Yang Liu
3323643

Lab Exercise 7

COMP6248
yl58n21@soton.ac.uk

1 Sequence-to-Sequence Modelling

1.1 Complete and train a sequence-to-sequence model

The snippet of code and the loss curve which is shown in **Figure 1** are as follows.

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

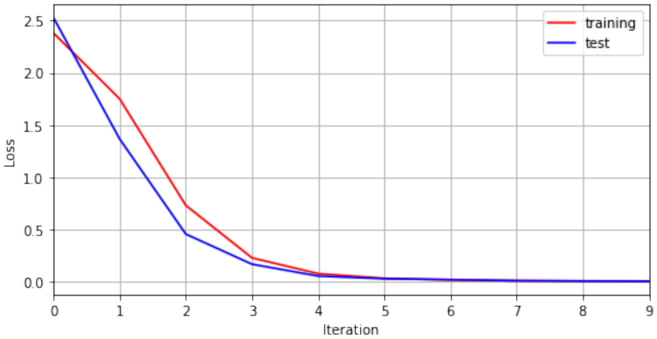

Figure 1: Loss

1.2 Now use it!

After decoding the code, we are expected to **answer the following**

1.2.1 why is the order of the output reversed

When we are translating one sentence, we go through from the beginning to the end of sentence and start to translate the beginning, but the translator might already forgotten it while still remember the end(just learned), so its better to start from the end which shows the reverse order of the output.

1.2.2 what is the point of teacher forcing

The teacher forcing is a way of training the RNN model that uses ground truth as input instead of output of the prior time step. Using output of prior time step of the model could result very large loss and slow the convergence because it usually starts with some random guesses and get punished for every words it generates.

1.3 Sequence Lengths

The decode method chunked each character into a block. By enlarging the block size(more characters), we find that the translation could keep most of the information in the middle of the sentence and translate them correctly however lose some characters at the beginning and the end of the sentence(mostly beginning). This shows our model works not so well with long sentence which is probably due to the short length of the training data.