

Experience Report: Project Setup and Execution

by Bhavya Jain, Rishabh Dhiman

Selecting the Laptop

We began by carefully selecting hardware that was optimal for the intended DLStreamer and OpenVINO applications. After research and consultation, we chose the Asus Zenbook S14 featuring the Intel Core Ultra 2 256V CPU with 2 performance cores and 6 efficiency cores (Lunar Lake architecture), Intel Arc 140V GPU, and integrated Intel NPU providing 47 TOPS. This ensured compatibility and optimal performance for our AI workloads.

Setting up Dual-Boot Ubuntu

We set up dual-booting by first preparing a bootable USB drive with Ubuntu 24.04 LTS. Using BIOS settings, we disabled Secure Boot and adjusted boot priorities. We successfully created partitions and installed Ubuntu alongside the existing Windows OS, ensuring we could use both environments interchangeably. This dual setup offered flexibility for development and testing.

Installing Essential Software

After Ubuntu installation, we updated the package repositories and installed critical tools:

- Python 3.12
- GStreamer 1.18
- OpenVINO 2025.2
- DLStreamer plugins
- NumPy 1.26

We extensively referred to official Intel OpenVINO and DLStreamer documentation to ensure a smooth installation process, following the detailed guidelines provided on their official websites and GitHub repositories.

Environment Configuration

We created and activated Python virtual environments to manage dependencies efficiently. We installed the necessary packages using pip:

```
python3 -m venv venv  
source venv/bin/activate  
pip install --upgrade pip
```

```
pip install openvino numpy
```

Downloading and Benchmarking Models

We cloned the OpenVINO Model Zoo using the provided documentation, downloading various pre-trained IR models specifically optimized for Intel hardware. Using custom Python scripts, we benchmarked each model's inference throughput, measured FPS, and calculated the maximum number of simultaneous streams supported.

Running the Project

The comprehensive benchmarking included testing various models under controlled conditions, conducting warm-up iterations, and capturing precise performance metrics. Using DLStreamer pipelines, we assessed real-world inference performance to validate theoretical maximum streams for different models.

Documentation and Troubleshooting

Throughout the setup, official Intel documentation proved invaluable. When issues arose—such as missing dependencies or version mismatches—we relied on documentation and community forums to resolve them promptly.

Conclusion

Selecting Intel-based hardware, carefully configuring dual-boot Ubuntu, installing appropriate software, and adhering closely to provided documentation were critical factors in the successful execution of our AI benchmarking project. This structured approach enabled us to achieve precise performance insights and efficiently utilize Intel's cutting-edge technologies.