

Tecto to AMR and translation

Ondřej Bojar, Silvie Cinková,
Ondřej Dušek, Tim O’Gorman,
Martin Popel, Roman Sudarikov,
Zdeňka Urešová

August 1, 2014

Introduction

Motivation

- ▶ We suspect parallel AMRs will generally be similar
- ▶ We believe the AMR and t-layer (tectogrammatical layer) are similar
- ▶ Being able to convert T-layer to AMR would give us lots of AMR data.
- ▶ Also would be useful for translation.

Looking at the AMR-AMR link

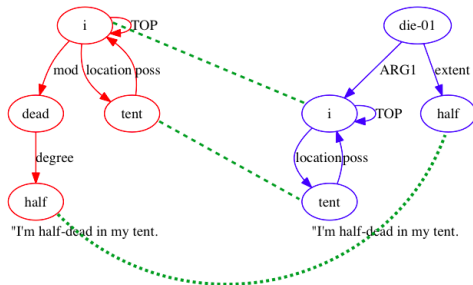
- ▶ Manual annotation of 100 sentences of Czech, Chinese and English suggest that there is a lot of similarity between languages.
- ▶ Smatch alone doesn't work for cross-lingual comparison.

AMRICA

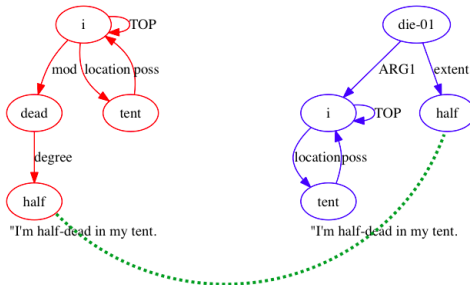
(AMR Inspector with Cross-language
Alignment)

Extension to Smatch (Cai & Knight 2012).

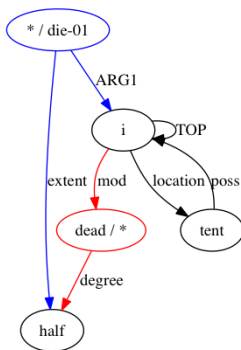
Smatch Classic



Smatch Classic

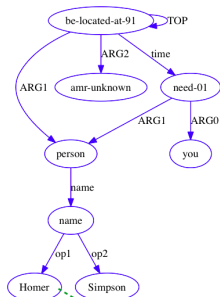


Smatch Classic

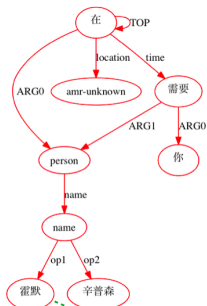


"I'm half-dead in my tent."

AMERICA

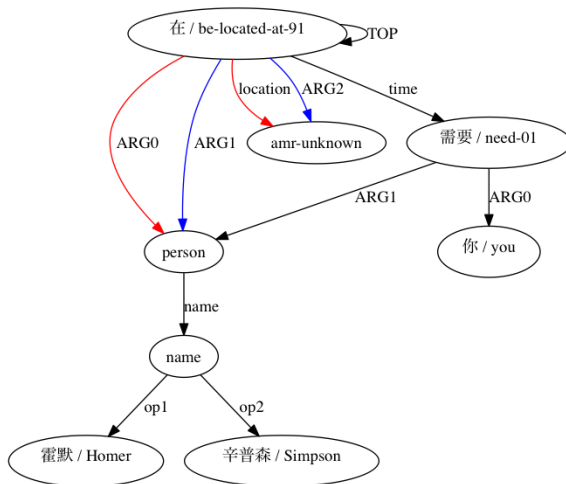


Where's Homer Simpson when you need him?



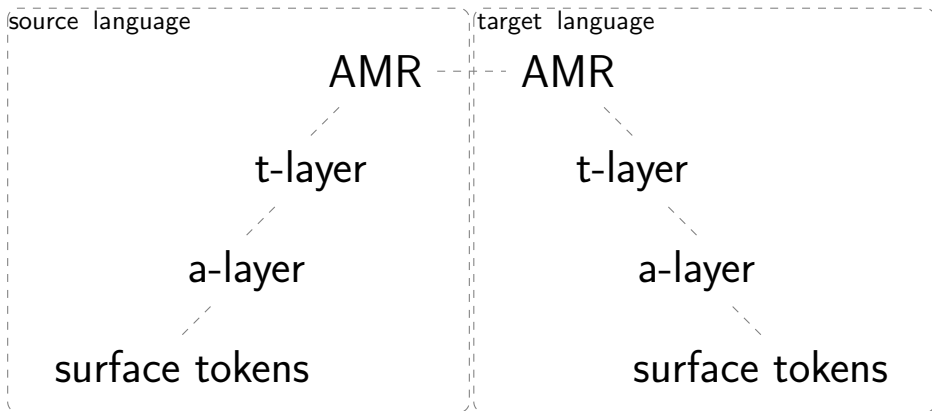
当你需要他时，霍默辛普森在哪里？

AMRICA



Where's Homer Simpson when you need him ?
当你需要他时，霍默·辛普森在哪里？

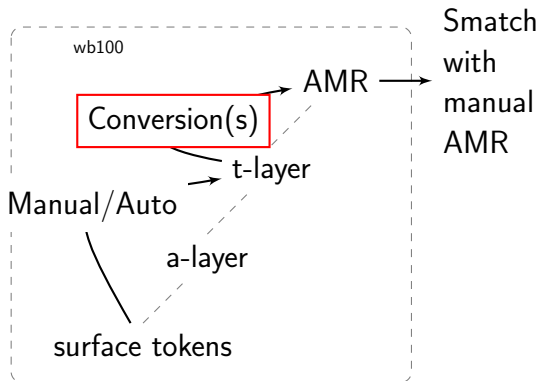
Simplistic View of the MT Plan



From Text to AMR

From Text to AMR

- ▶ Same goal as AMR parsing team but...
- ▶ Building on top of t-layer analysis.



General Conversion Idea

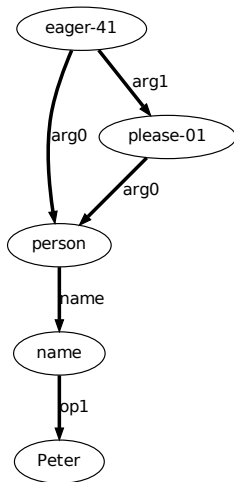
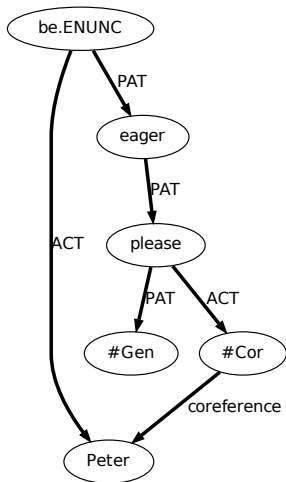
- ▶ AMR and PCEDT are very similar:
 - ▶ Both abstract away from syntax.
 - ▶ Both make all semantic links in a sentence in a graph format.
 - ▶ Both do coreference
 - ▶ Semantic roles must be converted.
 - ▶ Various minor structural differences.

General Conversion Idea

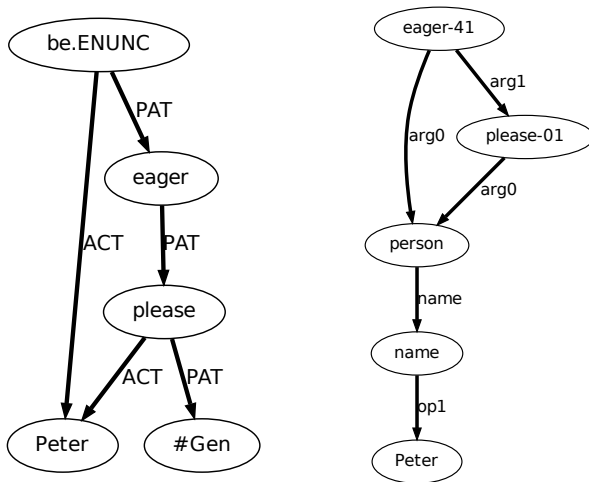
These are the main things that t-to-AMR conversion has to handle:

- ▶ PCEDT has many layers and features; AMR is all in one graph.
- ▶ Some additional data (named entities, inferrable predicates, etc) are added in AMR graphs
- ▶ Semantic roles must be converted.
- ▶ Various minor structural differences.

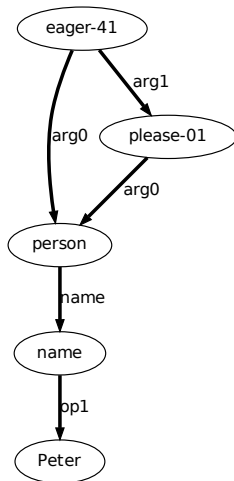
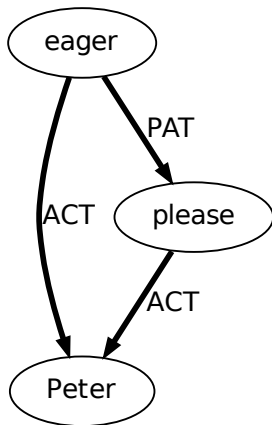
“Peter is eager to please”



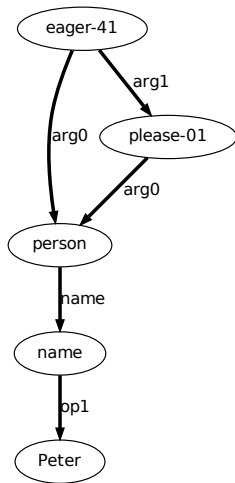
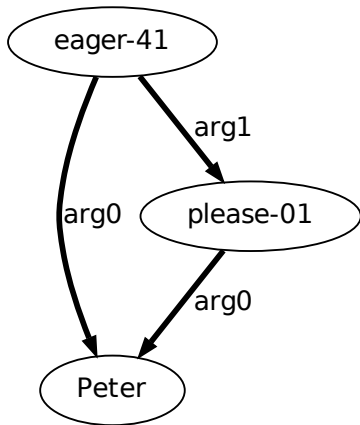
Merging of Coreferent Nodes



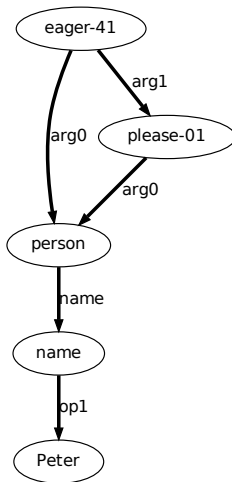
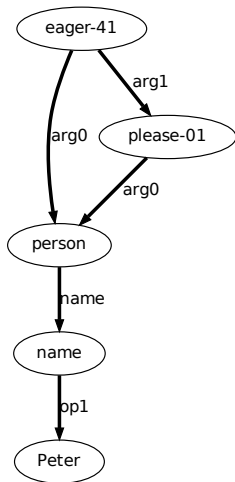
Elimination of semantically light words



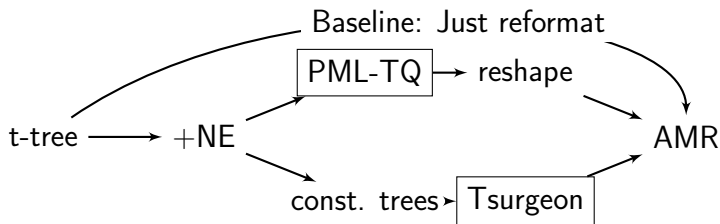
Semantic Roles and Senses



Named Entities



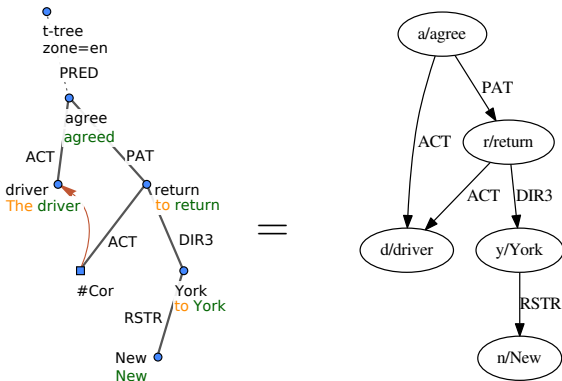
Conversion Procedures



- ▶ Handcoded rules in two toolkits:
 - ▶ PML-TQ to reflect AMR Guidelines (and tightly coupled with Prague t-layer approach)
 - ▶ Tree Surgeon to reflect Tim's knowledge (and closer to the target AMR format)

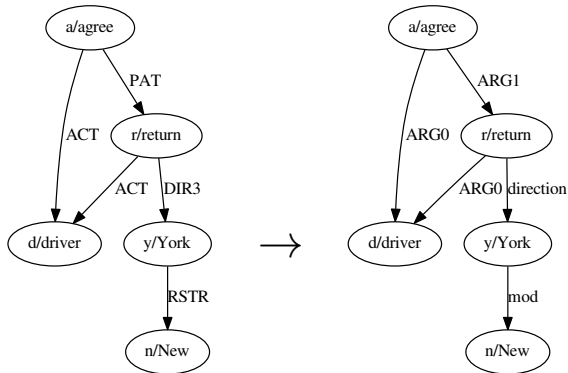
Baseline

- ▶ Just printing t-trees in AMR format
- ▶ No changes, just coreference replaced by reentrancies



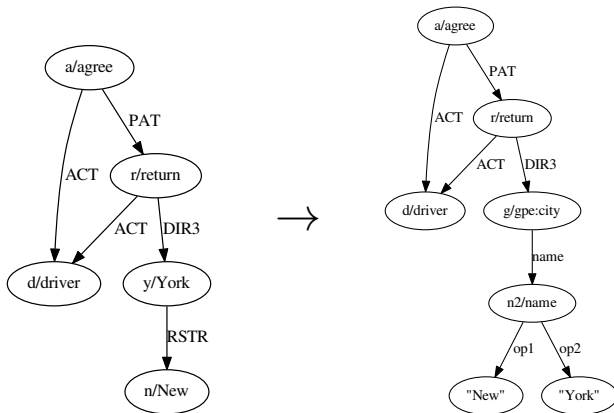
Edge Labels

- Deterministically mapped to AMR arc labels



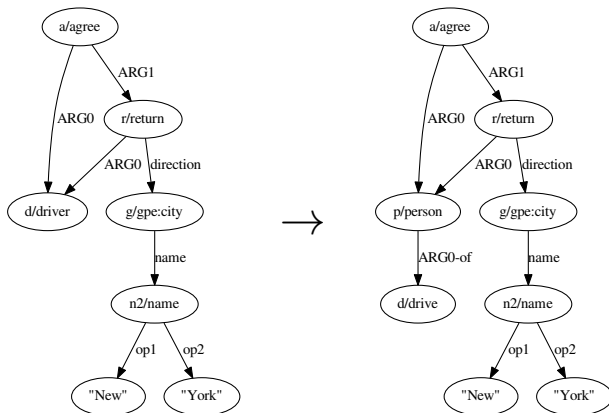
Named Entities

- ▶ AMR-style Named Entities added using Stanford NER (CoNLL'08) and NameTag (BBN-style)



Verbalization

- ▶ Based on a dictionary:
 - ▶ administration → administrate-01
 - ▶ beekeeper → person :ARG0-of keep-01
:ARG1 bee

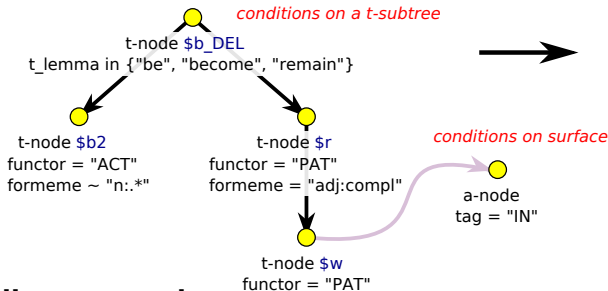


PML-TQ Rules

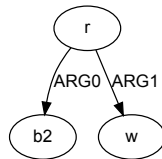
- ▶ Based on AMR guidelines (generalized)
- ▶ For copula, attributes, non-core roles ...

A PML-TQ rule

LHS (PML-TQ Query)



RHS (AMR Subtree)



Guidelines example:

The boy is responsible for the work.

PML-TQ Rules

Rule application

t-tree
zone=en

be.enunc
PREDv:fin
was

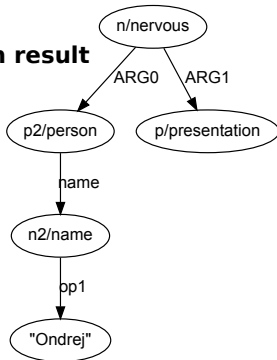
Ondrej
ACTn:subj
Ondrej

nervous
PATadj:compl
nervous

presentation
PATn:about+X
about the presentation



Conversion result



Matching t-tree

Matching sentence:

Ondrej was nervous about the presentation.

Guidelines-Based Conversion

- ▶ Create marking rules in PML-TQ from AMR guidelines examples.
 - ▶ Empirical generalizations of the examples.
 - ▶ Find a many sentences structurally similar to the example sentence as possible, not considering possible interactions between queries.
 - ▶ Phenomena: copula, attributes, non-core roles...
- ▶ Mark t-trees with these rules.
- ▶ Change each t-tree to mimic the AMR in the guideline example.

Sample PML-TQ

Ex.: Iraq launched a missile attack.

```
101a-Iraq_launched_a_missile_attack
t-node $launch_DEL := [
  t_lemma in {"launch", "begin", "start", "perform", "do", "make"},
  is_passive !=1,
  t-node $a_attack_01 := [
    (gram/sempos~"^n..*$" or formeme~"ger",
     t_lemma in {"N_V"},
     functor in {"CPHR", "DPHR", "PAT"}) or
    functor="CPHR",
    t-node $m_missile := [
      functor in {"MEANS", "RSTR"},
      gram/sempos="n.denot",
      ord<$a_attack_01.ord]
  ],
  t-node $c_country_n_name := [gram/sempos~"^n..*$", functor="AC"]
]
```

Constructing Rules (LHS)

- ▶ query nodes indexed to map on the resulting AMR
- ▶ _DEL = delete node in conversion
- ▶ What makes a word similar to
 - launch, v?* heuristic Wordlist of performative light verbs, based on PEDT
 - attack, n?* heuristic structure template (part of a phraseme or *-ing* form or being listed as a predicate-invoker in AMR Editor

Applying Rules

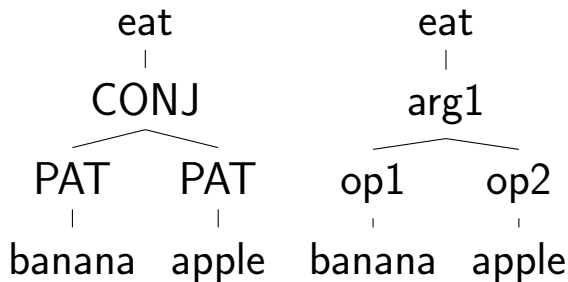
- ▶ PML-TQ specifies the LHS of the replacement.
- ▶ RHS is usually just taken from AMR Guidelines
- ▶ Top-bottom, clone nodes from original t-tree to tamr-tree
- ▶ For each node apply all possible rules:
 - ▶ Add modifier from AMR node
 - ▶ Change parent if necessary

Tsurgeon tree transformation

- ▶ Other approach is to convert t-layer (with many features) into constituency trees (reentrancies repeated as variables)
- ▶ This is so that we can use a tree transformation tool, **tsurgeon** (Levy and Andrews 2006) to quickly implement hand-written rules.
- ▶ Essentially does regular expressions over trees, with commands like “move”, “insert”, etc.

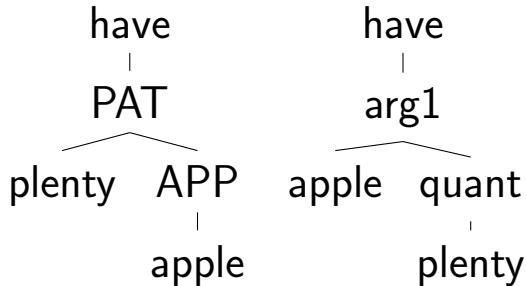
Tree transformation

- ▶ Many of the structural differences are pretty deterministic



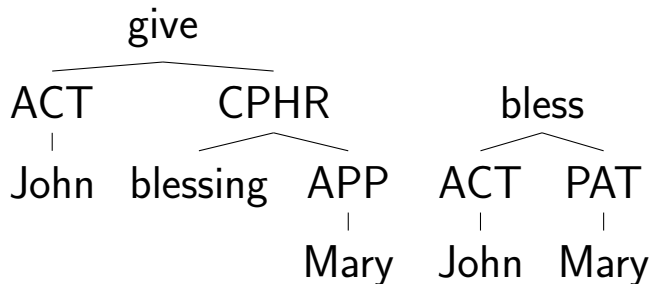
Transformer tree transformation

- Other differences are meaningful, but relatively consistent: AMR makes "apple" the head of "plenty of apples".



Adding and removing nodes using lexical lists

- ▶ Have lists of mappings (manually generated at ISI and Boulder) we use for converting complex patterns.



Predicate abstraction and semantic role conversion

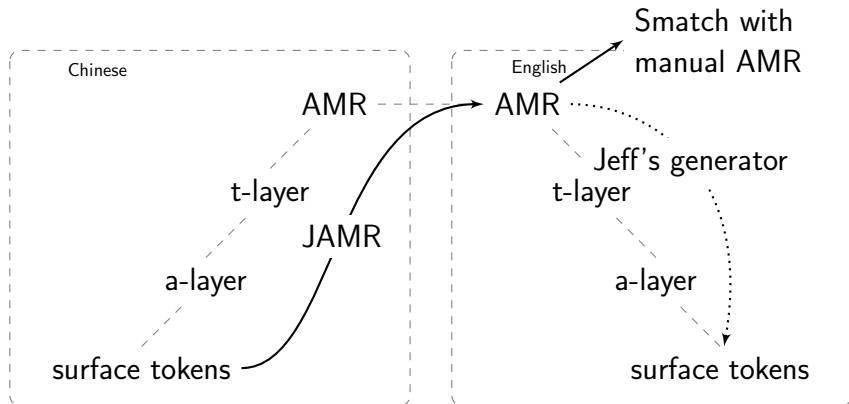
- ▶ AMR verbalizes predicates, so “destruction” is “destroy-01”, etc
- ▶ Propbank “unification” frame files mark this.
- ▶ Vallex mappings from t-layer roles to AMR numbered arguments.

Results of EN t-to-AMR Conv

incl. verb sense tags →	Smatch (%)	
	yes	no
Baseline	20	28
+labels	33	41
+labels +NE	37	45
+labels +PML-TQ +verbaliz.	35	43
+labels +PML-TQ +verbaliz. +NE	38	47
+labels +verbaliz. +NE	40	48
+Tsurgeon + NE	44	52

MT as Cross-Lingual Deep Parsing

Overview of Cross-Lingual Parsing



Chinese to English (cross-lingual AMR parsing)

Jeff and Adi will report on their experiment