Predicting Discourse Trees from Transformer-based Neural Summarizers

Wen Xiao, Patrick Huber and Giuseppe Carenini
Department of Computer Science, University of British Columbia
xiaowen3@cs.ubc.ca

Motivation
Discourse tree is important for extractive summarization task. [1]

Q: Do Summarizers Learn Discourse Information?

Idea: Does Summarizers’ Attention Align with Human-annotated Trees?
1. Build discourse trees based on the attention matrices of trained extractive summarization model,
2. Verify whether and how they are aligned with human-annotated discourse trees.

Step 0: Train Summarizer
Structure:
- BERT EDU Encoder
- Transformer-based Document Encoder
- Decoder
Dataset: CNNDM, NYT

Step 1: Get Attention
1. Average over each layer
2. Attention matrices per head per layer

Step 2: Build Discourse Trees
Constituency Tree Generation
CKY Algorithm: dynamic programming bottom-up alg.
Eisner Algorithm: dynamic programming alg., can only produce projective trees.
CLE Algorithm: find the maximum spanning tree in the graph, and can produce both projective or non-projective trees.

Dependency Tree Generation

A: Extractive summarization models do learn discourse information implicitly
- More dependency information is learnt
- Most of the discourse information is concentrated on a single head.
- The generated trees have similar properties as the ground-truth trees, as they both can capture both local dependencies and long-distance dependencies.
- The results are consistent across datasets and models → the learned discourse information is general and transferable inter-domain.

Step 3: Experiments
Settings
Datasets: RST-DT, Instruction, GUM
Evaluation Metric:
- Constituency Tree: RST-Parseval Score
- Dependency Tree: Unlabeled Attachment Score
Constraints:
- No Constraint / Sentence Constraint

Overall
--- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | ---
RST-DT | 61.2 / 59.7 | 76.2 / 74.6 | 23.7 / 24.8 | 28.3 / 28.2 | 21.6 / 17.5 | 29.3 / 19.6
CNNDM-2-1 | 60.3 / 59.9 | 75.4 / 75.0 | 23.8 / 20.5 | 13.8 / 27.8 | 7.3 / 17.3 | 16.1 / 28.5
Random | 58.6 / 51.1 | 74.1 / 51.0 | 11.2 / 20.2 | 20.3 / 20.2 | 1.7 / 0.08 | 18.7 / 0.1

Structure Properties (Best Head)
Per-head

* More detailed results and analysis on generated trees can be found in the paper.