Recognising entailment using Minimal Recursion Semantics

A case study

Elisabeth Lien

Department of Informatics, UiO

22nd October 2013

PhD project: MRSes in entailment recognition

- Examining advantages/disadvantages of using MRSes in entailment recogition
- Not previously used in entailment recognition (to my knowledge)
- Rich representations - a source of bounty or too complex?

Minimal Recursion Semantics (MRS)

- Computational semantic *framework* (not theory)
- Semantic representation for computational grammars
- Can be implemented in typed feature structure formalisms
- Integrated with the English Resource Grammar (ERG)

MRS example

He would wake up.

```
[ LTOP: h1
  INDEX: e3
  RELS: <
  [ pron_rel LBL: h4 ARG0: x5 ]
  [ pronoun_q_rel LBL: h6 ARG0: x5 RSTR: h7 BODY: h8 ]
  [ _would_v_modal_rel LBL: h2 ARG0: e3 ARG1: h9 ]
  [ _wake_v_up_rel LBL: h10 ARG0: e11 ARG1: x5 ] >
  HCONS: < h1 qeq h2 h7 qeq h4 h9 qeq h10 > ]
```

The PETE shared task

- evaluating parsers by using their output to decide entailment
- indirect parser evaluation: entailment system acts as intermediary
- entailmen decision is not the goal in itself, it's just a tool for parser evaluation
- purely syntactic entailments, no need for background knowledge or reasoning ability

The PETE data: text-hypothesis pairs

- Dev set: 66 pairs. Test set: 301 pairs.
- Text sentences: Penn & Unbounded Dependency Corpus
- Hypothesis: short sentence built around two syntactically related content words
- *He has a point he wants to make, and [...]* $\Rightarrow$ *Somebody wants to make a point.*
- *point* and *make*

Results!

On 301 PETE test data pairs

gold	YES: 156			NO: 145			
systems	YES	NO	acc	NO	YES	acc	tot acc
Camb	98	58	62.8	120	25	82.8	72.4
My sys	78	78	50.0	135	10	93.1	70.7
SCHWA	125	31	80.1	87	58	60.0	70.4

Characteristics of the PETE dev data

- The hypothesis (H) is always shorter that the text (T)
- In some cases, H is completely included in T
- Mostly, H is a substructure of T with some alterations:
  - active $\rightarrow$ passive
  - verb argument has been replaced by somebody/-one/-thing
  - H corresponds to a complex NP in T
- Negative pairs: Mostly combination of elements from T that don't match syntactically.
- Some pairs do require reasoning.

My system

- English Resource Grammar (ERG), version 1212
- PET parser
- Entailment system developed on both treebanked and 1-best MRSes.

My heuristic

- Take all relations in H whose ARG0 is an event variable...
- ...find the same or similar relations in T...
- ...and check whether their arguments can be matched
- Matching arguments:
    - Pronoun in H matches any NP in T.
    - Empty argument position in H matches any NP in T.
    - "handle relations" in H are a subset of "handle relations" i T.
- If matching succeeds for all event relations in H, we have entailment.

If ERG can't parse → entailment decision is NO.

My heuristic, cont.

...all relations in H whose ARG0 is an event variable...

- Lexical relations: adjectives, adverbs, prepositions, verbs, unknown words.
- Generalised relations: cardinal/ordinal numbers, negation, measure, possessive marker, ...
- Grammatical relations: apposition, compounds, ellipsis, subordination, ...

Maybe checking too much!

Example

- T: *He could also hear the stream which he had seen from his position.*
  H: *Someone had seen the stream.*
- `[ _see_v_1_rel LBL: h2 ARG0: e3 ARG1: x5 ARG2: x9 ]` in H matches same relation in T.
- Arguments:
  - x5/*someone* matches *he* in T.
  - x9/*the stream* matches *the stream* in T.

Cambridge system

- Parser: C&C parser
- Parser outputs *grammatical relations* (GRs) according to the Stanford Dependency scheme: `(nsubj tired man)`
- Entailment system: checking if GRs(H) are a subset of GRs(T)
- Entailment system seems to have been developed on 1-best output only.

Cambridge heuristic

Checking if GRs(H) are a subset of GRs(T):

- Any GR i H with a token that's not in T is ignored $\Rightarrow$ passive auxiliaries, pronouns, determiners, expl. subj. are ignored
- Passive subjects are equated with direct objects $\Rightarrow$ handles active to passive alterations
- Only subject and object relations (core relations) are considered
- T and H have to have some GRs in common

To sum up: YES if $\text{core(H)} \subseteq \text{core(T)}$ and $\text{grs(H)} \cap \text{grs(T)} \neq \emptyset$
No parse $\rightarrow$ entailment decision is NO

Let's look at those results again

On 301 PETE test data pairs

gold	YES: 156			NO: 145			
systems	YES	NO	acc	NO	YES	acc	tot acc
Camb	98	58	62.8	120	25	82.8	72.4
My sys	78	78	50.0	135	10	93.1	70.7
SCHWA	125	31	80.1	87	58	60.0	70.4

Compare: GRs vs. MRSes

Grammatical relations in Cambridge:

- Word dependencies typed with grammatical relations. Explicit relations → easy to pick out which ones to check.

MRSes:

- MRS EPs: a mixture of lexical predicates and relations, and various syntactic/semantic relations.
- GRs sometimes corresponds to an EP, sometimes to argument-value pairs in an EPs.
- Don't always know which grammatical relation each argument position of an EP corresponds to → we're not checking gram. rels. as precisely as Cambridge.

Compare: GRs vs. MRSes

Example: the object relation between *have* and *something* in *He would have something to work on.*

- GR: `(dobj have something)`
- MRS: `[ _have_v_1_rel LBL: h10 ARG0: e11 ARG1: x5 ARG2: x12 ]`
- `x12` is `_some_q_rel` and `thing_rel`

Compare: Cambridge heuristic vs. my heuristic

Cambridge:

- Checks whether the subject and object relations in H are also in T.
- GRs with tokens that are in H but not in T are ignored → even subject and object relations can be ignored
- The heuristic is maybe a bit “shallow”?

My system:

- Has explicit rules for matching arguments that are different.
- Makes the system more vulnerable to unseen cases.
- But also makes the positive entailment decisions more well founded.
- Heuristic is maybe too detailed. Should maybe have restricted the checking to a “core” set of relations.

Compare: Example

He would wake up in the middle of the night and fret about it. ⇒ *He would wake up.*

- Cambride: YES decision is based only on the single GR match (`nsubj would he`). The other GRs are ignored because they are non-core according to the heuristic.
- My system: YES decision is based on matching of both `_would_v_modal_rel` with scopal argument over `_wake_v_up_rel`, and `_wake_v_up_rel` itself with its pronoun argument.

Causes of incorrect decisions in my system

(Apart from errors and missing functionality in the code!)

- Incompatible MRS analyses.
- Error in MRS.
- PETE pair requires reasoning.
- PETE pair requires coreference resolution.
- PETE sentence is structurally ambiguous.

But the complexity of the MRSes was not a direct cause of error.

Beyond shared task results: Refining on test data

PETE development set is small: 66 pairs.

- Split the test set (301 pairs) in two, refine heuristic on one half, test on the other half.
- Could correct errors in the code, and include cases not covered by the code.
- Found several PETE pairs that require reasoning.
- Acc. YES: 72.8 %, acc. NO: 95.6 %, tot.acc.: 83.3 %

Final thoughts

- MRSes: Very rich in information, but some constructions are complex to process.
- Same construction can receive very different analyses.
- Lack of documentation $\rightarrow$ hard to predict analyses.
- Hard to predict analyses $\rightarrow$ hard to write code that covers all cases.

References

L. Rimell and S. Clark. Cambridge: Parser Evaluation using Textual Entailment by Grammatical Relation Comparison. *Proceedings of the 5th International Workshop on Semantic Evaluation, ACL 2010*