Experience Report: Building an SSD Object Detector

By: Parth Prasad Awasthi

Overview

This report outlines my hands-on experience implementing an **SSD (Single Shot MultiBox Detector)** object detection model using a **ResNet50** backbone, as required in the course assignment. The project involved designing a modular deep learning pipeline in PyTorch and dealing with a variety of technical and conceptual challenges throughout the process.

1. Challenges Faced During Implementation

One of the most significant challenges I encountered during the development and implementation of this project was memory management. My system's GPU repeatedly ran out of memory while training the SSD model. After considerable code optimization and adjustments — including reducing batch size and modifying data pipelines — I managed to stabilize the training process. However, during this phase, my system crashed twice due to the intensive computational load. This issue was closely tied to the nature of the project itself, which is demanding on hardware resources and requires careful handling to maintain system performance.

2. How I Used AI Tools (ChatGPT, Google AI Studio, etc.):

I actively utilized AI tools like **ChatGPT** and **Google AI Studio** during the development process. These tools were particularly useful in helping me understand complex parts of the implementation and in debugging various issues. They assisted in identifying the causes of errors and suggesting potential solutions, enabling smoother progress. My goal was to approach the project methodically and ensure system stability.

3. The Balance Between Writing Code Myself vs. Using AI Assistance:

The availability of AI-assisted tools has greatly enhanced the efficiency of coding tasks. While I was responsible for the overall architecture and logic of the project, AI tools played a supporting role in clarifying concepts and resolving technical issues. This allowed me to focus more on designing the solution and aligning it with the intended outcome. The process became more about making strategic decisions and less about repetitive troubleshooting, improving overall productivity.

4. What I Learned:

- I gained a deep understanding of SSD architecture and its dependencies, including anchor generation, IoU calculations, and loss functions.

- I learned the importance of memory management and how to work with limited hardware resources during model training.

- The project strengthened my debugging skills, especially in dealing with complex tensor operations and shape mismatches.

- I also developed a more structured approach to problem-solving using AI tools for support when necessary.

5. Suggestions for Improving the Assignment:

It would’ve helped if there were more visible examples of combinations — like which detection frameworks + datasets are lightweight enough to run on low-end devices without causing system meltdowns.

Also, some starter code for tricky parts like IoU matching or loss calculation scaffolds would make the learning curve slightly less vertical. But all said and done, this project was super fun, very challenging, and felt like building a boss-level AI tool from scratch.

Cheers!