# Intension, Attitude, and Tense Annotation in a High-Fidelity Semantic Representation

Gene Kim and Lenhart Schubert

Presented by: Gene Kim
April 2017

# Project Overview

Project: Annotate a large, topically varied dataset of sentences (e.g. Brown corpus) with *unscoped logical form* (ULF) representations.

- ULF: captures semantic type structure and marks scoping and anaphoric ambiguity

Goal: Develop a reliable, general-purpose ULF transducer, including attitudes, quantifiers, modifiers, tense, etc.

# Project Overview

Project: Annotate a large, topically varied dataset of sentences (e.g. Brown corpus) with *unscoped logical form* (ULF) representations.

- ULF: captures semantic type structure and marks scoping and anaphoric ambiguity

Goal: Develop a reliable, general-purpose ULF transducer, including attitudes, quantifiers, modifiers, tense, etc.

Example Annotation

"Alice thinks that John nearly fell"

```
[Alice.prp (<pres think.v>
                (that [John.prp (nearly.adv <past fall.v>)]))]
```

# Project Overview

Project: Annotate a large, topically varied dataset of sentences (e.g. Brown corpus) with *unscoped logical form* (ULF) representations.

- ULF: captures semantic type structure and marks scoping and anaphoric ambiguity

Goal: Develop a reliable, general-purpose ULF transducer, including attitudes, quantifiers, modifiers, tense, etc.

Example Annotation

"Alice thinks that John nearly fell"

```
[Alice.prp (<pres think.v>
                (that [John.prp (nearly.adv <past fall.v>)]))]
```

Intensional modifier

# Project Overview

Project: Annotate a large, topically varied dataset of sentences (e.g. Brown corpus) with *unscoped logical form* (ULF) representations.

- ULF: captures semantic type structure and marks scoping and anaphoric ambiguity

Goal: Develop a reliable, general-purpose ULF transducer, including attitudes, quantifiers, modifiers, tense, etc.

Example Annotation

"Alice thinks that John nearly fell"

```
[Alice.prp (<pres think.v>
```

Attitude predicate

```
                (that [John.prp (nearly.adv <past fall.v>)]))]
```

# Project Overview

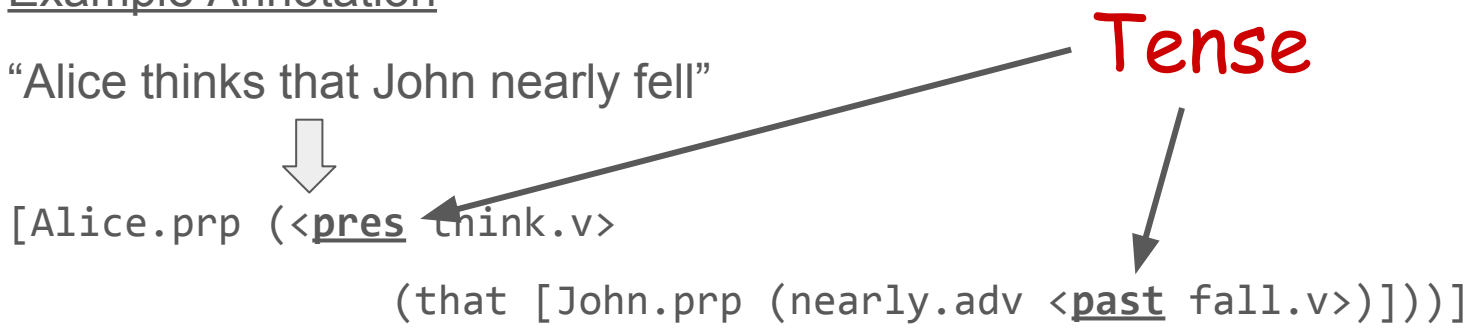
Project: Annotate a large, topically varied dataset of sentences (e.g. Brown corpus) with *unscoped logical form* (ULF) representations.

- ULF: captures semantic type structure and marks scoping and anaphoric ambiguity

Goal: Develop a reliable, general-purpose ULF transducer, including attitudes, quantifiers, modifiers, tense, etc.

Example Annotation

"Alice thinks that John nearly fell"



# Expected Inferences

*Intension*

John nearly fell ⇏ John fell

Surprisingly, Koko is intelligent ≠ Koko is surprisingly intelligent

# Expected Inferences

*Intension*

John nearly fell ⇏ John fell

Surprisingly, Koko is intelligent ≠ Koko is surprisingly intelligent

***Not possible by intersective modification (e.g. OWL-DL)***

# Expected Inferences

*Intension*

John nearly fell ⇏ John fell

Surprisingly, Koko is intelligent ≠ Koko is surprisingly intelligent

*Attitude*

Alice {thinks,believes,claims} that John nearly fell ⇏ John nearly fell

# Expected Inferences

*Intension*

John nearly fell ⇏ John fell

Surprisingly, Koko is intelligent ≠ Koko is surprisingly intelligent

*Attitude*

Alice {thinks,believes,claims} that John nearly fell ⇏ John nearly fell

***Hobbesian Logical Form conflates events and propositions***

# Expected Inferences

*Intension*

John nearly fell ⇏ John fell

Surprisingly, Koko is intelligent ≠ Koko is surprisingly intelligent

*Attitude*

Alice {thinks,believes,claims} that John nearly fell ⇏ John nearly fell

*Tense*

John nearly fell ⇒ Sometime in the past w.r.t. utterance, the event “John nearly falls” occurred

# Expected Inferences

*Intension*

John nearly fell ⇏ John fell

Surprisingly, Koko is intelligent ≠ Koko is surprisingly intelligent

*Attitude*

Alice {thinks,believes,claims} that John nearly fell ⇏ John nearly fell

*Tense*

John nearly fell ⇒ Sometime in the past w.r.t. utterance, the event “John nearly falls” occurred

***Tense not represented in AMR***

# Expected Inferences

*Intension*

John nearly fell ⇏ John fell

Surprisingly, Koko is intelligent ≠ Koko is surprisingly intelligent

*Attitude*

Alice {thinks,believes,claims} that John nearly fell ⇏ John nearly fell

*Tense*

John nearly fell ⇒ Sometime in the past w.r.t. utterance, the event “John nearly falls” occurred

- *We will see how the annotation and EL semantics achieve these*

# Current Project State

- We don’t have any annotations at the current stage since the annotation guidelines are under revision and the annotation tools are under construction.

- We performed preliminary annotations which indicated that our framework can semantically capture the information we seek to annotate, but needs to be made more transparent to reduce annotator burden.
    - On Brown and Little Prince corpus

# Episodic Logic (EL)

- Extended FOL.
- Closely matches expressivity of natural languages.
- Suitable for deductive, uncertain, and Natural-Logic-like inference (Morbini and Schubert, 2009; Schubert and Hwang, 2000; Schubert, 2014). A fast and comprehensive theorem prover, EPILOG, is already available.
- An effective representation for encoding verb gloss axioms from WordNet that enable intuitive inferences (Kim and Schubert, 2016).
    - Greater expressivity shown to appropriately handle intensional modification where many other methods fail.

# Current Limitation of Using EL

So EL sounds like a great representation, but...

# Current Limitation of Using EL

So EL sounds like a great representation, but...

**the current hand-crafted EL interpreter is too error-prone.**

# Current Limitation of Using EL

So EL sounds like a great representation, but...

**the current hand-crafted EL interpreter is too error-prone.**

**1 in 3** EL interpretations of glosses contained errors in Kim and Schubert's verb gloss axiom generation system.

- Many linguistic phenomena went unhandled because they didn't appear in the EL interpreter development set.

# Why ULF?

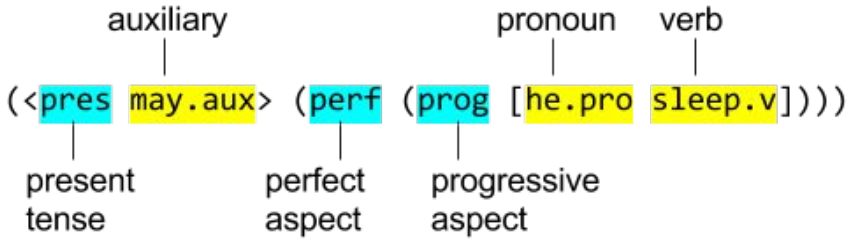
- ULF is a preliminary EL representation with syntactic marking of ambiguity. ULF primarily captures the semantic type structure.
- Semantic type structure is recoverable at a sentence level.
- Replacing indexical expressions and disambiguating quantifier scopes, word senses, and anaphora generally *require the sentence context to resolve*.

# ULF Syntax

- Atoms
  - *w/ POS suffix* - lexical entries
  - *w/o POS suffix* - operators corresponding to morpho-syntactic phenomena.

- 3 types of brackets
  - *round brackets* - prefixed operators
  - *square brackets* - infixed operators (only used for sentential formulas)
  - *angle brackets* - unscoped (prefixed) operators

*"He may have been sleeping"*



# Intension, Attitude, and Tense Semantics in EL/ULF

# Semantics of Intensional Modifiers

- Predicate modifiers map predicate meanings to predicate meanings.
- Predicates interpreted as functions from individuals and a situation to truth values
  - Arguments are curried with the situation applied last
- Enables proper interpretation of non-intersective modifiers (e.g. *very, fairly, big*) and in particular, intensional ones (e.g. *nearly, fake*).

```
(all x [[x (fake.a flower.n)] ⇒
    [(not [x flower.n]) and.cc [x (resemble.v flower.n)]]])
```

# Semantics of Intensional Modifiers

- Intensional sentence modifiers map sentence intensions to sentence intensions.

  “John is probably angry”
  `(probably.adv [John.prp (<pres be.v> angry.a)])`

  “According to the NYT, John is angry”
  `((adv-s (according_to.a <the.d _NYT.n>)) [John.prp (<pres be.v> angry.a)])`

- Extensional sentence modifiers become simple predications about episodes upon “deindexing”.

  “Most people left at dawn”
  `((adv-e (at.p dawn.n)) [<most.d (plur person.n)> <past leave.v>])`

# Semantics of Attitude Predicates

Attitude predicates (e.g. *assert, believe,* and *assume*) are relations between an individual and a proposition.

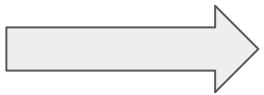
Proposition ≠ Episode in EL

- Proposition: reified sentence intension - informational entities
- Episode: real entities occupying time intervals.

Once a proposition is formed from a sentence with the **`that`** operator, it has the semantic type of an individual.

# Semantics of Tense

- Tenses are extensional sentence modifiers. They become simple predications about episodes upon “deindexing”.

| ULF | | EL (after deindexing) |
|---|---|---|
| `(past Φ)` |  | `[[Φ’ ** e] and.cc [e before NOW]]` |
| `(pres Φ)` | | `[[Φ’ ** e] and.cc [e at-about NOW]]` |

- Treat *will* as a present-tense modal auxiliary rather than “future” tense. “will” becomes `<pres will.aux>` (Hwang & Schubert ‘94).

# Annotating Intension, Attitude, and Tense in ULF

# Annotating Intension

- Predicate and sentence modifiers are different semantic types!
- Most adverbials can only be one of the two types.
  - Predicate-only: manner adverbs (e.g. confidently, awkwardly)
  - Sentence-only: speaker commentary (e.g. undoubtedly, in my opinion)
- But some can be both!
  - can, may, could, surprisingly, …. (lots of auxiliaries!)
  - Depends on the lexical entries as well as the syntax

1a. “Mary confidently spoke up”
1b. “Mary undoubtedly spoke up”
2a. “Koko is surprisingly intelligent”
2b. “Surprisingly, Koko is intelligent”

# Annotating Intension

Guidelines for distinguishing predicate and sentence modifiers

- Predicate modifiers - modified predicate affects what is said about the subject
  - obligation and permission (e.g. I **can** run, You **may** sit down)
  - modification dependent on the predicate (e.g. That's a **fake** diamond)

- Sentence modifiers - modifier only affects what is said about the sentence
  - necessity and possibility (e.g. That volcano **could** erupt)
  - temporal and frequency modalities (e.g. I run **regularly**)

# Annotating Intension

- Annotate *predicate* modifiers by scoping them around the modified predicate.

  “Mary confidently spoke up”
  `[Mary.prp (confidently.adv <past speak_up.v>)]`

- Annotate *sentence* modifiers by scoping them around the modified sentence.

  “Mary undoubtedly spoke up”
  `(undoubtedly.adv [Mary.prp <past speak_up.v>])`

# Annotating Attitudes

Recognize when a sentence is functioning as a proposition and annotate with `that` operator.

Propositions

- We know *that there's water on Mars*.
- I'm sure *(that) you've heard of him*.

Not Propositions

- He's the man *that I met yesterday*. (relative clause)
- I ate so much *that I got a stomachache*. (adverbial clause)

# Annotating Attitudes

Recognize when a sentence is functioning as a proposition and annotate with `that` operator.

Propositions

- We know *that there's water on Mars*.

```
[we.pro <pres know.v>
    (that ((adv-e (on.p Mars.prp))
           [there.pro <pres be.v> (k water.n)]))]
```

- I'm sure *(that) you heard him*.

```
[i.pro (<pres be.v> sure.a)
    (that [you.pro <past hear.v> him.pro])]
```

# Annotating Aspect

Aspect is generally captured by lexical entries (e.g. *daily, used to*)...

# Annotating Aspect

Aspect is generally captured by lexical entries (e.g. *daily, used to*)...

They’re Sentence Modifiers!

*We just saw how to handle this.*

# Annotating Aspect

Special Cases - marked morpho-syntactically in English, so we introduce special operators.  They're sentence modifiers like the lexicalized aspect operators.

- Perfect - `perf`
  - Marked with "have" + VB past participle
- Progressive - `prog`
  - Marked with "be" + VB-ing

# Annotating Tense

- Tense regarded as an unscoped operator to stay close to surface form.
- Tense annotated on the verb that bears the tense inflection in surface text.
  *This is always the first verb of a tensed verb phrase.*
  - “He is sleeping”
    `(<pres prog> [he.pro sleep.v])`
  - “He has left Rome”
    `(<pres perf> [he.pro (leave.v Rome.c)])`
  - “He had left Rome”
    `(<past perf> [he.pro (leave.v Rome.c)])`
  - “He has been sleeping”
    `(<pres perf> (prog [he.pro sleep.v]))`
  - “He may have been sleeping”

```
(<pres may.aux>
     (perf (prog [he.pro sleep.v])))
```

# Reducing Annotator Burden (on-going)

# Simplifications

- Phrasal bracketing driven annotation

```
(Mary (confidently (spoke up))) →
(Mary.nnp (confidently.rb (spoke.vbd up.prt))) →
[Mary.prp (confidently.adv-a <past speak_up.v>)]
```

- Relax well-formedness constraints where the real formula is recoverable
- Introduce macros to eliminate word reordering

# Phrasal Bracketing Driven Annotation

"Alice thinks that John nearly fell"

1. Group syntactic constituents

```
(Alice (thinks (that (John (nearly fell)))))
```

2. POS tagging

```
(Alice.nnp (thinks.vbz (that.in (John.nnp (nearly.rb fell.vbd)))))
```

3. Convert POS to logical types and separate morpho-syntactic markings as logical operators

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

(post-process) Update parentheses

```
[Alice.prp (<pres think.v> (that [John.prp (nearly.adv-a <past fall.v>)]))]
```

# Conclusions

- We introduced an on-going project of developing a ULF transducer to enable robust and scalable applications using EL.
- We presented annotation representations for intension, attitude and tense in ULF and discussed challenges.
- We discussed some strategies for reducing the burden on the annotators that we are currently exploring to generate reliable annotations.

# Acknowledgements

The work was supported by a Sproull Graduate Fellowship from the University of Rochester, DARPA CwC subcontract W911NF-15-1-0542, and NSF grant IIS-1543758.


# Semantic Representation Details

(Hobbs, 2008)[1] - Hobbsian Logical Form (HLF)

- Conflates events and propositions

  ***John's telling of his favorite joke** would make most listeners laugh; **the proposition that he did so** would not.*

- Interpretation of quantifiers in terms of "typical elements" can lead to contradiction

  **"Typical elements" of sets are defined as individuals that are not members of those sets, but have all the properties *shared* by members of the sets.**

  Consider S = {0,1}.
  Share property of being in S.
  **Typical element must be in S, but by definition, not in S!!!**

# Semantic Representation Details

(Allen et al. 2013)[2] - Description Logic (OWL-DL)

- OWL-DL: Web Ontology Language - Description Logic
  - Designed for ontologies, not full natural language
- Handling of predicate/sentence reification, predicate modification, self-reference, and uncertainty is unsatisfactory
  - Intersective predicate modification
    ***"whisper loudly"*** → ***whisper*** ⊓ $\forall_{of}$***-1.(loudly)*** → ***speak*** ⊓ $\forall_{of}$***-1.(softly)*** ⊓ $\forall_{of}$***-1.(loudly)***
  - Tree-shaped models requirement
    - *partOf* and *contains* relations in opposite directions not possible
    - review: "refresh one's memory" - self-reference
  - Reification
    - Classes and individuals are disjoint → can't refer to a class as an individual