Explanation of code

Overview

In this project, we developed an automated pipeline to extract and validate entity values from images. The primary goal was to identify numeric values associated with various attributes (e.g., weight, volume, dimensions) and ensure that these values were accurately extracted along with their corresponding units of measurement. We leveraged Optical Character Recognition (OCR) and Natural Language Processing (NLP) techniques to achieve this, using multiple OCR engines and a sophisticated method to filter and prioritize relevant values. The extracted entity values are used in fields like e-commerce, healthcare, and content moderation.

The project pipeline can be divided into three main components:
1. Image downloading and preprocessing.
2. Text extraction using PaddleOCR.
3. Entity value extraction using a custom numeric extraction function.

Detailed Approach

1. Image Downloading and Preprocessing

The first stage of our pipeline involved downloading the images provided through URLs in the dataset. We implemented a function to handle potential issues, such as broken links, by creating placeholder images to ensure the integrity of the dataset remains intact. The function downloads each image, and if a download fails after multiple attempts, a black placeholder image is created. This method prevents missing image data from halting subsequent stages of processing. To expedite downloading, we incorporated multiprocessing, allowing multiple image downloads to occur in parallel, significantly reducing the processing time.

2. Text Extraction using PaddleOCR

Text extraction was handled using **PaddleOCR**, an advanced OCR engine that supports GPU acceleration, making it ideal for large datasets. The OCR process captures any relevant text from the images, which is crucial for entity value extraction. We selected PaddleOCR over other OCR tools like EasyOCR and Tesseract due to its enhanced ability to accurately recognize text in various orientations and fonts, particularly for structured data like product packaging and labels.

Each image is processed individually, and the extracted text is stored in a dedicated column in the dataset. This text serves as the foundation for the entity extraction process. The OCR engine is designed to handle different image resolutions and textual angles, ensuring that even complex image layouts can be parsed effectively.

3. Entity Value Extraction

The core of this project revolves around accurately extracting numeric values and their corresponding units from the OCR-processed text. The dataset contains multiple entity types, such as **item weight**, **volume**, **height**, and **width**, each with its own set of valid units. To handle this, we created a custom function that:

- Identifies numeric values followed by units.
- Filters the extracted units to ensure they match the expected unit for each entity type.
- Prioritizes the most relevant numeric values based on specific keywords.

Entity-Unit Mapping

For each entity, we defined a valid set of units (e.g., grams, kilograms for weight, and litres, millilitres for volume). To ensure flexibility, we also created an extensive unit correction mapping to handle common abbreviations and unit variations (e.g., "g" for grams, "kg" for kilograms). This allowed us to standardize the extracted values, ensuring consistency across the dataset.

Prioritization of Values

Given that images often contain multiple numeric values, some of which may not be relevant, we introduced a prioritization mechanism. For entities such as **item weight**, specific keywords like "net weight" or "net wt" are given higher priority when multiple values are extracted. This ensures that the most accurate and contextually relevant values are selected for further processing.

Regular Expression Matching

We designed a robust regular expression pattern to capture numeric values followed by valid units. The function processes the extracted text, identifies all potential matches, and applies the prioritization mechanism to return the most likely correct value. The result is a clean, standardized value associated with the given entity type.

4. Model Evaluation and Submission Preparation

To assess the performance of our extraction model, we compared the predicted values against the ground truth in the training set using the **F1 score**, a metric that balances precision and recall. The F1 score was calculated in a macro-averaged manner, ensuring fair evaluation across different entity types. The model performed well in extracting relevant entity values, particularly in cases with high variability in unit representation.

Finally, we applied the extraction function to the test set and prepared the submission by including the extracted predictions for each image, ensuring the format aligns with the hackathon requirements.