Our model is an attention based encoder-decoder model, which takes a preprocessed representation of the RDF triple graph as input, and directly outputs lexicalizations of the input graph. Instead of using the heuristic delexicalization approach used in the baseline model, we opt for subword representations, which have recently been a key component of state of the art results in Neural Machine Translation. The use of subword encoding allows us to ensure that we never encounter unknown tokens in either training or test data, and also reduces the sparsity of the small training dataset.

We made the decision early on to focus on the raw text within the XML and avoid the use of delexicalization. We start by extracting modified triple sets and lexicalizations from the XML. Within the triple sets we removed underscores from the subject and object. We then broke apart the camel casing of the predicates. We created a dictionary for subword encoding from the triple sets and lexicalizations in the training dataset to be used in the subsequent byte pair encoding step. Finally we applied the moses tokenizer and byte pair encoding to the text. The triples were then chained together using special tokens that we call "tuple delimiters":,
e.g. <*subject*> __predicate_start__ <*predicate*> __predicate_end__ <*object*> __triple__.

For training we used Nematus, an attention-based encoder-decoder model for neural machine translation. Our encoder is a single layer bidirectional GRU, and our decoder also uses GRU cells.

We take advantage of several features of Nematus: encoder and decoder embeddings are tied, and we use *data* dropout on both the input and output sequences. Because this dataset is very small when compared with the parallel datasets used to train state-of-the-art machine translation models, we hypothesized that data dropout would help to regularize the model, and prevent overfitting, and our experiments showed that this was indeed the case.

We use the adadelta optimizer during training, and apply dropout to all feed forward parameters, and layer normalization for the encoder and decoder GRU layers. After training for several hundred epochs we continued training with minimum risk training, using sentence level BLEU score as the metric for computing the expected score from a sample size of 100 for each input.

Finally, we chose the model with the best validation Bleu score, 53.3, for submission. The final outputs were decoded from the best model using the Nematus beam search implementation with beam size 12.

Citations

Dzmitry Bahdanau, Kyunghyun Cho: "Neural Machine Translation by Jointly Learning to Align and Translate", 2014; arXiv:1409.0473.
Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun: "Minimum Risk Training for Neural Machine Translation", 2015; arXiv:1512.02433.
Rico Sennrich, Barry Haddow: "Neural Machine Translation of Rare Words with Subword Units", 2015; arXiv:1508.07909.

Jimmy Lei Ba, Jamie Ryan Kiros: “Layer Normalization”, 2016; arXiv:1607.06450.
Rico Sennrich, Orhan Firat, Kyunghyun Cho, Alexandra Birch, Barry Haddow, Julian Hitschler, Marcin Junczys-Dowmunt, Samuel Läubli, Antonio Valerio Miceli Barone, Jozef Mokry: “Nematus: a Toolkit for Neural Machine Translation”, 2017; arXiv:1703.04357.