NLP Assignment 2
Report

Harman Singh 2019042
Yash Bhargava 2019289

—---

Q1.

Preprocessing steps:

- Lowercasing
- Substitution of certain words (E.g., "aren't": "are not")
- Tokenization
- Removing punctuations ('''!()-[|]`{};:'"\,<>./?@#$=+%^&*_~''')
- Removing URLs, usernames
- Stripping extra white spaces

Bigram LM Model creation:

- Created a 2-D dictionary representing the co-occurrence matrix of bigram counts.
- Created a dictionary representing the unigram counts
- Laplace smoothing:
  Calculated the bigram probabilities using

$$P(w_i \mid w_{i-1}) = \frac{count(w_{i-1}w_i) + 1}{count(w_{i-1}) + V}$$

where V represents the vocabulary size.

Formulating \beta:

$$\text{Prob}(w_i \mid w_{i-1}) = (\text{count}(w_{i-1}\, w_i) / \text{count}(w_{i-1})) + \beta$$

- Approach 1:
  - \beta = k*|vader_sentiment_score (w_i-1 +" "+w_i) for sentence generation|

- Approach 2:
  - \beta = k*positive vader_sentiment_score (w_i-1 +" "+w_i) for positive sentence generation
  - \beta = k*negative vader_sentiment_score (w_i-1 +" "+w_i) for negative sentence generation

  where k is of the order 10^-4

Sentence Generation:

- First word : chosen randomly from vocabulary
- Subsequent words:
    - prediction using bigram score of the previous word and the current word
    - Approach 1: choosing the word with highest score
    - Approach 2: choosing a word from top k highest score words

Sentiment score using Vader:

- Used Vader sentiment analysis (using python library) to score the generated sentences. Assumption: neutral predictions are labeled as positive.

Q2.

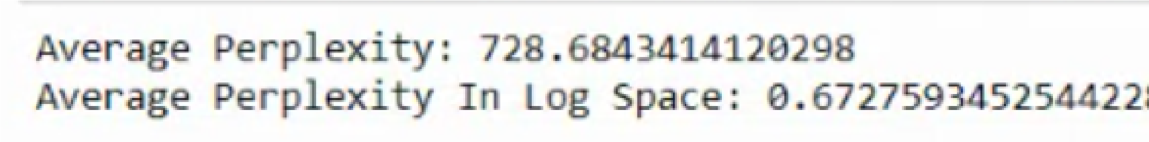
1. Intrinsic Evaluation
    a) `def calculate_sentence_perplexity_in_log_space`
    `def calculate_sentence_perplexity`
    The above two functions have been written from scratch to calculate the perplexity of a sentence in log space and normally. Perplex

    $$\mathrm{PP}(W) = \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P(w_i|w_{i-1})}}$$

    In log space PP(W) is calculated as following:

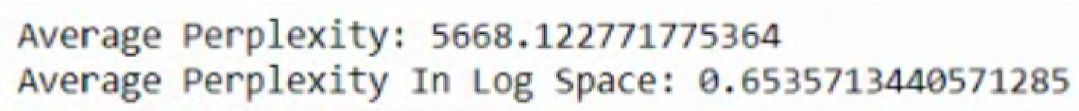
    $$\log\left[\Pi_{i=1}^{n} \quad P(w_i\,|\,w_{i-1}))\right] = \sum_{i=1}^{n} \quad \log\left[P(w_i\,|\,w_{i-1}))\right]$$

    b) Average perplexity of 500 sentiment oriented sentences:
    - Approach 1:



```
Average Perplexity: 728.6843414120298
Average Perplexity In Log Space: 0.672759345254422
```

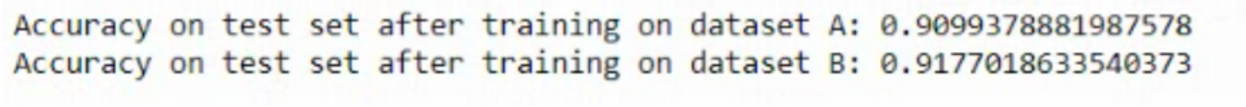
    - Approach 2:



```
Average Perplexity: 5668.122771775364
Average Perplexity In Log Space: 0.6535713440571285
```

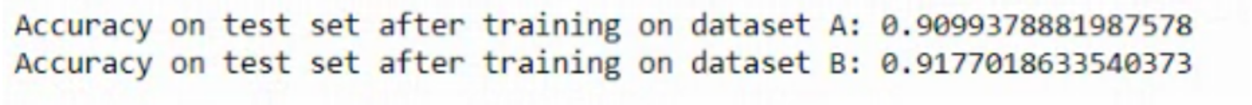
2. Extrinsic Evaluation

a) Approach 1: After training ML model on dataset A, acc_A on test set is and after training ML model on dataset B, acc_B on test set is



```
Accuracy on test set after training on dataset A: 0.9099378881987578
Accuracy on test set after training on dataset B: 0.9177018633540373
```

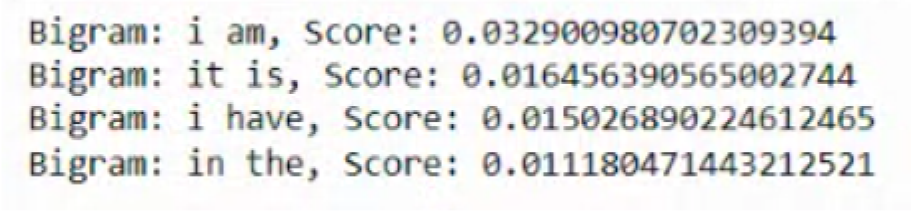
b) Approach 2: After training ML model on dataset A, acc_A on test set is and after training ML model on dataset B, acc_B on test set is



```
Accuracy on test set after training on dataset A: 0.9099378881987578
Accuracy on test set after training on dataset B: 0.9177018633540373
```

Desired Outcomes:

- Part A
  - a) Saved at : Link (2-D dictionary)
  - b) Top-4 bigrams and their score (before beta):



```
Bigram: i am, Score: 0.03290098070230939
Bigram: it is, Score: 0.016456390565002744
Bigram: i have, Score: 0.015026890224612465
Bigram: in the, Score: 0.011180471443212521
```

  - c) Accuracy on the test set for dataset A is : **90.99%**

- Part B
  - a) Method: Approach 1 and approach 2
  - b) Saved CSVs file in zip file
  - c) Average Perplexity of generated 500 sentences: Mentioned above
  - d)
    - Approach 1:
      - 5 positive samples:
        - rachelengland wonder what a good morning all the best thing i am not have
        - tehy toured with my friends in the best thing i am
        - politician on the best thing i am have
        - ribs with my friends in the best thing i am not have a
        - mondayyyyyyy yay my friends in the best thing i am not have a good
      - 5 negative samples:
        - hangover sucks i am not have a good

        - arrived crap i am not have a good morning all the
        - dan less features but i am not have a good
        - officially lost her i am not have a good morning
        - sickies i am not have a good morning

- □ Approach 2:
    - 5 positive samples:
        - razzle freedom vip excited favorites vip loved promoting loved favorites honestly promoting excited
        - achieved paradise favorites promoting promoting promoting loved favorites loved
        - twits love the same love to go back to
        - rounders glorious sunny honestly loved honestly honestly honestly
        - pouring ily favorites favorites honestly favorites excited i will be happy
    - 5 negative samples:
        - mission kill me up my friends raping pressure fails
        - bubble hell i do you can you have the day with no
        - boiling killed i am going to the day
        - office tragedy 7 hates jerk pressure hates hates jerk hates 7 pressure
        - redic fucked i was not have not a great weekend

e) Accuracy on test set for dataset B is : Mentioned above