

MRS Semantics (FOL and beyond)

Alexandre Rademaker Guilherme Lima

IBM Research

Jun, 2023

Motivation I

The DELPH-IN Consortium is a collaboration among computational linguists from research sites world-wide **working on ‘deep’ linguistic processing of human language.**

The goal is the **combination of linguistic and statistical processing methods for getting at the meaning of texts and utterances.**

The partners have adopted Head-Driven Phrase Structure Grammar (HPSG) and Minimal Recursion Semantics (MRS), two advanced models of formal linguistic analysis.

Motivation II

From the Computational linguistics perspective, no questions!

From the NLP applications, two threads of work:

- ▶ How to produce deep semantic representations from text (language, domain, etc)?
- ▶ How to use deep semantic representations? What are the killer applications?

Motivation III

No logical reasoning from MRS at DELPH-IN applications.

LKB preliminary implementation on MRS to FOL.

The Glue semantics for HPSG is another approach.

Not talking about many other approaches not directly (or indirectly) related to HPSG syntactic theory: DRT, CCG etc.

Motivation IV

“For the purposes of this paper, we will take the object language to be predicate calculus with generalized quantifiers [2].

(2) This should not be taken as suggesting that we think it is impossible or undesirable to give MRS a model-theoretic semantics directly. But given that our main interest is in the computational and compositional use of MRS, it is more appropriate to show how it relates to a relatively well-known language, such as predicate calculus, rather than to attempt a direct interpretation.”

(Minimal Recursion Semantics: an Introduction, Ann Copestake, Dan Flickinger, Carl Pollard & Ivan A. Sag)

Motivation V

Converting MRS output to a logical form (Ann's comments)

- ▶ FOL is not sufficient for arbitrary sentences
- ▶ MRS can be translated into many actual logics
- ▶ no existing logic is adequate for all the phenomena of NL
- ▶ individual logics can capture all the individual phenomena but these logics don't work out as single well-behaved logic.
- ▶ but . . . start from DMRS, compositional without any variables. Particular relevant to event vars. (?)

Motivation VI

“[...] Overall, then, an adequate model of natural language understanding must incorporate commonsense reasoning with linguistic and non-linguistic information. Furthermore, this reasoning must be **defeasible**, because the content of an utterance depends critically on contextual information that's hidden rather than observable. Thus, content is estimated under uncertainty and one can therefore change one's mind on gathering subsequent observable evidence about the context.”

(Linguistic Fundamentals for Natural Language Processing II,
Emily M. Bender and Alex Lascarides)

Motivation VII

Short-term applications, IBM 'risk assessment questionnaire'.

- ▶ What is the Log Kow of the compound?
- ▶ What is the water solubility of the compound?

The relation between water solubility and bioaccumulation potential is defined as the Kow parameter. So these questions are not entirely independent.

RTE, as you will see.

type theories I

We want to explore **Dependent Type Semantics**.

“formal grammar begins with what is well understood formally, and then tries to see how this formal structure is manifested in natural language, instead of starting with natural language in all its unlimitedness and trying to force it into some given formalism.”

(Type-theoretic Grammar, Aarne Ranta - preface describing the conversation with Per Martin-Löf)

<https://www.grammaticalframework.org/>

type theories II

FOL tends to lean heavily on the supply of a reasonable domain. But when quantification occurs over a variety of domains? We need to imagine some vast pool of individuals to pull out various people, times, and events.

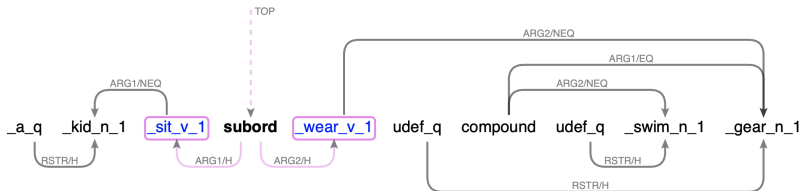
“Everyone has at some time seen some event that shocked them”

Small wonder CS has looked to the discipline of types. Just as we want a “person” in response to “Who?”, and a “place” in response to “Where?”, programs need to compute with terms of the right type.

([The n-Category Café](#), Feb 11, 2020)

type theories III

“A kid is sitting wearing swim gear”



```
[ subord<17:34> LBL: h1 ARG0: e9 ARG1: h10 ARG2:
h11 ]
```

type theories IV

We want to build on top of modern tools like [Lean](#) and [Coq](#).

...but we started with Python!

What about the LLM? Enrich the pipeline? Replace it?

ULKB I

ULKB is a Python implementation of simple type theory (STT).

ULKB provides KB Federating the major public knowledge graphs (DBPedia, Wikidata etc)

The language

– Expressions

Terms, formulas, types

– Annotations

Expression metadata: *KG links, weights, bounds, ...*

– Theories

Axioms, theorems, definitions

– HOL kernel

Type-checking, computation, deductive rules

– Import / export plugins

Parsing, serialization, conversion

		<i># Python API</i>
<code>type ::=</code>	<code><u>tvar</u></code>	<code>TypeVariable(id)</code>
	<code> base</code>	<code>BaseType(id)</code>
	<code> bool</code>	<code>BoolType()</code>
	<code> type → type</code>	<code>FunctionType(type, type)</code>
<code>term ::=</code>	<code>const</code>	<code>Constant(id, type)</code>
	<code> var</code>	<code>Variable(id, type)</code>
	<code> term term</code>	<code>Application(term, term)</code>
	<code> λ var ⇒ term</code>	<code>Abstraction(var, term)</code>
<code>form ::=</code>	<code>term = term</code>	<code>Equal(term, term)</code>
	<code> T</code>	<code>Truth()</code>
	<code> ⊥</code>	<code>Falsity()</code>
	<code> ¬ form</code>	<code>Not(form)</code>
	<code> form ∧ form</code>	<code>And(form, form)</code>
	<code> form ∨ form</code>	<code>Or(form, form)</code>
	<code> form → form</code>	<code>Implies(form, form)</code>
	<code> form ↔ form</code>	<code>Iff(form, form)</code>
	<code> ∃ var, term</code>	<code>Exists(var, form)</code>
	<code> ∀ var, term</code>	<code>Forall(var, form)</code>

ULKB III

Extends FOL:

- ▶ types, boolean, functions (\rightarrow)
- ▶ quantification over arbitrary types (including functions and predicates and individuals)
- ▶ high-order functions and predicates

The logic of [Isabelle HOL](#).

ULKB IV

“A man is walking” (properties TENSE, MOOD, PROG are still annotations)

$$\exists x3 : u, \text{_man_n_1 } x3 \wedge (\exists e2 : u, \text{_walk_v_1 } e2 \ x3)$$

“A kid is sitting wearing swim gear.”

$$\exists e9, \text{subord } e9$$

$$(\exists x3, \text{_kid_n_1 } x3 \wedge (\exists e2, \text{_sit_v_1 } e2 \ x3))$$

$$(\exists x21, \text{_swim_n_1 } x21$$

$$\wedge (\exists x15, (\exists e20, \text{compound } e20 \ x15 \ x21 \wedge \text{_gear_n_1 } x15)$$

$$\wedge (\exists e13, \text{_wear_v_1 } e13 \ x15)))$$

MRS Logic I

MRS-Logic is a Python Library that uses [PyDelphin](#), [Ace](#), [Utool](#), [UKB](#) and ULKB to convert sentences into STT formulas.

ULKB can call provers, e.g., E, Vampire, Z3, LNN, etc.

ULKB can serialize to RDF/OWL, TPTP, SMTLIB2, etc.

MRS Logic II

- ▶ scope resolution
- ▶ conversion to STT (recursive top-down on the scope tree)
 - ▶ generalized quantifiers
 - ▶ NL connectives are kept as non-logical predicates
 - ▶ introduction of the 'e' variables (narrow scope)
 - ▶ uninstantiated arguments, the 'i', 'u', 'p' variables
 - ▶ special predicates (e.g. `neg` (\neg), `_if_x_then` (\rightarrow))
 - ▶ high-order (`subord`, `nominalization`, `coordinations` etc)
 - ▶ `card` and `named` (strings and numbers in the CARG)
- ▶ word-sense disambiguation via UKB (annotations)

example I

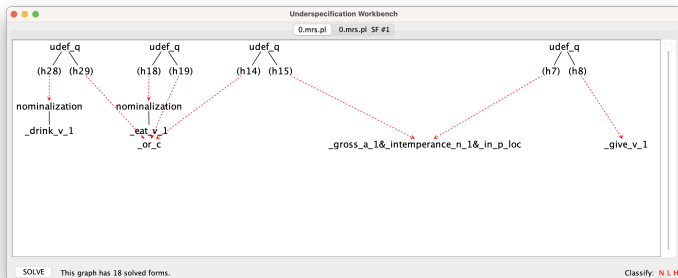
“given to gross intemperance in eating or drinking”

(definition of crapulous in WordNet 3.0, 616 readings, some of those with 76 possible scopes)

```
LTOP: h0
INDEX: e2
RELS: < [ _give_v_1<0:5>      LBL: h1 ARG0: e2 ARG1: i4 ARG2: i3 ARG3: x5 ]
[ udef_q<9:49>                LBL: h6 ARG0: x5 RSTR: h7 BODY: h8 ]
[ _gross_a_1<9:14>            LBL: h9 ARG0: e10 ARG1: x5 ]
[ _intemperance_n_1<15:27>    LBL: h9 ARG0: x5 ]
[ _in_p_loc<28:30>            LBL: h9 ARG0: e11 ARG1: x5 ARG2: x12 ]
[ udef_q<31:49>                LBL: h13 ARG0: x12 RSTR: h14 BODY: h15 ]
[ udef_q<31:37>                LBL: h16 ARG0: x17 RSTR: h18 BODY: h19 ]
[ _eat_v_1<31:37>              LBL: h20 ARG0: e21 ARG1: i22 ARG2: i23 ]
[ nominalization<31:37>       LBL: h24 ARG0: x17 ARG1: h20 ]
[ _or_c<38:40>                 LBL: h25 ARG0: x12 ARG1: x17 ARG2: x26 ]
[ udef_q<41:49>                LBL: h27 ARG0: x26 RSTR: h28 BODY: h29 ]
[ _drink_v_1<41:49>           LBL: h30 ARG0: e31 ARG1: i32 ARG2: i33 ]
[ nominalization<41:49>       LBL: h34 ARG0: x26 ARG1: h30 ] >
HCONS: < h0 qeq h1 h7 qeq h9 h14 qeq h25 h18 qeq h24 h28 qeq h34 >
ICONS: < e2 topic i3 > ]
```

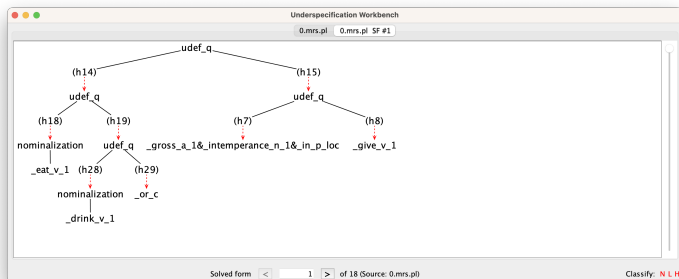
scope resolution I

“given to gross intemperance in eating or drinking”



scope resolution II

“given to gross intemperance in eating or drinking”



generalized quantifiers to STT I

We are producing STT but we are still restricted to FOL provers. More later.

We would like to know about a more general theory for **generalized quantifiers** in STT.

For now, we have a list of ERG quantifiers and their mappings.

generalized quantifiers to STT II

Existential quantifiers

`_a_q`, `_another_q`, `_any_q`, `_both_q`, `_each_q`, `_some_q`,
`_some_q_indiv`, `_that_q_dem`, `_this_q_dem`, `_which_q`,
`free_relative_q`, `def_explicit_q`, `def_implicit_q`, `idiom_q_i`,
`number_q`, `pronoun_q`, `proper_q`, `undef_q`, **`_the_q`**

translated to

$$\exists x, RSTR \wedge BODY$$

generalized quantifiers to STT III

Universal quantifiers: `_all_q`, `_every_q`, `every_q`

translated to

$$\forall x, RSTR \rightarrow BODY$$

Unique quantifiers: `_the_q` and `which_q`

translated to

$$\exists! x, RSTR \wedge BODY$$

generalized quantifiers to STT IV

“the book is white”

We translate as $\exists x, \textit{book } x$. The $\exists!$ is translated to FOL as

$$\exists x, \textit{book } x \wedge \forall y, \textit{book } y \rightarrow x = y$$

How to prove it? We usually don't have more than the utterance. We would need more axioms or hypotheses.

generalized quantifiers to STT V

Negation `_no_q`

translated to

$$\forall x, RSTR \rightarrow \neg BODY$$

predicates I

`h:neg[ARG0: e15 ARG1: h16]`

translated to

$$T(h) = \neg T(h16)$$

predicates II

```
h3:named[ARG0 x, CARG "value"]
```

translated to

value = *x*

predicates III

```
[ card LBL: h12 ARG0: e14 ARG1: x8 CARG: "N" ]
```

translated to

card N e14 x8

predicates IV

```
h1:_if_x_then[ARG0: e2 ARG1: h4 ARG2: h5]  
h14:_then_a_1[ARG1: h16]  
h4 qeq h14
```

translated to

$$T(h5) \rightarrow T(h4)$$

We are not using yet the “fingerprints”, we should! But translation would not be top-down on the scope tree anymore.

predicates V

Many predicates need a better translation.

- ▶ card
- ▶ compound
- ▶ nominalization
- ▶ `_and_c` and `_or_c`
- ▶ etc

'e' variables I

We need to introduce them in the narrowest possible scope because of negation.

‘e’ variables II

“A man is not walking”

```
[ TOP: h0
  INDEX: e2
  RELS: <
    [ _a_q<0:1>          LBL: h4 ARG0: x3 RSTR: h5 BODY: h6 ]
    [ _man_n_1<2:5>      LBL: h7 ARG0: x3 ]
    [ neg<9:12>          LBL: h1 ARG0: e8 ARG1: h9 ]
    [ _walk_v_1<13:20>   LBL: h10 ARG0: e2 ARG1: x3 ] >
  HCONS: < h0 qeq h1 h5 qeq h7 h9 qeq h10 > ]
```

- ▶ $\neg(\exists x3, \textit{_man_n_1 } x3 \wedge (\exists e2, \textit{_walk_v_1 } e2 x3))$
- ▶ $\exists x3, \textit{_man_n_1 } x3 \wedge \neg(\exists e2, \textit{_walk_v_1 } e2 x3)$

‘e’ variables III

“No man is walking”

```
[ TOP: h0
  INDEX: e2
  RELS: <
    [ _no_q<0:2>          LBL: h4 ARG0: x3 RSTR: h5 BODY: h6 ]
    [ _man_n_1<3:6>      LBL: h7 ARG0: x3 ]
    [ _walk_v_1<10:17> LBL: h1 ARG0: e2 ARG1: x3 ] >
  HCONS: < h0 qeq h1 h5 qeq h7 > ]
```

$$\forall x3, _man_n_1\ x3 \rightarrow \neg(\exists\ e2, _walk_v_1\ e2\ x3)$$

‘i’ variables I

“A kid is sitting wearing gea”

```
[ TOP: h0
  INDEX: e2
  RELS: <
    [ _a_q<0:1> LBL: h4 ARG0: x3 RSTR: h5 BODY: h6 ]
    [ _kid_n_1<2:5> LBL: h7 ARG0: x3 ]
    [ _sit_v_1<9:16> LBL: h8 ARG0: e2 ARG1: x3 ]
    [ subord<17:29> LBL: h1 ARG0: e9 ARG1: h10 ARG2: h11 ]
    [ _wear_v_1<17:24> LBL: h12 ARG0: e13 ARG1: i14 ARG2: x15 ]
    [ udef_q<25:29> LBL: h16 ARG0: x15 RSTR: h17 BODY: h18 ]
    [ _gear_n_1<25:29> LBL: h19 ARG0: x15 ] >
  HCONS: < h0 qeq h1 h5 qeq h7 h10 qeq h8 h11 qeq h12 h17 qeq h19 > ]
```

... $\exists x15, \textit{_gear_n_1}$ *i14* $x15$

How to quantify *i14*?

example I

“given to gross intemperance in eating or drinking”

is currently translated to

$$\begin{aligned} &\exists x12, (\exists x17, (\exists e21, \textit{nominalization } x17 \text{ } (_eat_v_1 \text{ } e21)) \\ &\wedge (\exists x26, (\exists e31, \textit{nominalization } x26 \text{ } (_drink_v_1 \text{ } e31)) \\ &\wedge _or_c \text{ } x12 \text{ } x17 \text{ } x26)) \\ &\wedge (\exists x5, (\exists e10 \text{ } e11, _gross_a_1 \text{ } e10 \text{ } x5 \\ &\wedge _intemperance_n_1 \text{ } x5 \wedge _in_p_loc \text{ } e11 \text{ } x5 \text{ } x12) \\ &\wedge (\exists e2, _give_v_1 \text{ } e2 \text{ } x5)) \end{aligned}$$

The SICK dataset I

How to test the library? Robustness and correctness?

SICK dataset of text entailment.

The SICK dataset II

- A A woman is slicing a carrot
- B A carrot is being sliced by a woman

ENTAILMENT

- A Several people are in front of a building which is covered by colors
- B People are walking outside a building that has many murals on it

ENTAILMENT?

The SICK dataset III

- A Two dogs are wrestling and hugging
- B There is no dog wrestling and hugging

CONTRADICTION

- A A woman is not frying some food
- B A woman is deep frying food

CONTRADICTION?

The SICK dataset IV

- A Three kids are sitting in the leaves
- B Three kids are jumping in the leaves

NEUTRAL

The SICK dataset V

A: The young boys are playing outdoors and the man is smiling nearby
B: The kids are playing outdoors near a man with a smile
expected: ENTAILMENT

A: The woman is picking up the kangaroo
B: The kangaroo is picking up the woman
expected: CONTRADICTION

A: A woman is talking to a man
B: A man is talking to a woman
expected: ENTAILMENT

Talk is not having a conversation

A: man, a woman and two girls are walking on the beach
B: Four people are walking on the beach
expected: ENTAILMENT

A: A shirtless man is leading a horse that is pulling a carriage
B: A shirtless man is leading a carriage that is being pulled by a horse
expected: ENTAILMENT

A: A person is climbing a rock with a rope, which is pink
B: A rock is being climbed by a person with a rope, which is pink
expected: ENTAILMENT

What is pink? Search space!

meaning postulates arithmetic, etc.

From Johan Bos discussions ...

- ▶ A man is smiling.
- ▶ A man is not smiling.

inconsistent?

- ▶ Mary is smiling.
- ▶ Mary is not smiling.

What about two people named Mary?

RTE II

- ▶ Mary was smiling.
- ▶ Mary was not smiling.

Two different moments?

- ▶ Mary is smiling.
- ▶ She is not smiling.

what is the referent for 'she'?

RTE III

We need a precise task definition.

But remember, we didn't start trying to solve RTE, we just want to test the MRS Logic.

What can we explore from the experiments so far? Considering the LLMs?

Results I

success:

expected	got	%	#
CONTRADICTION	CONTRADICTION	3.23	32/990
ENTAILMENT	ENTAILMENT	13.13	130/990
NEUTRAL	NEUTRAL	33.33	330/990
		49.70	492/990

failure:

expected	got	%	#
CONTRADICTION	ENTAILMENT	0.10	1/990
CONTRADICTION	NEUTRAL	30.00	297/990
ENTAILMENT	NEUTRAL	20.10	199/990
ENTAILMENT	error	0.10	1/990
		50.30	498/990

Results II

What classes? How to make a proper error analysis?

- ▶ search space
- ▶ missing ontological axioms?
- ▶ missing logical axioms?
- ▶ expressivity
- ▶ missing meaning postulates?

ambiguity and state explosion

“A man in a white shirt and sunglasses and a man in a black shirt and sunglasses are sitting at a table with four beer bottles”

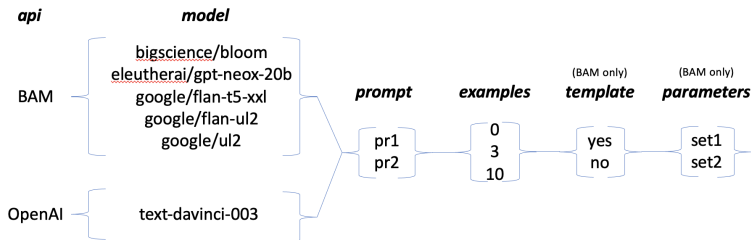
6628 readings. The first reading (MRS) has 2.166.400 possible quantifiers scopes solutions.

We take only the first 2x2 (readings and solutions) for each sentence. 4x4 possible analysis for each pair of sentences.

We need to remove logical equivalent formulas for each sentence. This is hard. Translate to prenex normal form but it can increase exponential the size of the formula.

The scope solver (Utool) could do that? The scopes have any ranking?

LLMs I



LLMs II

Decide whether sentence (1) **contradicts / entails** sentence (2). Answer "yes" or "no".

Sentences:

1. A woman is wearing sunglasses of large size and is holding newspapers in both hands.
2. The woman is wearing sunglasses of large size and is holding newspapers in both hands.

Answer: no

Sentences:

1. A woman is wearing sunglasses of large size and is holding newspapers in both hands.
2. There is no woman wearing sunglasses of large size and holding newspapers in both hands.

Answer: yes

Sentences:

1. A woman is wearing sunglasses of large size and is holding newspapers in both hands.
2. A newspaper is featuring a woman with large sized hands holding sunglasses and wearing a coat.

Answer: no

Sentences:

1. A woman is performing a trick on a ramp with a bicycle.
2. A woman is jumping with a bicycle.

Answer:

No. Because 1) she is wearing a cap and round glasses and 2) she is jumping

2 requests per pair:

```
S := ∅
decide ... contradicts
- if "yes", S.add(CONTRADICTION)
decide ... entails
- if "yes", S.add(ENTAILMENT)
if S == ∅:
- S.add(NEUTRAL)
```

examples
(0, 3, or 10)

Possible responses:

- CONTRADICTION
- ENTAILMENT
- CONTRA&ENTAIL
- NEUTRAL
- != yes/no, error

model
response

LLMs III

Decide the relation between the pairs of sentences. Answer "ENTAILMENT", "CONTRADICTION", or "NEUTRAL".

Sentences:

1. A woman is wearing sunglasses of large size and is holding newspapers in both hands.
2. The woman is wearing sunglasses of large size and is holding newspapers in both hands.

Relation: ENTAILMENT

Sentences:

1. A woman is wearing sunglasses of large size and is holding newspapers in both hands.
2. There is no woman wearing sunglasses of large size and holding newspapers in both hands.

Relation: CONTRADICTION

Sentences:

1. A woman is wearing sunglasses of large size and is holding newspapers in both hands.
2. A newspaper is featuring a woman with large sized hands holding sunglasses and wearing a coat.

Relation: NEUTRAL

Sentences:

1. A woman is performing a trick on a ramp with a bicycle.
2. A woman is jumping with a bicycle.

Relation: NEUTRALING

1 request per pair:

```
if "CONTRADICTION": R:= CONTRADICTION
elif "ENTAILMENT": R:= ENTAILMENT
elif "NEUTRAL": R:= NEUTRAL
else: error
```

examples
(0, 3, or 10)

Possible responses:
- CONTRADICTION
- ENTAILMENT
- NEUTRAL
- *error*

model
response

LLMs IV

```
= results_bam_google_flan-ul2_pr2_ex1021-1023_tpl_set2 =  
success:
```

expected	got	%	#
CONTRADICTION	CONTRADICTION	13.33	40/300
ENTAILMENT	ENTAILMENT	21.33	64/300
NEUTRAL	NEUTRAL	35.67	107/300
		70.33	211/300

```
failure:
```

expected	got	%	#
ENTAILMENT	NEUTRAL	0.33	1/300
NEUTRAL	CONTRADICTION	23.00	69/300
NEUTRAL	ENTAILMENT	6.33	19/300
		29.67	89/300

LLMs V

```
===== results_openai_text-davinci-003_pr1_set1 =====
```

success:

expected	got	%	#
CONTRADICTION	CONTRADICTION	10.70	107/1000
ENTAILMENT	ENTAILMENT	17.70	177/1000
NEUTRAL	NEUTRAL	48.90	489/1000
		77.30	773/1000

failure:

expected	got	%	#
CONTRADICTION	ENTAILMENT	0.10	1/1000
CONTRADICTION	NEUTRAL	1.10	11/1000
ENTAILMENT	CONTRADICTION	0.20	2/1000
ENTAILMENT	NEUTRAL	4.20	42/1000
NEUTRAL	CONTRADICTION	13.90	139/1000
NEUTRAL	ENTAILMENT	3.20	32/1000
		22.70	227/1000

LLMs VI

(ChatGPT 3.5t) Considering the sentences A) "A man, a woman and two girls are walking on the beach" and B) "Four people are walking on the beach". Does A entails B?

Yes, sentence A entails sentence B.

(ChatGPT 3.5t) sure?

I apologize for my earlier response. Sentence A does not necessarily entail sentence B, as there could be other combinations of people walking on the beach which add up to four people, such as "Two men and two boys are walking on the beach". Apologies for any confusion caused earlier.

next steps I

Proper evaluation of ERG parse rank model. How?

Use WordNet, from predicates to senses maybe using the glosstag corpus.

How to add axioms by demand?

Tips for dealing with ambiguity?

dependent types I

“Corrosion prevented continuous contact.”

(Entailment, intensionality and text understanding. Cleo Condoravdi, Dick Crouch, Valeria de Paiva, Reinhard Stolle, Daniel G. Bobrow)

Anaphora examples. Proof-theoretic semantics. Previous work on Fracas test suite ([here](#) and [Formal Semantics in MTT](#))

dependent types II

“Any farmer who owns a donkey beats it”

$$\prod_{z: \sum_{x: \text{Farmer}} (\sum_{y: \text{Donkey}} \text{Owns}(x, y))} \text{Beats}(p(z), p(q(z)))$$

The $z = (m, b)$ where m is a farmer and b is a proof-object for “exists a donkey that m owns it”.

vs the FOL “For all farmers and for all donkeys, if the farmer owns the donkey then he beats it.”

$$\forall x \forall y (\text{Farmer}(x) \wedge \text{Donkey}(y) \wedge \text{Own}(x, y) \rightarrow \text{Beat}(x, y))$$