Brain Tumor Image Classification Using Support Vector Machine

Akangsha Deb[1]   Tanu Tiwari[1]   Sanjana Raj Kumari[1], Jaydeep Swarnakar*[1]

[1]Department of Electronics & Communication Engineering, School of Technology, North-Eastern Hill University, Shillong-793022, Meghalaya, India.

akangshadeb96@gmail.com, tanukt20@gmail.com, sanjanarajk14@gmail.com, *jaydeepswarnakar@gmail.com

ABSTRACT

This paper is focused on developing a model for classifying MRI images of brain tumors including Glioma Tumor, Meningioma Tumor, Pituitary Tumor and no tumor. The dataset has been collected from Kaggle. Then, it has been organized and uploaded to GitHub. The different libraries of Python namely Matplotlib, NumPy, and Sci-Kit Learn have been capitalized in this work. Google Collaboratory has been used for the execution environment with a limited storage requirement of 108 GB. The images are cloned from GitHub repository named "Brain-Tumor" to Google Collaboratory. This repository contains a dataset corresponding the images of all the specified types of tumors. The Support Vector Machine algorithm has been used for classification. The accuracy of the proposed model ranges in between 84% to 92%.

Keywords: Google Collaboratory, Image Classification, MRI images, Machine Learning, Repository, Support Vector Machine (SVM).

I. INTRODUCTION

A brain tumor is an abnormal cell growth in or around the brain. It is classified in two categories-primary and secondary. Primary tumors affect nearby structures, while secondary tumors spread from other body parts. There are various types of primary tumors, including noncancerous (benign) and cancerous (malignant) ones. Brain cancers exhibit rapid growth, invading and destroying brain tissue. The tumors are of different sizes and some of the tumors are detected at early stage due to noticeable symptoms. Other tumors grow extensively before discovery. Inactive brain regions may delay to develop symptoms if the tumor originates there. The detection of tumors necessitate treatment which depends on tumor's type, size, and location. Common treatments include surgery and radiation therapy. In this work, a model has been developed to detect certain types of tumors from an individual's MRI scans. The model is easily deployable in scanners that will produce MRI images of the relevant tumor. The motivation of the proposed work is to automate the prediction of tumor (if any) with substantial accuracy through eliminating the manual interventions. The work is based on image classification. Image classification is the field in which similar images are provided under a label, and then learning is made on the basis of the group of images provided [1]. The proposed work uses MRI images for brain tumor prediction. We have created a model titled Brain Tumor Image classification using support vector machine algorithm which will ask user for the MRI images and accordingly it will predict whether that person is having tumor or not. If tumor is detected, then it will be classified under the three classes which are Glioma tumor, Meningioma tumor and Pituitary tumor. By deploying the model in scanners, it aims to accurately identify the corresponding tumor type for a given MRI image, reducing the risk of false predictions or misdiagnosis. The model primarily relies on image classification techniques. Image classification involves grouping similar images under specific labels and training a model based on these image sets. Several techniques can be used for image classification including maximum likelihood, minimum-distance, principal components, support vector machine (SVM) [2]. iso cluster, k-means, and convolutional neural networks (CNN) [3-6]. In this model, the Support Vector Machine algorithm is utilized due to its ability to handle extreme objects in a group known as support vectors. Support-vector networks are supervised learning models used in machine learning for classification and regression analysis. They employ specific learning algorithms to analyze data.

- SVMs were developed by Vladimir Vapnik at AT&T Bell Laboratories in 1993. [7,8]

- SVMs not only excel at linear classification but also possess the capability to efficiently handle non-linear classification through the utilization of the kernel trick. This technique allows SVMs to implicitly map input data into high-dimensional feature spaces.
- The Support Vector Machine (SVM) algorithm operates based on the concept of support vectors. These support vectors represent the objects within a group that are least similar to the rest of the group.

The term "vectors" is used because these objects are represented in a two-dimensional chart, akin to vectors in a two-dimensional space.

A. **Machine Learning Approach Used**

As previously mentioned, the utilized Machine Learning Algorithm or methodology is Support Vector Machine (SVM).

The SVM training algorithm constructs a model that assigns new examples to specific categories. It functions as a non-probabilistic binary linear classifier, although techniques like Platt scaling can enable SVM to be used in a probabilistic classification setting.

- SVM maps training examples to points in space to maximize the gap between the two categories.
- Subsequently, new examples are mapped into the same space and classified based on which side of the gap they fall on.

Support vector machine algorithm ineffectively categorizes unlabeled data and has gained widespread usage in various industrial applications, making it one of the most commonly employed clustering algorithms.
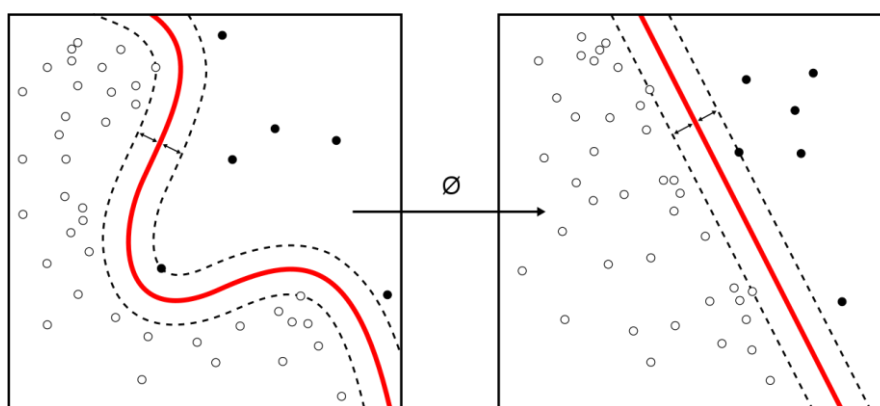

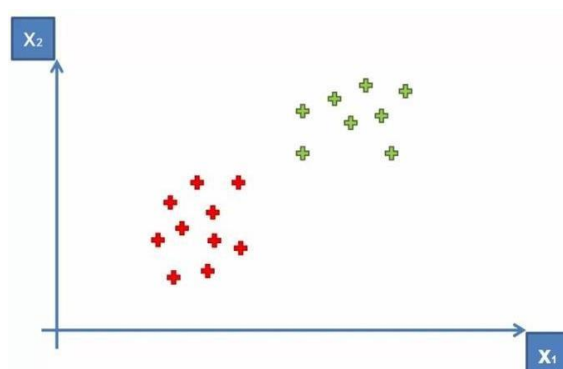**Figure 1: Non-Linear Classification**   **Figure 2: Linear Classification**

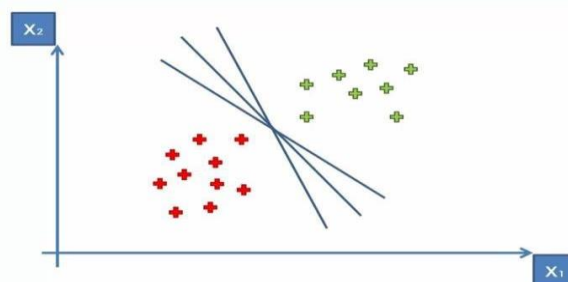**Figure 3: Representation of 2 groups in 2D space**

Figure 4: Fitting a line between boundaries between two groups

After representing the groups on a chart, boundaries are established based on the concept of Maximum Margin. Maximum margin refers to the largest sum of distances between the objects within a group and the boundary. Support vectors within the group play a crucial role in determining this Maximum Margin. The diagrams below visually depict this phenomenon. The Maximum Margin Hyperplane is a plane that effectively separates the two groups while maximizing the margin between them.
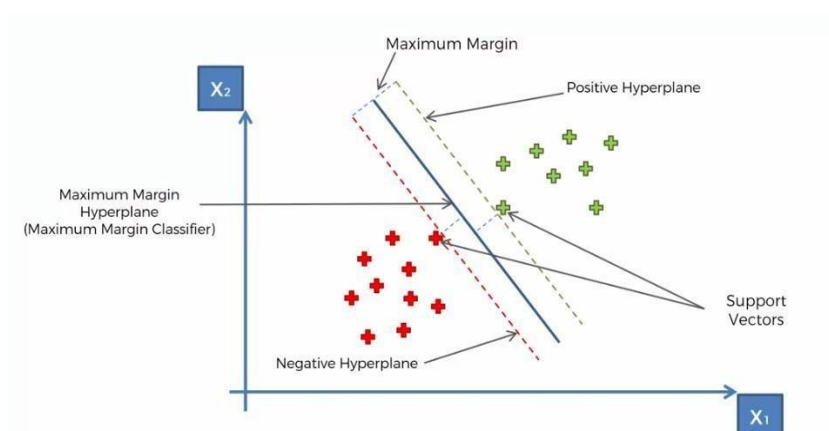

Figure 5: Hyperplane Fitting and Support Vectors

The actual implementation of SVMs is significantly more complex and involves a substantial amount of mathematical calculations.

To simplify the laborious task of performing these calculations, the Sci-Kit learn library offers a SVM module that provides a pre-built model. This library reduces the burden of manual mathematical computations and streamlines the implementation process.

II. METHODOLOGY

The following section provides a description of the Model's methodology-

- Firstly abundant number of MRI images of patients are collected from the internet using various sources such as Kaggle, Google search etc. There are around 700 images under each label of data with a total of 2963 images.
- A repository is created in GitHub and images are uploaded.
- These images are then organized and numbered and put in various folders labelled with the name of the type of images of the tumor.
- They are properly scaled and preprocessed in the code using CV2 and NumPy Python libraries.
- After pre-processing of images, a temporary data set is generated and using Pickle library data is stored in the same directory.
- The Dataset is used further to extract the preprocessed image data for further splitting in test set and train set with a test size of 11 percent.
- The train set is fed to the supervised machine learning algorithm called from Scikit learn library. As the images are highly complex the model will consume some time to build learning on the correlation of the images in the provided group.

- After the Model is successfully generated, the accuracy, confusion matrix and the F1 score of the Model will be displayed.
- A test set of images will now be loaded into the model, which is now prepared for testing.
- .Final Deployment is done and type of brain tumor would be detected.

The brief description of working of the Model is given along with the flowchart-

The image dataset for the model is collected from Kaggle. The dataset comprises X-Ray images of different tumor types, including Glioma Tumor, Meningioma Tumor, Pituitary Tumor, and a set of images representing the absence of tumors. With approximately 700 images per set and hence total of 2963 images, the dataset is properly scaled and preprocessed using CV2 and NumPy Python libraries [9, 10]. The preprocessed images are temporarily stored in a pickle file within the same directory. Next, the dataset is split into a training set and testing set, with a test size of 11%. The training set is fed into the SVM algorithm, which takes some time to build correlations between the complex images in the dataset. Once the model is successfully generated, it displays the accuracy, confusion matrix and F1 score [11-13] of the current learning. The model is also deployed to interact with users who can input MRI images for prediction and the results will be shown accordingly.

The dataset required for the model consists of processed MRI images [14,15] depicting the Glioma Tumor, Meningioma Tumor, and Pituitary Tumor. Additionally, there is a fourth set of images representing the absence of tumors.
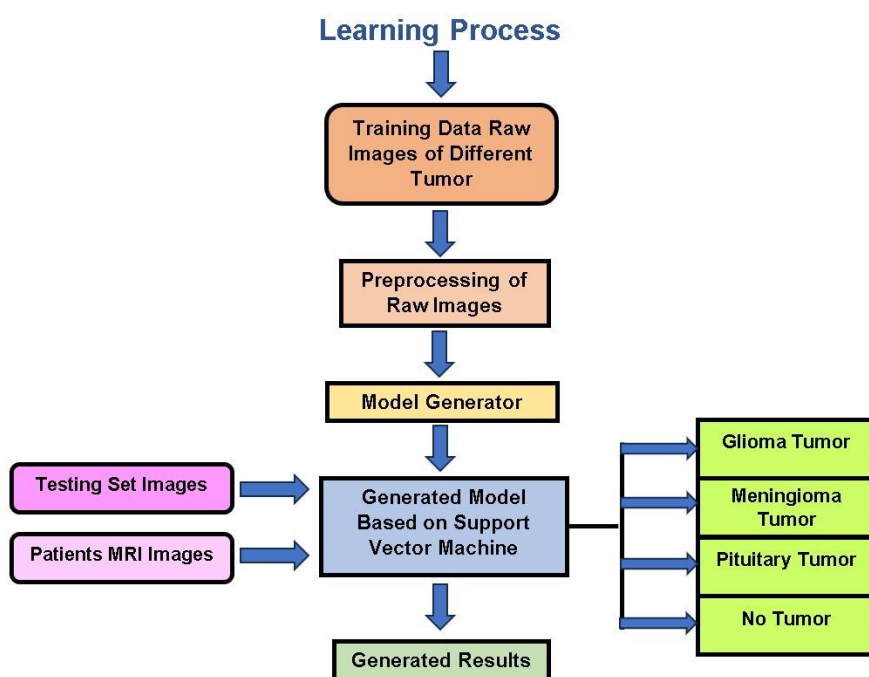
Figure 6: Flowchart of the Model

III. SIMULATIONS AND RESULTS

Performance Matrix of the model gives three output result variables which tells the performance of the project namely Accuracy, F1 score, and Confusion Matrix.

A. **Accuracy**

Accuracy is a metric that quantifies the proportion of correct predictions out of the total number of test cases. In our scenario, the accuracy of the model ranges from 84% to 90%, depending on the specific iteration of the model run.

Figure 7: Accuracy Result of the Model

B. **Confusion Matrix**

Accuracy can be misleading when the number of observations are unequal in a given dataset which is the case in our problem. The images for the 4 sets are not of same amount. In these kinds of cases accuracy is not able to summarize the model performance correctly. To overcome this, confusion matrix comes into picture. A confusion matrix is a technique for summarizing the performance of a classification algorithm. A confusion matrix is used to describe the performance of a classification mode. Confusion matrix is the plot of the predicted values against actual values. The 4 major parameters in a confusion matrix are-True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) parameters make up the confusion matrix. Detailed description of four parameters is given below:

i)True Positive (TP):
True Positive represents the number of instances that are correctly predicted as positive by the classification model. In other words, it is the count of the cases where the actual class is positive, and the model also predicted it correctly as positive.

ii)True Negative (TN):
True Negative represents the number of instances that are correctly predicted as negative by the classification model. It is the count of the cases where the actual class is negative, and the model correctly predicted it as negative.

iii)False Positive (FP):
False Positive represents the number of instances that are incorrectly predicted as positive by the classification model. It is the count of cases where the actual class is negative, but the model predicted it as positive.

iv)False Negative (FN):
False Negative represents the number of instances that are incorrectly predicted as negative by the classification model. It is the count of cases where the actual class is positive, but the model predicted it as negative.

Figure 8: Confusion matrix of the model

C. **F1 Score**

To define F1 score we need to calculate two perimeters Precision and Recall.
i)Precision:

Precision is a measure of how accurately the model predicts the positive class. It is calculated as the ratio of true positives (TP) to the sum of true positives and false positives (TP + FP). In other words, precision quantifies the percentage of positive predictions that are actually correct. A high precision indicates that the model has a low rate of false positives.

Precision = TP / (TP + FP)

ii)Recall:

Recall, also known as sensitivity or true positive rate, measures the ability of the model to correctly identify positive instances. It is calculated as the ratio of true positives (TP) to the sum of true positives and false negatives (TP + FN). Recall captures the percentage of positive instances that the model correctly classified. A high recall indicates that the model has a low rate of false negatives.

Recall = TP / (TP + FN)

The F1 score combines precision and recall into a single value that reflects the balance between these two measures. It is the harmonic mean of precision and recall, calculated as:

F1 Score = 2 * (Precision * Recall) / (Precision + Recall)

The F1 score ranges from 0 to 1, where a score of 1 represents a perfect model, and a score of 0 indicates poor performance. By considering both precision and recall, the F1 score provides a comprehensive evaluation of the model's ability to make accurate positive predictions while minimizing false positives and false negatives.

```
[ ] #for calculating  f1_score
    print("f1_score =" ,f1_score(testing_labels,y_pred, average='macro'))

    f1_score = 0.8496080816945232
```

Figure 9: F1 score result of the model

IV. CONCLUSION

In this paper, the proposed model focuses on classifying different types of brain tumor using MRI images. In the traditional approach, doctors manually examine MRI scans and provide their diagnoses, which introduces a high possibility of human errors. To address this, the proposed model automates the examination of MRI scans using machine learning techniques. The results of the model demonstrate that automating the examination of MRI scans using the Support Vector Machine approach achieved an average accuracy of 87 percent. To further enhance accuracy, the model can be tested using alternative machine learning approaches or a larger dataset of real-life images. The developed machine learning model could be deployed in real-life MRI scanners in hospitals. The proposed model can provide a preliminary indication of the type of tumor diagnosed and its prediction has shown promising results for preliminary assessments. Additionally, this model can be extended to predict other diseases that require scanning and imaging including fractures, breast cancer cell detection, lung cancer diagnosis and cardiovascular disease assessment. Considering the past COVID-19 pandemic which severely affected the lungs of patients, this model can also perform lung imaging which can differentiate between infected and healthy individuals. Additionally, imaging and prediction methods can help in figuring out the infection stage, allowing for preventive action and possibly saving more lives. By training the proposed model with various cancer images and CT scan images, it has the potential to fully automate the prediction of cancers at earlier stages, resulting in significant life-saving benefits.

REFERENCES

[1] M. Alfonse and A.B.M. Salem, "An automatic classification of brain tumors through MRI using support vector machine," Egyptian Computer Science Journal, vol. 40(3), pp.1110-2586, September 2016.

[2] E.I. Zacharaki, S. Wang, S. Chawla, D.S. Yoo, R. Wolf, E.R. Melhem and C. Davatzikos, "MRI-based classification of brain tumor type and grade using SVM-RFE," in proceedings of IEEE International Symposium on Biomedical Imaging: From Nano to Macro, IEEE, pp.1035-1038, 2009.

[3] M.M. Badža and M.Č. Barjaktarović, "Classification of brain tumors from MRI images using a convolutional neural network," Applied Sciences, vol. 10(6), p.1999, 2020.

[4]   M. Nazir, S. Shakil and K. Khurshid, "Role of deep learning in brain tumor detection and classification," A review. Computerized Medical Imaging and Graphics, vol. 91, p.101940, 2021.

[5]   S. Bakas, M. Reyes, A. Jakab, S. Bauer, M. Rempfler, A. Crimi, R.T. Shinohara, C. Berger, S.M. Ha, M. Rozycki and M. Prastawa, "Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge," arXiv preprint arXiv:1811.02629, 2018.

[6]   R.H. Ramdlon, E.M. Kusumaningtyas and T. Karlita, "Brain tumor classification using MRI images with K-nearest neighbor method," in proceedings of International Electronics Symposium (IES), IEEE, pp.660-667, September 2019.

[7]   J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio and V. Vapnik, "Feature selection for SVMs," Advances in neural information processing systems, p.13, 2000.

[8]   M.O. Stitson, J.A.E. Weston, A. Gammerman, V. Vovk and V. Vapnik, "Theory of support vector machines," University of London, vol. 117(827), pp.188-191, 1996.

[9]   I. Maglogiannis, E. Zafiropoulos and I. Anagnostopoulos, "An intelligent system for automated breast cancer diagnosis and prognosis using SVM based classifiers," Applied intelligence, vol. 30, pp.24-36, 2009.

[10]   Aastha, S. Mishra and S. Mohanty, "Integration of machine learning and IoT for assisting medical experts in brain tumor diagnosis," Smart healthcare analytics: state of the art, pp.133-164, 2022.

[11]   T. Maruyama, N. Hayashi, Y. Sato, S. Hyuga, Y. Wakayama, H. Watanabe, A. Ogura and T. Ogura, "Comparison of medical image classification accuracy among three machine learning methods," Journal of X-ray Science and Technology, vol. 26(6), pp. 885-893, 2018.

[12]   Z.A. Sejuti and M.S. Islam, "An efficient method to classify brain tumor using CNN and SVM," in proceedings of 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), IEEE, pp. 644-648, 2021.

[13]   G. Çınarer and B.G. Emiroğlu, "Classification of brain tumors by machine learning algorithms," in proceedings of 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), IEEE, pp. 1-4, 2019.

[14]   R. Bhattacharjee and M. Chakraborty, "Brain tumor detection from MR images: Image processing, slicing and PCA based reconstruction," in proceedings of Third International Conference on Emerging Applications of Information Technology, IEEE, pp.97-101, 2012.

[15]   S. Krishnapriya and Y. Karuna, "Pre-trained deep learning models for brain MRI image classification," Frontiers in Human Neuroscience, p.17, 2023.