Rahul Thapa
Paper: Neural Machine Translation by Jointly Learning to Align and Translate

Summary

In this paper, the authors present a newly emerging approach to machine translation called neural machine translation. Unlike the traditional phrase-based translation system which consists of many small sub-components that are tuned separately, neural machine translation attempts to build and train a single, large neural network that reads a sentence and outputs a correct translation.

Most of the proposed neural machine translation models belong to a family of encoder-decoders. A potential issue with this encoder-decoder approach is that a neural network needs to be able to compress all the necessary information of a source sentence into a fixed-length vector which may make it difficult for the network to cope with long sentences. Therefore, in order to deal with this issue, the authors introduce an extension to the encoder-decoder model which learns to align and translate jointly. Each time the proposed model generates a word in a translation, it (soft-)searches for a set of positions in a source sentence where the most relevant information is concentrated. The model then predicts a target vector word based on the context vectors associated with these source positions and all the previously generated target words. The most distinguishing feature of this approach is it encoded the input sentence into a sequence of vectors and chooses a subset of these vectors adaptively while decoding the translation. This frees a neural translation model from having to squash all the information of a source sentence, regardless of its length, into a fixed-length vector. For this, the authors implement a mechanism of attention in the decoder which decides part of the source sentence to pay attention to. By letting the decoder have an attention mechanism, the authors relieve the encoder from the burden of having to encode all information in the source sentence into a fixed-length vector.

The authors would also like the annotation of each word to summarize not only the preceding words, but also the following words. Therefore, they propose to use bidirectional RNN. Bidirectional RNN consists of forward and backward RNN's. The forward RNN reads the input sequence as it is ordered and calculated a sequence of forward hidden states. The backward RNN reads the sequence in the reverse order resulting in a sequence of backward hidden states. They obtain an annotation of each word by concatenating the forward hidden state and the backward one. In this way, the annotation contains the summaries of both the preceding words and the following words and will be focused on the words around the target word.

The authors trained two types of models. First, an RNN Encoder-Decoder which they called RNNencdec and the other their proposed model which they called RNNsearch. The encoder and decoder of RNNencdec have 1000 hidden units. The forward and backward RNN of the encoder of RNNsearch each has 1000 units. The decoder of RNNsearch also has 1000 units. In both cases, they used a multilayer network with a single maxout hidden layer to compute the conditional probability of each target word.

Quantitatively, the RNNsearch performed better than conventional RNNencdec. The authors used the BLEU score as a comparing metric. Their model also performed comparably to conventional phrase-based translation system (Moses), which is a significant achievement considering that Moses uses a separate monolingual corpus in additional to the parallel corpora the authors used to train their model. They also noticed that the performance of the RNNencdec dramatically drops as the length of the sentence increases. However, their RNNsearch models are more robust to the length of the sentences. Specially, RNNsearch-50 shows no performance deterioration even with sentences of length 50 or more. From the qualitative analysis where they investigated the (soft-)alignment generated by the RNNsearch, they were able to conclude that the model can correctly align each target word with the relevant words, or their annotations, in the source sentence as it generated a correct translation.

Strengths

- The paper is nicely organized with enough explanation in each section.
- The authors do a good job of explaining why their model performs better than the older encoder-decoder approach. They back this assertion by some substantial qualitative and quantitative results.
- The example of the long sentences and its translation as given by their model and the encoder-decoder approach really helps understand the conclusion better. Therefore, that section adds to the strength of the paper.

Weaknesses

- Some of the math is hard to understand and he explains it out of order.
- The paper is lacking a better neural network architecture and an explanation of architecture other than from the mathematical perspective.

Confusions

- I felt it hard to follow through the math that the authors include in the paper.
- I also wanted to learn a little bit more about how the neural encoder-decoder based approach works. I was a little confused while reading the paper because I did not have a background on that topic.

Discussions

- What is the purpose of training a bidirectional RNN? How does it help in making the translation better?
- What is the advantage of jointly learning to align and translate over the encoder-decoder approach?
- How does the architecture of a bidirectional RNN look like?
- How does pretraining the model on monolingual data first might affect the performance of the network?