CSE508 Information Retrieval
Winter 2024
Assignment - 4

Due Date: April 20, 2024 ; 23:59 **Max. Marks:** 100

Instructions:

1. The assignment is to be attempted individually.
2. The proposed solutions will be evaluated during your code demo and viva.
3. Institute plagiarism policy will be strictly followed.
4. The programming language allowed is Python.
5. Ensure that your code is thoroughly documented for clarity and understanding.
6. You can utilize libraries such as NLTK and BeautifulSoup for data preprocessing in your Python code.
7. You are required to use version control via GitHub:
    a. Make a GitHub repository with the name:
    CSE508_Winter2024_A4_<Roll_No.>.
    b. Add your assignment TA as a contributor. The TA assigned (along with their GitHub handle) to you for this assignment will be released to the classroom.
8. You must make a detailed report with the name **CSE508_Winter2024_A4_<Roll_No>_Report.pdf** with a brief overview of your approach, methodologies, assumptions, and results for each problem.
9. Steps for submission:
    a. A zipped folder **CSE508_Winter2024_A4_<Roll_No.>** consisting of all your code files, dumped files and **Report.pdf**
    b. A text file **CSE508_Winter2024_A4_<Roll_No.>.txt** consisting of the link to your GitHub repository.
10. If it has been mentioned to code a solution from scratch, using any library is strictly not allowed.

Review Summarization using GPT2 [100 Marks]

1. Use the Amazon Fine Food Reviews dataset
2. Clean and preprocess the 'Text' and 'Summary' column from the dataset.

Model Training

1. Initialize a GPT-2 tokenizer and model from Hugging Face.
2. Divide the dataset into training and testing (75:25)
3. Implement a custom dataset class to prepare the data for training.
4. Fine-tune the GPT-2 model on the review dataset to generate summaries.
5. Experiment with different hyperparameters such as learning rate, batch size, and number of epochs to optimize the model's performance.

Evaluation

After training, compute ROUGE scores on the test set to assess the model's overall performance i.e. compute ROUGE score for every predicted summary vs the actual summary.

For Example

Given Review Text: "The Fender CD-60S Dreadnought Acoustic Guitar is a great instrument for beginners. It has a solid construction, produces a rich sound, and feels comfortable to play. However, some users have reported issues with the tuning stability."
Given Summary: "Good for beginners but has tuning stability issues."

Generated Summary: "The Fender CD-60S Acoustic Guitar is suitable for beginners, but there are reported tuning stability issues."

Rouge Scores

ROUGE-1: Precision: 0.75, Recall: 0.80, F1-Score: 0.77
ROUGE-2: Precision: 0.50, Recall: 0.67, F1-Score: 0.57
ROUGE-L: Precision: 0.67, Recall: 0.75, F1-Score: 0.71
(These are random values)

Relevant links

https://www.kaggle.com/code/changyeop/how-to-fine-tune-gpt-2-for-beginners
https://www.youtube.com/watch?v=nsdCRVuprDY
https://youtu.be/CDmPBsZ09wg?t=2558

NOTE: If you're facing computational issues in fine-tuning the model on the entire corpus, kindly use a smaller random sample of the dataset. You can choose the sample size according to whatever doesn't cause computational issues but do keep in mind that it should be decently large for the model to train properly.