Abstract Meaning Representation for Sembanking

Aristeidis Tsialos

March 2, 2015
Outline

1. **A Semantic Treebank for English**
   - Abstract Meaning Representation for Sembanking
   - Motivation

2. **Defining AMRs**
   - The Building Blocks
   - Types of Relations
   - Inverse Relations

3. **Building & Evaluating AMRs**
   - Creating AMRs
   - Evaluation
   - Limitations
Outline

1 A Semantic Treebank for English
   - Abstract Meaning Representation for Sembanking
     - Motivation

2 Defining AMRs
   - The Building Blocks
   - Types of Relations
   - Inverse Relations

3 Building & Evaluating AMRs
   - Creating AMRs
   - Evaluation
   - Limitations
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)."
Abstract Meaning Representation (AMR)\cite{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 (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).”
1 A Semantic Treebank for English
   - Abstract Meaning Representation for Sembanking
   - Motivation

2 Defining AMRs
   - The Building Blocks
   - Types of Relations
   - Inverse Relations

3 Building & Evaluating AMRs
   - Creating AMRs
   - Evaluation
   - Limitations
In the past couple of decades NLP researchers have had tremendous success in building accurate syntactic parsers.
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.
The Motivation

A sembank of English could have the same positive impact on automatic semantic annotation that syntactic treebanks had for statistical parsing.
1. A Semantic Treebank for English
   - Abstract Meaning Representation for Sembanking
   - Motivation

2. Defining AMRs
   - The Building Blocks
   - Types of Relations
   - Inverse Relations

3. Building & Evaluating AMRs
   - Creating AMRs
   - Evaluation
   - Limitations
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
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
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
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
1. A Semantic Treebank for English
   - Abstract Meaning Representation for Sembanking
   - Motivation

2. Defining AMRs
   - The Building Blocks
   - Types of Relations
   - Inverse Relations

3. Building & Evaluating AMRs
   - Creating AMRs
   - Evaluation
   - Limitations
Types of relations

AMR uses approximately 100 relations, such as:

\[
(s \ / \ \text{hum-02} \\
\quad :\text{arg0} (p \ / \ \text{person} \\
\qquad :\text{name} (n \ / \ \text{name} \\
\qquad \quad :\text{op1} "\text{John}" \\
\qquad \quad :\text{op2} "\text{Smith}")) \\
\quad :\text{beneficiary} (g \ / \ \text{girl}) \\
\quad :\text{time} (w \ / \ \text{walk-01} \\
\qquad :\text{arg0} g \\
\qquad :\text{destination} (t \ / \ \text{town}))
\]

Example:
John Smith hummed to the girl as she walked to town.
AMR uses approximately 100 relations, such as:

\[
\text{(s / hum-02} \\
\quad \text{:arg0 (p / person} \\
\quad \quad \text{:name (n / name} \\
\quad \quad \quad \text{:op1 "John"} \\
\quad \quad \quad \text{:op2 "Smith"))} \\
\quad \text{:beneficiary (g / girl)} \\
\quad \text{:time (w / walk-01} \\
\quad \quad \text{:arg0 g} \\
\quad \quad \text{:destination (t / town))}
\]

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

Frame arguments, following PropBank conventions[3].
Types of relations

AMR uses approximately 100 relations, such as:

\[(s / \text{hum-02})
  \text{:arg0 (p / person)}
  \text{:name (n / name}]
  \text{:op1 "John"}
  \text{:op2 "Smith")}]

\text{:beneficiary (g / girl)}
\text{:time (w / walk-01}]
\text{:arg0 g}
\text{:destination (t / town))}

Frame arguments, following PropBank conventions\[3\].

General semantic relations.

Example:
John Smith hummed to the girl as she walked to town.
AMR uses approximately 100 relations, such as:

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

Example:
John Smith hummed to the girl as she walked to town.
Types of relations

AMR uses approximately 100 relations, such as:

\[(s \ / \ \text{hum-02} \quad :\text{arg0} (p \ / \ \text{person} \quad :\text{name} (n \ / \ \text{name} \quad :\text{op1} "\text{John}" \quad :\text{op2} "\text{Smith}")) \quad :\text{beneficiary} (g \ / \ \text{girl}) \quad :\text{time} (w \ / \ \text{walk-01} \quad :\text{arg0} g \quad :\text{destination} (t \ / \ \text{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:

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

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.
1. A Semantic Treebank for English
   - Abstract Meaning Representation for Sembanking
   - Motivation

2. Defining AMRs
   - The Building Blocks
   - Types of Relations
   - Inverse Relations

3. Building & Evaluating AMRs
   - Creating AMRs
   - Evaluation
   - Limitations
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 semantically clauses with different top concepts (AMRs).

\[
(s / \text{sing-01} \quad (b / \text{boy} \\
: \text{arg0} (b / \text{boy} \\
: \text{source} (c / \text{college})) \\
: \text{polarity} -)
\]

The boy from the college didn’t sing.

\[
(b / \text{boy} \\
: \text{arg0-of} (s / \text{sing-01} \\
: \text{polarity} -) \\
: \text{source} (c / \text{college}))
\]

The college boy who didn’t sing...

The college boy 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.

\[(s / \text{sing-01} :\text{arg0} (b / \text{boy} :\text{source} (c / \text{college)}) :\text{polarity} -))\]

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

\[(b / \text{boy} :\text{arg0-of} (s / \text{sing-01} :\text{polarity} -) :\text{source} (c / \text{college}))\]

The college boy who didn’t sing...
# A Semantic Treebank for English

- Abstract Meaning Representation for Sembanking
- Motivation

## Defining AMRs

- The Building Blocks
- Types of Relations
- Inverse Relations

## Building & Evaluating AMRs

- Creating AMRs
- Evaluation
- Limitations
Creating AMRs

AMRs are manually constructed by human annotators.
AMRs are manually constructed by human annotators.

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

1. A Semantic Treebank for English
   - Abstract Meaning Representation for Sembanking
   - Motivation

2. Defining AMRs
   - The Building Blocks
   - Types of Relations
   - Inverse Relations

3. Building & Evaluating AMRs
   - Creating AMRs
   - Evaluation
   - Limitations
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.
1. A Semantic Treebank for English
   - Abstract Meaning Representation for Sembanking
   - Motivation

2. Defining AMRs
   - The Building Blocks
   - Types of Relations
   - Inverse Relations

3. Building & Evaluating AMRs
   - Creating AMRs
   - Evaluation
   - 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.
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.
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.
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.
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.
