AMAZON ML CHALLENGE

TEAM NAME- **THE BIT LORDS**

Introduction

This document outlines the approach taken for extracting entity values from images in a machine learning challenge. The primary tool employed was PyTesseract, an optical character recognition (OCR) library. The goal was to accurately extract relevant information, such as weight, volume, voltage, wattage, and dimensions, from product images.

Data Exploration and Preprocessing

The dataset consisted of 260,000 training images and 131,000 test images. No preprocessing steps were applied to the images prior to OCR. This decision was primarily due to the computational constraints of working without an external GPU, which would have slowed down preprocessing operations.

OCR with PyTesseract

PyTesseract was used to extract text from the images. The library's default settings were used without any significant modifications. While PyTesseract proved effective in many cases, its performance was hindered by the quality of some images. Many images had low resolution, unclear text, or light-colored text, making it difficult for the OCR engine to accurately recognize characters.

Entity Recognition and Unit Conversion

Entity recognition and unit conversion were implemented using custom algorithms and string functions. A list of potential unit abbreviations was compiled, and logic was applied to identify and extract relevant information from the extracted text.

Results and Evaluation

The model achieved an accuracy of 49% on the training set and 56% on the test set. However, the prediction process took approximately 800 minutes due to the lack of an external GPU.

The final evaluation by the competition organizers yielded an F1-score of 0.319, indicating room for improvement. The primary factors affecting performance were the quality of the images and the limitations of the OCR library.

Challenges and Future Directions

The primary challenges encountered were related to image quality and the computational limitations of working without an external GPU. To improve performance, future work could explore:

- **Image Preprocessing:** Implementing preprocessing techniques such as resizing, denoising, and contrast enhancement to improve image quality.
- **OCR Model Optimization:** Experimenting with different OCR models or fine-tuning PyTesseract for specific image characteristics.
- **Custom Entity Recognition:** Developing more sophisticated entity recognition algorithms tailored to the specific domain and entity types.
- **Hardware Acceleration:** Utilizing a GPU or TPU to accelerate the OCR process and reduce prediction time.

Conclusion

While PyTesseract proved to be a valuable tool for image-based entity extraction, its performance was limited by the quality of the images and computational constraints. Future research and advancements in OCR technology could address these challenges and lead to more accurate and efficient solutions.

TEAM MEMBERS

GORRE DINESH CHANDAN

SANGEETH

JUNUBALA ROHITH

ANNADI ASHRITHA