebQuestions, CommonsenseQA…

Dialogue: Amazon Alexa Challenge, Google Home, Microsoft Cortana

Inferring from Language: JOCI, SNLI, MultiNLI,...

Semantic Parsing: AMR, DRS Parsing (IWCS-2019 Shared Task), Cross-lingual Semantic Parsing (SemEval 2019 Shared Task 1)

Others: GLUE

Problems

Current state-of-the-art systems end up modeling artifacts rather than learning robust representations

Problems

Current state-of-the-art systems end up modeling artifacts rather than learning robust representations

Question Answering/Reading Comprehension ***(Jia & Liang 2017)***

Question: "What is the name of the quarterback who was 38 in Super Bowl XXXIII?"
Article: Super Bowl 50
Paragraph: "Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager."

Original Prediction: John Elway

Accuracy: 80.0%

Problems

Current state-of-the-art systems end up modeling artifacts rather than learning robust representations

Question Answering/Reading Comprehension ***(Jia & Liang 2017)***

Question: “What is the name of the quarterback who was 38 in Super Bowl XXXIII?”
Article: Super Bowl 50
Paragraph: “Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver’s Executive Vice President of Football Operations and General Manager.”

Original Prediction: John Elway

Accuracy: 80.0%

Unrelated Information

Question: “What is the name of the quarterback who was 38 in Super Bowl XXXIII?”
Article: Super Bowl 50
Paragraph: “Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver’s Executive Vice President of Football Operations and General Manager. *Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV.*”
Prediction under adversary: Jeff Dean

Accuracy: 34.2%

Problems

Current state-of-the-art systems end up modeling artifacts rather than learning robust representations

Inferring from Language *(Gururangan et al., 2018)*

Premise
Two dogs are running through a field

Entailment	**Neutral**	**Contradiction**
There are animals outdoors	Some puppies are running to catch a stick	The pets are sitting on a couch

Problems

Current state-of-the-art systems end up modeling artifacts rather than learning robust representations

Inferring from Language ***(Gururangan et al., 2018)***

Premise
Two dogs are running through a field

Entailment	**Neutral**	**Contradiction**
There are animals outdoors	Some puppies are running to catch a stick	The pets are sitting on a couch

Ignoring Premise

Entailment Artifacts
Generalization & Shortening

Neutral Artifacts
Modifiers & Purpose Clauses

Contradiction Artifacts
Negation & Dog-to-Cat

Accuracy:
52.3% (MultiNLI)
67.0% (SNLI)

Solutions?

A few approaches to deal with these problems are being explored

1. ***Inducing bias***

   Bias toward relevance, style, repetition, and entailment… somehow

2. ***Common sense***

   Current system look like a “mouth without a brain”, let’s add a brain

3. ***Evaluate the model on unseen tasks***

   Check if the model generalizes beyond the exact dataset format

Solutions?

A few approaches to deal with these problems are being explored

1. ***Inducing bias***

   Bias toward relevance, style, repetition, and entailment… somehow

2. ***Common sense***

   Current system look like a “mouth without a brain”, let’s add a brain

3. ***Evaluate the model on unseen tasks***

   Check if the model generalizes beyond the exact dataset format

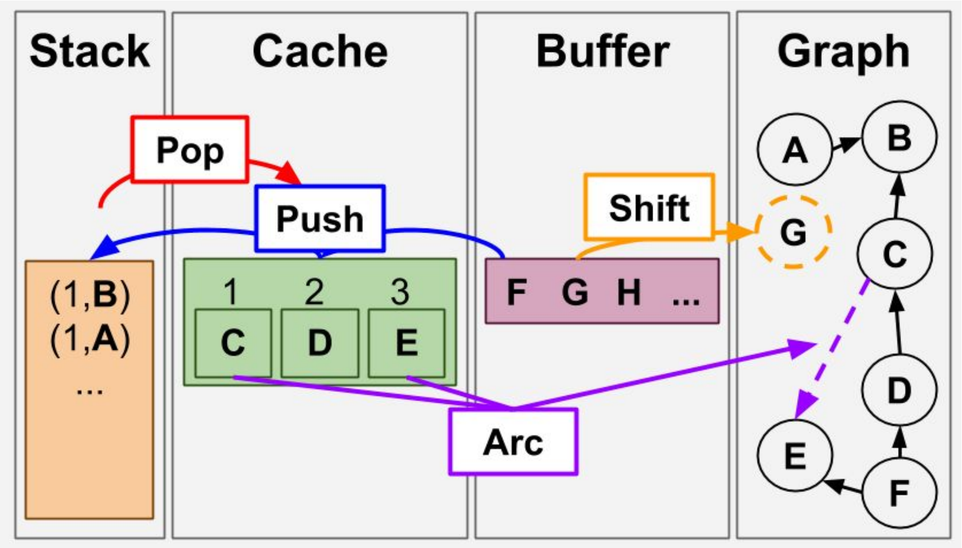
(All of the above assume a core neural/machine learning architecture)

4. ***Symbolic semantic representation***

   Directly encode linguistic information and logical reasoning through the representation

Cache Transition Parser

A transition system for parsing graphs using a fixed-sized cache.



Pop: pops the top element from stack to its indexed position in cache

Shift: moves the front of the buffer by one and adds a vertex to the graph for the front element

Push: moves the front of the buffer to the cache and pushes the old cache value to the stack

Arc: forms an arc between a given index of the cache and the rightmost element of the cache

Annotation and Parsing

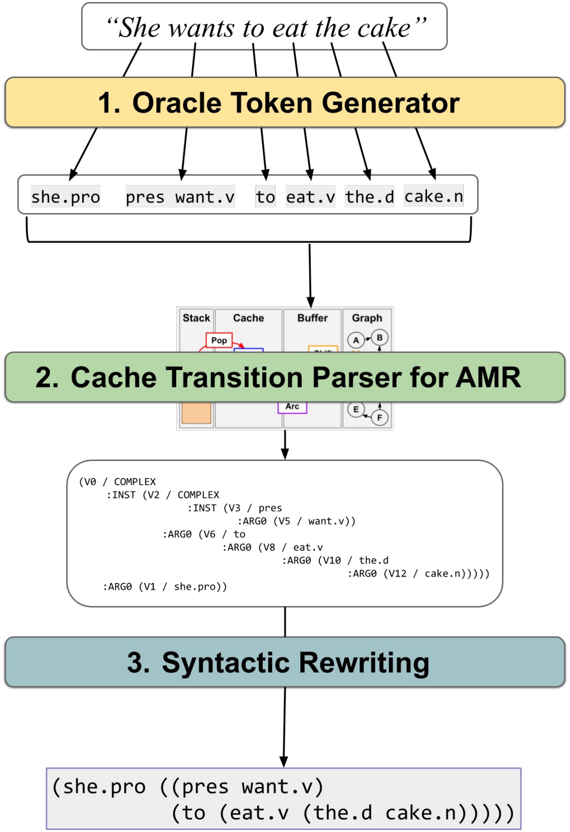
Human ULF annotations...

- are fast (~8 min/sent)
- are consistent (up to 0.88 IAA)
- number over 2000 sentences

Preliminary parsing experiment

- Based on an AMR cache transition parser (Peng et al. 2018)

System Outline



Annotation and Parsing

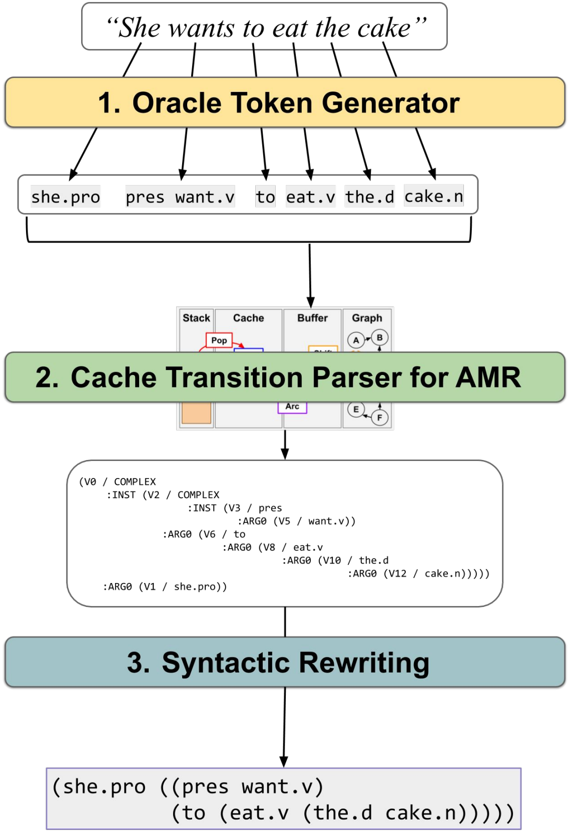
Human ULF annotations...

- are fast (~8 min/sent)
- are consistent (up to 0.88 IAA)
- number over 2000 sentences

Preliminary parsing experiment

- Based on an AMR cache transition parser (Peng et al. 2018)
- No added assumptions about ULF structure

System Outline



Annotation and Parsing

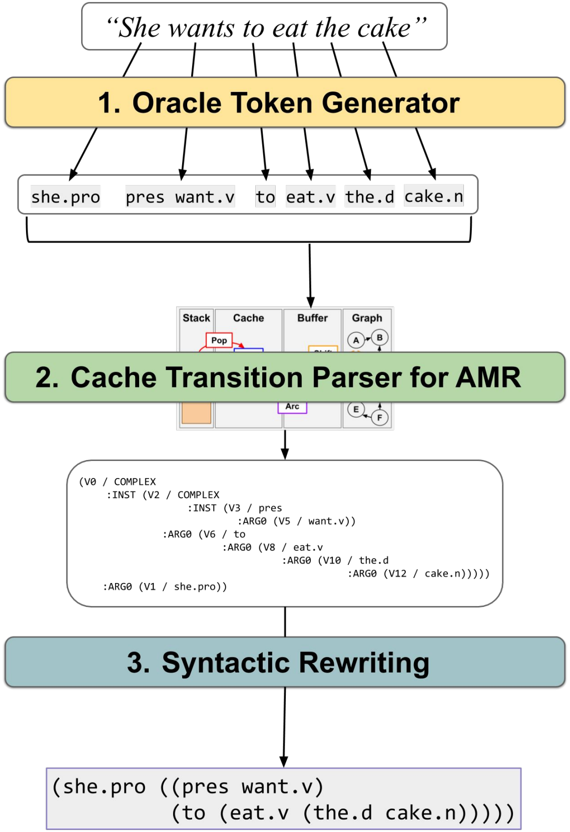
Human ULF annotations...

- are fast (~8 min/sent)
- are consistent (up to 0.88 IAA)
- number over 2000 sentences

Preliminary parsing experiment

- Based on an AMR cache transition parser (Peng et al. 2018)
- No added assumptions about ULF structure
- Dataset of 900 sentences

System Outline



Annotation and Parsing

Human ULF annotations...

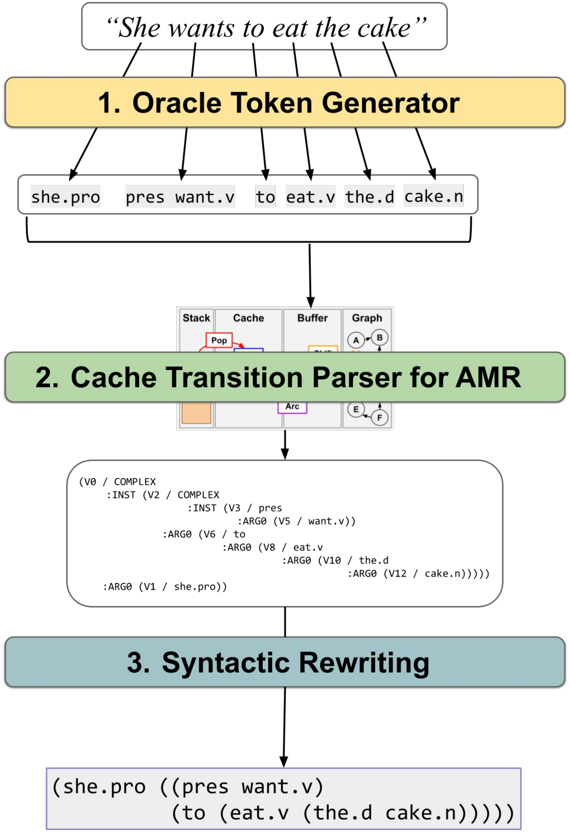
- are fast (~8 min/sent)
- are consistent (up to 0.88 IAA)
- number over 2000 sentences

Preliminary parsing experiment

- Based on an AMR cache transition parser (Peng et al. 2018)
- No added assumptions about ULF structure
- Dataset of 900 sentences

0.738 Average partial match

System Outline



What is ULF?

Relaxations/Macros

TODO