The Meaning Factory: Producing AMRs with Boxer

Johannes Bjerva Johan Bos Hessel Haagsma

university of groningen

SemEval-2016: Shared Task 8

- Produce Abstract Meaning Representations (AMRs) for sentences
- Formal semantics system – Discourse Representation Theory (DRT)
- Can the representations from DRT, Discourse Representation Structures (DRSs), be easily converted into AMRs?

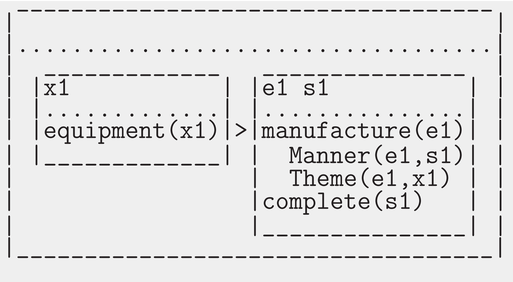
Boxer, a semantic parser based on formal semantics

- The semantic parser that we employed is Boxer based on CCG
- Semantic representations based on DRT: Discourse Representation Structures (DRSs), first-order logic representation
- Various notations are possible, such as boxes, which display scopes of discourse referents, contain properties and two-place relations

Gold-standard AMR

```
(m / manufacture-01
   :ARG1 (e2 / equipment
             :mod (a2 / all))
   :ARG1-of (c / complete-02))
```

DRS, as produced by Boxer, for the same sentence



What DRS and AMR have in common

- neo-Davidsonian event semantics
- recursive meaning representations
- normalization of date expressions expected

Important differences between DRS and AMR

- AMR has no explicit means for quantification and negation
- AMR expects different thematic role labels (Boxer uses VerbNet)
- AMR assigns no scope for propositional meanings
- AMR is strongly event-oriented (verbalization)
- AMR has flat lists of coordinated structures
- AMR has symbol grounding by wikification (for named entities)

Method

- Pre-processing and Tokenisation
- Lexical anticipation
- Conversion from DRS to AMR using a recursive translation function
- Re-labelling
- Wikification with DBPedia Spotlight

Results

Scores on *test* part of development data

	DFA	Xinhua	Consensus	Bolt	Proxy
Boxer	39.9	**57.2**	45.8	47.0	56.0
JAMR	**47.5**	52.8	**49.6**	**48.7**	**60.2**

- F-score on official test data: 47% (50% unofficially)

Conclusions

- DRSs and AMRs have many similarities and differences!
- Overall results are perhaps disappointing: Much lower F-scores than state-of-the-art supervised semantic parsers
- However, with relatively little effort reasonable output is produced.
- For notoriously hard constructions such as control and coordination Boxer performs well
- AMR is not a replacement for DRS, as it has less expressive power, but the ability to switch between the two formats would be a welcome feature

Error analysis (Boxer output left, gold AMR right)

```
(e6 / and                                  (a / and
 :op1 (k1 / thug                            :op1 (t / thug
    :domain (x1 / they))                       :domain (t2 / they))
 :op2 (k2 / deserve-01                      :op2 (d / deserve-01
    :ARG0 x1                                   :ARG0 t
    :ARG1 (x2 / bullet)))                      :ARG1 (b / bullet)))
```
They are thugs and deserve a bullet. (#111, F-score: 90.9)

```
(e1 / arrest-01                            (a / arrest-01
:ARG1 (x1 / person                         :ARG1 (p / person :quant 1
  :ARG0-of (v1002 / protest-01)))            :ARG0-of (p2 / protest-01)))
```
A protester was arrested. (#710, F-score: 92.3)

```
(e1 / create-01                            (c3 / create-01
 :ARG1 (x1 / force                          :ARG0 (a / and
    :mod (s1 / country                         :op1 (c2 / country
       :name (p1002 / name                        :wiki "United_States"
        :op1 "afghanistan")                       :name (n2 / name :op1 "US"))
    :poss (x2 / security))                     :op2 (c4 / coalition))
 :ARG0 (x3 / and                           :ARG1 (f / force
    :op1 (x4 / organization                    :purpose (s / security)
       :name (n3 / name                        :mod (c / country
        :op1 "us")                              :wiki "Afghanistan"
        :wiki "United_States" )                 :name (n / name
    :op2 (x5 / coalition)))                     :op1 "Afghanistan"))))
```
The Afghan security force was created by the US and the coalition. (#300, F-score: 90.9)

```
(e1 / tell-01                              (t / tell-01
 :ARG0 (x1 / they)                          :ARG0 (t2 / they)
 :ARG1 (p1 / and                            :ARG1 (a / and
    :op1 (k1 / avoid-01                        :op1 (a2 / avoid-01
     :ARG0 (x2 / she)                           :ARG0 s
     :ARG1 (x3 / cafeteria))                    :ARG1 (c / cafeteria))
    :op2 (k2 / take-01                         :op2 (p / participate-01
     :ARG0 x2                                   :polarity -
     :ARG1 (x4 / part)                          :ARG0 s
     :polarity -                                :ARG1 (h / homecoming)))
     :in (x5 / homecoming)))                 :ARG2 (s / she))
 :ARG2 x2)
```
They told her to avoid the cafeteria and not take part in homecoming. (#151, F-score: 85.0)

```
(e1 / fall-01                              (f / fall-07
 :ARG0 (x1 / man                            :ARG1 (m / man
    :mod (s1 / innocent))                      :ARG1-of (i / innocent-01)
 :ARG1 (x2 / victim)                           :mod (a / another))
 :to (x3 / machine))                        :ARG2 (v / victimize-01
                                              :ARG0 (m2 / machine)
                                              :ARG1 m))
```
Another innocent man falls victim to the Machine. (#1024, F-score: 26.1)