NLP Programming Assignment 3

Diwakar Mahajan – dm3084

Run the script file for running the code for all 3 questions and obtaining the results.
To Run:
sh run.sh

The script generates the following:

- Extracted corpus: corpus.de, corpus.en
- For Q4, devwords_top10.txt & alignment_model1.txt
- For Q5, alignment_model2.txt
- tvalModel1.pkl – model storing t parameters generated from IBM Model 1
- qvalModel.pkl - model storing q parameters generated from IBM Model 2

Q4. IBM_Model1.py implements the IBM Model 1 EM Algorithm.

The program accepts gz files of the two parallel corpora of English and Foreign language and produces t values after 5 iterations. t values are generated only for the possible word combinations that are found in the corpora. t values are saved as tvalModel1.pkl to be used later.

The program first extracts the files and then proceeds with training the model. The program also has inbuilt functions for generating the top words for English words (function 'getTopWords') and get alignments (function 'align').

Results of devwords can be found in file devwords_Top10.txt

Results of alignments can be found in the file aligment_model1.txt
They match with the given sample 100%

Q5. IBM_Model2.py implements the IBM Model 2 EM algorithm.

The program accepts gz files of the two parallel corpora of English and Foreign language along with the t values as generated by IBM Model 1. The program then evaluates q values of all the alignments in the corpora. q values are saved as qvalModel.pkl to be saved later.

Results of alignments can be found in the file aligment_model2.txt
They match with the given sample 100%

Comparison:
The two models produce alignments, which improve with the IBM Model 2. This can easily be attributed to introduction of q parameters and also initialization of t parameters with the values derived from IBM Model 1 rather than using 0 as initialization.

Example:

English: ‘resumption of the session’
German: ‘wiederaufnahme der sitzungsperiode’

We get the following alignments:
Model 1: [1, 2, 4]
Model 2: [1, 3, 4]
For the Model 2 the alignment improves and German word *der* now aligns *the* rather than *of,* which is an improvement. This can be attributed to the reasons discussed before.

Q6. The task is to find the best suitable sentence translation of a German sentence in a scrambled file of English sentences. This task can be easily achieved using the model generated in the previous question (i.e. t parameters and q parameters).

However, we encounter the problems of unseen words and unseen alignments. For such cases we use log probability to handle zeros probabilities and also we replace log(0) (which is negative infinity) with a large negative value.

We obtain accuracy of 93% by running the given script.