Model Improvement

Learn strategies for improving an encoder-decoder model.

Chapter Goals:

- Learn strategies for improving an encoder-decoder model
- Run the encoder-decoder model in inference mode

A. Training strategies

Good encoder-decoder models tend to have a large number of weight parameters, since they consist of large LSTM/BiLSTM layers. Because of this, it usually takes a long time to train an encoder-decoder model to convergence.

To speed up training, it might be beneficial to use a larger batch size during the initial stages of training, then reduce the batch size once the model begins to show considerable improvement in reducing the loss function. A larger batch size allows the model to get through more data, and as long as we reduce the batch size at an early enough stage the model won't miss its convergence point.

Another strategy to speed up training is to start off with a larger learning rate, then gradually decrease the learning rate as the model trains.

B. Domain-specific strategies

The encoder-decoder model can be used for basically any seq2seq task. This includes domains like machine translation, text summarization, and dialog systems (e.g. chatbots). However, each of these domains has their own special features, which may require tweaks to how the encoder-decoder model is used.

For example, in text summarization it is often a good idea to truncate the input text to a maximum length. This is because input text can be incredibly long (think huge news articles) and also because much of the important information in a text can be found in the first few paragraphs. Similarly, we

would want to set a maximum decoding length when creating summaries, so that the summaries don't just replicate the entire article.

When training a dialog system such as a chatbot, the data we use and process is incredibly important. For example, we would want to train a customer service chatbot on data that contains dialogue specific to customer service. Furthermore, we need to be careful to prune out profanity and other words that we don't want the dialog system to output.

In general, it is important to understand the seq2seq task's domain prior to building an encoder-decoder model for the task. This way you can make the necessary data and model tweaks that will generate the best performance.