

Power Learn Project – AI Module Assignment Report

Submitted by: Omope Elizabeth

Module: AI Fundamentals

Part 1: Theoretical Understanding

Q1: Explain the primary differences between TensorFlow and PyTorch. When would you choose one over the other?

TensorFlow and PyTorch are leading deep learning frameworks. TensorFlow, developed by Google, is preferred for production-level applications and supports both low-level and high-level APIs via Keras. PyTorch, developed by Meta, offers a more Pythonic and flexible experience, making it ideal for research and experimentation. TensorFlow is often chosen for deployment, while PyTorch is favored for model development and academic research.

Q2: Describe two use cases for Jupyter Notebooks in AI development.

Jupyter Notebooks are widely used for: (1) interactive data exploration and visualization, and (2) building and documenting AI models in an iterative, transparent way. They combine code, results, and explanations in one place for easy collaboration and learning.

Q3: How does spaCy enhance NLP tasks compared to basic Python string operations?

spaCy offers pre-trained NLP pipelines for tasks like tokenization, POS tagging, and named entity recognition (NER), which go beyond basic string operations. It understands linguistic context, allowing more accurate and efficient text analysis.

Comparative Analysis: Scikit-learn vs TensorFlow

Scikit-learn is designed for classical machine learning models (like Decision Trees and SVMs), while TensorFlow is specialized for deep learning and neural networks. Scikit-learn is easier for beginners and small datasets; TensorFlow is ideal for large-scale, production-ready models.

Part 2: Practical Implementation

Task 1: Classical ML with Scikit-learn (Iris Dataset)

A Decision Tree Classifier was trained on the Iris dataset to predict flower species. The process included data preprocessing, encoding, and model evaluation using accuracy, precision, and recall metrics.

■ *Insert Screenshot 1: Accuracy, Precision, Recall Output*

Task 2: Deep Learning with TensorFlow (MNIST Dataset)

A CNN model was built to classify handwritten digits using the MNIST dataset. The model achieved over 95% accuracy and visualized predictions on sample test images.

■ *Insert Screenshot 2: Accuracy/Loss Graph or Prediction Samples*

Task 3: NLP with spaCy (Amazon Product Reviews)

Using spaCy, named entity recognition (NER) extracted product names and brands, while a rule-based sentiment system categorized reviews as positive or negative.

■ *Insert Screenshot 3: NER and Sentiment Output*

Part 3: Ethics & Optimization

Ethical Considerations: Models can reflect bias from training data. For example, MNIST may not represent all handwriting styles, and Amazon reviews can contain cultural or language bias. Tools like TensorFlow Fairness Indicators and spaCy's rule-based filters help detect and mitigate these biases.

Troubleshooting Challenge: Debugging errors like tensor dimension mismatches and incorrect loss functions ensures model correctness. Common fixes include reshaping inputs and using the correct activation/loss functions.

Conclusion

This assignment provided valuable hands-on experience with Scikit-learn, TensorFlow, and spaCy, deepening understanding of AI development, ethics, and deployment. It also highlighted the importance of fairness and transparency in building responsible AI systems.