Deep Learning Lab 7

Jack Hilton-Jones (jhj1g23@soton.ac.uk)

I. Sequence to Sequence Model

The sequence-to-sequence model is finished by the completing the forward method, after this the model is trained and the plot of the training loss to validation can be seen in Figure 1.

```
class Encoder(nn.Module):
    def __init__(self, input_dim, emb_dim, hid_dim):
        super().__init__()

        self.hid_dim = hid_dim
        self.embedding = nn.Embedding(input_dim, emb_dim)
        self.rnn = nn.LSTM(emb_dim, hid_dim)

    def forward(self, src):
        embedded = self.embedding(src)
        outputs, (hidden, cell) = self.rnn(embedded)
        return hidden, cell
```

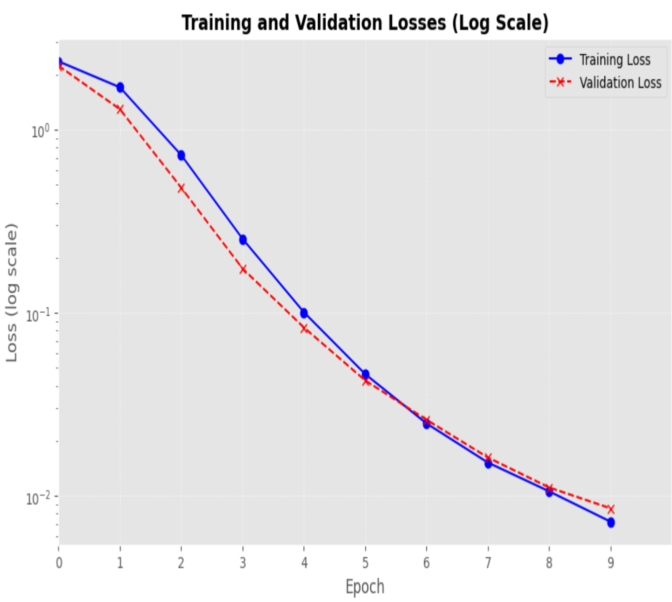

Fig. 1. Sequence-to-Squence Model Training and Validation Loss

After the model is trained, the morse code can be debugged with the given code from the lab sheet plus a couple of lines to keep the proper dimensions.

```
def decode(code):
    out = ''
    for chunk in code.split(' '):
        num = ds.encode_morse('^ ' + chunk + ' $').unsqueeze(1)
        pred = model(num.cuda(), maxlen=2)
        pred_dim = [*pred.shape][-1]
        pred = pred[1:].view(-1, pred_dim).argmax(-1)
        out += ds.decode_alpha(pred.cpu())[::-1]
    return out
```

The output of the following code is:

$$\text{psprbefflfng} \tag{1}$$

$$\text{hfshecderfmheffefrsed} \tag{2}$$

$$\text{hlshefcnfnpherfccin} \tag{3}$$

The data is split into smaller segments that contain a portion of the data which are processed by the model and the chunk boundaries are positioned between the spaces of the Morse code. The performance of the model on the chunk size is dependent on the characteristics of the data the model was trained on. If the chunk is too small, this will not provide enough context for the model to make an accurate judgement. However, if the chunks are too large, this can exceed the model's training capacity.

The performance of the model is tested on different chunk sizes. The results are in Table I.

Chunk Size	**Decoded Text Output**
2	'psprbelcolng' 'blhbbpdeplthlcfefrseed' 'jlhb pbgocpheroccian'
5	'rp lbflolc^' 'll bhcf bpy lofefrse^^' 'll bhc locle lfoccin'
10	'^^lcol^^^^^' '^lof bpbjfelfe^^^^^^^' '^ocf lochpfcn'

TABLE I
DECODED TEXT OUTPUTS FOR DIFFERENT CHUNK SIZES.

As the chunk size is increased the ^ symbols increase in the decoded output which represents the padding, this indicates the model is less certain about the translation from Morse code. With longer chunks, the model outputs more incorrect symbols, as it is difficult to capture the relationships in the data with longer sequences. This relates to the training data in the sense that if the model was trained with short sequences, the model weights would be optimised for that particular length. If longer sequences are not present, the model will not recognise and learn to decode them correctly. This is what is happening in this model, as the model has been trained on short sequences, when increasing the chunk size the model's accuracy decreases.

Depending on how the model is trained can affect the performance, as LSTMs can cause problems with longer sequences due to vanishing gradient issues. In addition, the hidden state may be too small to be able to remember long sequences.