

hager_XAIProject1 README

Hager Ali (202202011)

Hager Ali
Student ID: 202202011

Introduction

This document provides instructions to set up and run the code in `hager_XAIProject1_.ipynb`, a Jupyter notebook for processing and analyzing brain scan images from the BraTs18 dataset using a convolutional neural network (CNN) and visualization techniques.

Prerequisites

Before running the code, ensure you have the following:

- **Python 3.11:** The notebook was developed using Python 3.11. Install it from python.org or use a version manager like `pyenv`.
- **Jupyter Notebook:** Required to run the `.ipynb` file. Install it with:

```
pip install jupyter
```

- **Hardware:** A system with sufficient memory and CPU/GPU resources (e.g., at least 8GB RAM, GPU recommended for faster training).
- **Dataset:** The BraTs18 dataset in a zip file named `BraTs18.zip`, containing brain scan images and masks in the structure `brats18/images/` and `brats18/masks/`.

Setup Instructions

Follow these steps to set up the environment and run the code:

1. Download the Notebook

Download the `hager_XAIProject1_.ipynb` file to your local machine or clone the repository if hosted on a platform like GitHub.

2. Create a Virtual Environment (Optional but Recommended)

Create a virtual environment to manage dependencies:

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

3. Install Dependencies

Install required Python libraries:

```
pip install scikit-image numpy opencv-python sklearn matplotlib
seaborn tensorflow
```

Ensure the latest version of scikit-image if needed:

```
pip install scikit-image --upgrade
```

4. Download and Extract the Dataset

Place the BraTs18.zip file in the same directory as the notebook. The notebook extracts it to /content/BraTs18 in Google Colab. For local use, modify the path:

```
extract_path = './BraTs18'
```

Ensure the zip file contains:

- brats18/images/ for brain scan images (e.g., image_1.png).
- brats18/masks/ for corresponding mask files (e.g., mask_1.png).

5. Launch Jupyter Notebook

Start Jupyter Notebook:

```
jupyter notebook
```

Open hager_XAIProject1_.ipynb in your browser.

Running the Code

Execute the cells in the notebook in order:

1. Install Libraries & Dependencies

Run the initial cells to install scikit-image and import libraries:

```
import os
import numpy as np
import cv2
import glob
import random
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt
import seaborn as sns
from skimage.feature import graycomatrix, graycoprops
import tensorflow as tf
from tensorflow.keras import layers, models
```

2. Extract the Dataset

Run the extraction cell:

```
import zipfile

zip_file_path = '/content/BraTs18.zip'
extract_path = '/content/BraTs18'

with zipfile.ZipFile(zip_file_path, 'r') as zip_ref:
    zip_ref.extractall(extract_path)

print(f"Files extracted to: {extract_path}")
```

Adjust zip_file_path and extract_path for local use.

3. Data Preprocessing

Run the preprocessing cell to load images and masks:

```
image_paths = sorted(glob.glob(os.path.join(extract_path, 'brats18/
    images/*.png')))
mask_paths = sorted(glob.glob(os.path.join(extract_path, 'brats18/
    masks/*.png')))
```

This loads 5042 images and masks.

4. Additional Steps

Execute further cells for preprocessing, model training, and visualization. If using visualization techniques (e.g., Grad-CAM), initialize the model with a dummy input to avoid errors:

```
simplified_model(tf.zeros((1, 112, 112, 1), dtype=tf.float32))
```

Download Loop

To repeatedly download and process the dataset or iterate over the code execution:

1. **Download the Dataset:** If the BraTs18.zip file is missing or needs updating, download it again from the source and place it in the working directory.
2. **Extract and Verify:** Run the extraction cell to unzip the file. Check the output to confirm files are extracted to the specified `extract_path`.
3. **Loop Execution:** For iterative processing (e.g., training with different parameters or visualizing multiple images):

- Modify the code to loop over image and mask pairs using `image_paths` and `mask_paths`.
- Example loop to process all images:

```
for img_path, mask_path in zip(image_paths, mask_paths):  
    img = cv2.imread(img_path, cv2.IMREAD_GRAYSCALE)  
    mask = cv2.imread(mask_path, cv2.IMREAD_GRAYSCALE)  
    # Add preprocessing and model prediction here  
    print(f"Processed: {img_path}")
```

- Run the notebook cells sequentially for each iteration, adjusting parameters as needed.
4. **Automation:** For continuous looping, consider scripting the process in a Python file and using a loop with a delay:

```
import time  
while True:  
    # Run extraction and processing  
    time.sleep(3600) # Wait 1 hour before next loop
```

Troubleshooting

- **Missing Dataset:** Ensure BraTs18.zip is in the correct directory and update `zip_file_path`.
- **Library Errors:** Install missing libraries manually (e.g., `pip install tensorflow`).
- **Memory Issues:** Reduce batch size or use a subset of the 5042 images if memory is limited.
- **Model Initialization:** If you see `AttributeError: The layer sequential_5 has never been called`, add the dummy input call as shown above.

Outputs

- Dataset extracted to `/content/BraTs18` (or your `extract_path`). - Loads 5042 images and masks for training and visualization. - Visualization outputs saved as PNG files (e.g., `gradcam_output.png`).

Notes

- The notebook assumes a Google Colab environment (`/content/` paths). Adjust paths for local execution. - Ensure the model is trained before visualization, and input data matches the expected shape (e.g., `(112, 112, 1)`).

For further assistance, refer to the comments in the notebook or library documentation (e.g., TensorFlow, scikit-image).