IITians_In_De_North – Amazon ML '24

EDA

The following was the distribution of different entities asked in the "test" dataset: {'height': 32282, 'width': 26931, 'depth': 28146, 'item_weight': 22032, 'maximum_weight_recommendation': 7028, 'wattage': 5447, 'voltage': 5488, 'item_volume': 3833}. After inspecting data of all these entities in the "train" dataset, we arrived at following conclusion:

Entity	Train Data Quality	Quantity
Height, Width	Good	43597 + 44183
Depth	Bad	45127
Weight	Moderate/Good	102786
Max Weight Recommend	Moderate	3263
Volume	Moderate	7682
Wattage, Voltage	Moderate/Bad	9466 + 7755

The depth training data was pretty bad with various false positives. The major reason of false positives being that most of the objects in consideration were 2D. Similar composition of 2D objects was also observed in the test dataset.

Methods Tried

The following metrics were noted when trying each approach:

- Inference time.
- Performance on each entity.
- Finetune capability (if required).

The following approaches were tried:

- EasyOCR + Parser + BertQA: OCR was run on each test sample and BertQA was used to extract entity value from parsed OCR text.
- OCR + YOLO: Approach inspired from D-Extract paper by Amazon (link). YOLO v7 was finetuned to classify the bounding box of OCR as width/height/depth to extract dimensions from images. Train dataset along with D Extract Dataset was used.
- InternVL2-1B, MoonDream2, Phi-3-Instruct, Vila, PaliGemma: The VLMs were used to extract entities from images by ImageQA.
- DimenExtract Finetuned MoonDream2: Moondream2 was finetuned on width+height train dataset.
- Volt+Watt Finetuned MoonDream2: Moondream2 was finetuned on volt+watt train dataset.

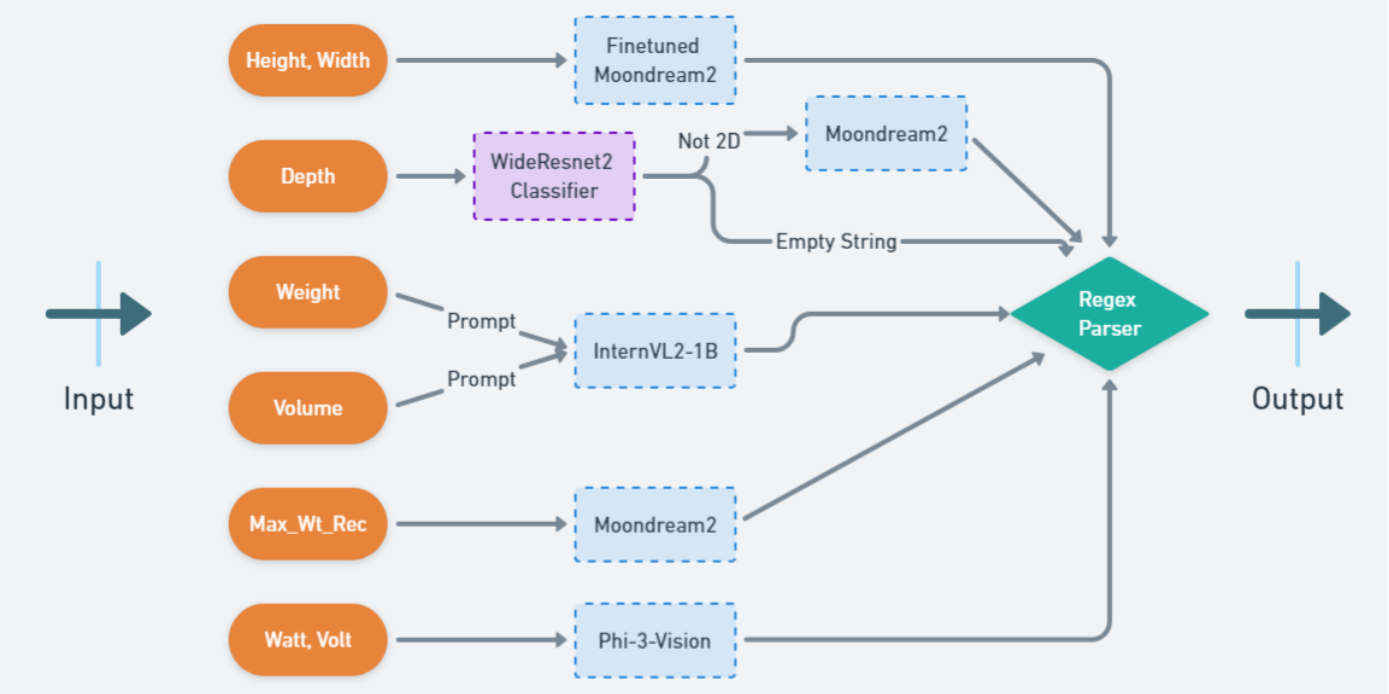
Initial Observation/Evaluation of Methods:

1. Parsed EasyOCR + BertQA was pretty fast, but was very unpredictable and unstable. Different types of prompts and parser protocols were required for different groups because of OCRs inability to extract text without mistakes.
2. Finetuned YOLO got overfitted and classified most of the bounding boxes as "width" despite of balanced nature of training data.
3. Vila was very accurate but had huge inference time. PaliGemma and Phi-3-Vision had moderate inference time but Phi-3 performed better in each entity class. Both being big models, were difficult to fine tune with limited compute availability.

4. InternVL2-1B had decent performance across all entities except in dimensions where it hallucinated, its amazing OCR capabilities coming in handy when required to extract weight from image. It was fast (~1/2 sec per image). The bigger InternVL2 models were very better but required heavier compute.
5. Moondream2 was lightweight, insanely fast (0.5s / img) and showed decent performance across all domains given its size. Its performance was best in max_weight_recommended but it gave many false positives in volt+watt , depth and used to confuse and interchange between height and width.

Final Approach

After carefully analysing all the methods and train+test dataset we decided to finetune moondream2 for width+height as it was fast, light and training dataset contained good and numerous samples of width, height. Here is the link for our finetuned model on huggingface: https://huggingface.co/Meghnad/moondream2-dimextract-ft-464-10 . Moondream2 was also used to infer max_weight_recommendation. InternVL2-1B, given its superior performance in OCR and its speed was used to extract weight and volume from images. We finetuned moon dream to help with it with its false positive problem in voltage and wattage data, link of finetune: https://huggingface.co/Meghnad/moondream2-voltwatt-ft-82-8 . But the moderate quality of training dataset didn't help to improve the performance a lot. As the voltage + wattage samples in test dataset were only 17221, we decided to run them through Phi-3-Vision given its superior reasoning ability to handle false positives. Coming to the depth dataset, which was most notorious and contained lots of false positives in training dataset. Even in the test dataset, most of the depth samples were 2D images. Only heavy models like InternVLM2-4B, vila, Phi-3-vision were able to output correct results, all other light models were hallucinating even after implying prompt engineering techniques. But given the huge size of depth training dataset, we couldn't use heavy VLMs. As a result, we applied transfer learning on Wide_Resnet_v2 and finetuned it on manually annotated data to classify weather an image was 2D or not. Depth of all 2D images was returned as null and the other images were passed to moondream2. A regex parser pipeline was deployed to parse all the responses to suitable formats.



********** End Of The Report **********