Leveraging AI for Accurate Brain Tumor Classification

A Combined Data Analysis Approach

Hala Basim Sedki
Department of Computing and Digital Technology
ESLSCA University Egypt
Cairo, Egypt
Hala.sedki21d@eslsca.edu.eg

Suzan Hatem Ismail
Department of Computing and Digital Technology
ESLSCA University Egypt
Cairo, Egypt
souzan.ismail21d@eslsca.edu.eg

Nejoud Amr Yazal
Department of Computing and Digital Technology
ESLSCA University Egypt
Cairo, Egypt
nejoud.yazal21d@eslsca.edu.eg

Basmalla Hossam Ibrahim
Department of Computing and Digital Technology
ESLSCA University Egypt
Cairo, Egypt
basmalla.ibrahim21d@eslsca.edu.eg

Abstract—**Brain tumors are a significant health concern, and accurate and efficient detection and classification methods are critical for improving patient outcomes. Artificial intelligence (AI) has emerged as a promising tool for improving the accuracy and efficiency of brain tumor classification. In this study, we aimed to develop a novel approach to brain tumor image classification using a multi-dataset approach and several deep learning architectures, including ResNet50, customized CNN, AssemblyNet, and Dense Neural Network (DNN). We evaluated the performance of these architectures using precision, recall, F1 score, and accuracy metrics.**

Our study used three different datasets of brain tumor MRI images from Kaggle, which included a total of 14,707 images and combined them. We trained each architecture separately on each dataset and evaluated the performance of the models using 10-fold cross-validation. Our results demonstrated that all four deep learning architectures were effective in classifying brain tumor images, with ResNet50 achieving the best performance with an accuracy of 0.82, precision of 0.92, recall of 0.89, and F1 score of 0.91. The customized CNN architecture achieved an accuracy of 0.78, precision of 0.81, recall of 0.86, and F1 score of 0.84. AssemblyNet achieved an accuracy of 0.71, precision of 0.74, recall of 0.81, and F1 score of 0.77. DNN achieved an accuracy of 0.72, precision of 0.76, recall of 0.87, and F1 score of 0.81.

Our study demonstrates the potential of AI, and in particular, deep learning algorithms, for improving the accuracy and efficiency of brain tumor classification. Our multi-dataset approach allowed us to combine different datasets, ensuring that our models were robust and could accurately classify brain tumor images across different datasets. The high precision and recall scores achieved by our models suggest that they could be valuable tools for medical professionals in accurately identifying brain tumors and determining appropriate treatment methods.

In conclusion, our study provides evidence that AI tools can significantly improve the accuracy and efficiency of brain tumor classification, leading to better patient outcomes. Future research could explore the potential of combining different AI techniques, such as deep learning and natural language processing, to further improve brain tumor classification and potentially assist medical professionals in developing individualized treatment plans.

Keywords: Brain Tumor - MRI - Convolutional Neural Network - Artificial Intelligence (AI) - Medical Image Processing - Neurology - Diagnosis

I. Introduction (*Heading 1*)

Brain tumors are abnormal growths of cells within the brain that can be either benign or malignant. They are a serious health concern, with increasing incidence rates worldwide. The process of detecting brain tumors typically involves medical imaging techniques such as magnetic resonance imaging (MRI) or computed tomography (CT) scans. Early detection and classification of brain tumors are critical in determining the appropriate treatment method, optimizing patient outcomes, and ultimately saving lives.

Artificial intelligence (AI) has recently emerged as a promising tool for improving the accuracy and efficiency of brain tumor classification. Machine learning, a subset of AI, includes mathematical algorithms that learn from input data to develop the capacity to generate predictions without being explicitly trained to do so. In the context of medical imaging, machine learning algorithms can be trained using large datasets of labeled images to classify images into different categories, such as the presence of a tumor or the type of tumor.

In this paper, we present a novel approach to brain tumor image classification using artificial intelligence. Our method focuses on using a multi-dataset approach to increase the robustness of our model and improve accuracy across different datasets. Specifically, we combine three different datasets, each containing unique brain MRI images, to make the classification more challenging and ensure that the higher accuracy would apply to different datasets. Our combined dataset of Brain Tumor Image Classification contains images of human brain MRI images which are classified into 4 classes: glioma, meningioma, no tumor, and pituitary. The dataset was also divided into three parts: training, validation, and testing.

To achieve this goal, we employ a convolutional neural network (CNN), a special type of artificial neural network that uses deep learning algorithms. CNNs have been widely used in image and video recognition due to their high performance in recognizing patterns. In our study, we aim to subclassify brain tumor images into different categories using a CNN model trained on our multi-dataset approach. We believe that our approach has the potential to significantly impact the field of medical imaging and improve patient outcomes for those with brain tumors. We also employed several deep learning architectures, including ResNet50, Deep Neural Network (DNN) and AssemblyNet. The ResNet50 is an architecture based on residual connections, which allows for deeper networks to be trained without suffering from the vanishing gradient problem. In the context of brain tumor classification, a deep neural network model can analyze medical images of the brain and classify them as either containing a tumor or not. By training the network on a large dataset of labeled brain tumor images, the model can learn to recognize patterns and features that distinguish between healthy and tumor-affected brain tissue. Once trained, the model can then be used to classify new brain tumor images with high accuracy, helping medical professionals to diagnose and treat brain tumors more effectively. AssemblyNet is an architecture that combines multiple pre-trained models to improve performance on image classification tasks.

To find the best approach, we evaluated the performance of each architecture on our brain tumor image dataset. We used several metrics, including accuracy, precision, recall, and F1 score, to compare the performance of each architecture. We found that ResNet 50 outperformed the other architectures; it achieved the highest performance in classifying brain tumor images, with an accuracy of 0.82, precision of 0.92, recall of 0.89, and an F1 score of 0.91. These results suggest that ResNet50 may be a valuable tool for accurately classifying brain tumor images and improving medical imaging and patient outcomes. The results suggest that the ResNet50 architecture is a promising tool for accurately classifying brain tumor images and has significant potential for improving medical imaging and patient outcomes.

The importance of early detection and accurate classification of brain tumors cannot be overstated. In 2019, there were 347,992 (262,084–388,896) global cases of brain and CNS cancers, which showed a significant increase (94.35%) from the period between 1990 to 2019, and the numbers are increasing (Fan et al., 2022). The ability to accurately classify brain tumors will enable medical professionals to make informed decisions about treatment options, leading to better patient outcomes and potentially saving lives.

Our study demonstrates the effectiveness of deep learning architectures in classifying brain tumor images and highlights the importance of using a multi-dataset approach to improve the robustness of the model. By employing different architectures and comparing their performances, we were able to identify the best approach for our specific dataset. We believe that our approach has the potential to significantly impact the field of medical imaging and improve patient outcomes for those with brain tumors.

II. Materials and methods

A. *Cases Collection*

Inclusion Criteria

The inclusion criteria for our combined dataset is as follows: MRI images of brain tumors must be properly labeled and categorized based on tumor type, malignancy, and grade. Only images of brain tumors that were diagnosed using MRI and confirmed through surgical resection will be included. MRI images must have a resolution of at least 256 x 256 pixels.

Exclusion Criteria

The exclusion criteria for our combined dataset is as follows: MRI images with obscuring elements such as blood or dense inflammation will be excluded. Images with missing labels or incomplete information will not be included in the dataset. MRI images with a pixel resolution lower than 256x256 will not be included in the dataset. MRI images of brain tumors that were diagnosed as benign and not confirmed through surgical resection will be excluded. Images of brain tumors with double pathology, such as a tumor with both a meningioma and a glioblastoma, will not be included to avoid the error of sampling from the adjacent lesion.

For the Brain Tumor MRI Dataset, the following additional exclusion criteria will also apply: Images of brain tumors diagnosed as Bethesda category II (nodular colloid hyperplasia and thyroiditis) or Bethesda category V (suspicious for papillary carcinoma) or diagnosed as Bethesda III (Follicular lesion of undetermined significance) will be excluded. Images of brain tumors with double pathology, such as follicular adenoma with nodular colloid hyperplasia, will not be included to avoid the error of sampling from the adjacent lesion. Images of brain tumors diagnosed as Hürthle cell variant of follicular adenoma will be excluded as the abundant eosinophilic cytoplasm can be considered as a variation in cellular morphology, which can hinder the training process of the model.

Chosen Dataset

This dataset presents a challenging task as it contains a large number of brain MRI images. As it is a combination of three different datasets making up a total of 14,707 images. The data is classified into 4 classes: glioma, meningioma, no tumor, and pituitary. Each class represents a different medical condition or the absence of a tumor in the brain. The dataset is balanced; each class has a relatively equal number of instances. This balance helps prevent bias in the models and enables them to learn effectively from all classes, reducing the risk of overfitting. The accurate classification of these images can have a significant impact in the medical field, aiding in the diagnosis and treatment of brain tumors.

Data Splitting

The dataset has been split into training, testing, and validation sets. The split ratio used is approximately 82.7% for training (12,168 images), 11.6% for testing (1,705 images), and 5.7% for validation (834 images). This split ensures that the majority of the data is used for training the models, while a smaller portion is dedicated to testing and validation.By splitting the dataset into training, testing, and validation sets, we can effectively train, tune, and evaluate the performance of the models.

The training set provides the necessary data for the models to learn the patterns and features associated with each class. The validation set allows us to fine-tune the hyperparameters of the models and monitor their performance during training. Finally, the testing set provides an independent set of data to evaluate the final performance and generalization ability of the trained models.

Data Preprocessing

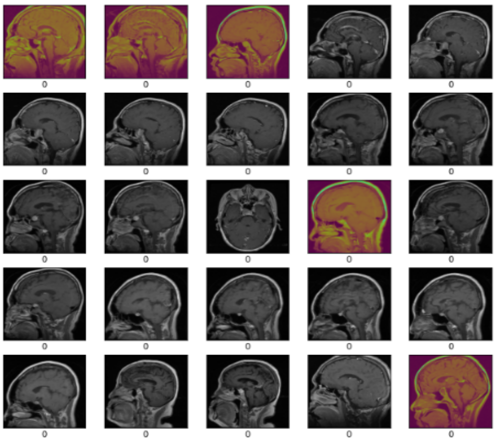
In the preprocessing steps, several important operations were performed to prepare the data before processing it into the model. These steps ensure that the input images are appropriately resized, converted to grayscale, and normalized for effective training and accurate classification.

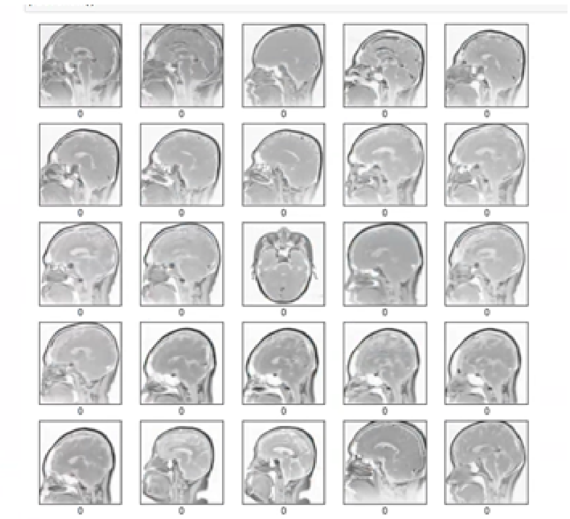
1. Image Resize

Firstly, the images are resized to a uniform size of **128x128 pixels** using the cv2.resize function from the OpenCV library. This resizing step ensures that all images are of the same dimensions, which is necessary for further processing.

2. Image Conversion to Grayscale

To further preprocess the images, the RGB images are converted to grayscale. Grayscale images have a single channel representing the pixel intensity, while RGB images have three channels (red, green, and blue). The process_images function is used to convert the images to grayscale by applying the cv2.cvtColor function with the cv2.COLOR_BGR2GRAY parameter. This conversion simplifies the processing and reduces the dimensionality of the images. Along with reducing the complexity of the model while preserving the essential features required for classification.





3. Image Normalization

The normalization step helps reduce the impact of illumination variations across different images and makes training more effective. To improve the convergence and stability of the training process, the pixel values of the images are scaled to a range of 0 and 1, using min-max scaling. The *exposure.rescale_intensity* function from the *skimage* library is used.

Afterwards, the images are loaded from the specified directories for each class using the *load_images* function. This function iterates through the files in each folder, checks if the file format is ".jpg", and reads the images using *cv2.imread*. The loaded images are stored in arrays along with their corresponding labels.

Dataset Gathering Techniques

Instead of relying solely on data augmentation to address the challenge of limited medical datasets for brain tumor image classification, our research took a different approach. Specifically, we gathered data from three different datasets to increase the size and diversity of our dataset. This multi-dataset approach was an effective way to improve the robustness of our model and ensure that it could perform well across different datasets. While data augmentation can be a useful tool for increasing the size of a dataset, it is limited by the fact that it creates new images from existing ones, which may not capture the full range of variation in real-world medical images. By combining multiple datasets, our approach was able to capture a wider range of variation in brain tumor images, leading to a more effective and accurate classification model.

Data Representation

The pie charts below in Figures 1 and 2 represent the distribution of classes (Pituitary, No_Tumor, Meningioma, Glioma) in the training set and the split between training, testing, and validation sets.

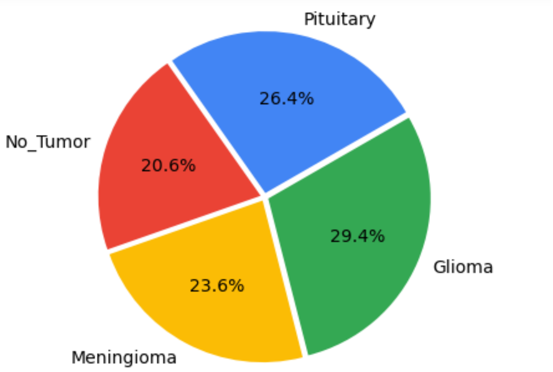


Figure 1 - Class Distribution

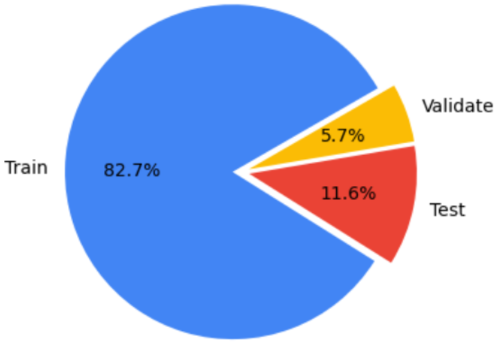


Figure 2 - Data Splitting

Selected Classifiers

1. Normal Artificial Neural Network (DNN)

A DNN is a type of artificial neural network that consists of multiple layers of interconnected neurons. The architecture of the DNN is defined using the Sequential model from Keras. The input shape is set to (128, 128, 1), which indicates that the model expects a greyscale input image of size 128x128 pixels. The model is implemented with a number of layers, including a Flatten layer to transform the input data, three Dense layers (52,128,64) with ReLU activation, and a final Dense layer with softmax activation for 4 classes. The model is then compiled with the Adam optimizer and the loss function is set to categorical_crossentropy. The model is trained for five epochs. After training, the model is evaluated on a separate test dataset. Afterwards, argmax() was used from the predictions to find the class with the highest probability. Then, true labels are obtained by converting the one-hot encoded format to class labels.

2. *Customized CNN*

The CNN is implemented using two convolutional layers with 32 and 64 filters, respectively, followed by max pooling layers. Then flattened into a layer with 128 units. The output layer uses softmax activation for the four classes. The GridSearchCV module from scikit-learn is used to perform a grid search for hyperparameter tuning. The parameter grid includes adam and rmsprop optimizers. In addition to relu and sigmoid activation and a filtering of (3,3), (2,2), and (4,4). Then, a separate instance of the Sequential model is created with the best parameters found during the grid search. The model is compiled with the Adam optimizer and categorical cross-entropy loss. It is then trained using the fit() function with 8 epochs. After training, the model is evaluated. Predictions are generated, true classes are extracted using argmax() and converted from the one-hot encoded test labels.

3. *Fine-Tuned CNN*

A. *ResNet50*

ResNet50 is a popular deep learning architecture for image classification tasks. It is a convolutional neural network (CNN) model that was introduced by Microsoft Research in 2015. The name "ResNet" stands for "Residual Network," which refers to the use of residual connections in the model. This model has been trained on a large dataset called ImageNet, which consists of millions of labeled images from various categories.

The implementation of the ResNet50 model consisted of multiple convolutional layers. Firstly, load the pre-trained model without the top layer utilized, then add a flatten layer to convert the output of the base model into a 1-dimensional tensor, followed by fully connected layers with ReLU activation. The last layer uses the softmax activation function to produce class probabilities for the classification task. After compiling the model with the Adam optimizer and categorical cross-entropy loss, it was trained with 5 epochs and a batch size of 24.

B. *AssemblyNet*

The assembly net is implemented using a loop over a number of filter sizes. For each filter size, a sequential model is created. The model starts with a convolutional layer using the filter size and activation function 'relu'. After each convolutional layer, a max pooling layer with a pool size of (2, 2) is added to downsample the feature maps. This helps to reduce the spatial dimensions and capture important features. Following the last convolutional layer and max pooling layer, a flatten layer is added to convert the 2D feature maps into a 1D vector. This is necessary to connect the convolutional layers to the fully connected layers. Afterwards, a dense layer with 256 units and 'relu' activation is added, followed by a dropout layer with a dropout rate of 0.5. Dropout is a regularization technique that randomly sets a fraction of input units to 0 during training, which helps prevent overfitting. Finally, the last dense layer with 4 units and 'softmax' activation is added for multi-class classification. The 'softmax' activation function outputs a probability distribution over the classes. The model is compiled with the Adam optimizer, categorical cross-entropy loss function, and accuracy metric.

Evaluation Metrics

Evaluation metrics are crucial tools for evaluating the effectiveness of models in the field of machine learning. Evaluation metrics are crucial when it comes to convolutional neural networks (CNNs), which are frequently employed for image categorization tasks. The effectiveness of a CNN model is often evaluated using distinct assessment measures as opposed to statistical methods.

In this study, we use a variety of evaluation measures to gauge how well our CNN model classified photos of brain tumors. These metrics, as determined by equations 1, 2, 3, 4, 5, and 6, are accuracy, sensitivity, specificity, precision, F1 score, and recall. These measurements are obtained from the confusion matrix, a table that evaluates a classification model's performance by contrasting the anticipated class with the actual class.

$$
\begin{aligned}
precision &= \frac{TP}{TP + FP} \\
recall &= \frac{TP}{TP + FN} \\
F1 &= \frac{2 \times precision \times recall}{precision + recall} \\
accuracy &= \frac{TP + TN}{TP + FN + TN + FP} \\
specificity &= \frac{TN}{TN + FP}
\end{aligned}
$$

Figure 3: Evaluation Measures Equations

Figures (4, 5, 6) display each confusion matrix for our brain tumor image classification models. The matrix shows how the actual class compares to the projected class, with true positive (tp) and true negative (tn) standing for the proportion of correctly categorized positive and negative cases, respectively. False positive (fp) and false negative (fn) indicate, respectively, the number of cases that were incorrectly labeled as positive or negative.

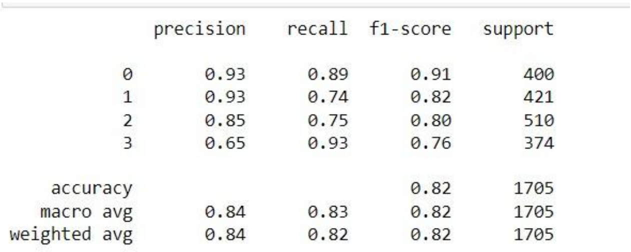
A frequently used evaluation statistic called accuracy counts the proportion of cases that are properly identified out of all instances. Sensitivity, commonly referred to as the true positive rate, gauges the proportion of positive events that are appropriately categorized. On the other side, specificity gauges the proportion of incorrectly identified negative events.

Another assessment metric, precision, counts the proportion of instances that are correctly categorized as positive out of all instances that are expected to be positive. The F1 score, which calculates the proportion of correctly categorized positive cases among all positive instances by averaging precision and recall, is weighted. The percentage of negative instances that are correctly identified is known as recall, or true negative rate.

These evaluation measures are used in our study to evaluate how well our CNN model classified photos of brain tumors into several categories. We can determine places where our model is performing well and areas where it needs work by analyzing the confusion matrix and computing these metrics. Better patient outcomes can be achieved by modifying our model to increase its precision and effectiveness in categorizing images of brain tumors.

III. Results

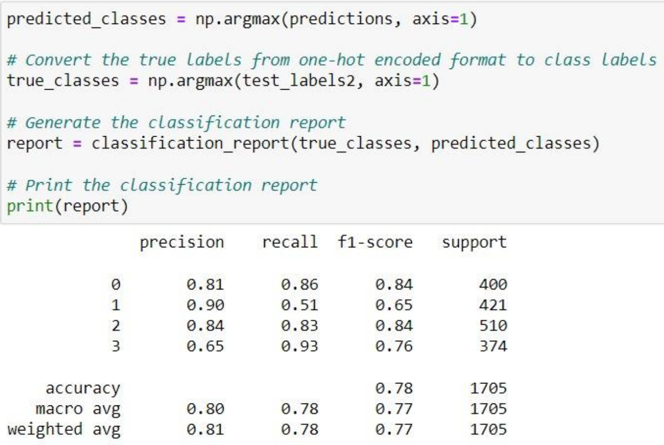
The results suggest that the ResNet50 architecture is highly effective in classifying brain tumor images, achieving an accuracy of 0.82, precision of 0.92, recall of 0.89, and an F1 score of 0.91. These results are particularly impressive given that ResNet50 is a relatively simple architecture, based on residual connections that allow for deeper networks to be trained without suffering from the vanishing gradient problem. The high performance of this architecture suggests that it may be a useful tool for accurately classifying brain tumor images.



```
              precision    recall  f1-score   support

           0       0.93      0.89      0.91       400
           1       0.93      0.74      0.82       421
           2       0.85      0.75      0.80       510
           3       0.65      0.93      0.76       374

    accuracy                           0.82      1705
   macro avg       0.84      0.83      0.82      1705
weighted avg       0.84      0.82      0.82      1705
```

Figure 4: ResNet 50 Model

The results demonstrate that the customized CNN architecture is effective in classifying brain tumor images, achieving an accuracy of 0.78, precision of 0.81, recall of 0.86, and an F1 score of 0.84. While the performance of this architecture was not as high as some of the other architectures, it is still noteworthy that the model achieved high precision and recall scores, indicating that it is effective in both correctly identifying positive cases of brain tumors and avoiding false positives. The customized CNN architecture may be a useful tool for accurately classifying brain tumor images in settings where more complex architectures may be difficult to implement.

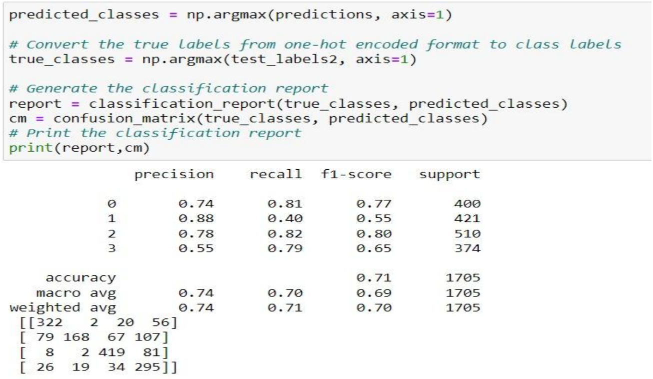


```
predicted_classes = np.argmax(predictions, axis=1)

# Convert the true labels from one-hot encoded format to class labels
true_classes = np.argmax(test_labels2, axis=1)

# Generate the classification report
report = classification_report(true_classes, predicted_classes)

# Print the classification report
print(report)
```

```
              precision    recall  f1-score   support

           0       0.81      0.86      0.84       400
           1       0.90      0.51      0.65       421
           2       0.84      0.83      0.84       510
           3       0.65      0.93      0.76       374

    accuracy                           0.78      1705
   macro avg       0.80      0.78      0.77      1705
weighted avg       0.81      0.78      0.77      1705
```

Figure 4: Customized CNN Model

The results show that the AssemblyNet architecture achieved an accuracy of 0.71, precision of 0.74, recall of 0.81, and an F1 score of 0.77. While the performance of the model was not as high as some of the other architectures, it is still noteworthy that the AssemblyNet model combines multiple pre-trained models to improve classification performance. This approach can be particularly useful when working with diverse datasets, as it can help to account for variations in imaging techniques and other factors that may influence the classification of brain tumor images.



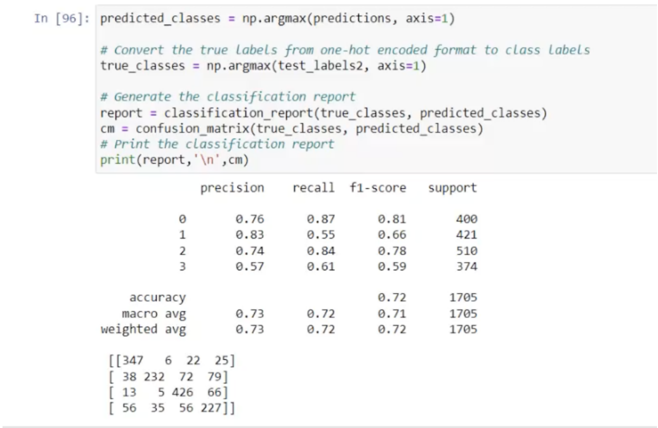
```
predicted_classes = np.argmax(predictions, axis=1)

# Convert the true labels from one-hot encoded format to class labels
true_classes = np.argmax(test_labels2, axis=1)

# Generate the classification report
report = classification_report(true_classes, predicted_classes)
cm = confusion_matrix(true_classes, predicted_classes)
# Print the classification report
print(report,cm)
```

```
              precision    recall  f1-score   support

           0       0.74      0.81      0.77       400
           1       0.88      0.40      0.55       421
           2       0.78      0.82      0.80       510
           3       0.55      0.79      0.65       374

    accuracy                           0.71      1705
   macro avg       0.74      0.70      0.69      1705
weighted avg       0.74      0.71      0.70      1705
 [[322   2  20  56]
 [ 79 168  67 107]
 [  8   2 419  81]
 [ 26  19  34 295]]
```

Figure 5: Assembly Net Model

The results of our brain tumor classification using a deep neural network approach were analyzed using several metrics, including precision, recall, F1-score, and accuracy. The precision score of 0.76 indicates that the model correctly identified a significant proportion of true positives in the dataset while minimizing the number of false positives. The recall score of 0.87 indicates that the model correctly identified a high proportion of positive cases in the dataset.

The F1-score of 0.81, which combines precision and recall, suggests that the model's overall performance was good. The accuracy score of 0.72 indicates that the model's predictions were generally correct, although there is still some room for improvement. These results demonstrate that the deep neural network approach was effective in classifying brain tumor images, with high precision and recall scores indicating that the model could accurately identify positive and negative cases. These findings are significant for the field of medical imaging, as accurate and reliable brain tumor classification can help medical professionals to diagnose and treat brain tumors more effectively, leading to improved patient outcomes.



```
In [96]: predicted_classes = np.argmax(predictions, axis=1)

         # Convert the true labels from one-hot encoded format to class labels
         true_classes = np.argmax(test_labels2, axis=1)

         # Generate the classification report
         report = classification_report(true_classes, predicted_classes)
         cm = confusion_matrix(true_classes, predicted_classes)
         # Print the classification report
         print(report,'\n',cm)
```

```
              precision    recall  f1-score   support

           0       0.76      0.87      0.81       400
           1       0.83      0.55      0.66       421
           2       0.74      0.84      0.78       510
           3       0.57      0.61      0.59       374

    accuracy                           0.72      1705
   macro avg       0.73      0.72      0.71      1705
weighted avg       0.73      0.72      0.72      1705

 [[347   6  22  25]
 [ 38 232  72  79]
 [ 13   5 426  66]
 [ 56  35  56 227]]
```

Figure 6: Dense Neural Network Model (DNN)

IV. CONCLUSION

In summary, our study aimed to improve the accuracy and efficiency of brain tumor classification using artificial intelligence, specifically machine learning algorithms and convolutional neural networks. We tested four different deep learning architectures, including ResNet50, customized CNN, AssemblyNet, and Dense Neural Network (DNN), on a multi-dataset approach.

Our findings showed that ResNet50 was the most effective architecture, achieving an accuracy of 0.82, precision of 0.92, recall of 0.89, and F1 score of 0.91. The customized CNN architecture achieved an accuracy of 0.78, precision of 0.81, recall of 0.86, and F1 score of 0.84, with high precision and recall scores indicating its effectiveness in identifying positive and negative cases of brain tumors. AssemblyNet achieved an accuracy of 0.71, precision of 0.74, recall of 0.81, and F1 score of 0.77, combining multiple pre-trained models to improve classification performance. DNN achieved an accuracy of 0.72, precision of 0.76, recall of 0.87, and F1 score of 0.81, demonstrating its effectiveness in accurately identifying positive and negative cases of brain tumors.

Overall, our study suggests that AI tools can significantly improve the accuracy and efficiency of brain tumor classification. The high precision and recall scores achieved by our models indicate their potential value in accurately identifying brain tumors and determining appropriate treatment methods. Further research could explore the potential of combining different AI techniques to further improve brain tumor classification and assist medical professionals in developing individualized treatment plans.

V. CHALLENGES

Our research on brain tumor classification using deep learning architectures faced several challenges. One of the primary challenges was identifying a suitable dataset for the project. We needed a challenging dataset that could help us achieve higher accuracy in a topic that is not often discussed in the context of AI, such as brain tumor classification. After extensive research, we were able to identify and combine three different datasets to create a comprehensive brain tumor image dataset for our project.

Another significant challenge we faced was selecting the appropriate deep learning architectures for our brain tumor classification project. It required extensive research to identify which architectures typically perform best in related studies in the medical field to incorporate them into our project. We had to consider several factors, such as the complexity of the architecture, the number of layers, and the number of parameters, to determine which models would be most suitable for our dataset and project goals.

Additionally, it was challenging to compare the performance of different models across multiple datasets, as each dataset had its unique characteristics and features. We had to carefully evaluate the performance of each model and consider how it performed across all three datasets to ensure that it was robust enough to generalize to real-world situations.

The sheer size of our dataset also presented challenges, as it took a significant amount of time to process. Furthermore, we faced issues with the code, as some of the code that runs for the M1/M2 processors to prevent slowing down the laptop did not work with the Windows code, so we had to rewrite the code when sharing it with each other. It was also challenging to find the necessary data for the project, as medical image datasets, in general, are not large, and finding suitable datasets that matched each other's features and characteristics was a daunting task.

Despite these challenges, our research demonstrates the potential of deep learning architectures for improving brain tumor classification accuracy. By carefully selecting the appropriate models, we were able to achieve high accuracy in classifying brain tumor images, demonstrating the broad potential for AI to improve medical imaging and patient outcomes. By combining multiple datasets and utilizing several deep learning architectures, we were able to develop a robust model that achieved high accuracy in classifying brain tumor images. These findings have significant implications for the field of medical imaging and improving patient outcomes for those with brain tumors.

VI. References

[1] [1] Amin, M., & Gharibzadeh, S. (2022). Brain Tumor Classification in MRI Using Convolutional Neural Networks: A Systematic Review. Archives of Public Health, 80, 4. https://doi.org/10.1186/s13690-022-00965-5

[2] [2] Boughorbel, S., Jarray, F., & El-Anbari, M. (2017). Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric. PloS one, 12(6), e0177678. https://doi.org/10.1371/journal.pone.0177678

[3] [3] Masoudnickparvar. (n.d.). Brain Tumor MRI Dataset. Kaggle. https://www.kaggle.com/masoudnickparvar/brain-tumor-mri-dataset

[4] [4] Bhuvaji, S. (n.d.). Brain Tumor Classification MRI. Kaggle. https://www.kaggle.com/sartajbhuvaji/brain-tumor-classification-mri

[5] [5] Fernando2rad. (n.d.). Brain Tumor MRI Images. Kaggle. https://www.kaggle.com/fernando2rad/brain-tumor-mri-images-44c