CSC321: Assignment #3

Xiangyu Kong
kongxi16

March 22, 2018

Problem 1

1. The model will not perform well on long sequences. This is because after the encoder compresses the long sequence of inputs into the fixed length vector h_v, h_v will have to convey much more information. This makes the vector harder to interpret for the decoder layer, and the decoder will less likely give a correct result.

2. The added words are 'team', 'problematic', 'ink', 'obviously', 'shy', 'philosophical', and 'supercalifragilisticexpialidocious'

   The predicted results are shown in below.

   As we can see, for shorter words like team and ink, the translator does a good job. However, for longer words like 'problematic', 'philosophical' and 'supercalifragilisticexpialidocious', the translator does a very poor job.

   Listing 1: Translated Results

```
team --> eamtay
problematic --> opserarcepray
ink --> inkway
obviously --> odloy-eylway
shy --> ytsay
philosophical --> isorecarcalclay
supercalifragilisticexpialidocious --> afessfsesesssipicici
```

Problem 2

1. The problem with teacher forcing is that during training, the decoder are given in ground truth token as input to next time unit. However, when training, the decoder has to predict the next letter given the previous prediction. If the previous prediction is incorrect, then the error will be enlarged through time because every time step, the decoder is predicting based on a false previous input.

2. A solution would be to generate a token that contains necessary and important information for predicting the next result to use as the next time step's input instead of simply using the previous result.

Problem 3

See model.py for implementation

Problem 4

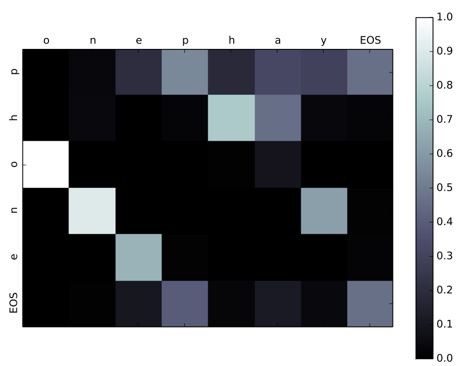
See model.py for implementation

Problem 5

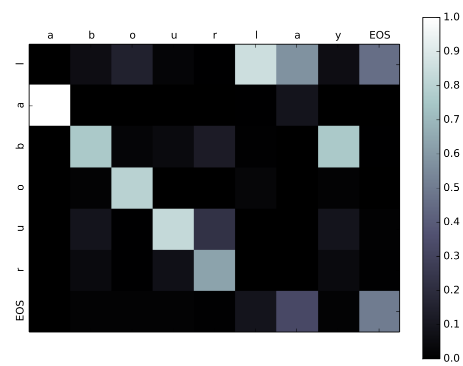
The words and their translations are given in the listing below. As we can see, simple short words are more likely to produce correct translations. For example, “cake”, “labour”, “drink” and “phone” all have the right translations and the visualizations are given in Fig1. In the visualizations, we see that the network goes linearly through the letters after the first consonant (block). This shows that the network can correctly capture these simple structures of the word.
However, the network starts to make mistakes when it gets unusual and long words. For example, “p-value”, “well-done”, “anthropocene”, etc are translated incorrectly. The visualizations are given in Fig2. We can see that for “p-value”, the network understands that “p” should be placed at the back but it does not know what to do with the dash “-”. For “anthropocene”, the network successfully understands that no letter needs to be placed at the back but gets lost after the second “a”. This is because the network was not trained enough with these specific structures and thus does not know how to translate them.

Listing 2: words and predictions

```
roomba --> ooodrray
cake --> akecay
labour --> abourlay
drink --> inkdray
phone --> onephay
bouquet --> ouquerbay
anthropocene --> anthanthayentay
p-value --> -ueyeydray
well-done --> elay-onelyyday
thimolystically --> ipthallystitesthay
supercalifragilisticexpialidocious --> upercacalicalicaalic
```

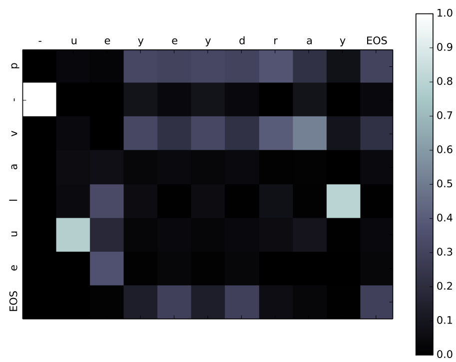


(a) phone

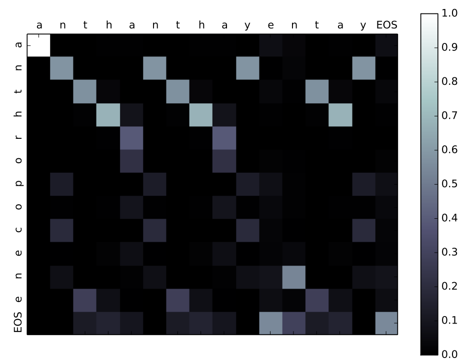


(b) labour

Figure 1: Correct Visualizations



(a) p-value



(b) anthropocene

Figure 2: Incorrect Visualizations