Assignment 4

CS 421: Natural Language Processing

Due: April 23, 2025 (11:59 p.m. CST)

1 Introduction

Welcome to Assignment 4 (Bonus)! In this assignment, you will explore the application of transformer models in the task of text summarization. You will load a preprocessed dataset, utilize pre-trained models from Hugging Face's Transformers library, and evaluate the performance of the model using the summarization pipeline. Specifically, you will use three core Hugging Face libraries: transformers for model and pipeline setup, datasets to load the CNN/DailyMail dataset, and evaluate computing performance metrics.

The goal is to get hands-on experience with large language models in a practical setting and compare different summarization outputs using qualitative and quantitative methods.

You will complete the code and run your code on two pre-trained models of your choice. Then compare the performance of the two models on a small set of data using both automatic metric (ROUGE-1) and human evaluation.

Don't hesitate to reach out on Piazza or during office hours with any questions you have as you complete it! Happy Coding!

2 Instructions

Each part of this deliverable is labeled as Code or Written. The guidelines for these two types of components are provided below.

Note on Bonus points: The assignment will be graded out of 45 points. The 45 points will be scaled to contribute a maximum of **3 points** towards your final course grade.

2.1 Code

The questions need to be completed using Python (version 3.10+). There are **no external packages** required to complete this assignment. If you want to use an external package for any reason, you are required to get approval from the course staff on Piazza prior to submission. Templates are provided for each Code question (.ipynb files) as supplementary material. Do not modify the structure of the starter notebook unless instructed. All necessary code should

be added in the designated sections of the notebook. These templates may also contain important information and/or examples in comments so please read them carefully. This part of the assignment will be graded manually using Gradescope.

To submit your solution for Code questions, you need to submit the following files:

☐ `Assignment_4.ipynb`

Submit this `zip` file on Gradescope under `Assignment 4`. All specified files need to be submitted to receive full credit.

2.2 Written

You are required to submit all Written questions in a single PDF file. You may create this PDF using Microsoft Word, scans of your handwritten solution, LaTeX or any other method or design tool you prefer. To submit your solution for Written questions, you need to provide answers to the following questions in a single PDF.

☐ Q5

Before submission, ensure that all pages of your solution are present and in order. Submit this PDF on Gradescope under `Assignment 4`.

3 Questions

3.1 Code (30 points)

Q1 (5): Load the Dataset
Q1 (5): Load the Dataset Use the Hugging Face datasets library to load *CNN Daily News* Summarization dataset[1] for summarization.
Supplementary material: `translation.ipynb`

Q2 (10): Summarization with a Pre-trained Model
Set up a Hugging Face summarization pipeline[2] using a two pre-trained model `t5-small`[3], and `distillbart-cnn`[4]. Use the pipeline() utility from the transformers library to construct a functional summarization pipeline.
Supplementary material: `translation.ipynb`

[1] `https://huggingface.co/datasets/abisee/cnn_dailymail`
[2] `https://huggingface.co/docs/transformers/en/main_classes/pipelines#transformers.SummarizationPipeline`
[3] `https://huggingface.co/sshleifer/distilbart-cnn-12-6`
[4] `https://huggingface.co/google-t5/t5-small`

Q3 (10): Summary Generation
Utilize the pipeline to generate summaries for 20 news articles, from test split of the dataset. Use your summarization pipeline to generate the summaries and store both the original articles and their corresponding summaries in Python lists. These lists will be used later for performance evaluation.
Supplementary material: `translation.ipynb`

Q4 (5): Evaluation and Output Storage
Evaluate the generated summaries using the ROUGE metric, focusing specifically on the ROUGE-1 F1-score as the evaluation metric. Use the Hugging Face evaluate library to load and compute the average ROUGE-1 score[5].
Supplementary material: `translation.ipynb`

3.2 Written (15 points)

Q2 (15): Qualitative Evaluation of Summaries
Select 5 news articles from your generations and compare the summaries generated by both models for each article. For each article, you will therefore have two summaries (one from each model). Assess each pair of summaries using the following Likert Scale questions (1 to 5):

1. Is the summary fluent and grammatically, correct?
2. Does it retain the key points of the original text?
3. Is the summary factually consistent with the original text (i.e., does it remain faithful to the source content)?
4. How would you rate the overall quality of the summary based on the above three assessments?

Following the evaluation, answer the following reflection question:

5. Based on the ROUGE-1 scores and your manual evaluation, which evaluation method provides a better indicator of summary quality, and why?

You should present your results clearly in your written report (PDF), for example using tables or bullet points, and provide a brief explanation for each of your ratings.

[5] `https://huggingface.co/spaces/evaluate-metric/rouge`

4 Rubric

This assignment will be graded according to the rubric below. Partial points may be awarded for rubric items at the discretion of the course staff.

Q1 (5 points possible)
+5
Q2 (10 points possible)
+10
Q3 (10 points possible)
+10
Q4 (5 points possible)
+5
Q5 (15 points possible)
+15