Course: CSCI 5622 **Project final report** **Due: December 13, 2021**

Brain Tumor Radiogenomic Classification

Group members
- MyeongSeon Lee (Myeongseon.Lee@colorado.edu)
- Varun Manjunath (vama5032@colorado.edu)
- Shubham Rathi (shubham.rathi@colorado.edu)

1. Problem space

A brain tumor is the growth of abnormal cells in the tissues of the brain. It can be benign (not cancer) or malignant (cancer) [1]. A malignant brain tumor is an aggressive life-threatening condition and early detection is very important for treatment and also for survival. However, detection of brain tumors is very costly, time-consuming, painful (surgery or biopsy), and in need of specialized human resources such as neuropathologist and radio neurologist. One of the popular techniques to detect brain tumors is Magnetic Resonance Imaging (MRI) that solves some of these issues (like removing pain) addressed above, however other issues, especially specialized human resources, are still to be solved. Another drawback of the MRI approach is that it cannot detect below 3 mm size [2]. Hence, developing new approaches for accurate detection and analysis of Brain tumors is necessary in Medical Science.

Computer science, especially in the area of machine learning, has been drastically developed for the last few decades. Much effort has been made to develop algorithms to detect brain tumors by analyzing MRI scans. There are many different methods for classification of image data and each method has advantages and disadvantages. Most of these methods revolve around the Convolutional Neural Network. In our project, instead of focusing on one only method to develop an algorithm with high performance, we tried several different methods such as K-means clustering, Convolutional Neural Network (CNN), CapsNet, Recurrent Neural Network (RNN) to compare which ones are good for classification of this type of image before focusing on the method for further development.

In our project, we predict MGMT_value(1 denotes the presence and 0 denotes the absence of MGMT). MGMT is a DNA repair enzyme that rescues tumor cells from alkylating agent-induced DNA damage and leads to resistance to chemotherapy with alkylating agents.[3] MGMT promoter promotes production of MGMT protein to repair DNA. When MGMT promoter is methylated, MGMT gene is sliced, MGMT protein expression is decreased, DNA repair activity is reduced, leading to human carcinogenesis. Microscopic genetic changes may manifest as macroscopic morphological changes in the brain tumors that can be detected using magnetic resonance imaging (MRI), which can serve as noninvasive biomarkers for determining methylation of MGMT regulatory regions.

2. Dataset

2.1. Data Source:

https://www.kaggle.com/c/rsna-miccai-brain-tumor-radiogenomic-classification/data

2.2. Contents of the dataset

Our training data set is brain MRI images taken from 582 subjects and, among them, 306 (52.6%) subjects have MGMT_value of 1. Total of 276 (47.4%) subjects have an MGMT_value of 0 and 306 have MGMT_value of 1, so the dataset is well balanced. There are a total of 258641 MRI scans available. The scans were taken in various angles, positions using four different MRI sequences (T1w, T1wCE, T2W, and FLAIR, refer to 2.4). The shape of each scan is 512 x 512 pixels and all pixels will be used as features.

2.3. MRI sequences

- **T1w (T1-weighted)**: The sequence weighting highlights differences in theT1 relaxation time of tissues. T1 relaxation time is a measure of how quickly the net magnetization vector recovers to its ground state, the return of excited nuclei from the high energy state to the low energy or ground state.

- **T1wCE (T1-weighted contrast-enhanced)**: T1w with contrast agent (gadolinium) to enhance the contrast of MRI scans.

- **T2w (T2-weighted)**: The sequence weighting highlights differences in the T2 relaxation time of tissues. T2 is the time it takes for the transverse magnetization vector to decay to 1/e or 37% of its initial magnitude.

- **FLAIR (Fluid attenuated inversion recovery):** A special inversion recovery sequence with a long inversion time. This removes the signal from the cerebrospinal fluid in the resulting images.

2.4. Example Images

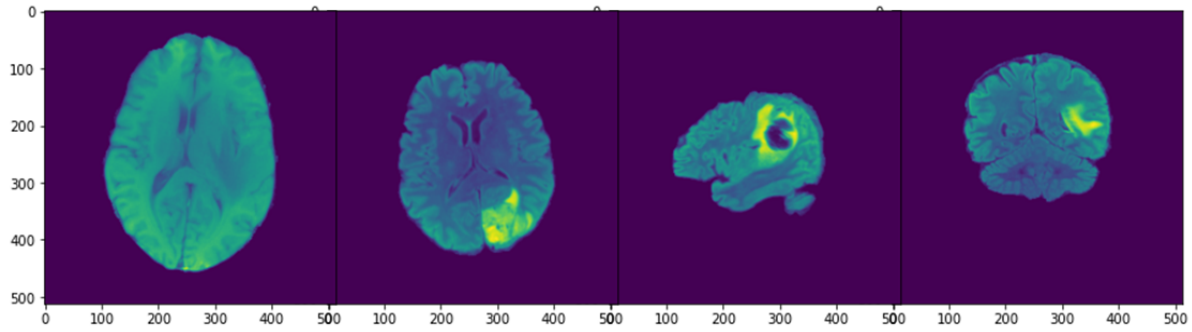

Fig. 1. MRI scans from subjects without (A) or with (B, C and D) a brain tumor.

2.5. Output (MGMT_value)

Our Train_labels.csv file contains the target MGMT_value (presence of methylation on MGMT (O^6-methylguanine-DNA methyltransferase) promoter) for each subject in the training data. A higher MGMT score means more possibility of the presence of the tumor. Therefore, MGMT_value of 1 will represent the patient is likely to have a tumor and otherwise a value of 0.

3. Implementation Flow

3.1 Data Preprocessing

In our data, for each kind of MRI(T1w, T1wCE, T2w, T1wCE), there is a different count for each patient that ranges from 15 to 700. For example, our dataset for a patient has 100 images of type FLAIR, 120 of type T1w, 60 of type T2w, and 150 of type T1wCE. We have taken following measures to prepare our data:

- There were a lot of blank images associated with +ve and -ve output. Those images were not providing any features and introducing noise in our data. So, we filtered out those images.

- CT Scan images are a bunch of continuous images and our data contains those images in sequence. This means that some of the images (that are continuous) will almost not differ at all from the next image. Therefore we decided to take one image out of 3 images from the input and discarded the other two. This solved two of our problems. First loading less images means less memory consumption and second, it avoids overfitting by removing similar kinds of images that don't contain similar features.

- As these all images are taken in a controlled environment, the main part of the image lies in the middle of the image. Therefore, we cropped the side part(taking [80:400, 100:420]), which gave us the image of 320x320. Further, we resize the image to 128x128.

- The dicom image pixels use 16 bits each, that is pixel value ranging from 0 to 65536. After analyzing our data, we found out that most of the image pixels are ranging from 0 to 8000 values. So, we normalized our data by dividing each pixel by 4095.

3.2 Modeling/Training:

Now after data preparation we have training data for each type of MRI image with dimension 128x128. As a part of this project, we aimed to compare different models on the basis of accuracy, f1 score and auc. The next section this contains the metrics for each approach mentioned below :

- *Bag of words:*

In our approach first we used the ORB feature extraction algorithm that extracts keypoints and descriptors from the image.The number of keypoints the algorithm is given is 40 and the

descriptor size is 32. So, the size of an image is 40*32. For each patient there are 4 kinds of scans. We perform ORB feature extraction on all the images in all the 4 types of scans and append the result to a list. We do this for every patient and then combine the lists obtained for all the patients by adding these lists to another list. This bigger list is then flattened. We obtain a set of feature descriptors.Then K-means++ clustering is performed on these feature vectors with the number of clusters set to 2500.Then for every patient we perform the prediction with the help of k-means on his feature vectors. Further, we store the count(number of occurrences of the predicted class) in a histogram of size 2500. We use this histogram as the input data to Support vector machines and MLP algorithms.

- *Long Short Term Memory(LSTM):*

The LSTM models were built with various numbers of layers (3-6 layers) and various numbers of nodes (32-256 nodes) in each layer and the training process was performed with various epochs (50-100 epoches) and learning_rates (1e-2 - 1e-5) to find out the best model. Following is the layer design that gave us the best accuracy and f1 score.

```
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 bidirectional_39 (Bidirecti  (None, 256)              263168
 onal)

 dense_68 (Dense)             (None, 64)               16448

 dense_69 (Dense)             (None, 64)               4160

 dense_70 (Dense)             (None, 1)                65

=================================================================
Total params: 283,841
Trainable params: 283,841
Non-trainable params: 0
```

- *Convolution Neural Network(CNN):*

We tried two different types of CNN models to predict the output and evaluated the results. First, we tried the 3d-CNN model. In this, our single input consists of 24 images placed in a bunch followed by Convolutional and pooling layers. So, our input shape is 128x128x24.The width and height of the input shape are the dimensions of the image and the depth is the number of images placed together in a bunch. Second, we implemented a 2d-CNN model. In this, we feeded each image separately to the model. So, our input shape is 128x128. We got better accuracy while using the 2d-CNN model. Fig.2 represents the design of each layer in the 2d-CNN model(that gave us the best accuracy).

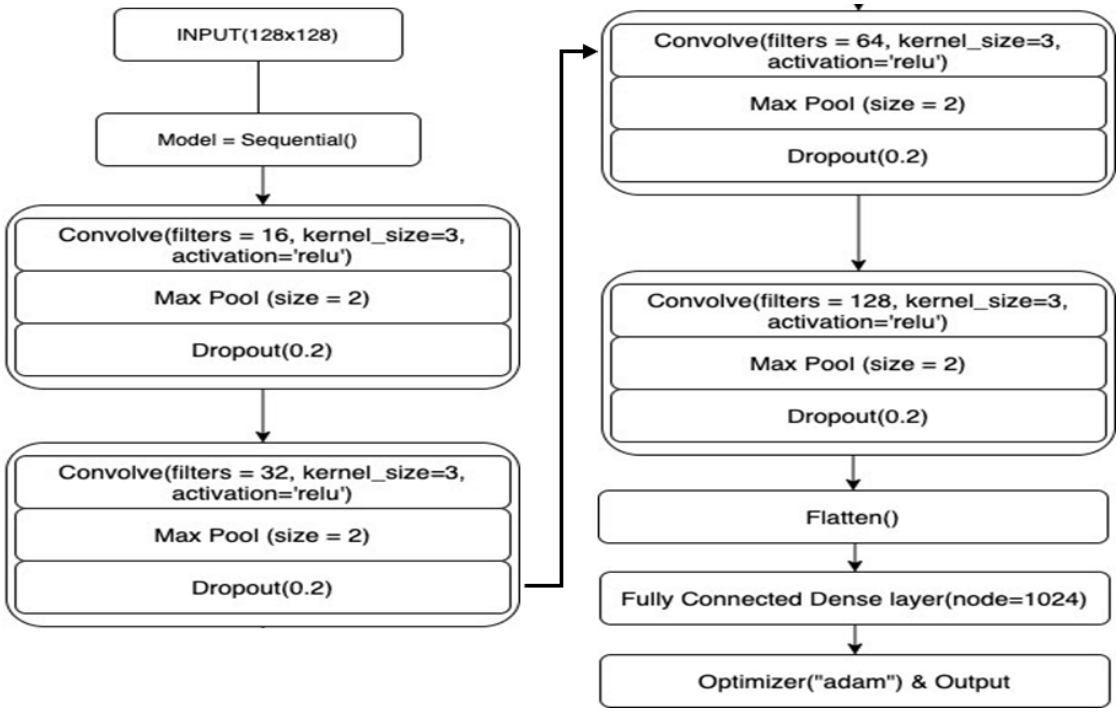

Fig. 2. 2d - CNN Model Layer Design

4. Results

For each model we measured metrics that include accuracy, f1-score and auc. Following are the metrics with best accuracy configuration of each model:

	Accuracy	**F1-score**	**AUC**
Bag of words + SVM(kernel=linear)	0.53	0.60	-
LSTM	0.79	0.78	0.77
CNN(2D-model)	0.86	0.85	0.91
CNN(3D-Model)	0.55	0.45	0.74

Note: These metrics are denoted the best configuration for each model observed by us.

In addition to this, we also measured accuracy and loss for every epoch. Fig.3 and Fig.4 represent the plots for every model with the best configuration observed by us.

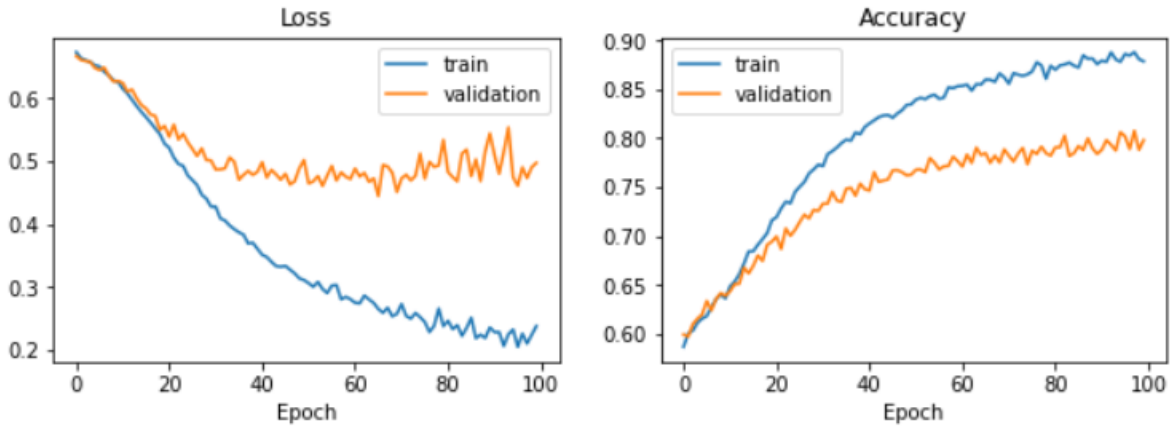

Fig 3. Loss and accuracy for LSTM

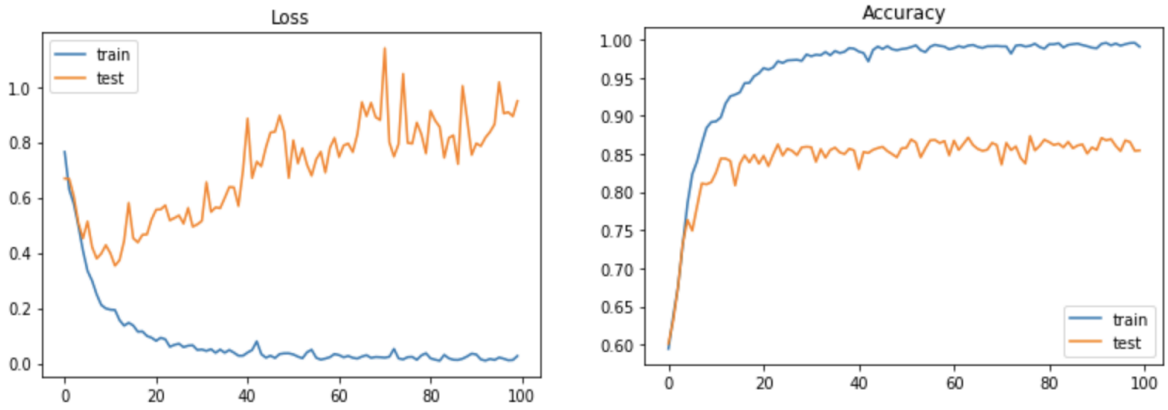

Fig 4. Loss and accuracy for CNN(2d-model)

5. Discussion

All the above methods show us their learning behavior with our dataset. We tried four different kinds of models: SVM, LSTM, CNN(2d), CNN(3d). From the above results, we can conclude that the 2-d CNN model outperformed all other models. We got better accuracy and f1 score for the same. Apart from this, the results for these models have surprised us a bit. Before implementation, we had expected better results from the 3d-CNN model but it did not come out that well. Maybe, we can explore more ways of combining the images to 3d and evaluate our model. On the other hand, we did not expect 79% accuracy from LSTM.

As a course project we were restricted by the time and resources. For given resources, one can try processing each image to train the model with different contrasts and sizes of each image. Currently, we used 128x128 image size from 512x512 for 2d-CNN and LSTM. One can try out 224x224 or 256x256 as well. Another model Capsule Neural Net, which had planned to do but could not implement it due to time and resource constraints. Capsule Neural Net is an implementation based on CNN itself that has been promising in giving good results for image classification tasks.

Besides the model that we implemented(or planned to), there are a lot of other models that are being developed and designed specifically for Biological classification tasks such as U-net. People have claimed that they have got better results by using these models for the tasks related to Biology. Given time and resources, one can explore these options and improve the accuracy of this task. As mentioned earlier, this will not only help in understanding machine learning but also have a potential to contribute to detection of Brain Tumor Cancer in an early stage.

6. References

1. https://www.cancer.gov/publications/dictionaries/cancer-terms/def/brain-tumor
2. https://ieeexplore.ieee.org/document/8748692
3. Wick W, Weller M, van den Bent M, *et al*. MGMT testing – the challenges for biomarker-based glioma treatment. *Nat Rev Neurol* 2014;10:372-85
4. Soejima H, Zhao W, Mukai T. Epigenetic silencing of the MGMT gene in cancer. *Biochem Cell Biol.* 2005;83(4):429–437.
5. Automatic Prediction of MGMT Status in Glioblastoma via Deep Learning-Based MR Image Analysis
6. [2104.01149] Glioblastoma Multiforme Prognosis: MRI Missing Modality Generation, Segmentation and Radiogenomic Survival Prediction
7. Brain tumor segmentation based on deep learning and an attention mechanism using MRI multi-modalities brain images | Scientific Reports
8. Deep-Learning Convolutional Neural Networks Accurately Classify Genetic Mutations in Gliomas