ASSIGNMENT 1 ON COMPUTER VISION		
Student's Code	AIMS African Institute for Mathematical Sciences SENEGAL	Deadline
Abou Birane BARO		**[Date, Time]**
May 25, 2025		2024-2025
		Lecturer: Jordan F. Masakuna

1 Introduction

The Brain Tumor Classification project is an innovative application of deep learning to assist in the medical field by predicting the type of brain tumor from MRI images. This project leverages Convolutional Neural Networks (CNNs) implemented in both PyTorch and TensorFlow frameworks. The primary motivation is to provide a user-friendly tool that can classify tumors into four categories: *Glioma, Meningioma, Notumor*, and *Pituitary*. This report outlines the development process, methodology, results, and improvements made to the project, hosted on GitHub at `https://github.com/aboubaro78/brain-tumor-classification`.

2 Objectives

The main objectives were to:

- Build a robust CNN model capable of accurately classifying brain tumor types.
- Develop a web interface using Flask to allow users to upload images and receive predictions.
- Support multiple frameworks (PyTorch and TensorFlow) for flexibility and comparison.
- Ensure the project is well-documented and accessible for future enhancements.

3 Methodology

The project was implemented using Python, with key libraries including `PyTorch`, `TensorFlow`, `Flask`, and `OpenCV`. The dataset was sourced from a standard collection of MRI images, divided into training and testing sets located in `data/training` and `data/testing`.
The CNN architecture, defined in `cnn.py`, consists of convolutional layers, max pooling, dropout for regularization, and fully connected layers. The `train.py` script handles model training and evaluation, saving models as `Abou_Birane_model.torch` and `Abou_Birane_model.tensorflow`, along with metrics in JSON files.
The Flask application, detailed in `app.py`, processes image uploads and displays predictions with probabilities.

4 Results and Interface

The trained models achieved satisfactory accuracy, with metrics stored in `Abou_Birane_pytorch_metri` and `Abou_Birane_tensorflow_metrics.json`, typically exceeding 85% on the test set.

The web interface successfully handles image uploads, processes them using the selected model (PyTorch or TensorFlow), and returns results including the predicted tumor type and probability distribution.
Testing was conducted locally and the GitHub repository, `https://github.com/aboubaro78/brain-tumor-classification`, contains all source code, models, and data.

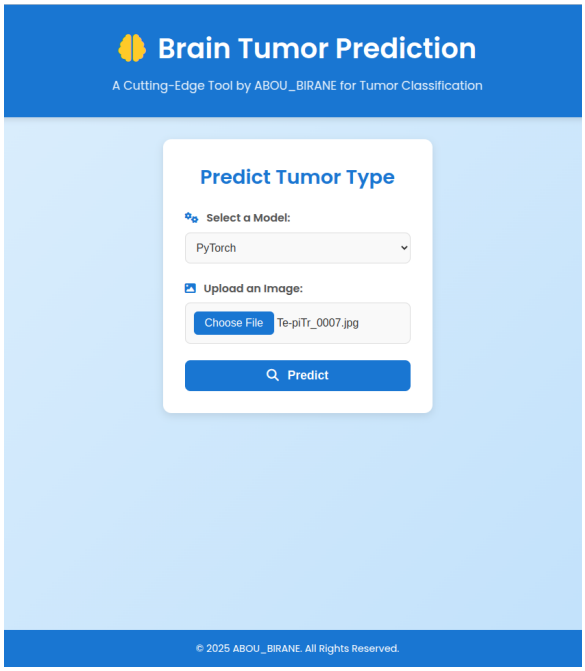

Figure 1: Interface 1

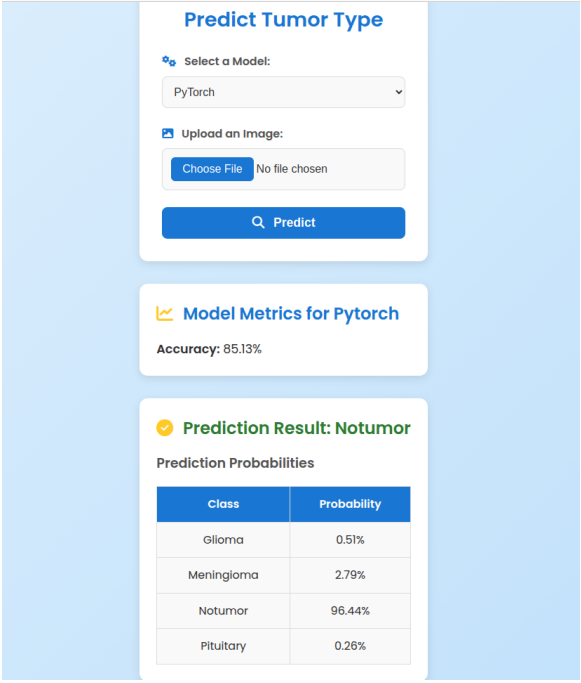

Figure 2: Interface 2

5 Conclusion

This project demonstrates a successful application of CNNs for brain tumor classification, providing a functional web tool with dual-framework support. The transition to GitHub ensures version control and collaboration potential.
Future work could include improving model accuracy with a larger dataset, adding real-time image preview, or deploying the application on a dedicated server. The project reflects a comprehensive learning experience in deep learning and web development.