Span-based Hierarchical Semantic Parsing for Task-Oriented Dialog: Supplementary Material

A Span embedding features

We slightly modify the span embedder from Lee et al. (2017) as follows. To compute the embeddings of spans $x_{i:j}$ in an utterance $x = (x_0, \dots, x_{n-1})$, We first add boundary tokens x_{-1} = `<BOS>` and x_n = `<EOS>`. We then embed each token x_i as a vector e_i, and then apply multi-layered bidirectional LSTMs. Let h_i^{F} and h_i^{B} be the ith forward and backward hidden states, and let h_i be their concatenation. The embedding of span $x_{i:j} = (x_i, \dots, x_{j-1})$ is then a concatenation of the following vectors:

- Endpoint hidden states: h_i and h_{j-1}.
- Uniform average of hidden states:

$$h_{\mathrm{uni}} = \frac{1}{j-i}(h_i + \cdots + h_{j-1}). \qquad (1)$$

- The difference in hidden state after reading the span in each direction (Cross and Huang, 2016; Stern et al., 2017): $h_{j-1}^{\mathrm{F}} - h_{i-1}^{\mathrm{F}}$ and $h_i^{\mathrm{B}} - h_j^{\mathrm{B}}$.
- The attention-weighted average of token embeddings (Lee et al., 2017; He et al., 2018): we compute attention weights over the $j - i$ positions:

$$a_k \propto \exp\,[w_{\mathrm{a}}^\top h_k]. \qquad (2)$$

  Then we average the token embeddings:

$$h_{\mathrm{att}} = \sum_{k=i}^{j-1} a_k e_k. \qquad (3)$$

- Span length (Lee et al., 2017; He et al., 2018): we bin the length into buckets [1, 2, 3, 4, 5–7, 8–15, 16–31, 32–63, 64+] and use a 20-dimension embedding to represent each bucket.

B Hyperparameters and training details

Tokens appearing less than 2 times in training data are converted into `UNK` tokens. We also perform word dropout with probability proportional to the frequency of the word in training data.

To compute the node scores f_{n} and edge scores f_{e}, we apply 2-layer feedforward networks with hidden sizes of 200 over the span embeddings. Label embeddings in f_{e} are 150 dimensional. The parameters are trained using Adam (Kingma and Ba, 2015) with the initial learning rate of 5×10^{-4} and early stopping. We apply dropout with probability 0.2 before each LSTM and feedforward layer.

References

James Cross and Liang Huang. 2016. Span-based constituency parsing with a structure-label system and provably optimal dynamic oracles. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Luheng He, Kenton Lee, Omer Levy, and Luke S. Zettlemoyer. 2018. Jointly predicting predicates and arguments in neural semantic role labeling. In *Association for Computational Linguistics (ACL)*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

Kenton Lee, Luheng He, Mike Lewis, and Luke S. Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Mitchell Stern, Jacob Andreas, and Dan Klein. 2017. A minimal span-based neural constituency parser. In *Association for Computational Linguistics (ACL)*.