Amazon ML Challenge - Team DBkaScam

Arnav Goel, Medha Hira, Mihir Agarwal, A. S. Poornash

September 2024

1 Problem Statement

Given the images and entity name, we were tasked to create a machine-learning model that extracts entity values from images. The task was particularly challenging as we had to ensure the correct unit and value predictions.

2 Approaches Explored

We were provided with a train and test datasets of images extracted from the Amazon Database.

1. Although our initial intuition led us to approach this problem by performing a classification task for the units (based on the entity name) and a regression task for calculating the values, we quickly realized this approach would yield suboptimal results on the F1 metric, which considers True Positives, False Positives, and False Negatives. Given the maximum allowed deviation of $\pm 2\%$, this method would result in poor performance using regression.

2. We also experimented with Tesseract for OCR to extract text from images. The results were sub-optimal on the train set. Tesseract struggled with accurately parsing images, particularly when the text was too small. Additionally, it processed text strictly from left to right, often overlooking closely situated units and values, which led to numerous inaccuracies in the extracted data.

3. Finally, we explored the use of Vision Language models. This has been further elaborated in Section:3.

On taking a closer look at the dataset and performing the necessary EDA, we realized the problem at hand involved reading the dimensions of : width, depth, height, item weight, maximum weight recommendation, voltage, wattage, and item volume from the given datasets to make accurate predictions.

3 Visual Language Model - based approach

Vision Language Models (VLMs) have recently emerged as powerful tools in Optical Character Recognition (OCR), improving the accuracy and versatility of extracting text from images. These models leverage both visual and linguistic information, making them superior to traditional OCR systems, which rely solely on visual features.

Below are the VLMs that we tried for the task at hand:

- Qwen2-VL-7B
- InternVL2-8B (8B)
- MiniCPM-V-2.6 (8B)

Different approaches tried with VLMs:

- Zero Shot Prompting: We leveraged the model's pre-trained knowledge, focusing on crafting better prompts to achieve consistent and accurate outputs.
- Supervised Fine Tuning: We fine-tuned Qwen2-VL on roughly 150,000 training samples using Low Rank Adaptation but observed minimal F1 score improvement, likely due to the lack of empty string cases causing hallucinations when no correct unit was present in images.
- Dynamic Few Shot Learning: We then shifted to this approach, designing **four curated examples of edge cases for each entity name** and using them as examples in the prompt. This strategy improved the model's clarity and accuracy.

During zero-shot inference with MiniCPM, we faced unit variation noise like "cm," "centimeter," and "centimetre," leading to over 1,000 unique units. We implemented regex and a rule-based system to standardize unit formats. Additionally, when the model output two units, the one next to the value was prioritized. We also managed edge cases like missing values or invalid output and recorded the upper limit for range values. Moreover, extra post-processing converted fractions and mixed fractions into decimal format for uniformity.

Focus on prompt design helped us decrease the cases of noise and our final Improved Prompt is given below:

Improved Prompt

Please extract the item weight and its unit of measurement from the image, providing them separately. Ensure that the unit is one of the following: {str_units}. Format your response as follows:
Value: <only the numerical value>
Unit: <unit of measurement from the specified list>

BEST PERFORMANCE: Dyanmic few shot with Improved prompt on Qwen2-VL-7B!