Report for assignment 5 by Linsen Li

1. Description of data set

This task is to build a machine translation model to translate English to Germany. We download the data set from the website and store it in a txt file. We have chosen the number of 30000 English-to-Germany pairs from the txt file. The following figure shows some examples of the data set (the left is English and the right is Germany):

```
# Show these pair of data
for i in range(3000, 3010):
    print('[' + clean_pairs[i, 0] + '] => [' + clean_pairs[i, 1] + ']')
```

```
[tom is spirited] => [tom ist temperamentvoll]
[tom is stalking me] => [tom stalkt mich]
[you broke your leg] => [du hast dir das bein gebrochen]
[its a good school] => [das ist eine gute schule]
[tom will continue] => [tom wird weitermachen]
[the signal was red] => [das signal war rot]
[tom is marys son] => [tom ist der sohn marias]
[what did you learn] => [was haben sie gelernt]
[tom is undressing] => [tom entkleidet sich]
[youve upset tom] => [du hast tom verargert]
```

Figure1: 10 examples of the English-to-Germany pairs

Then we split the data into training set, validation set and test set:

```
# Split the data into train,val,test
np.random.shuffle(clean_pairs)
training, test = clean_pairs[:27000,:], clean_pairs[27000:,:]
np.random.shuffle(training)
train, val = training[:25000,:], training[25000:,:]
print('The shape of training set is' + str(train.shape))
print('The shape of validation set is' + str(val.shape))
print('The shape of test set is' + str(test.shape))
```

```
The shape of training set is(25000, 2)
The shape of validation set is(2000, 2)
The shape of test set is(3000, 2)
```

Figure2: The shape of training set, validation set and test set

2. Description of the model

2.1 Seq2seq models

For the Seq2seq models, we build encoder and decoder separately. Then we connect these two parts and formulate our Seq2seq models.

2.1.1 Seq2seq models with LSTM

We use loss function as 'categorical_crossentropy' with the default learning rate 0.001. We choose optimizer as 'rmsprop'. The following is the summary and structure of the model:

```
Model: "model_training"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
encoder_input_x (InputLayer)    (None, None, 28)     0
__________________________________________________________________________________________________
decoder_input_x (InputLayer)    (None, None, 30)     0
__________________________________________________________________________________________________
encoder (Model)                 [(None, 512), (None, 583680      encoder_input_x[0][0]
__________________________________________________________________________________________________
decoder_lstm (LSTM)             [(None, None, 512),  1112064     decoder_input_x[0][0]
                                                                 encoder[1][0]
                                                                 encoder[1][1]
__________________________________________________________________________________________________
decoder_dense (Dense)           (None, None, 30)     15390       decoder_lstm[1][0]
==================================================================================================
Total params: 1,711,134
Trainable params: 1,711,134
Non-trainable params: 0
__________________________________________________________________________________________________
```

Figure3: The summary of the Seq2seq model with LSTM

```python
# Show the structure of the nerual network
from keras.utils.vis_utils import plot_model
plot_model(model, show_shapes=True, to_file='model_LSTM.png')
```

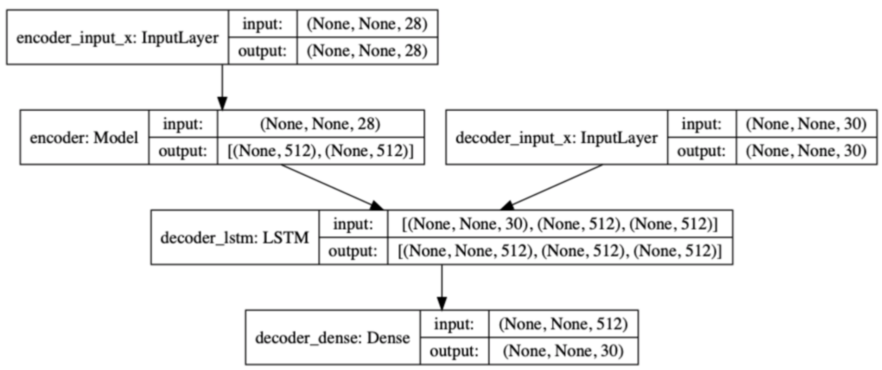

Figure4: The structure of the Seq2seq model with LSTM

We train the model for 100 epochs, and get the following loss plot:

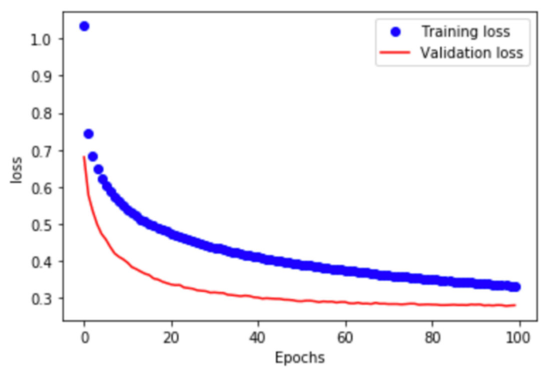

Figure5: The loss plot of the Seq2seq model with LSTM

We test the BLEU score on the validation set, and get the BLEU score:

```python
# Compute the BLEU on validation set
blue_score(input_val_texts, target_val_texts)
```

```
Average BLEU score 0.1134
```

Figure6: The BLEU score of the Seq2seq model with LSTM on validation set

2.1.2 Seq2seq models with GRU

We use loss function as 'categorical_crossentropy' with the default learning rate 0.001. We choose optimizer as 'rmsprop'. The following is the summary and structure of the model:

```
Model: "model_1"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
Encoder_Input (InputLayer)      (None, None, 28)     0
__________________________________________________________________________________________________
Decoder_Input (InputLayer)      (None, None, 30)     0
__________________________________________________________________________________________________
Encoder_GRU (GRU)               [(None, 256), (None, 218880      Encoder_Input[0][0]
__________________________________________________________________________________________________
Decoder_GRU (GRU)               [(None, None, 256),  220416      Decoder_Input[0][0]
                                                                 Encoder_GRU[0][1]
__________________________________________________________________________________________________
DecoderOutput (Dense)           (None, None, 30)     7710        Decoder_GRU[0][0]
==================================================================================================
Total params: 447,006
Trainable params: 447,006
Non-trainable params: 0
__________________________________________________________________________________________________
```

Figure6: The summary of the Seq2seq model with GRU

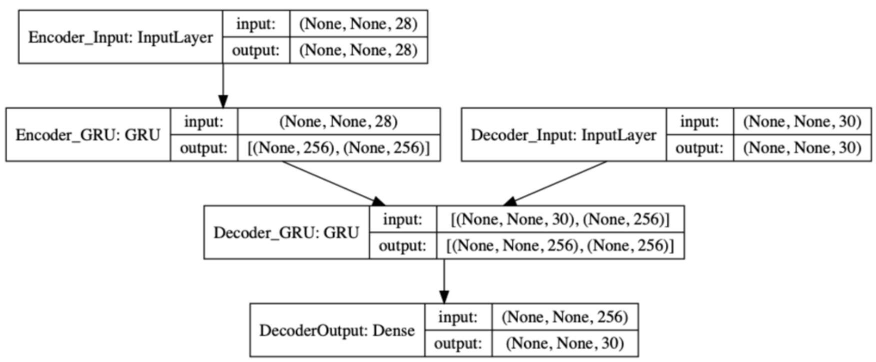

Figure7: The structure of the Seq2seq model with GRU

We train the model for 50 epochs, and get the following loss plot:

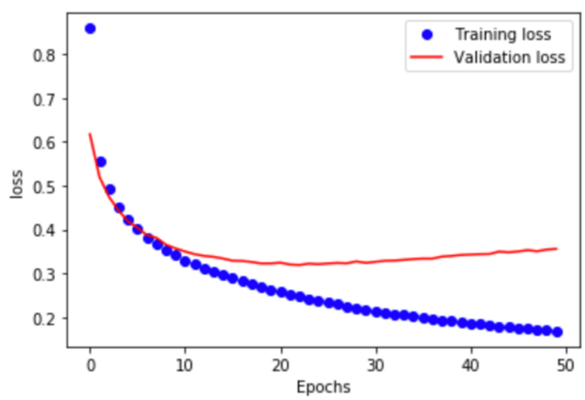

Figure8: The loss plot of the Seq2seq model with GRU

We test the BLEU score on the validation set, and get the BLEU score:

```
# Compute the BLEU on validation set
blue_score(input_val_texts, target_val_texts)

Average BLEU score 0.1024
```

Figure9: The BLEU score of the Seq2seq model with GRU on validation set

2.1.3 Seq2seq models with SimpleRNN

We use loss function as 'categorical_crossentropy' with the default learning rate 0.001. We choose optimizer as 'rmsprop'. The following is the summary and structure of the model:

```
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
encoder_input_x (InputLayer)    (None, None, 28)     0
__________________________________________________________________________________________________
decoder_input_x (InputLayer)    (None, None, 30)     0
__________________________________________________________________________________________________
encoder (Model)                 [(None, 256), (None, 72960       encoder_input_x[0][0]
__________________________________________________________________________________________________
decoder_RNN (LSTM)              [(None, None, 256),  293888      decoder_input_x[0][0]
                                                                 encoder[1][0]
                                                                 encoder[1][1]
__________________________________________________________________________________________________
decoder_dense (Dense)           (None, None, 30)     7710        decoder_RNN[1][0]
==================================================================================================
Total params: 374,558
Trainable params: 374,558
Non-trainable params: 0
__________________________________________________________________________________________________
```

Figure10: The summary of the Seq2seq model with SimpleRNN

```
# Show the structure of the nerual network
from keras.utils.vis_utils import plot_model
plot_model(model_SimpleRNN, show_shapes=True, to_file='model_SimpleRNN.png')
```

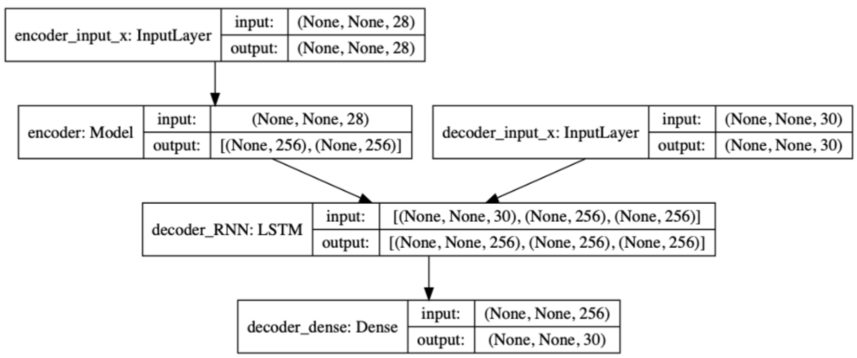

Figure11: The structure of the Seq2seq model with SimpleRNN

We train the model for 50 epochs, and get the following loss plot:

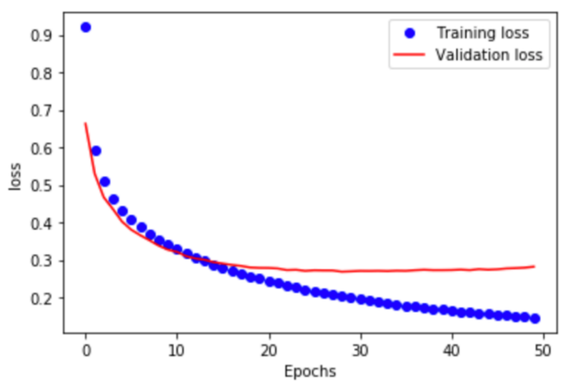

Figure12: The loss plot of the Seq2seq model with SimpleRNN

We test the BLEU score on the validation set, and get the BLEU score:

```
# Compute the BLEU on validation set
blue_score(input_val_texts, target_val_texts)

Average BLEU score 0.0877
```

Figure13: The BLEU score of the Seq2seq model with SimpleRNN on validation set

2.1.4 Seq2seq models conclusion

We have tried Seq2seq model with different RNN cells: LSTM, GRU and simple RNN. We find that LSTM performs the best on BLEU score.

2.2 Seq2seq models with attention

We use loss function as 'categorical_crossentropy' with the default learning rate 0.001. We choose optimizer as 'rmsprop'. The following is the summary and structure of the model:

```
Model: "model_3"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_4 (InputLayer)            (None, None, 30)     0
__________________________________________________________________________________________________
input_3 (InputLayer)            (None, None, 28)     0
__________________________________________________________________________________________________
lstm_7 (LSTM)                   (None, None, 512)    1112064     input_4[0][0]
__________________________________________________________________________________________________
bidirectional_11 (Bidirectional [(None, None, 512),  583680      input_3[0][0]
__________________________________________________________________________________________________
dropout_7 (Dropout)             (None, None, 512)    0           lstm_7[0][0]
__________________________________________________________________________________________________
dropout_6 (Dropout)             (None, None, 512)    0           bidirectional_11[0][0]
__________________________________________________________________________________________________
dot_3 (Dot)                     (None, None, None)   0           dropout_7[0][0]
                                                                 dropout_6[0][0]
__________________________________________________________________________________________________
activation_2 (Activation)       (None, None, None)   0           dot_3[0][0]
__________________________________________________________________________________________________
dot_4 (Dot)                     (None, None, 512)    0           activation_2[0][0]
                                                                 dropout_6[0][0]
__________________________________________________________________________________________________
concatenate_16 (Concatenate)    (None, None, 1024)   0           dot_4[0][0]
                                                                 dropout_7[0][0]
__________________________________________________________________________________________________
dense_2 (Dense)                 (None, None, 30)     30750       concatenate_16[0][0]
==================================================================================================
Total params: 1,726,494
Trainable params: 1,726,494
Non-trainable params: 0
__________________________________________________________________________________________________
```

Figure14: The summary of the Seq2seq model with attention

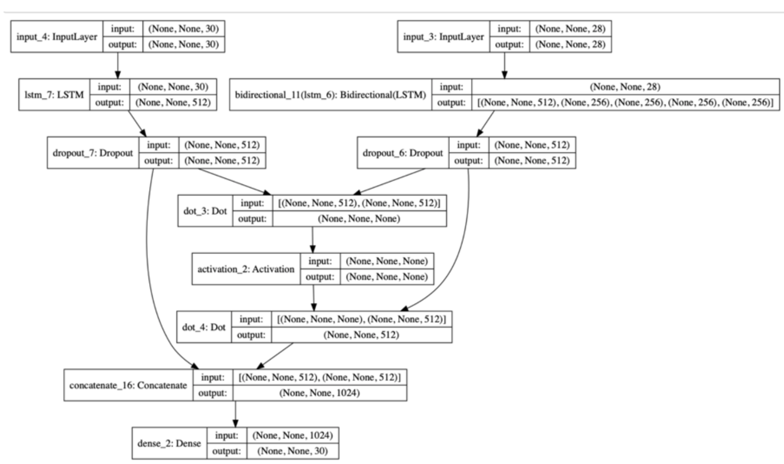

Figure15: The structure of the Seq2seq model with attention

We train the model for 50 epochs, and get the following loss plot:

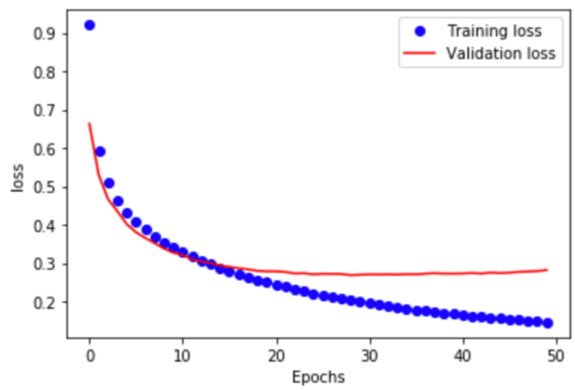

Figure16: The loss plot of the Seq2seq model with attention

3. Compare model on test set

Since previously we have found out that LSTM performs the best on BLEU score on validation set. So, we choose LSTM as our best Seq2seq model. In order to improve the Seq2seq model, we change it to Bi-LSTM. Note that we can only use Bi-direction LSTM for the encoder part. At last we get our best Seq2seq model. We use this to compare with the attention model on test set, the result is following:

(e1)Compute the BLEU scores of the best seq2seq model from step (c)

```
]: blue_score(input_test_texts, target_test_texts)
Average BLEU score 0.1124
```

(e2)Compute the BLEU scores of the attention model from step (d) ¶

```
]: blue_score_attention(input_test_texts, target_test_texts)
Average BLEU score 0.1434
```

Figure17: The BLEU on test set

The result show that the Seq2seq with attention performs better than normal Seq2seq.

4. Show some translation examples

```
----------------------------------------
English:        tom is easygoing
Germany (true):  tom ist gelassen
Germany (pred):  tom ist langweilig
Germany (pred_attention): tom ist ein geter nann
----------------------------------------
English:        i hate my eyebrows
Germany (true):  ich hasse meine augenbrauen
Germany (pred):  ich hasse meinen rocher
Germany (pred_attention): ich habe einen hand auf
----------------------------------------
English:        what did i just do
Germany (true):  was habe ich gerade getan
Germany (pred):  was habe ich heraucht getan
Germany (pred_attention): was hat tom gesagt
----------------------------------------
English:        im nervous
Germany (true):  ich bin nervos
Germany (pred):  ich bin ungerecht
Germany (pred_attention): ich bin nicht auf
----------------------------------------
English:        its worth a try
Germany (true):  es ist einen versuch wert
Germany (pred):  es ist eine groe arbeit
Germany (pred_attention): es ist ein geter hand
----------------------------------------
English:        come back here
Germany (true):  kommt wieder her
Germany (pred):  komm schon schon hier
Germany (pred_attention): komm her sich rein
----------------------------------------
English:        tom was kidding
Germany (true):  tom machte witze
Germany (pred):  tom hat geschummelt
Germany (pred_attention): tom war nicht getangen
----------------------------------------
```

```
----------------------------------------
English:        its urgent
Germany (true):  es ist dringlich
Germany (pred):  es ist freundlich
Germany (pred_attention): es ist ein gelentig
----------------------------------------
English:        she hates carrots
Germany (true):  sie hasst karotten
Germany (pred):  sie hasst karotten
Germany (pred_attention): sie hat einen hand geschafft
----------------------------------------
English:        can i see this one
Germany (true):  kann ich mir dieses hier ansehen
Germany (pred):  kann ich diesen stift benutzen
Germany (pred_attention): kann ich mit dir helfen
----------------------------------------
English:        keep it short
Germany (true):  fasse dich kurz
Germany (pred):  behalten sie es auf
Germany (pred_attention): behalte dich
----------------------------------------
English:        once isnt enough
Germany (true):  einmal reicht nicht
Germany (pred):  einmal ist nicht geizg
Germany (pred_attention): setz die euch auf
----------------------------------------
English:        youre very good
Germany (true):  ihr seid echt gut
Germany (pred):  du bist sehr gut
Germany (pred_attention): du bist ein gescheiden
----------------------------------------
English:        toms angry
Germany (true):  tom ist wutend
Germany (pred):  tom ist gerangert
Germany (pred_attention): tom ist geschaftigt
----------------------------------------
```

Figure18: Some translation examples