Fine-tune BERT for Extractive Summarization

Scott Chase Waggener
Pankaj Kr Jhawar
Yuan Chiang

November 3, 2019

1 Citation

References

[1] Yang Liu. *Fine-tune BERT for Extractive Summarization*. 2019. arXiv: `1903.10318 [cs.CL]`.

2 Task

Liu (2019) attempted to extend the BERT masked-language model to the task of extractive summarization. While BERT is a more powerful architecture for representing the complex features involved in extractive summarization, BERT's raw inputs and outputs are not suitable for summarization. Liu proposes modifications that resolve these issues.

3 Data

Two benchmark datasets were used:

1. CNN/DailyMail news highlights dataset (Hermann et al., 2015)
2. New York Times Annotated Corpus (Sandhaus 2008).

Both datasets consist of publications and an associated summary or bullet points.

4 Approach

The paper first proposes modifications to the inputs of BERT to circumvent the incompatibility of BERT's token-grounded inputs with summarization's sentence-grounded inputs. Specifically, [CLS] and [SEP] tokens are added to indicate the beginning and end of a sentence, and interval segment embeddings were used to further distinguish adjacent sentences. The interval segment embeddings were specifically conditioned on whether a sentence index is odd or even. BERT was then fed these token, position, and interval segment embeddings as input.

Summarization layers were added to the output of BERT to capture document level features. The author tried multiple architectures for the summarization layers including a simple classifier, LSTM, and transformer. In each case the final layer was a sigmoid classifier that would label an input sentence depending on whether or not it should appear in the final summary.

5 Evaluation

Experiments were run using PyTorch, OpenNMT, and the bert-base-uncased pretrained BERT model. The training, testing, and validation splits were performed as described in Hermann et al. (2015) and Durrett et al. (2016). Model checkpoints were saved every 1000 steps, and the best model was selected from the top-3 checkpoints. Performance of BERTSUM and other notable networks were examined using intrinsic evaluation with the ROUGE-1, ROUGE-2, and ROUGE-L metrics. The simple classifier, LSTM, and transformer variants of BERTSUM were all benchmarked. LEAD was used as a baseline, which uses only the first three sentences of a document as the summary. The performance of PGN, DCA, REFRESH, and NEUSUM were presented for comparison using the performance values cited in each network's respective paper. BERTSUM achieved the best performance on each metric, with the transformer variant being the best performer among BERTSUM variants. The LSTM variant did not improve performance versus the simple classifier variant.