TASK 3 - Development History

Task 2 Timeline

Date: 21/03/2023

Contribution:

Group member 1:

- Read the instruction for Task 2 and started this part.

Step 1: Data Import:

- Each excel file contains multiple worksheets. Data are positioned differently in each worksheet.
- Combined all data together and removed any duplicates to perform the next step.

Step 2: Text Extraction and Cleaning:

- Extracted the 'textOriginal' fields in all top-level comments.
- Removed emojis and normalized the text into lowercase for further analysis.
- Removed emojis using utf-8 format and normalized text.
- Extracted the vocab and countvec lists for English comments from Channels that have at least 15 English comments using the langdetect library with DetectorFactory.seed = 0. Note: Deciding the language on a comment level, not a sentence level.

Date: 25/03/2023

Contribution:

Group member 1:

- Generate csv file (create func to generate csv and detect the language) ● Generate a csv file that contains unique channel ids along with the counts of top level comments(all language, and english).
    - start coding for step 4 start by tokenizing the text data using the regular expression "[a-zA-Z]+" to extract words containing only alphabetic characters. Next, I remove both context-independent and context-dependent stopwords from the vocabulary. For context-independent stopwords, I utilize a provided list (stopwords_en.txt). For context-dependent stopwords, I set a threshold based on words that appear in more than 99% of

channel IDs with at least 15 English comments. After tokenization and stopword removal, I stem the tokens using the Porter stemmer to reduce them to their root forms. Then, I handle rare tokens by removing those that appear in less than 1% of channel IDs with at least 15 English comments. Additionally, tokens with a length less than 3 are removed from the vocabulary. To include meaningful bigrams, I use the Pointwise Mutual Information (PMI) measure to identify the first 200 collocations that frequently occur within the same comment. Finally, I calculate the vocabulary containing both unigrams and bigrams to be used for further analysis and modeling.

Date: 28/03/2023

Contribution:

Group member 2:

- Fix detect language func because it didn't follow assignment's specification.
- Fix code for remove context dependent

Date: 29/03/2023

Contribution:

Group member 2:

Date: 30/03/2023

Contribution:

Group member 2:

- Fix code for remove rare token
- Try to reorder step 4

Date: 31/03/2023

Contribution:

Group member 2:

- Try to reorder step 4

Date: 1/04/2023

Contribution:

Group member 2:

- Separate code for remove rare token and stopwords

- Try to reorder step 4

Date: 3/04/2023

Contribution:

- Separate code for remove rare token and stopwords
- Try to reorder step 4

Group member 2:

Date: 9/04/2023

Contribution:

- Merge code for remove rare token and common token
- Try to reorder step 4

Group member 1:

Date: 10/04/2023

Contribution:

Group member 1:

- Try to count bigram and unigram in dataframe (step 5)

Date: 12/04/2023

Contribution:

Group member 1:

- Fix a bug in extracting channelId and textOriginal
- Regenerate channel_list.csv
- Add Testing script

Date: 13/04/2023

Contribution:

Group member 1 & 2:

- Check step 1-5 code
- Fix code that filter at least 15 english comment
- Generate countvec.txt

Date: 16/04/2023

Contribution:

Group member 1:

- Fix step in generate bigram and unigram
- Fix generate bigram using PMI 200

Date: 17/04/2023

Contribution:

Group member 2:

- Found a bug in Bigram Func. After generate bigram, we should not remove it from dataframe.

Date: 18/04/2023

Contribution:

Group Member 1& 2:

- Reorder step 4
- Fix generate bigram (I forgot to put validate bigram within same comment)
- Merge bigram using _ in dataframe
- Write justification for generate bigram and unigram

Date: 19/04/2023

Contribution:

Group Member 1& 2:

- Final Testing & Checking
- Add final documentation
- Prepare for Submission

Google Colab Workbook Link

Task 1 link text

Task 2 link text