Introduction to Web Search & Mining

Group Project

Final Report, Code Due: June 11th
Demo Due: June 9th

Introduction:

This is a group-based project. Each group should contain maximum 4 students. In this project, there are four options. Each one of you should choose a preferred option and find partners to form a group, and then register your name and student ID to the corresponding slot on our shared Tencent Doc.

WSM project groups:

Tencent Documentation:

https://docs.qq.com/sheet/DZW9VbUZHQ0RoTWVp?tab=BB08J2.

Each Option can only be chosen by 12 groups at most, so if a project idea is chosen by too many groups, it will be allocated on a first-come, first-serve basis. The deadline of project selection is April 16th.

Project A

In this project, you are asked to build a simple news search.

1. Dataset

All documents indexed by the search engine should come from the real-news-like folder of the C4 dataset (about 15G):
https://huggingface.co/datasets/allenai/c4/tree/main/realnewslike

2. Search system

A Query can be any item, person, event, or question related to one or several pieces of news. Your search engine should support the following two forms of search:

1) Boolean Search (25%): Users provide search keys and operations between keys. The system needs to return all the original documents. The query language must include operations such as AND, OR, NOT.
2) ranked Search (20%): Given a search query, the search system is supposed to return a ranked list of search results (origin documents). You need to consider factors like semantic relevance and freshness. To implement this, you may use any ranking method.

3. Search result processing

In addition to the original documents that were required to be returned in the previous section, we now want to return the result of processing or understanding those documents. A good search engine should also support some advance functions:

1) Multi-news summarization (15%): This is an advanced feature of Ranked search. Group news from the same event into one category and generate a summary. Different news events related to query should generate separate summaries. For each generated

summary, we must also know which original documents were generated for that summary.

2) QA (15%): For example, when you query "How old is Donald Trump?" A good search engine will return "76" as the first result for this question, and link this answer to the document collection that implies the answer. Note that although our corpus does not contain knowledge data such as wiki, we can still conduct QA part within the scope of news (eg. " How many casualties in xxx incident? ").

Note that this section is open, so you can implement either of the above two functions, or you can implement other functions related to the understanding/processing of search results as you wish. When you are demoing, state what you did, and the best overall result will get higher marks.

4. UI and report:

Implement GUI Interface for demo and project report (25%). The functionality of Section 3 is based on rank search, in other words, boolean search as a separate interface. These new features are supported only in rank search.

References:

1. https://github.com/Alex-Fabbri/Multi-News
2. https://arxiv.org/abs/2110.08499
3. https://arxiv.org/pdf/2112.07916v2.pdf
4. https://paperswithcode.com/task/multi-document-summarization
5. https://paperswithcode.com/task/question-answering

Project B:

The project aims to search images based on text queries. It is allowed to use pre-trained models to generate embeddings and fine-tune with additional datasets, but using existing pre-built text-image search tools is not permitted.

1. Dataset

The dataset contains 3000 images(about 3.4G). The image dataset can be downloaded from this URL:
https://jbox.sjtu.edu.cn/v/link/view/78b4522e29bb436aad57fb7b12f3cd19.

2. Search engine

1) high search accuracy (30%). The project requires a high search accuracy where the search engine should return images that match the input query. This can be achieved by using pre-trained models to generate embeddings for both images and text, and then using the similarity between these embeddings to retrieve relevant search results.
2) fast response time (10%). The search engine should also have a fast response time and should return search results in a short amount of time.
3) lightweight (20%). The search engine should be lightweight, achieved by reducing the model's parameters using knowledge distillation. The goal is to make the search engine run on a personal laptop, and extra points will be awarded if the search engine can run on a mobile phone.
4) scalability (20%). The search engine's scalability is essential which should be ensured by adding up to 500 new images(depending on your device) to the image database

on-site and testing whether text queries can retrieve the newly added images.

3. UI and report:

Implement GUI Interface for demo and project report (20%). The image search results should be displayed in a sorted order. Please design interface yourself and display a reasonable number of images for each search.

References:

1. Radford A, Kim J W, Hallacy C, et al. "Learning transferable visual models from natural language supervision." International conference on machine learning. PMLR, 2021.
2. Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: visual question answering. CoRR, abs/1505.00468.
3. Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. CoRR, abs/2010.11929.
4. Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2019. Tinybert: Distilling BERT for natural language understanding. CoRR, abs/1909.10351.
5. Ren S, Zhu K Q. Leaner and Faster: Two-Stage Model Compression for Lightweight Text-Image Retrieval. NAACL, 2022 arXiv:2204.13913.
6. Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network.
7. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Advances in Neural Information Processing Systems, volume 30. Curran Associates, Inc.