CSE508 Information Retrieval
Winter 2024
Assignment-2

Due Date: Mar 6, 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_A2_<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 on the classroom.
8. You must make a detailed report with the name **CSE508_Winter2024_A2_<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_A2_<Roll_No.>** consisting of all your code files, dumped files and **Report.pdf**
    b. A text file **CSE508_Winter2024_A2_<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.

Refer to the dataset linked here for the assignment - Dataset **.** The dataset consists of links to images and corresponding text reviews for a given Product ID. Fetch the images from the given URLs and follow the instructions below.

Question - Perform the tasks specified below to make a Multimodal Retrieval System using Text as well as Images as the Input Data [100 Marks]

1. Image Feature Extraction [25]
    a. Use basic image pre-processing techniques as altering contrast, resizing, geometrical orientation, random flips, brightness and exposure or any other relevant operation.
    b. Use a pre-trained Convolutional Neural Network Architecture as ResNet, VGG16, Inception-v3, MobileNet (or any other CNN , preferably pre-trained on ImageNet Dataset), to extract relevant features from the images in the given training Set. Choose only one of the networks for your final pipeline.
    c. Normalize the extracted features.
2. Text Feature Extraction [25]
    a. Implement relevant pre-processing techniques as Lower-Casing, Tokenization, removing punctuations, Stop Word Removal, Stemming and Lemmatization on the given text reviews in the data
    b. Calculate the Term Frequency-Inverse Document Frequency (TF-IDF) scores for the textual reviews.

    Note: Please make sure to save your extracted features and the TF-IDF score using the pickle module so that you can run your code quickly in the demo
3. Image Retrieval and Text Retrieval [25]
    a. For the input (image, review) pair, find the most similar images (*preferably your top three*) to your input based on extracted image features/embeddings using a similarity measure (cosine similarity) and a suitable data-structure.
    b. For the input (image, review) pair, find the most similar reviews (*preferably your top three*) to your input review based on TF-IDF scores using a similarity measure (Cosine Similarity)
    c. Save your results using Python's **pickle** module to save and load your results.
4. Combined Retrieval (Text and Image)
    a. Get a composite similarity score (average) for the pairs generated in 3a) and 3b)
    b. Rank the pairs based on the composite similarity score.
5. Results and Analysis
    a. Present the top-ranked (image, review) pairs along with the cosine similarity scores.
    b. Observe which out of the two retrieval techniques gives a better similarity score and argue the reason.
    c. Discuss the challenges faced and potential improvements in the retrieval process.

Note: You are free to use any similarity metric. Just make sure it's consistent throughout the assignment and kindly mention it explicitly in your report.

6. Sample Test Case: [Please note that the output values are dummy values; The test case is given just to comprehend the format.]
    a. **Input:**

    Image and Text Query Input :

    Image:

    https://images-na.ssl-images-amazon.com/images/I/71bztfqdg+L._SY88.jpg

    Review: I have been using Fender locking tuners for about five years on various strats and teles. Definitely helps with tuning stability and way faster to restring if there is a break.

    b. Output:

    ——----------

    USING IMAGE RETRIEVAL

    1) ***Image URL:*** *[List of Image URLs]*
    Review: *Corresponding review*
    Cosine similarity of images - xx.xxxx
    Cosine similarity of text - xx.xxx

    2) ***Image URL:*** *[List of Image URLs]*
    Review: *Corresponding review*
    Cosine similarity of images - xx.xxxx
    Cosine similarity of text - xx.xxx

    3) ***Image URL:*** *[List of Image URLs]*
    Review: *Corresponding review*
    Cosine similarity of images - xx.xxxx
    Cosine similarity of text - xx.xxx

Composite similarity scores of images: xx.xxxx
Composite similarity scores of text: xx.xxxx
Final composite similarity score: xx.xxxx

—----------

USING TEXT RETRIEVAL

1) ***Image URL:*** *[Corresponding List of Image URLs]*
   Review: *Extracted Review*
   Cosine similarity of images - xx.xxxx
   Cosine similarity of text - xx.xxx

2) ***Image URL:*** *[Corresponding List of Image URLs]*
   Review: *Extracted Review*
   Cosine similarity of images - xx.xxxx
   Cosine similarity of text - xx.xxx

3) ***Image URL:*** *[Corresponding List of Image URLs]*
   Review: *Extracted Review*
   Cosine similarity of images - xx.xxxx
   Cosine similarity of text - xx.xxx

   Composite similarity scores of images: xx.xxxx
   Composite similarity scores of text: xx.xxxx
   Final composite similarity score: xx.xxxx

—----------

7. You must run your code during the demo.