Condenser and Adaptive Batch Scheduling for OpenQA Passage Retrieval

Zikun Cui, Xiaoying Yang, Yiting Zhao

1 Department of Civil and Environmental Engineering, Stanford University
2 Department of Computer Science, Stanford University
3 Department of Management Science & Engineering, Stanford University

Methods

Adaptive Batch Scheduling
- Query: Training dataset \( T = \{q_1, q_2, \ldots, q_n \} \), where \( q_i \) is the question and passage pair (q, p)
- Method
- Calculate Hardness Scores:
  - Scores training instances in the initialized sample if this is the first batch of training
  - Higher hardness score indicates a more difficult instance
- Co-conversion
- Augment training examples with new text
- Loss Function for each batch

Baseline Pre-trained Language Models
- BERT: Bidirectional transformer pre-trained on masked language modeling
- ELMo: Representations of words as contexts-dependent word vectors
- Co-conversion

Condenser
- Early and final layers
- Input: query, paragraph, and passage
- Output: passage scores
- Architectures: CNN, attention

GCoCondenser
- Augment the Condenser with additional losses
- Loss Function for each batch

Analysis

Main Findings
- Condenser
- Adaptive Batch Scheduling
- Generate better query and passage encoding
- Batch size: BERT + ELMo
- Decrease in BLEU score
- Simpler model architecture reduces perplexity
- Early fusion

Conclusion

Future Work
- Expand batch size of BERT
- Improve query and passage encoding
- Test with different model architectures
- Further improve batch size

Limitations
- Lack of hardware resources