Advanced Methods in NLP Lecturer: Jonathan Berant

Final Project - RST Parser

1 Data - RST Discourse Treebank

The RST-DT is a large corpus of documents annotated with EDU segmentation (EDU is the atomic unit of RST schema) and full text level rhetorical structure. It contains 385 articles from the Wall Street Journal. These articles are a subset of the Penn Treebank corpus.
Your are provided with 347 samples from the RST-DT, which were divided into train and dev sets. The remaining 38 trees (the test set of RST-DT) will be used to evaluate your parser performance. The provided data includes 3 files for each train/dev sample:

1. `*.out` - the full original text
2. `*.out.edus` - the segmentation of the full text into EDUs
3. `*.out.dis` - the discourse tree

You can download the dataset as well as supporting code from `~omrikosh/advanced_nlp/rst_project/`

Carlson and Marcu [2001] provides detailed explanation regarding the RST schema, including many examples for different relation types and nuclearity. The class presentation `http://www.cs.tau.ac.il/~joberant/teaching/advanced_nlp_spring_2018/files/RST.pdf` includes materials regarding building RST Shift-Reduce parser.

1.1 Data Preprocessing

The original RST schema includes 78 relation types. Following existing parsers, we will partition those types into 19 clusters (so relations in the same cluster share some rhetorical meaning). The method `map_to_cluster` in `utils.py` maps an original relation type into its cluster.

Advice: Make sure you understand the difference between same-unit, span and all other clustered relation types.

2 Parser Evaluation

We will use the following definitions when evaluating RST parsers (all examples in this section are based on tree `0600` from the train set):

- $span(t)$ - span set for tree t (including its leaves, without its root). $span(t_{0600}) = \{(1,1),(2,2),(3,3),(2,3)\}$

- $nuc(t)$ - span and nuclearity set for tree t. $nuc(t_{0600}) = \{(1,1,N),(2,2,N),(3,3,S),(2,3,S)\}$
- $realtion(t)$ - span, nuclearity and relation set for tree t: $nuc(t_{0600}) = \{(1,1,N,span),$ $(2,2,N,span),(3,3,S,elabortaion),(2,3,S,elaboration)\}$

RST parsers are evaluated using three F1 metrics:

- $F1_{span} = \frac{|span(t_{gold}) \cap span(t_{pred})|}{|span(t_{gold})|}$
- $F1_{nuc} = \frac{|nuc(t_{gold}) \cap nuc(t_{pred})|}{|nuc(t_{gold})|}$
- $F1_{relation} = \frac{|relation(t_{gold}) \cap relation(t_{pred})|}{|relation(t_{gold})|}$

The $F1_{span}$ metric measures how accurate the discourse parser is in finding the right structure of the discourse tree, while the $F1_{nuc}$ and $F1_{relation}$ metrics measure the parser's ability to find the right labels (i.e., nuclearity or relation labels) in addition to the right structure. Morey et al. [2017] provides explanation regarding the evaluation procedure, as well as comparison between existing notable parsers.

2.1 Evaluating Predicted Discourse Trees

You can evaluate your predicted discourse trees by executing the script `evaluation.py` with the following arguments:

1. The first argument is a path to the folder containing the ground truth trees. Each tree should have a separate file (with the tree number as file name), containing the tree structure in the format defined by $realtion(t)$. You are provided with such files for trees 0600 and 0603.
2. The second argument is a path to the folder containing the predicted trees. The format of the files should be the same as in (1).

The script evaluates the accuracy of the predicted trees using the three F1 metrics that were previously described.

References

Lynn Carlson and Daniel Marcu. Discourse tagging reference manual. *ISI Technical Report ISI-TR-545*, 54:56, 2001.

Mathieu Morey, Philippe Muller, and Nicholas Asher. How much progress have we made on rst discourse parsing? a replication study of recent results on the rst-dt. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1319–1324, 2017.