Document: Workflow for Table Detection, Extraction, and OCR

Overview

This document outlines the steps performed to detect, extract, and process tables from images using a series of advanced machine learning models. The workflow includes table detection, table extraction, removal of tables from images, Optical Character Recognition (OCR), and conditional processing based on the type of query. Additionally, a specialized model is utilized for efficient handling of these tasks on a specific hardware configuration. Below are the detailed steps:

Step 1: Table Detection

Model Used: Table Transformer (DETR)
Training Dataset: PubTables1M

The first step involves detecting tables within an image using the Table Transformer (DETR) model. DETR is a Transformer-based object detection model, and in this case, it has been specifically trained on the PubTables1M dataset for enhanced accuracy in recognizing table structures in documents.

- **Key Details:**
  - **Transformer Architecture:** DETR
  - **Normalization:** The "normalize before" setting is applied, which means Layer Normalization is performed before self- and cross-attention mechanisms within the Transformer model.

Step 2: Table Extraction

Model Used: Table Transformer (TATR)
Training Datasets: PubTables1M and FinTabNet

After detecting the tables, the next step is to extract the content of the tables using the Table Transformer (TATR) model. This model is trained on a combination of PubTables1M and FinTabNet datasets to ensure it can accurately capture and structure the table data from diverse document formats.

Step 3: Table Removal

In this step, all detected tables are removed from the image. This process is essential to prepare the document for OCR by eliminating the table structures, which can interfere with text recognition.

Step 4: Optical Character Recognition (OCR)

Once the tables have been removed, OCR is performed on the document. OCR is crucial for converting the text within the image into machine-readable text. This step ensures that all text information, excluding tables, is accurately captured.

Step 5: Table to HTML Conversion

After extracting the tables, the next step is to convert the extracted table data into HTML format. This conversion allows to pass the table structure as text to the LLM. The table extraction results are formatted into HTML table elements (<table>, <tr>, <td>, etc.).

Step 6: Conditional Processing Based on Query

The final step involves conditional processing of the image and OCR results based on the nature of the query:

- **Image Graph/Logo Related Queries:** If the query requests information related to images, graphs, logos, or other non-textual elements, both the image (with OCR applied) and the OCR text data are used to generate the response.
- **Textual Queries:** If the query pertains only to text, then only the OCR text data is used for generating the response.

Hardware and Model Optimization

Model Used: LLaVA-13B Quantized in 4-bit
Hardware: NVIDIA GeForce RTX 3090

To efficiently handle the processing on a GeForce RTX 3090 GPU, the LLaVA-13B model is quantized to 4-bit precision. This quantization allows the model to run efficiently on the hardware, providing a balance between performance and resource utilization while maintaining accuracy in tasks such as table detection, extraction, and OCR.

By following this workflow, we ensure accurate detection, extraction, and processing of tables from images while also providing flexible query handling to meet various information retrieval needs. The use of LLaVA-13B quantized to 4-bit enables effective utilization of the 3090 GPU, optimizing the overall processing pipeline.