Abstract Meaning Representation for Sembanking

Aristeidis Tsialos

March 2, 2015

Outline

Outline

Abstract Meaning Representation

Abstract Meaning Representation (AMR)[1] is a semantic representation language for creating a sembank of English.

Abstract Meaning Representation

Abstract Meaning Representation (AMR)[1] is a semantic representation language for creating a sembank of English.

> "AMRs are rooted, labeled graphs that are easy for people to read, and easy for programs to traverse."

Abstract Meaning Representation

Abstract Meaning Representation (AMR)[1] is a semantic representation language for creating a sembank of English.

> "AMRs are rooted, labeled graphs that are easy for people to read, and easy for programs to traverse."

> "AMRs abstract away from morpho-syntactic idiosyncrasies such as word category (POS), word order, and function words (determiners, some prepositions)."

Outline

Syntactic Treebanks

A success story

In the past couple of decades NLP researchers have had tremendous success in building accurate syntactic parsers.

Syntactic Treebanks

A success story

In the past couple of decades NLP researchers have had tremendous success in building accurate syntactic parsers.

The main reason are syntactic treebanks; simple files of sentences paired with rooted, labeled syntactic trees, that provide statistical parsers with reliable training and testing data sets.

A universal framework for semantic annotatation

The Motivation

A sembank of English could have the same positive impact on automatic semantic annotation that syntactic treebanks had for statistical parsing.

Outline

The Building Blocks

AMR concepts are either English words, PropBank framesets[3] or special keywords.

```
(w  /work-01
    :arg0 (b / boy)
    :manner (h /hard))
```

Example:
the boy is a hard worker
the boy works hard

The Building Blocks

AMR concepts are either English words, PropBank framesets[3] or special keywords.

```
(w  /work-01
    :arg0 (b / boy)
    :manner (h /hard))
```

Example:
the boy is a hard worker
the boy works hard

The Building Blocks

AMR concepts are either English words, PropBank framesets[3] or special keywords.

```
(w  /work-01
    :arg0 (b / boy)
    :manner (h /hard))
```

Example:
the boy is a hard worker
the boy works hard

The Building Blocks

AMR concepts are either English words, PropBank framesets[3] or special keywords.

```
(w  /work-01
    :arg0 (b / boy)
    :manner (h /hard))
```

Example:
the boy is a hard worker
the boy works hard

Outline

Types of relations

AMR uses approximately 100 relations, such as:

```
(s  / hum-02
    :arg0 (p / person
             :name (n / name
                      :op1 "John"
                      :op2 "Smith"))
    :beneficiary (g / girl)
    :time   (w  / walk-01
                :arg0 g
                :destination (t / town)))
```

Example:

John Smith hummed to the girl as she walked to town.

Types of relations

AMR uses approximately 100 relations, such as:

```
(s  / hum-02
    :arg0 (p / person
             :name (n / name
                      :op1 "John"
                      :op2 "Smith"))
    :beneficiary (g / girl)
    :time   (w  / walk-01
                :arg0 g
                :destination (t / town)))
```

Frame arguments, following PropBank conventions[3].

Example:

John Smith hummed to the girl as she walked to town.

Types of relations

AMR uses approximately 100 relations, such as:

```
(s  / hum-02
    :arg0 (p / person
             :name (n / name
                      :op1 "John"
                      :op2 "Smith"))
    :beneficiary (g / girl)
    :time   (w  / walk-01
                :arg0 g
                :destination (t / town)))
```

Frame arguments, following PropBank conventions[3].

General semantic relations.

Example:

John Smith hummed to the girl as she walked to town.

Types of relations

AMR uses approximately 100 relations, such as:

```
(s  / hum-02
    :arg0 (p / person
             :name (n / name
                      :op1 "John"
                      :op2 "Smith"))
    :beneficiary (g / girl)
    :time   (w  / walk-01
                :arg0 g
                :destination (t / town)))
```

Frame arguments, following PropBank conventions[3].

General semantic relations.

Relations for quantities

Example:

John Smith hummed to the girl as she walked to town.

Types of relations

AMR uses approximately 100 relations, such as:

```
(s  / hum-02
    :arg0 (p / person
             :name (n / name
                      :op1 "John"
                      :op2 "Smith"))
    :beneficiary (g / girl)
    :time   (w  / walk-01
                :arg0 g
                :destination (t / town)))
```

Frame arguments, following PropBank conventions[3].

General semantic relations.

Relations for quantities

Relations for date-entities.

Example:

John Smith hummed to the girl as she walked to town.

Types of relations

AMR uses approximately 100 relations, such as:

```
(s  / hum-02
    :arg0 (p / person
             :name (n / name
                      :op1 "John"
                      :op2 "Smith"))
    :beneficiary (g / girl)
    :time   (w  / walk-01
                :arg0 g
                :destination (t / town)))
```

Example:

John Smith hummed to the girl as she walked to town.

Frame arguments, following PropBank conventions[3].

General semantic relations.

Relations for quantities

Relations for date-entities.

Relations for lists.

Outline

The Top Concept of a Sentence

The top-level root of an AMR represents the focus of the sentence or phrase. Choosing a different top concepts is one of the few factors that can change the AMR of a sentence.

Here is an example of two similar semanticly clauses with different top concepts (AMRs).

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

The boy from the college didn't sing.
The college boy didn't sing.

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

The college boy who didn't sing...

Inverse Relations

In the previous example observe the use of :arg0-of for changing the focus of the sentence. This is an example of an *inverse relation*.
Such inverses are defined for all the types of relations.

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

The boy from the college didn't sing.
The college boy didn't sing.

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

The college boy who didn't sing...

Outline

Creating AMRs

AMRs are manually constructed by human annotators.

[1] AMR Editor: amr.isi.edu/editor.html

Creating AMRs

AMRs are manually constructed by human annotators.

An AMR power editor is available online[1], that offers users significant guidance for constructing AMRs.

[1] AMR Editor: amr.isi.edu/editor.html

Creating AMRs

AMRs are manually constructed by human annotators.

An AMR power editor is available online[1], that offers users significant guidance for constructing AMRs.

Using the AMR Editor, annotators are able to translate a full sentence into AMR in 7-10 minutes and postedit an AMR in 1-3 minutes.

[1] AMR Editor: amr.isi.edu/editor.html

Outline

Evaluation

A new metric was created to assess inter-annotator agreement (IAA).

SMatch

Smatch[2] reports the semantic overlap between two AMRs by viewing each AMR as a conjunction of logical triples. Smatch computes precision, recall, and F-score of one AMRs triples against the others.

Initial Results

4 expert AMR annotators annotated 100 newswire sentences and 80 web text sentences. The average annotator vs. consensus IAA (smatch) was 0.83 for newswire and 0.79 for web text.

When newly trained annotators doubly annotated 382 web text sentences, their annotator vs. annotator IAA was 0.71.

Outline

Limitations

AMR has a few noteworthy limitations:

Limitations

AMR has a few noteworthy limitations:

It is not an interlingua; AMR is heavily biased towards English.

Limitations

AMR has a few noteworthy limitations:

- It is not an interlingua; AMR is heavily biased towards English.
- It has no universal quantifier. Words like all modify their head concepts.

Limitations

AMR has a few noteworthy limitations:

- It is not an interlingua; AMR is heavily biased towards English.
- It has no universal quantifier. Words like all modify their head concepts.
- It does not distinguish between real events and hypothetical, future, or imagined ones.

Limitations

AMR has a few noteworthy limitations:

- It is not an interlingua; AMR is heavily biased towards English.
- It has no universal quantifier. Words like all modify their head concepts.
- It does not distinguish between real events and hypothetical, future, or imagined ones.
- It does not represent inflectional morphology for tense and number, and it omits articles.

Summary

Abstract Meaning Representations (AMRs) are rooted, directional and labeled graphs that abstract away from morpho-syntactic idiosyncrasies.

AMRs are easy to construct correctly, and there is a high level of consensus among annotators about the correct AMR for a sentence.

AMRs have several limitations but they could still offer a good base framework for the creation of statistical tools for the semantic parsing of English.

For Further Reading I


Laura Banarescu et al. *Abstract Meaning Representation for Sembanking*. 2013.


Shu Cai and Kevin Knight. *Smatch: an evaluation metric for semantic feature structures. submitted*. 2012.


Paul Kingsbury and Martha Palmer. "From TreeBank to PropBank." In: *LREC*. Citeseer. 2002.