Stanford CS224N NLP with Deep Learning

Winter 2021

Lecture 10 – Transformers and Pretraining

Subword Models

So far, during training, we have assumed a fixed vocabulary built from the training set. Under this assumption, novel words seen at test time are mapped to a single `<UNK>` token. This, however, does not make a lot of sense, especially in languages with complex morphology.

The answer to this is *subword modelling*, or reasoning about structure below the level of words. At training and testing time, each word is split into a sequence of known subwords.

Byte-Pair Encoding

Byte-pair encoding is an effective strategy for defining a subword vocabulary. We start with a vocabulary containing only single characters and an end-of-word token, and then add pairs of commonly adjacent characters iteratively.
Thus common words end up in the vocabulary, while rarer words are split into subwords.

Pretrained Embeddings

In modern NLP, almost all parameters in networks are initialised via pretraining. Pretraining methods hide parts of the input from the model, and train the model to reconstruct these parts.

We train a neural network on the language modelling task (word prediction). We save the parameters of this network.

These parameters serve as an *initialisation point* for the downstream task's network. They can be finetuned to the task during training.

The advantage of starting with $\text{argmin}_\theta \mathfrak{L}_{\text{pretrain}}(\theta) = \hat{\theta}$ is that this makes it easier to find $\text{argmin}_\theta \mathfrak{L}_{\text{finetune}}(\theta)$.

Pretraining Decoders

Decoders can be finetuned by training a classifier on the last word's hidden state, starting with randomly initialised parameters.

They can also be finetuned on sequence generation, by initialising the classifier (from hidden states to probability distributions over the vocabulary) from the pretrained parameters.

Pretraining Encoders

Encoders are pretrained by *masking* the input. Certain words in the input are replaced with a `<MASK>` token, and some are replaced with normal words. The model is trained to predict the true words.

BERT was pretrained on a *next sentence prediction* task.

Pretraining Encoder-Decoders

Encoder-decoder architectures are commonly trained by language modelling. Another variant of the task is to blank out spans from the input and train the model to predict the missing spans (called *span corruption*).