ASSIGNMENT – I

(N18140956 – İsmail Talha YILMAZ)

Assignment contains 3 parts, ngram modeling, linear interpolation and discounting.

Before starting all the parts, data cleaning is done. All of the sentences made lowercased. All of the non-word and non-number characters are eliminated. Finally, all of the words in sentences are arranged so that they have only 1 character spaces each.

In the first part, ngram models are generated. In a seperate function “generate_ngrams”, ngrams are generated. This function takes order of n-gram ‘n’ as an input and generates desired n-gram list of sentence as an output.

For each n-gram model, MLE is calculated.

For unigrams, MLE is calculated by counting occurences of unigram and it is divided by all number of words in corpus. The formula is presented below.

$$q_{ML} = \frac{\text{Count}(w_i)}{\text{Count}()}$$

For bigrams, MLE is calculated by counting occurences of bigram and it is divided by the count of occurence of previous word unigram. The formula is presented below.

$$q_{ML}(w_i \mid w_{i-1}) = \frac{\text{Count}(w_{i-1}, w_i)}{\text{Count}(w_{i-1})}$$

For trigrams, MLE is calculated by counting occurences of trigram and it is divided by the count of occurence of previous 2-word bigram. The formula is presented below.

$$q_{ML}(w_i \mid w_{i-2}, w_{i-1}) = \frac{\text{Count}(w_{i-2}, w_{i-1}, w_i)}{\text{Count}(w_{i-2}, w_{i-1})}$$

This probability calculations are done in prob method. This method returns the probability result of sentence.

For unigram model, Laplace smoothing is implemented. The difference of Laplace smoothing comes from it adds 1 value to count of unigram. Also in division, total number of unique numbers are also considered. Below is the formula of Laplace smoothing.

$$P_{+1}(w_i) = \frac{c_i + 1}{N + V}$$

In the formula, c states the count of unigram, N states for total word number and V stands for vocabulary size. It means the unique word numbers in the corpus. The result is the Linear interpolation probability.

Because of the lack of time, I couldn't implement Good-turing algorithm.

In the next step, perplexity calculation is done. Lower perplexity is better for model. Below formula shows how perplexity calculation is done.

$$P = 2^{-l}$$

$$l = \frac{1}{M} \sum \log_2 \{p(s_i)\}$$

Firstly l value is calculated. This calculated adding log probability of the model divided by the number of ords in dataset. In order to calculate perplexity, 2 to the power –l is calculated. Adding 2 operations together, result can be obtained by multiplying all probabilies and taking the Mth order of derivative. Perplexity calculation is done this way. For all models, perplexity can be calculated in the hw.

Linear interpolation is also implemented for bigrams and trigrams. Formula is presented below.

- Take our estimate $q(w_i \mid w_{i-2}, w_{i-1})$ to be:

$$= \lambda_1 \times q_{ML}(w_i \mid w_{i-2}, w_{i-1})$$
$$+\lambda_2 \times q_{ML}(w_i \mid w_{i-1})$$
$$+\lambda_3 \times q_{ML}(w_i)$$

There are 2 rules. Firstly, all lambda values should be bigger than 0. Secondly, all lambdas total sum must be equal to 1. For bigrams, I set lamdas to 0.5 and fot trigrams I set all lambdas to 0.33. As a result, interpolation becomes the weighted average of unigrams bigrams and trigrams.