Vietnam National University, Ho Chi Minh City
University of Information Technology
Falcuty of Computer Science

Final Project Report

Content-based image retrieval on the ILSVRC2012 dataset

Instructor: Dr. Thanh Duc Ngo
Class: CS336.O11.KHTN

Student name	*Student ID*
Nguyen Tran Viet Anh	21520006
Huynh Dang Vinh Hien	21520029
Le Thanh Minh	21520063
Huynh Pham Duc Lam	21521050
Tran Hoang Bao Ly	21521109

Ho Chi Minh City, February 2024

Contents

Addition information about the project:

- Source code: https://github.com/4ursmile/Massive-image-text-retrieval-system
- Demonstration video: https://youtu.be/NrrZ9378q-Y

1 Introduction

Since the introduction of digital cameras and their seamless integration into everyday smartphones, the world has witnessed an exponential increase amount of images captured daily. This avalanche, fueled by readily available internet technology and affordable digital sensors, has led to the creation of vast and diverse image databases across various applications. However, navigating and effectively retrieving specific images from this ever-growing sea of visual data presents a significant challenge.

To address this, the field of Content-Based Image Retrieval (CBIR) has gained immense traction, aiming to bridge the gap between the low-level visual features extracted from images and the high-level interpretations and intentions that resonate with human perception.

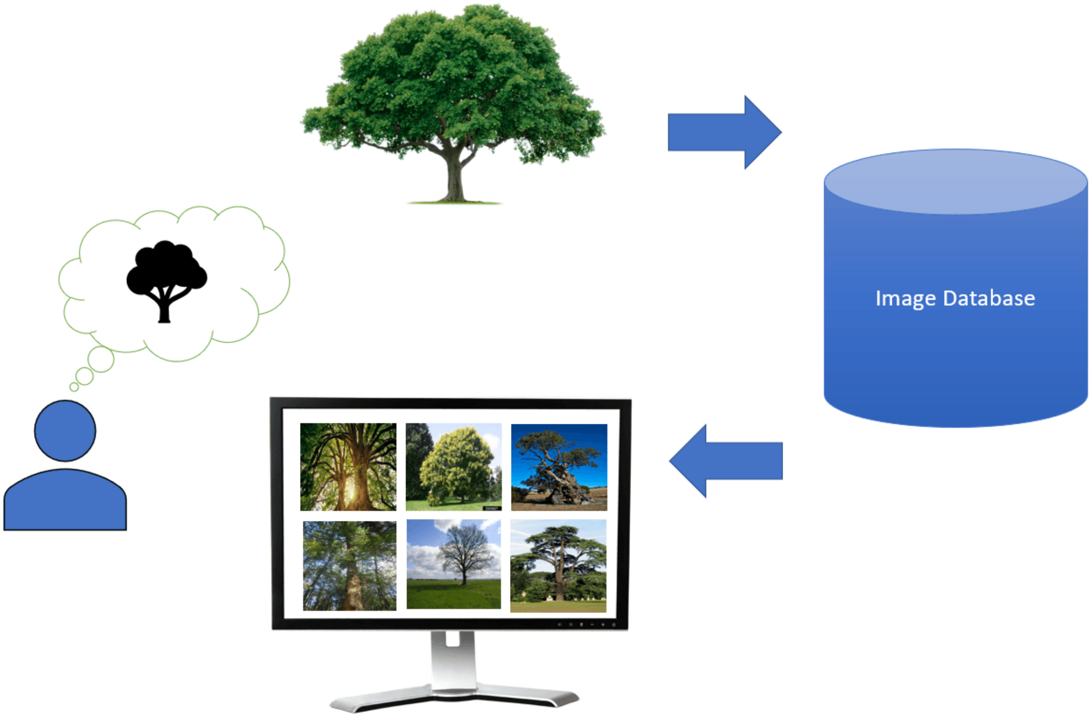

Figure 1: Image retrieval with image as a query

This work delves into simple CBIR on ImageNET Large Scale Visual Recognition Challenge 2012 (ILSVRC2012) [1] dataset and deploys onto a web application to grasp the hands-on experience making a retrieval system. We utilized CLIP [2] to generate embeddings for our image collections, and with the help of FAISS [3] we can easily return ranking list with high performance on both Query By Example (image query) and Semantic Retrieval (text query). On the composed image retrieval task, because of its sheer complexity, we opted to use pre-trained models such as BLIP [4] and Large Language Model to generate our final query.

2 Methodology

2.1 Contrastive Language-Image Pre-Training (CLIP)

Contrastive Language-Image Pre-Training, or CLIP, is a machine learning model published by OpenAI [2], based on ResNet [5] and Transformer [6] architecture. In the CLIP model, the input is represented as pairs of text and image, which will be encoded simultaneously by different modules:

- The text encoder uses Transformer [6] with byte-pair encoding (BPE).
- The image encoder either uses ResNet [5] and replaces the average pooling by an attention pooling mechanism, or VisionTransformer.

Then, the output of these encoders is merged into a single model to maximize the relevance between correct pairs of matching. Formally, suppose we have n pairs of elements $\{(t_1, p_1), (t_2, p_2), \ldots, (t_n, p_n)\}$ after encoding process, where t_i and p_i are the text and image representation of the i-th input. Let $s(t_i, p_j)$ mean the similarity between t_i and p_j. Hence, the model's goal is to maximize $s(t_i, p_j)$ where $i = j$, and minimize $s(t_i, p_j)$ otherwise.

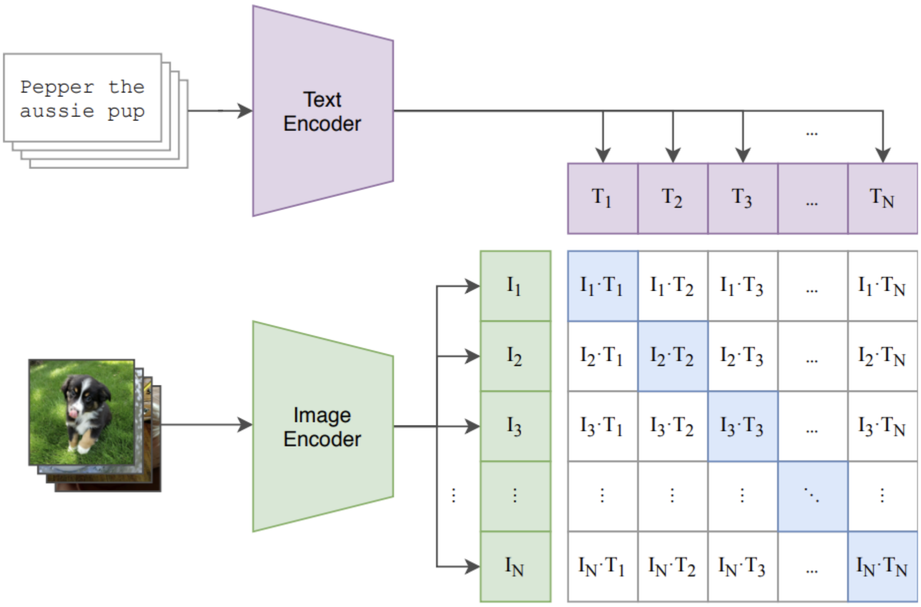

Figure 2: An overview of CLIP model architecture

In the concept of an information retrieval system, the CLIP model may be useful because of its organization of pre-trained data. For example, in the case of a given text T, where the information needed is to find the most relevant image from the corpus, we can simply select the image p_i that maximizes $s(T, p_i)$. The process is vice versa if an image P is provided instead.

2.2 Facebook AI Similarity Search (FAISS)

Searching for similar instances in a model is usually faced as a bottleneck, where sequential search or other traditional methods may not satisfy the performance requirements. Published in 2017, Facebook AI Similarity Search (FAISS) [3] introduced a searching technique that leverages the parallelization advantages of GPUs.

FAISS is built is solve the task that for each query $x \in \mathbb{R}^d$, we want to find the most k elements that have the highest similarity with x in a given collection. It is difficult to explain the approach of FAISS clearly. However, we can summarize it by some key features:

- The queries are processed *in batch*, not sequentially for each query.
- The distance/similarity equation is separated into parts, and the independent parts are pre-computed for further uses.
- The searching procedure is done via *heap* data structure. However, instead of using a single one, several *GPU heaps* are constructed in order to acquire the parallelization benefit.
- Sorting is done *in-register* to avoid any hardware delays.

2.3 BLIP

Solving the problem in the case of composed data requires a module to merge them into a single query. We decided to use BLIP [4], a language-image pre-training model based on bootstrapping to generate the right caption for a given image. First of all, BLIP is trained with a large quantity of composed data to learn the caption meaning of a wide range of images. Then, when a photo is given, CLIP uses its previously learned information to generate a caption for the photo. This approach takes the idea from NLP pre-trained language model, which has been shown significantly effective.

2.4 FastAPI

FastAPI (homepage) is a popular web framework developed by Snowflake Inc. FastAPI, implemented in Python, boasts an array of functions and APIs facilitating seamless input and output interactions with users. Additionally, its incorporation of various data visualization methods enhances user engagement and interaction. This framework's capabilities make it an optimal solution for professionals in the realms of machine learning and data science, aligning with the project's evolution towards web-based accessibility.

3 System overview

3.1 Image-based Image Retrieval

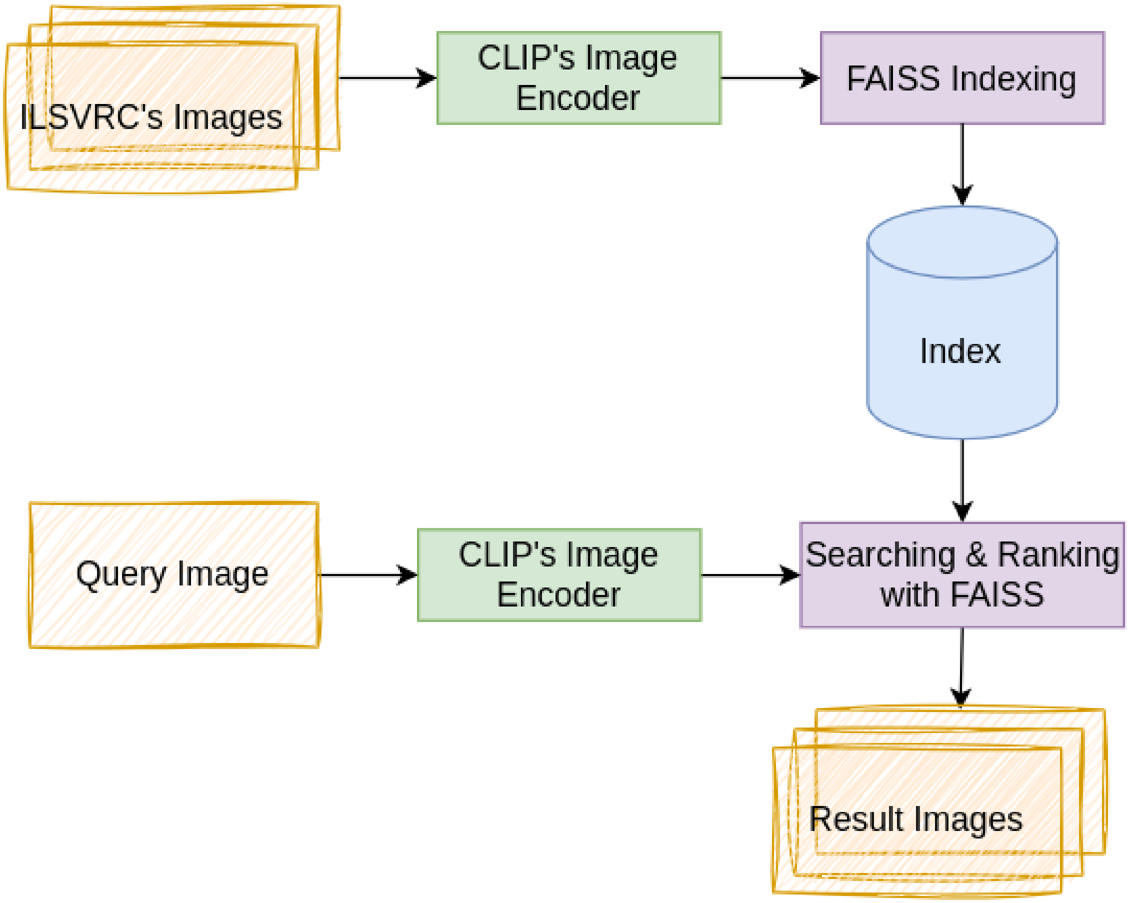

Figure 3: Model pipeline of the image-based system

Our model incorporates two main components: CLIP, an image encoder, and FAISS, an indexing algorithm.

First, CLIP takes an image as input and generates an embedding, which is a high-dimensional vector that captures the image's semantic content. Then, FAISS is used to create an index of these embeddings. This index allows for efficient retrieval of images from the database that are similar to a given query image.

When a user submits a query image, CLIP generates an embedding for it, and FAISS searches the index for images with similar embeddings, using cosine similarity. The most similar images are then returned to the user.

3.2 Text-based Image Retrieval

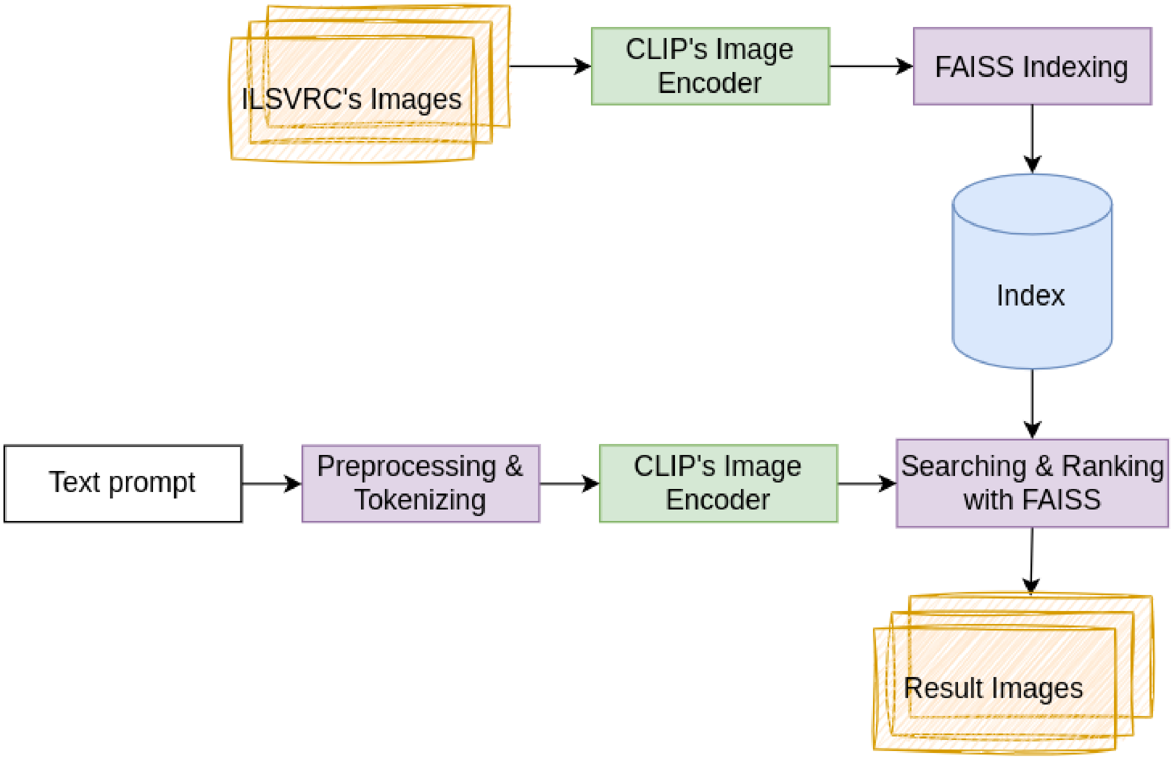

Figure 4: Model pipeline of the text-based system

Our text-based image retrieval model follows a similar structure to the image-based one, with the key difference lying in the query processing. When a user inputs text, we prepend it with the prompt `The photo of`. Tokenized text and prompt are then fed into CLIP's image encoder to generate an embedding, which is subsequently used for FAISS similarity search.

3.3 Composed Image Retrieval

Similar to the previous models, the composed image retrieval model differentiates from these models in how it processes the query. Instead of relying solely on text input, it first feeds the input image into BLIP to extract a textual caption. This image caption is then merged with the text query and both are processed by GPT3.5-Turbo [7] to generate a better text prompt. Finally, the resulting combined representation is used by CLIP to generate an embedding vector for retrieval.

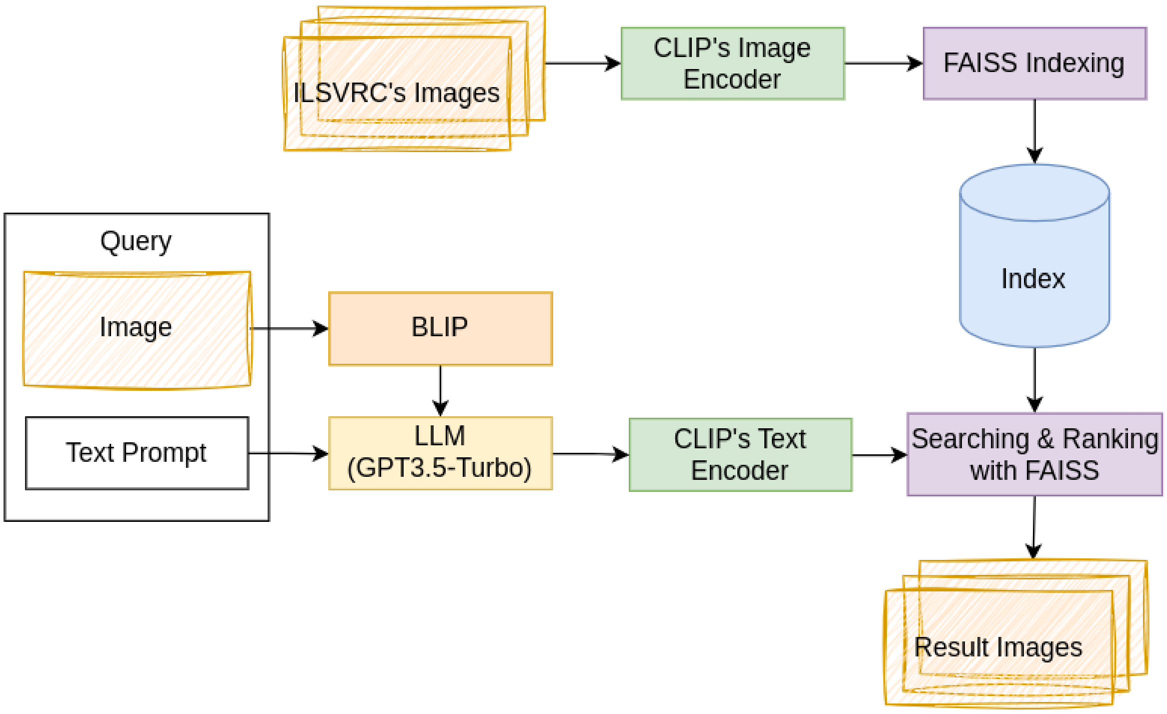

Figure 5: Model pipeline of the composed system

4 Interface framework

To transition the project from a terminal-based environment to a web-based system featuring a user-friendly interface, we leverage FastAPI to seamlessly link user queries to the system. Additionally, we implement a dedicated web page to facilitate direct interaction with users and streamline connectivity with the APIs.

5 Experiment

5.1 Dataset

We choose the validation part of the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) dataset, 2012 version as a main dataset to estimate the model performance. The dataset is built to solve the image classification task, which has 1000 object classes, in total. There are 1 281 167, 50 000, 100 000 in the training, validation and testing set, respectively. Because of the limitation of our project, only the validation set is kept.

Figure 6: Some examples from the ILSVRC Dataset

More information about the dataset can be found at homepage.

For the purpose of comparison, we also selected the Oxford Buildings (Oxford5k) dataset and the Paris (Paris6k) dataset to evaluate system performance crossing several domains. Both datasets contain images related to buildings and other infrastructure from the respective city.

5.2 Module configuration

For the CLIP model, we choose the VitL14 variant to be the encoder for image because it is proven to have competitive performance in extracting features from images and has been widely supported by popular machine learning libraries. The text encoder, as mentioned, remains stable from CLIP.

For similarity comparison, we leverage cosine similarity to measure the similarity between embedding vectors. Notably, this metric proves particularly effective in cases where vectors are already normalized. Additionally, cosine similarity boasts strong integration and efficient computation within the FAISS toolkit, further enhancing its practical suitability.

5.3 Evaluation metric

As this is an information retrieval task, we decided to select mean average precision at top k (mAP@K) to evaluate the effectiveness of model. As users usually pay attention only to some results that are shown at the top, we think this metric is the most suitable. The set of hyperparameter k chosen for mAP@k includes 5, 10, 20 and 50.

5.4 Results

The model is evaluated with 5000 images randomly selected from the validation set of ILSVRC 2012 and sent to the system as follows:

- For the image-based retrieval by image, the images are input directly.

- For the text-based retrieval task, we extract the gold labels from images and send the labels to the system.

mAP@	Image ILSVRC	Image Oxford	Image Paris	Text ILSVRC
5	95.5	91.3	93.4	68.1
10	89.2	80.8	85.6	65.5
20	81.3	69.0	78.6	61.3
50	69.6	54.9	67.9	53.7

Table 1: Experimental results from our system (mAP %)

Model	Dataset	Image
Delhumeau et al., 2013	Holidays	65.8
He et al., 2015	VOC-12	76.4

Table 2: Baselines from previous research (mAP %)

There is a limited number of research that treats this task as an information retrieval task. Hence, it is difficult to select a solution that uses mAP or other IR-based evaluation metrics. These two approaches are the most suitable baselines that we could find.

References

[1] Olga Russakovsky et al. "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision (IJCV)* 115.3 (2015), pp. 211–252. DOI: 10.1007/s11263-015-0816-y.

[2] Alec Radford et al. *Learning Transferable Visual Models From Natural Language Supervision*. 2021. arXiv: 2103.00020 [cs.CV].

[3] Jeff Johnson, Matthijs Douze, and Hervé Jégou. *Billion-scale similarity search with GPUs*. 2017. arXiv: 1702.08734 [cs.CV].

[4] Junnan Li et al. *BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation*. 2022. arXiv: 2201.12086 [cs.CV].

[5] Kaiming He et al. *Deep Residual Learning for Image Recognition*. 2015. arXiv: 1512.03385 [cs.CV].

[6] Ashish Vaswani et al. *Attention Is All You Need*. 2017. arXiv: 1706.03762 [cs.CL].

[7] Tom B. Brown et al. *Language Models are Few-Shot Learners*. 2020. arXiv: 2005.14165 [cs.CL].