Abstract Meaning Representations for Sembanking

Etienne de Crecy

University of Edinburgh

March 4, 2016

Overview

What is AMR (briefly)?

Abstract Meaning Representation (AMR) is a semantic representation language aiming to express the **meanings** of whole English sentences in a human and machine-readable way.

Why was it created?

AMR was created in response to the **fragmented** state of the semantic annotation field: many separate annotations exist for niche tasks, for example co-reference, Named Entities, discourse connectives, etc.
This results in **resources and efforts being split** over many different projects, which is an issue in particular with regard to training data.

Why was it created? - continued

The goal of the authors is to establish a 'simple readable sembank of English sentences paired with their **whole-sentence**, logical meanings' using AMR.
They believe such a **sembank** could have a similar **impact** on statistical Natural Language Understanding (NLU) and Generation (NLG) as the Penn TreeBank had on statistical parsing.

Basic principles

Abstract Meaning Representation relies on these basic principles, meant to ensure its suitability for sembanking:

- Easy to work with for both humans and computers
- Several syntactic forms, but one meaning? Still one AMR
- PropBank frames as a basis for the representation
- From strings to meanings or meanings to strings
- Not an Interlingua: AMR is language-specific

Easy to work with for both humans and computers

AMRs are represented as rooted, directional and labeled **graphs** that are 'easy for humans to read, and easy for programs to traverse'.
There are several different formats to work with: LOGIC format, AMR format, GRAPH format

$\exists$ w, b, g:
instance(w, want-01) $\wedge$ instance(g, go-01) $\wedge$
instance(b, boy) $\wedge$ arg0(w, b) $\wedge$
arg1(w, g) $\wedge$ arg0(g, b)

Figure: LOGIC format

```
(w / want-01
   :arg0 (b / boy)
   :arg1 (g / go-01
            :arg0 b))
```

Figure: AMR format

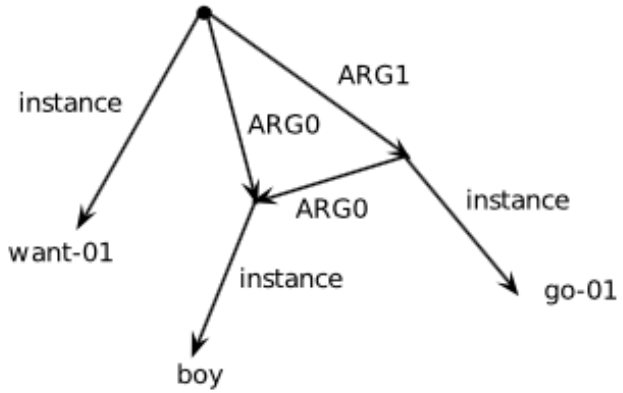

Figure: GRAPH format

One meaning, one AMR

AMR attempts to abstract away from syntactic representations: sentences with the same basic meaning should be assigned the same representation, **regardless of syntax variations**.

Example (One AMR, several syntaxes)

```
(d / describe-01
  :arg0 (m / man)
  :arg1 (m2 / mission)
  :arg2 (d / disaster))
```

The man described the mission as a disaster.
The man's description of the mission: disaster.
As the man described it, the mission was a disaster.

PropBank framesets as a basis for AMR

PropBank is a corpus annotated with verbal propositions and their arguments. Each of these verbal proposition along with its argument is called a **frameset**.
ANR makes use of these frameset to annotate meanings of sentences. However, contrary to the PropBank corpus, even phrases containing no verbs are annotated using PropBank framesets.

Example (Related verbs and nouns go to one frameset)

`bond investor` $\longrightarrow$ `invest-01`
`to invest`

From strings to meanings, or meanings to strings

AMR does not state any rules on how to derive meanings from sentences or sentences from meanings in order to make sembanking faster, as annotators can simply write down the meanings associated to sentences **without explaining the steps used to get there**.
It also 'allows researchers to explore their own ideas about how strings are related to meanings'.

Not an Interlingua: AMR is language-specific

AMR uses concepts (English words, PropBank framesets, or special keywords) **inherited from English**, and is therefore heavily biased towards it. It is not meant to be able to represent the meaning of sentences in other languages.
However, there have been work on developing AMR for other languages, and to use AMRs as an **additional transfer layer** in Machine Translation[1]

[1] For more information, see Xue et al, 2014

More about AMR graphs

We can distinguish two main elements in AMR: concepts and semantic relations between those concepts.

- AMR uses **variables to refer to instances** of a certain concept. Leaves of the graphs are labelled with concepts, so one labelled leave is an instance of a given concept (e.g. 'boy')
- AMR uses approximately a hundred relations: frame arguments (:arg0, :arg1, etc), general semantic relations (:age, :purpose, etc), relations for quantities, for date-entities, and for lists. Relations are labelled on the edges of the graph.

More about AMR graphs - continued

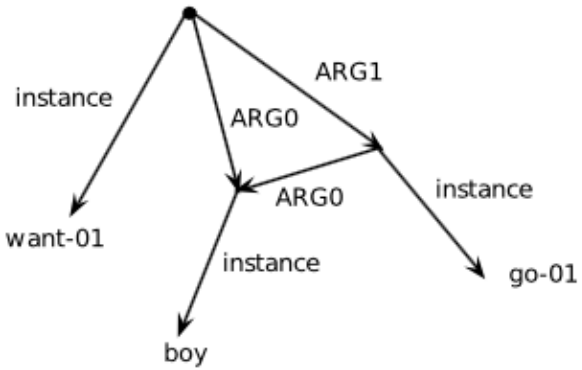

Figure: `The boy wants to go.`

Concepts instantiated in the graph:

- `boy`
- `want-01`
- `go-01`

Relations present in the graph:

- `:arg0`
- `:arg1`

```
(w / want-01
   :arg0 (b / boy)
   :arg1 (g / go-01
            :arg0 b))
```

The boy wants to go.

- w, b and g are variables, that is, instances of the concepts want-01, boy and go-01. This is denoted by the symbol '/'. Each concept and its arguments is enclosed by parenthesis.
- :arg0 and :arg1 are semantic relations, denoted by the symbol ':'. b is the 1st argument of both w and g, and g is the 2nd argument of w.

Examples of AMR representations - General semantic relations

```
(s / hum-02
   :arg0 (s2 / soldier)
   :beneficiary (g / girl)
   :time (w / walk-01
            :arg0 g
            :destination (t / town)))
```

The soldier hummed to the girl as she walked to town.

Examples of AMR representations - Inverse relations

```
(s / sing-01
  :arg0 (b / boy
    :source (c / college)))
```

The boy from the college sang.

```
(b / boy
  :arg0-of (s / sing-01)
  :source (c / college))
```

The college boy who sang.

The top-level root of an AMR represents the focus of the sentence. With the inverse relations `:arg0-of` and `:quant-of`, it becomes possible to build rooted structures, changing the focus of the representation as needed.

Examples of AMR representations - Modals and negation

```
(g / go-01
  :arg0 (b / boy)
  :polarity -)
```

The boy did not go.

```
(p / possible
  :domain (g / go-01
    :arg0 (b / boy))
  :polarity -)
```

It's possible for the boy not to go.

Negation is expressed with `:polarity` (note the `-`), and modals are expressed with concepts, such as `possible` or `obligate-01`. We can also see that copulas are expressed by `:domain` ('It **is** not possible...')

Examples of AMR representations - Questions

```
(f / find-01
   :arg0 (g / girl)
   :arg1 (a / amr-unknown))
```

What did the girl find?

The concept amr-unknown is used for wh-questions, yes/no questions and imperatives are handled differently with the relation :mode.

Quick question

```
(o / obligate-01
  :arg2 (g / go-01
    :arg0 (b / boy))
  :polarity -)
```

Quick question

```
(o / obligate-01
  :arg2 (g / go-01
    :arg0 (b / boy))
  :polarity -)
```

The boy doesn't have to go.

Quick question

```
(o / obligate-01
  :arg2 (g / go-01
    :arg0 (b / boy))
  :polarity -)
```

The boy doesn't have to go.

```
(f / find-01
  :arg0 (g / girl)
  :arg1 (t / toy
    :poss (a / amr-unknown)))
```

Quick question

```
(o / obligate-01
  :arg2 (g / go-01
    :arg0 (b / boy))
  :polarity -)
```

The boy doesn't have to go.

```
(f / find-01
  :arg0 (g / girl)
  :arg1 (t / toy
    :poss (a / amr-unknown)))
```

Whose toy did the girl find?

Examples of AMR representations - Verbs and nouns

```
(s / see-01
  :arg0 (j / judge)
  :arg1 (e / explode-01))
```

The judge saw the explosion.

```
(t / thing
  :arg1-of (o / opine-01
    :arg0 (g / girl)))
```

the girl's opinion

Most English verbs have a corresponding PropBank frameset, and it is also possible to express most nouns using framesets.

Examples of AMR representations - Named Entities

```
(p / person
  :name (n / name
    :op1  "Mollie"
    :op2  "Brown"))
```

Mollie Brown

Any name can be handled with `:name`. Additionally, there are approximately 80 standardized types of named entities.

Examples of AMR representations - Reification

```
(m / marble
  :location (j / jar))
```

the marble in the jar

```
(b / be-located-at-91
  :arg1 (m / marble)
  :arg2 (j / jar))
```

The marble is in the jar.

Reification allows us to use an AMR relation as a concept.

Limitations of AMR

AMR has a few issues (it does).

Limitations of AMR

AMR has a few issues (it does).

- No inflectional morphology for tense and numbers, no articles.

Limitations of AMR

AMR has a few issues (it does).

- No inflectional morphology for tense and numbers, no articles.
- No universal quantifiers.

Limitations of AMR

AMR has a few issues (it does).

- No inflectional morphology for tense and numbers, no articles.
- No universal quantifiers.
- No distinction between real, hypothetical, future or imagined events.

Limitations of AMR

AMR has a few issues (it does).

- No inflectional morphology for tense and numbers, no articles.
- No universal quantifiers.
- No distinction between real, hypothetical, future or imagined events.
- There are issues with the representation of some concepts, e.g. *history teacher* vs *history professor*.

Evaluation and inter-annotator agreement

The authors created a metric to evaluate inter-annotator agreement called *smatch*, that 'reports the **semantic overlap** between two AMRs by viewing each AMR as a conjunction of logical triples' and calculates precision, recall and F-score.

In a inter-annotator agreement study, four experts annotated 100 newswire sentences and 80 web text sentences and then created consensus annotations through discussion. The average annotator vs consensus *smatch* score was **0.83** for newswire and **0.79** for web text. Average inter-annotator agreement score amongst newly trained annotators is **0.71**.

Current AMR Bank

The AMR bank is composed of several thousand sentences and their annotations. Sources include the novel 'The Little Prince', news programs, CCCTV broadcast conversations.
It takes 7-10 minutes to annotate a full sentence, 1-3 minutes to post-edit it.

Applications, extensions to AMR

The authors' main goal is to constitute a large sembank for statistical NLU and MT applications.

A disjunctive AMR has recently been created in order to allow annotators to express the same content in different ways: `official talks` vs `state-sanctioned talks` vs `meetings sanctioned by the state`.

They also wish to include more relations, quantification, temporal relations, etc.

References


Banarescu et al. (2014)
Abstract Meaning Representation (AMR) 1.2 Specification
Proc. Linguistic Annotation Workshop


Banarescu et al. (2013)
Abstract Meaning Representation for Sembanking


Xue et al. (2013)
Not an Interlingua, But Close: Comparison of English AMRs to Chinese and Czech
Proc. LREC


Cai and Knight (2013)
Smatch: an Evaluation Metric for Semantic Feature Structures
Proc. ACL

Finally the end.

Thanks for listening.