

Final Project: AI Model Development and Deployment

Overview

For your final project in the AI fundamentals course, you will develop and deploy a machine-learning model using the skills and concepts learned. You have 2.5-3 hours to complete the project, working through several key phases of the AI development lifecycle. You will use one of the provided datasets or you can use a dataset you bring.

Project Phases and Instructions

Your task is to create a Jupyter notebook or script that documents each phase of the project. ClearML is optional, but if used, it can assist in tracking experiments and logging results.

Available Datasets

Select from the following datasets, all compatible with CPU-only environments:

- **Image Datasets:**

```
import torchvision.datasets as datasets

# CIFAR-10: Small image classification (10 classes)
cifar10_train = datasets.CIFAR10(root='./data', train=True, download=True)

# MNIST: Handwritten digit classification (10 classes)
mnist_train = datasets.MNIST(root='./data', train=True, download=True)

# COCO: Object detection and segmentation
coco_train = datasets.CocoDetection(root='./data/train2017', annFile='./data/annotations/instances_train2017.json')

# VOC 2012: Object detection
voc_train = datasets.VOCDetection(root='./data', year='2012', image_set='train', download=True)
```

- **NLP and Tabular Datasets:**

```
from sklearn.datasets import fetch_20newsgroups, load_iris, load_wine

# 20 Newsgroups: Text classification (NLP)
newsgroups_train = fetch_20newsgroups(subset='train')

# Iris: Classification of iris species (Tabular)
iris = load_iris()

# Wine: Classification of wine types (Tabular)
wine = load_wine()
```

Note: Use the **Wine** dataset instead of the deprecated **Boston Housing** dataset for tabular data.

Recommended Setup

To ensure a smooth development process, it's advised to set up a virtual environment (venv) and install the necessary packages. Here's how:

1. Create and Activate a Virtual Environment:

```
python -m venv myenv  
source myenv/bin/activate # On Windows: myenv\Scripts\activate
```

2. Install Required Packages:

```
pip install numpy pandas matplotlib scikit-learn torch torchvision xgboost jupyter
```

3. (Optional) Install ClearML for Experiment Tracking:

```
pip install clearml
```

Project Phases

Phase 1: Problem Identification, Data Collection, and Annotation

- **Objective:** Define the problem and understand the dataset.
- **Tasks:**
 1. **Select a Dataset:** Choose one from the provided image, NLP, or tabular datasets.
 2. **Problem Definition:** Clearly define the problem (e.g., image classification, text classification, tabular data classification).
 3. **Data Exploration:** Explore the dataset to understand its structure, characteristics, and any preprocessing needs.
 4. **Annotation (if applicable):** Describe the labeling process and how labels are used (especially for object detection tasks).
- **Output:** A Jupyter notebook or script with code and comments explaining the selected problem and dataset exploration.

Phase 2: Model Development and Initial Experimentation

- **Objective:** Develop and train an initial model.
- **Tasks:**
 1. **Model Selection:** Choose an appropriate model architecture (e.g., CNN for images, Logistic Regression or XGBoost for tabular data).
 2. **Model Implementation:** Implement the chosen model using PyTorch, scikit-learn, or XGBoost.
 3. **Training:** Train the model on the dataset and evaluate its initial performance using appropriate metrics.
 4. **(Optional) Experiment Logging:** Use ClearML or similar tools to log experiments and track performance metrics.
- **Output:** A Jupyter notebook or script demonstrating the model implementation, training process, and initial performance results.

Phase 3: Hyperparameter Tuning and Model Optimization (Optional)

- **Objective:** Enhance model performance through tuning and optimization.
- **Tasks:**
 1. **Hyperparameter Tuning:** Experiment with various hyperparameters (e.g., learning rate, number of trees for XGBoost) to optimize model performance.
 2. **Model Optimization:** Apply optimization techniques like pruning, quantization, or model compression (if applicable).
 3. **Validation:** Evaluate the improved model on a validation set and document the changes.
- **Output:** Updated Jupyter notebook or script showing the tuning and optimization processes and their impact on model performance.

Phase 4: Deployment Simulation and Performance Evaluation (Optional)

- **Objective:** Simulate deploying the model and assess its performance in a production-like environment.
- **Tasks:**
 1. **Model Conversion:** Convert the model to a deployment-ready format using TorchScript (for PyTorch models) or export models for use with APIs.
 2. **Deployment Simulation:** Set up a simple API using Flask or a deployment tool like TorchServe to serve the model.
 3. **Performance Evaluation:** Measure the model's inference speed, resource usage, and scalability in the simulated environment.
- **Output:** A Jupyter notebook or script detailing the deployment simulation and performance evaluation steps.

Phase 5: Final Presentation and Critical Review

- **Objective:** Summarize your project and reflect on the process and outcomes.
- **Tasks:**
 1. **Project Summary:** Write detailed comments in your notebook or script summarizing each phase of your project.
 - Problem definition and dataset selection.
 - Model development and initial results.
 - (Optional) Hyperparameter tuning and optimizations.
 - (Optional) Deployment simulation and performance evaluation.
 - Challenges faced and solutions.
 2. **Critical Review:** Provide a critical analysis of your approach within the notebook or script, discussing potential improvements and future work.
- **Output:** A well-documented Jupyter notebook or script that includes a summary and critical review of your project.

Completion

This project will help you practically apply the concepts learned in the course to develop and simulate the deployment of an AI model. Focus on understanding each phase and documenting your work thoroughly in the notebook or script. Enjoy the process of bringing your AI model from concept to a deployment-ready state!