Going from UD towards AMR

Kelvin Han, Siyana Pavlova

3rd June 2019

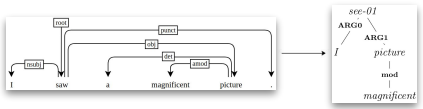
Université de Lorraine · Institut des sciences du Digital Management & Cognition · Loria Laboratoire lorrain de recherche en informatique et ses applications

Supervised Project with Bruno Guillaume and Maxime Amblard

Overview

Objective

Develop a system which transforms **Universal Dependencies (UD)** annotated sentences to **Abstract Meaning Representation (AMR).**

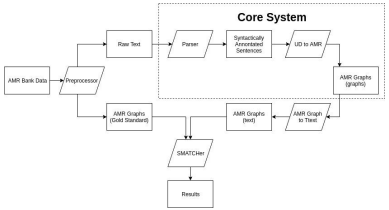


Motivation

- Limited semantically-oriented annotated data, especially for languages other than English.
- Such data is useful for many aspects of NLP.
- But large amount of UD annotated data in more than 70 languages.

=> Using UD annotated data can help reduce the effort and costs of producing semantically-oriented annotated data.

System Architecture



Frameworks

AMR

- **Semantic representation framework**
- Focuses on **predicate-argument structure**
- "Who does what to whom" captured in a single data structure
- Makes extensive use of PropBank predicate frames

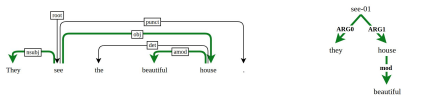
UD

- Framework for **syntactic annotation**
- Aimed at building a **cross-linguistically consistent** treebank
- Driving principles:
  - **Dependency**
  - **Lexicalism**
  - Preference for **content words**

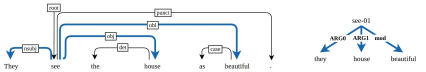
UD to AMR

Due to the nature of both frameworks, there are some parallels between UD graphs and AMR graphs.

"They see the beautiful house"



"They see the house as beautiful"

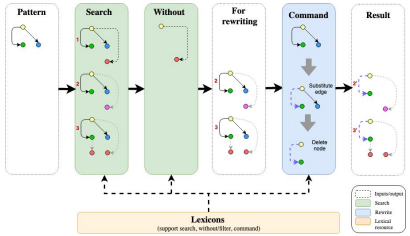


Walkthrough

Grew

- **Graph rewriting tool** for NLP, developed by researchers at Loria
- Uses the Ocaml programming language, Python package available
- Grew allows for:
  - **Searching** through graphs using **patterns**;
  - Applying **rewrite commands**;
  - Creating parameters for rules using **lexicons;**
  - Combining patterns into **strategies and packages**.

Graph Rewriting System (GRS)



Lexicons

Two main reasons lexicons are useful in the context of this project:

- Predicate sense disambiguation;
- Choosing the correct argument structure.

Our lexicon was constructed by manually annotating **834 PropBank predicates**, enriching them with semantic role labels.

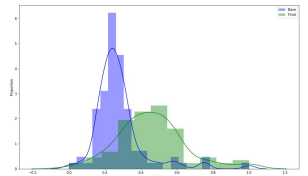
Results

System Evaluation

The system was evaluated on the first 100 sentences of the AMR Bank*. **Our Final GRS** encompases 183 GRS rules, grouped in 10 packages. 55 of the rules do not have a repetitive structure. A lexicon containing all predicates contained in the first 100 sentences was used too.

A comparison with **Base**, which contains a minimal set of preparatory rules and without a lexicon, shows a **0.19 increase** in F1-score.

	Base			Our Final GRS		
	Min	Max	Arithmetic Mean	Min	Max	Arithmetic Mean
Precision	0.00	1.00	0.27	0.00	1.00	0.47
Recall	0.00	1.00	0.28	0.00	1.00	0.46
F1-score	0.00	1.00	**0.27**	0.00	1.00	**0.46**



*The AMR Bank contains the 1,562 sentences of The Little Prince in English, annotated by human experts. It is provided by the Linguistics Data Consortium.

Lexicon Inter-Annotator Agreement

497 randomly selected PropBank predicates were annotated by the two authors. An **inter-annotator agreement** of **0.65** across the 1,273 annotated semantic roles was achieved. This was computed using Cohen's Kappa.

The code for this project is available at https://github.com/siyanapavlova/AMR_annotations.