CSC421 Programming Assignment 3

Yubo Wang 1002138377

Part 1

1. From my perspective, the model will not perform well on long sequences since after encoder need to compress the long sequence into a fix length vector, thereby the data cannot be interpreted by decoder accurately.

2. Using LSTM and GRU

3. When training with teacher force, since we are using the output token in the next step, a wrong output token could lead to total failure of the interpretation.
   This problem won't occur during training time since we are feeding the ground truth token, but at test time, since we don't have the ground truth token anymore, a wrong token will lead to significant error.

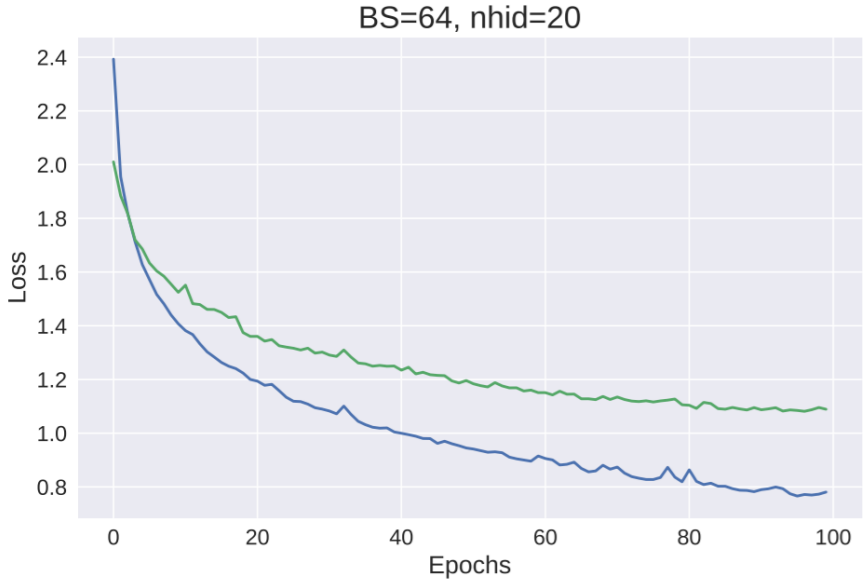
4. By feeding generated token instead of ground truth token can reduce the difference between training and testing, thereby enhance the performance of the network.

Part 3

1. As implemented in code

2. The result is not so good.

```
ethay airday ontioningmay inssay oknedpay
ethay airway ondishinguray isway olmisedway
ethay aircay ontirinedway issway orsktray
```



I ran the decoder section for three times and the result are above.
Accuracy of translated words:

The	Air	Conditioning	Is	Working
1	0.33	0	0.33	0

From the result we have, we can still conclude that the model can do well with short words, especially those short words begin with a consonant.

3.

```
source:           the air conditioning is working
translated: ethay airway onditingsray isway orkingway


source:           this model is not good
translated: isslay oderlay isway othay oodgay


source:           can i have some enjoyable cake and drink
translated: ancay iway avedray omedsay enonoundershay akentway andway inksay

source:           i am fixing this model
translated: iway amway ixingsray isslay oderlay
```

From these test sentence I have tried, I can conclude that the mode fails to translate long words and words begin with consonant letter. Only word (“I”, “am”, “are”, “good”) except words used in training are translated correctly.

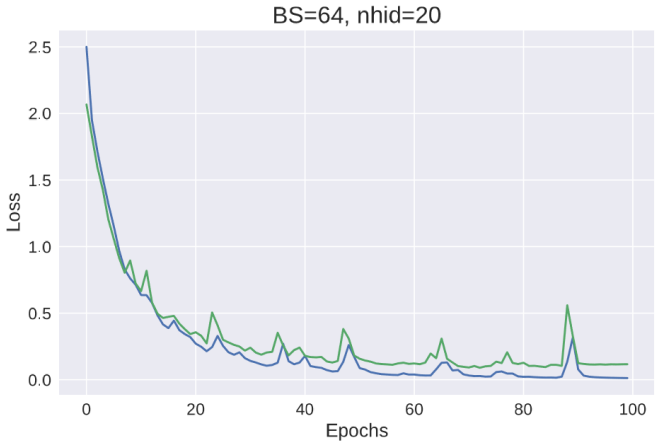
Part 4

1. In code

2. In code

3.
```
Epoch:  95 | Train loss: 0.016 | Val loss: 0.113 | Gen: etway airway
onditioningcay isway orkingway
Epoch:  96 | Train loss: 0.015 | Val loss: 0.116 | Gen: etway airway
onditioningcay isway orkingway
Epoch:  97 | Train loss: 0.014 | Val loss: 0.115 | Gen: etway airway
onditioningcay isway orkingway
Epoch:  98 | Train loss: 0.013 | Val loss: 0.116 | Gen: etway airway
onditioningcay isway orkingway
Epoch:  99 | Train loss: 0.012 | Val loss: 0.116 | Gen: etway airway
onditioningcay isway orkingway
source:       the air conditioning is working
translated:   etway airway onditioningcay isway orkingway

source:       can i have some enjoyable cake and drink
translated:   ancay iway avehay omesay enjoyableway akecay andway inkdray
```



The result from RNN attention decoder is much better than result from previous model. Most words in the sentence get translated correctly, and the model performs well on test sentence.
The training speed of one epoch is much slower then previous model. 6.82s for one epoch, compared with previous model that only need 2s for one epoch since more calculations are required when implement attention.

4.
```
Epoch:  97 | Train loss: 0.126 | Val loss: 0.355 | Gen: ethay airway
onditionengcay isway orkingway
Epoch:  98 | Train loss: 0.164 | Val loss: 0.282 | Gen: esthay airway
onditioningcay isway orkingway
Epoch:  99 | Train loss: 0.142 | Val loss: 0.319 | Gen: ethay airway
onditiongingcay isway orkingway
source:       the air conditioning is working
translated:   ethay airway onditiongingcay isway orkingway

source:       can i have some enjoyable cake and drink
translated:   ancay iway avehay omesay enjoymesleway akecay andway inkdray
```

The translation accuracy is lower than previous one's. The word conditioning is translated incorrectly. In my customized test sentence, the word enjoyable is also translated incorrectly, which can be implied that long words have a higher chance to be translated inaccurately.
I get 7.22s for one epoch when train with scaled dot, which is a little slower than training with additive attention. Theoretically scaled dot attention should compute faster then additive attention since it uses matrix operation which increases computation efficiency, but we get opposite result since we are running the model on a small dataset.

Part 5

1.

Advantage: The performance is better without scaling for larger values d_k of the formula for attention $\text{softmax}\left(\left(\frac{QK^T}{\sqrt{d_k}}\right)V\right)$

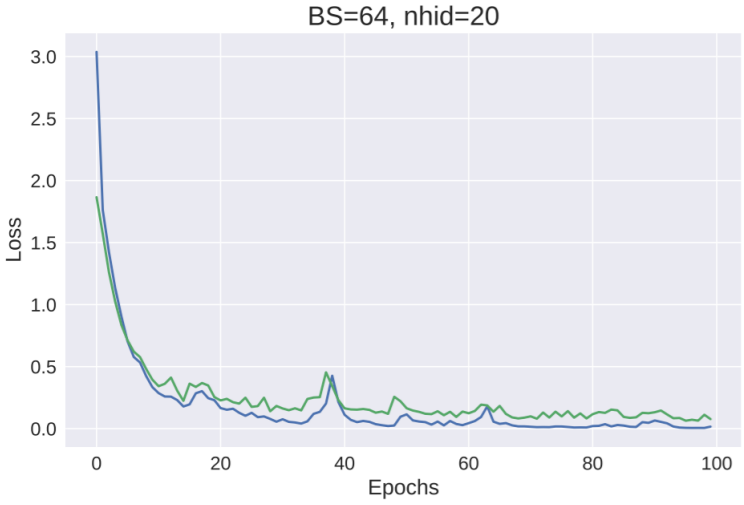
Disadvantage: Using additive attention is slower and it uses more space.

2.

In code

3.

```
Epoch:  97 | Train loss: 0.006 | Val loss: 0.064 | Gen: ethay iiway
onditiningcay isway orkingway
Epoch:  98 | Train loss: 0.006 | Val loss: 0.112 | Gen: ethay airray
onditiningcay isway orkingway
Epoch:  99 | Train loss: 0.016 | Val loss: 0.077 | Gen: ethay aiiwaaa
onditiningcay isway orkingway
source:          the air conditioning is working
translated: ethay aiiwaaa onditiningcay isway orkingway
```



The translation accuracy is higher compare to the previous decoders. (Still one word is translated incorrectly though)
Running time for one epoch is approximately 3.41s, which is faster than attention decoder.

4.

```
Epoch:  97 | Train loss: 0.309 | Val loss: 0.369 | Gen: ethay ay ongcay isway
orkingway
Epoch:  98 | Train loss: 0.317 | Val loss: 0.373 | Gen: ethay
airwairwairwairwairw onitay isway orkingway
Epoch:  99 | Train loss: 0.306 | Val loss: 0.359 | Gen: ethay ay oninay isway
orkingway
source:          the air conditioning is working
translated: ethay ay oninay isway orkingway
```

The translation result is worse compare with previous models. Some words lose letters during translation. This also happen on test sentence

```
source:          can i have some enjoyable cake and drink
translated: ay iway ay - enjoy akecakecakecakecakec ay inkdraay
```

The result without using causal is bad since scaled dot product attention cannot handle the complex input. The causal attention uses a mask to make the contexts easier to use in calculation, which is not implemented in the normal dot product attention.

5.

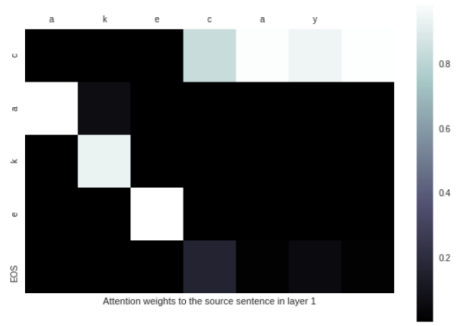
Because translation of pig latin and simple sentences are simple, thereby the network can memorize the ordering and produce the correct answer. In addition, self attentions, encoder attentions and residual connections can also preserve the positional information from the context so that positional encoding is not required to make our decoder work.

Part 6

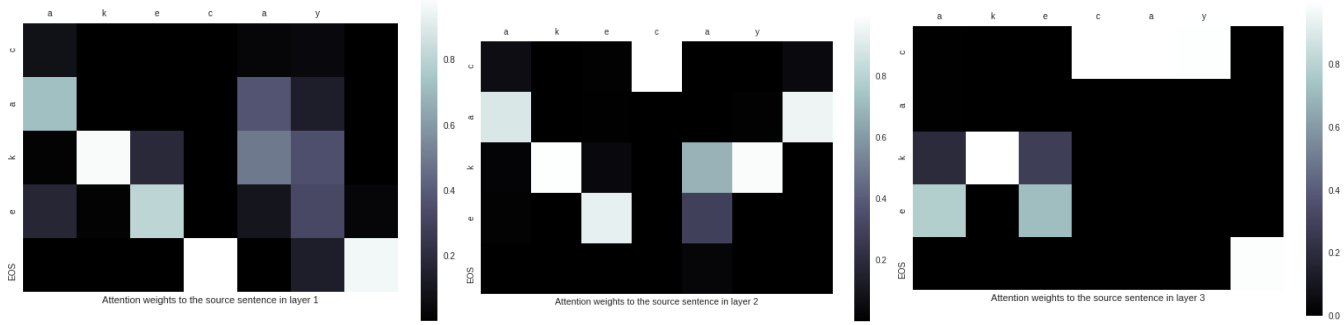
1.

For word cake:

RNN:



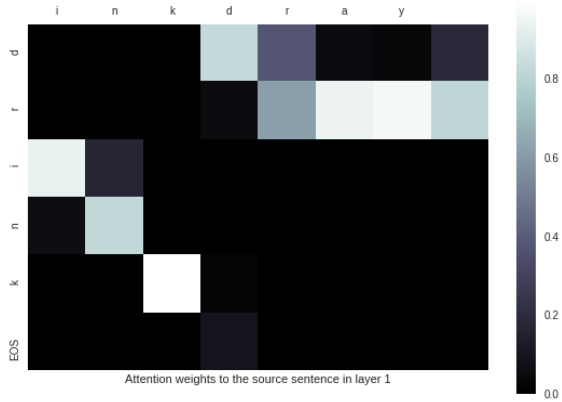
Transformer:



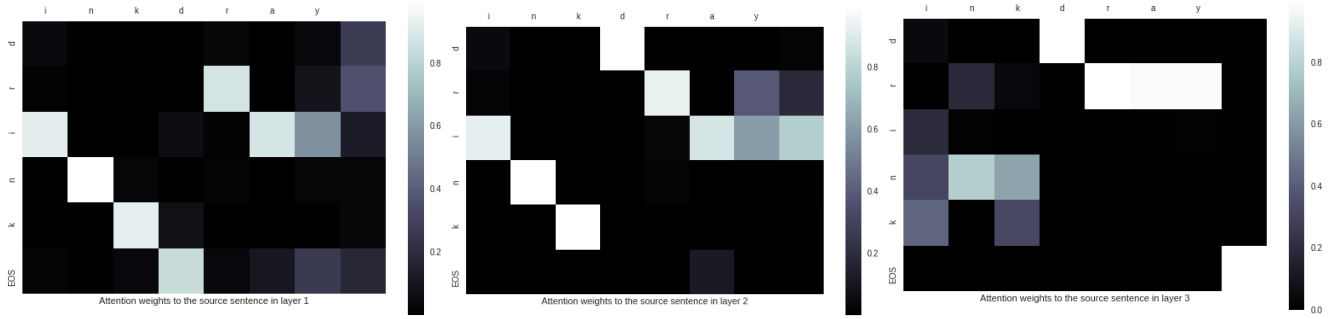
Word cake is translated successfully.

For word drink

RNN:



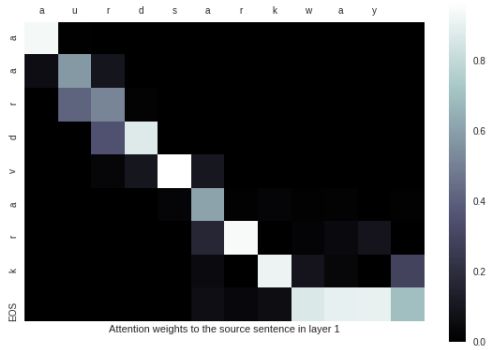
Transformer:



The word drink is translated successfully.

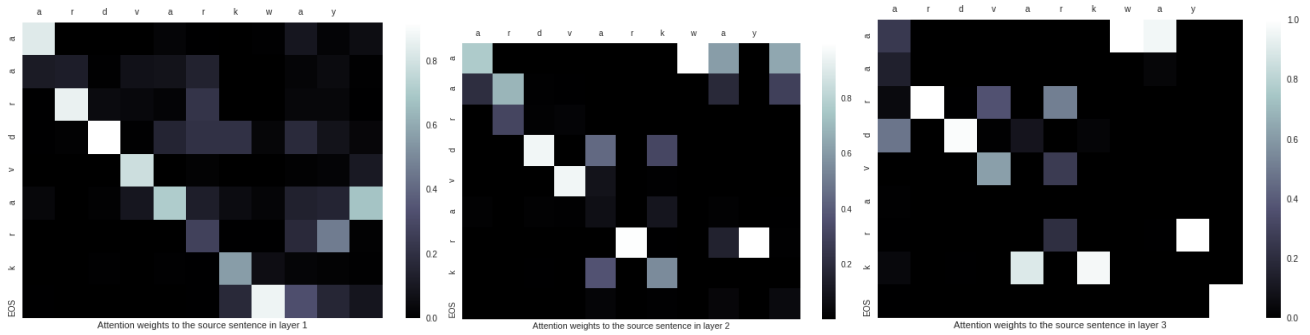
Word aardvark

RNN:



The word is translated into aurdsarkway, which is wrong.

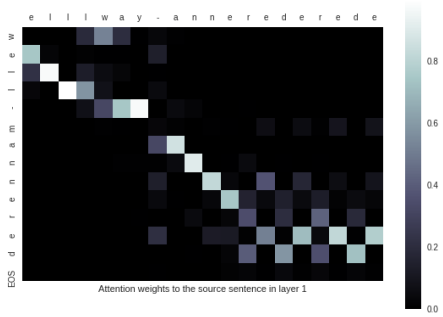
Transformer:



The word is translated into ardvarkway, which is wrong.

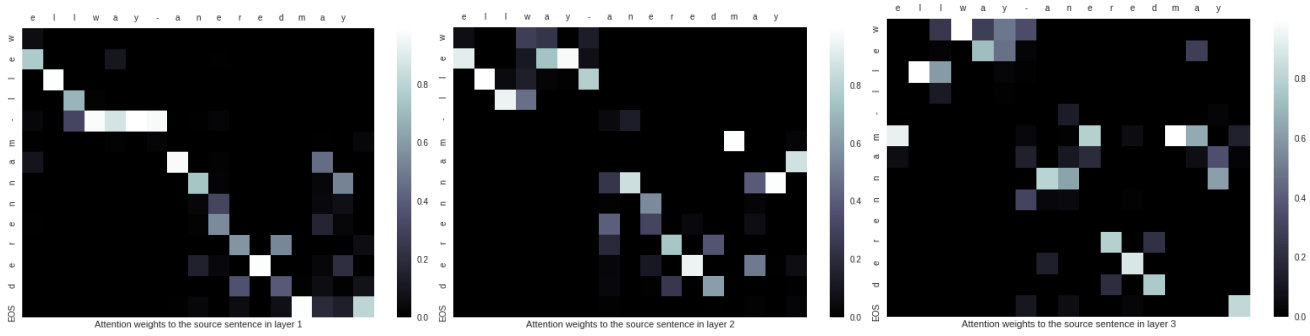
Word well-mannered

RNN:



The word is translated into elllway-annerederede, which is wrong.

Transformer:



The word is translated into ellway-aneredmay, which is correct.

From these 4 test cases, we can see that those two correct case all have several diagonal lines that translate the word in the correct order.
In these two wrong cases, in word aardvark, the RNN attention model translate second letter a into u for some reason, the transformer attention model pass second letter a in the translation process in layer1. When translate word well-mannered, the dash caused confusion in RNN attention model, which makes attention after dash low, and there is some bug in recursion process that repeat rede.
Therefore, we can conclude that long words that include – or begin with several vowel is hard to be translated correctly using our models. In general, short words are more likely to be translated accurately, while complicated words have lower translation accuracy.