Reply Generation

Communication through mails remains a poignant form of networking in any company. It always gives a sense of formal communication which adds value to the conversation happening. Hundreds of mail get exchanged in any company on a daily basis be it internal or external part of the company. Our objective is to create a reply generation model that will aid the users who are trying to reply to a mail by suggesting them suitable responses.

1. Data

The Enron email dataset contains approximately 500,000 emails generated by employees of the Enron Corporation. It was obtained by the Federal Energy Regulatory Commission during its investigation of Enron's collapse. This is the May 7, 2015 Version of dataset, as published at https://www.cs.cmu.edu/~./enron/

https://www.kaggle.com/shashichander009/inshorts-news-data

2. Data Wrangling

The data was available to us in the standard email format as shown below.

	file	message
0	allen-p/_sent_mail/10.	Message-ID: <15464986.1075855378456.JavaMail.e...
1	allen-p/_sent_mail/1000.	Message-ID: <13505866.1075863688222.JavaMail.e...
2	allen-p/_sent_mail/1002.	Message-ID: <30965995.1075863688265.JavaMail.e...
3	allen-p/_sent_mail/1004.	Message-ID: <17189699.1075863688308.JavaMail.e...
4	allen-p/_sent_mail/102.	Message-ID: <30795301.1075855687494.JavaMail.e...

The file column can give us details such as the sender and receiver of the mail. While the message column consists of the body of the mail, the subject, the date

and time of sending the mails. Our objective is to retrieve all the reply mails that were being sent. In order to do that we firstly worked to get the body of the mail as shown below.

	file	message	subject	body
0	allen-p/_sent_mail/10.	Message-ID: <15464986.1075855378456.JavaMail.e...	Re:	Traveling to have a business meeting takes the...
1	allen-p/_sent_mail/1000.	Message-ID: <13505866.1075863688222.JavaMail.e...		Randy,\n\n Can you send me a schedule of the s...
2	allen-p/_sent_mail/1002.	Message-ID: <30965995.1075863688265.JavaMail.e...	Re: Hello	Greg,\n\n How about either next Tuesday or Thu...
3	allen-p/_sent_mail/1004.	Message-ID: <17189699.1075863688308.JavaMail.e...	Re: PRC review - phone calls	any morning between 10 and 11:30
4	allen-p/_sent_mail/102.	Message-ID: <30795301.1075855687494.JavaMail.e...	FW: fixed forward or other Collar floor gas pr...	---------- Forwarded by Phillip K ...

Then we performed the data cleaning steps as follows:

1.) Remove the initial auto-generated 'Re:' from the body.
2.) Removing special characters using regular expressions.
3.) Tokenize the body into sentences
4.) Lowercasing the text.
5.) Removing unnecessary white spaces.

After cleaning the data we got the reply mails as shown below.

	Reply mail
0	traveling to have a business meeting takes the...
1	greg how about either next tuesday or thursday...
2	any morning between and
3	paula million is fine phillip
4	i think fletch has a good cpa i am still doin...

3. Pre-processing and modelling

. In our approach to solve this problem we wish to predict the next tokens based on the initial 20 tokens of the reply mail. For this we made two lists called ‘needed list’ and ‘rested list’. We will use the needed list to make the vocabulary and do all the pre-processing steps. The rested list will help us in the future to cross-check our results based on the actual next tokens.

Next we split the data-set (needed list) into training and testing set with 6000 samples for training and 1851 samples for testing purpose.

Out of these training samples we made a vocabulary of unique words. The vocabulary size amounted to 7284 unique words. Based on this vocabulary we transformed the testing data into categorical features such that we have the number of unique words as the number of categories, in our case 7284.

Then we made the Keras Sequential model summary of which is below. We used softmax function for output probabilities accompanied with relu activation.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (None, 19, 1)             7284
_________________________________________________________________
lstm (LSTM)                  (None, 19, 100)           40800
_________________________________________________________________
lstm_1 (LSTM)                (None, 100)               80400
_________________________________________________________________
dense (Dense)                (None, 100)               10100
_________________________________________________________________
dense_1 (Dense)              (None, 7284)              735684
=================================================================
Total params: 874,268
Trainable params: 874,268
Non-trainable params: 0
```

We then compiled the model to minimize the cross-entropy loss with an adam optimizer. We chose ‘accuracy’ as the performance metric for our problem. The model was then trained for 200 epochs with batches of size 256.

The training can be seen as below.

```
Epoch 186/200
24/24 [==============================] - 0s 13ms/step - loss: 1.5027 - accuracy: 0.6416
Epoch 187/200
24/24 [==============================] - 0s 13ms/step - loss: 1.5196 - accuracy: 0.6354
Epoch 188/200
24/24 [==============================] - 0s 13ms/step - loss: 1.4846 - accuracy: 0.6402
Epoch 189/200
24/24 [==============================] - 0s 13ms/step - loss: 1.5115 - accuracy: 0.6296
Epoch 190/200
24/24 [==============================] - 0s 12ms/step - loss: 1.5055 - accuracy: 0.6407
Epoch 191/200
24/24 [==============================] - 0s 13ms/step - loss: 1.4948 - accuracy: 0.6373
Epoch 192/200
24/24 [==============================] - 0s 13ms/step - loss: 1.4237 - accuracy: 0.6664
Epoch 193/200
24/24 [==============================] - 0s 12ms/step - loss: 1.4491 - accuracy: 0.6491
Epoch 194/200
24/24 [==============================] - 0s 13ms/step - loss: 1.4486 - accuracy: 0.6499
Epoch 195/200
24/24 [==============================] - 0s 13ms/step - loss: 1.4205 - accuracy: 0.6536
Epoch 196/200
24/24 [==============================] - 0s 14ms/step - loss: 1.4149 - accuracy: 0.6670
Epoch 197/200
24/24 [==============================] - 0s 14ms/step - loss: 1.3796 - accuracy: 0.6671
Epoch 198/200
24/24 [==============================] - 0s 14ms/step - loss: 1.4152 - accuracy: 0.6592
Epoch 199/200
24/24 [==============================] - 0s 13ms/step - loss: 1.3880 - accuracy: 0.6707
Epoch 200/200
24/24 [==============================] - 0s 13ms/step - loss: 1.4134 - accuracy: 0.6590
```

4. Predictions:

We made a customized function which will give us the most probable sequence by predicting the unique word present in the vocabulary.

Below were some of the tests that we did.

Test-1

```
X_test2[321]
```

```
'greetings since i get everything that you forward you may remove me from you distribution list hope all is well'
```

And our model generated this-

```
'we a fried of will'
```

Test-2

```
X_test2[213]
```

```
'steve peace is watching his political career vanish before his very eyes couldn t happen to a nicer guy rightly'
```

And our model generated this-

```
inary classification   (e.g. if it uses a `sigmoid` la
  warnings.warn('`model.predict_classes()` is deprecat
'statement either months hour impacted'
```

As can be seen from the predicted text, the results are quite average. The generated text seems random and does not seem to convey the fluency of written language and seems to be clearly generated by a computer program.

5. Future Improvements

Given that we have to solve the same problem, there are multiple things that can be done in future:

(i) The data cleaning process could be enhanced where the vocabulary of the text contains only proper English words and not short-forms or non-english words.

(ii) Transformer based models can be used instead of LSTM based model where the self-attention property can be utilized in remembering larger sentences.

(iii) Since we could use only around 8000 samples out of 50,000 ones, the model performance could be enhanced with utilization of more data.

6. Credits

I would like to thank my Springboard mentor Mr. Himanshu Jain for being super supportive and guiding me to solve this problem.