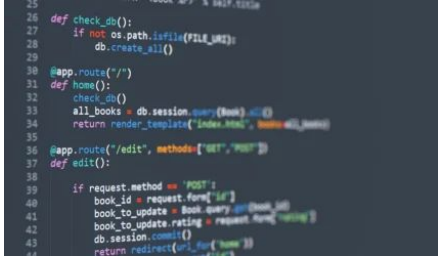
Brain Tumor Detection System using Convolutional Neural Networks

Final View and Delivery

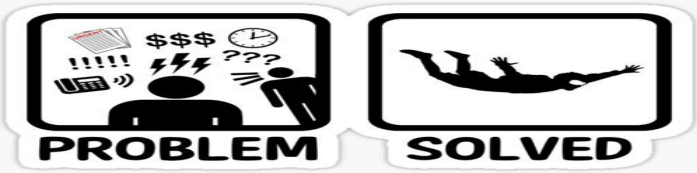
Abdullah Alshamrani
Senior CS Project 2024

Idea of the Project



“Brain Tumor Detection System” aims to make a diagnosis on Medical Images to detect if brain has a tumor or not. Either positive = “Yes tumor” OR “No tumor”. Our project automates the process of detecting brain tumors, **reducing time** and **expertise** required. Our Model **Achieved high accuracy(97.17%)** in identifying brain tumor regions minimizing risk of false positive and false negative. We have developed a user-friendly interface that has only two pages one for uploading and one for the result. Output of the result is Powered with AI to generate an advice depending on the result you get.

Challenges



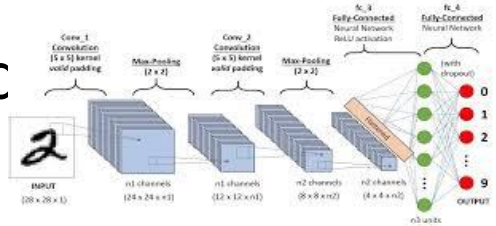
- Data Visualization and understanding

- Data collection

- Achieving High Accuracy

- Implementing OPENAI API KEY to Flask to generate an advice on the result and attaching it to the website

Overcoming problems: Data Visualization and Understanding



At the beginning of thinking to implement the brain tumor detection system project, I was thinking about how this could possibly be done, this was hard to my brain to envision. How we can we make a ML Model such this. But with the use of the internet I started to see papers that are related to this, and I have seen people doing projects like this using different algorithms to implement the detectors. This moved me to know the concepts needed and I started my journey to learn about and relearn some of the concepts one of them and an important one would be “Convolutional Neural Network”(CNN). Overcoming this problem was a long journey of learning through online resource(Research papers, Videos, Articles) to understand the idea about Neural Networks and how it works. This then moved me to make my own detector.

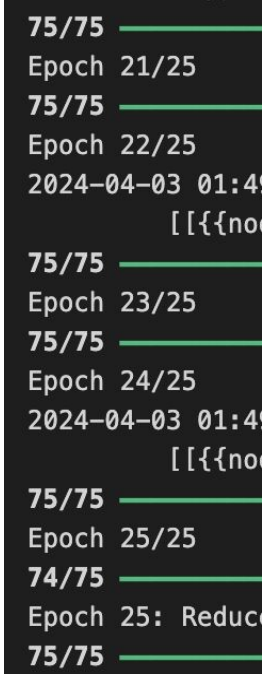
Overcoming problems: Data Collection

In the beginning of this project I did change the data That I worked on for many times, after choosing a good data that I can work with, I have published the reference of this data in my future presentation view #2. This was the first time for me to work with searching the data online. I finally used an available data published in Kaggle.com.

In this journey of learning how to collect good data, I spent many hours in finding good data, but that helps data scientists usually to work better with their Machine Learning Model.

Overcoming problems: Achieving High Accuracy

In achieving High accuracy while testing the model in CNN, I used many methods to get a higher accuracy. I remember starting with **80s %** at the beginning. Ending now where I have a much better/higher accuracy. Brainstorming going around articles, running code with different parameters inserting. Eventually I used epochs, and used **EarlyStopping** to test in epochs. When accuracy improves with each epoch it makes a record of it and when it declines when no improvement is coming anymore or a reduce in the accuracy with next epoch. This is why I trained it in 25 epoch and it was the best where it recorded an accuracy of **97.11%.**




```
scores = cnn_model.evaluate(X_test, y_test_encoded)
print(f'Test accuracy: {scores[1] * 100:.2f}%')
```

```
19/19 ━━━━━━━━━━━━━━━━━━━━ 0s 12ms/step - accuracy: 0.97
Test accuracy: 97.17%
```

Overcoming problems: OpenAI API KEY Implementation on Flask

In implementing the OpenAI API KEY, I had many problems in how to implement it in the code, watching tutorials in how to get an Open API KEY fixed a problem, but implementing it to the Flask web application was a little hard, but tutorials and reading articles fixed it.

After Implementing the whole website was in hold, and access was denied and it is just playing in the server even if you kill the port. Fix for killing the port to work and edit code was restarting the whole system, and this fixed it.

Getting back to the code, I noticed my error was coming because of an error connecting to the OpenAI API because of ‘OpenAI’ Library which I did not download in my environment. Finally and after hours of working this got it to be fixed.

```python
# Generating advice based on the result using AI through OpenAI API KEY
api_key = "sk-proj-VrSASXdvJ9GFArFNUjHYT3BlbkFJdWIJNQ5noRqPYLFJdFCp"


if result == 'Positive':
    advice = ask_openai_for_advice(api_key, result)
else:
    return "You teseted Negative."


return jsonify({'prediction': result, 'advice': advice})


def ask_openai_for_advice(api_key, keyword):
    openai.api_key = api_key


    # Setup the health advice query using the keyword
    query = f"Act as a health advisor, I have been diagnosed with a brain tumor and it
```

Demo of the project