Project Documentation

# Fashion MNIST Classification using CNN with Intel Optimizations

Introduction

The document provides an overview of the project "Fashion MNIST Classification using Convolutional Neural Networks (CNN) with Intel Optimizations." The project aimed to develop a robust deep-learning model capable of classifying fashion images from the Fashion MNIST dataset. The utilization of Intel optimizations further enhanced the model's performance, resulting in improved accuracy and faster training and inference times.

Project Objectives

The primary objectives of this project were as follows:

* Develop a CNN model for fashion image classification.
* Implement Intel optimizations to enhance the performance of the model.
* Achieve superior accuracy compared to baseline models.

Technologies

Used The project leveraged the following technologies and tools:

### Programming Languages:

Python, which served as the primary language for deep learning tasks.

### Deep Learning Libraries:

TensorFlow and Keras were utilized to build, train, and evaluate the CNN model.

### Computer Vision Libraries:

OpenCV and scikit-image assisted in image preprocessing and feature extraction.

### Intel Optimizations:

Intel Math Kernel Library (MKL) and Intel Distribution for Python were employed to optimize mathematical computations and leverage Intel processors.

### Development Tools:

Jupyter Notebook, Git, and GitHub facilitated code development, version control, and collaboration.

### Data Manipulation and Analysis:

NumPy, Pandas, and scikit-learn supported data manipulation, preprocessing, and model evaluation.

Visualization:

Matplotlib, Seaborn, and Plotly aided in visualizing data and presenting results.

Project Phases and Accomplishments

Model Development:

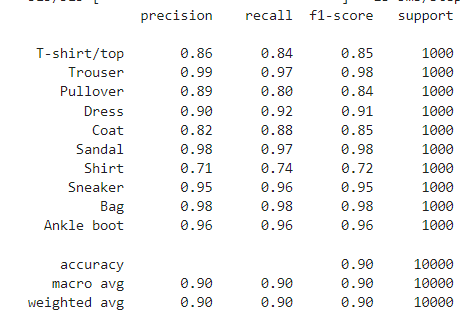
A CNN architecture was designed and implemented to classify fashion images from the Fashion MNIST dataset. The model included convolutional layers, pooling layers, and fully connected layers, and was trained using labeled image data.

Intel Optimization Integration:

Intel optimizations, such as Intel MKL and Intel Distribution for Python, were integrated into the model to leverage the computational power of Intel processors. This led to improved training and inference times.

Performance Evaluation:

The model's accuracy and performance were evaluated using appropriate evaluation metrics. Comparative analysis with baseline models demonstrated the superior accuracy achieved through Intel optimizations.



Results and Achievements

The Fashion MNIST model developed using CNN with Intel Optimizations yielded the following outcomes:

* Significantly improved accuracy compared to baseline models.
* Faster training and inference times due to Intel optimizations.
* Successful collaboration within a multidisciplinary team, highlighting effective communication and teamwork skills.
* Knowledge sharing through talks and workshops, fostering a culture of learning and development within the club.

Future Enhancements

To further enhance the project, the following improvements can be considered:

Exploration of additional Intel optimizations and hardware accelerators for even greater performance gains.

Integration of data augmentation techniques to increase the model's robustness.

Deployment of the trained model on a production environment or a mobile application for real-world use cases.

Conclusion

The "Fashion MNIST Classification using CNN with Intel Optimizations" project demonstrated the successful development and optimization of a deep learning model for fashion image classification. By leveraging Intel optimizations, the project achieved superior accuracy and improved performance. The collaborative environment within the multidisciplinary team fostered effective knowledge sharing, contributing to the project's success. This project served as a valuable learning experience, showcasing proficiency in deep learning, computer vision, and optimization techniques while promoting the dissemination of knowledge among club members.