Brain Tumor Classification with Weighted Shallow Convolutional Neural Networks

Lakshay Mehla
Mehta Family School of Data Science and Artificial Intelligence
Indian Institute of Technology Roorkee
Haridwar, India
lakshaymehla613@gmail.com

Chhavi
Mehta Family School of Data Science and Artificial Intelligence
Indian Institute of Technology Roorkee
Haridwar, India
panghalchhavi15@gmail.com

***Abstract*- Brain tumor classification is a very important and crucial element of medical sciences. Classification of brain tumor is a challenging task if we use manpower. In this work, we try to find an architecture that performs very well for the classification task to classify 3 types of brain tumor. Our proposed network is a weighted convolutional network in which we combined different kinds of convolutional operations between the same pair of layers by assigning them weights using popular neural architectural search technique, DARTS. Our results showed this much percent accuracy on the validation dataset as compared to existing pre-trained networks, VGG19, VGG16, Alex Net. Our key observations direct towards using shallow networks to gain satisfactory results for brain tumor classification.**

Keywords—Neural Architecture Search, Brain Tumor Classification, Weighted CNN

I. Introduction

A terrible class of diseases with high fatality rates are brain tumors. A fast and accurate diagnosis is crucial for choosing the best course of action. Classifying brain tumors has always involved skilled individuals visually analyzing medical images. But this method is labor-intensive, subjective, and prone to human error. The field of medical image analysis has seen a revolution because of recent advances in deep learning. CNNs, or convolutional neural networks have proven to be highly efficient at classifying brain tumors. This study investigates the use of deep learning to categorize brain cancers into three main groups: pituitary, glioma, and meningioma. Section II continues with the discussion with the literature of the classification problem. In Section III, we talk about convolutional neural networks and give a brief about a neural architecture search technique, DARTS and how it relates to our work. Section IV contains the details of various experiments performed and observations.

II. Related Work

Medical imaging is an emerging field due to some advantages like better accuracy, consistency, reduced bias, high speed, high scalability and quantification. One of the major problems of medical imaging is classification of brain tumor images. CNNs have been used since a long time in this area. The three explored ways to solve this problem are using meta-heuristic optimization algorithms, fine-tuning pretrained deep neural networks and employing a definite architecture built based on experience.

The three class classification of brain tumor is solved by Kharrat[6] by using genetic algorithm and SVM. The optimal set of texture features are obtained by spatial gray level dependence method which were then fed to an SVM. Genetic algorithm helped in extracting useful set of features.

Swati[7] used pretrained VGG19 and fine-tuned the network block-wise to enhance the classification results. This leads to classification accuracy of 94.82%. Saxena[5] used various networks for transfer learning and achieved highest accuracy of 95% in ResNet.

El-Wahab[4] proposed BTC-fCNN that uses 13 layers involving 3X3, 1X1 convolution layers followed by ReLU activation function. They trained 5 models and the best of them was chosen. The mean accuracy achieved is 93.52%.

Chitnis[2] proposed a Learning by Self Explanation architecture search method. It works like GANs having 2 models, one for predicting and one for validating the predictions. Four-level optimization problem was formed from this method which was solved to get the best architecture. 90.6% accuracy was obtained with 375M parameters.

III. Differentiable Architecture Search (DARTS)

Architectural search has always been a challenging task in deep learning. Finding the optimal architecture requires a lot of experience and trials. Automatic search for the architecture is very useful technique. This can be done by various optimization algorithms such as NASNet, PNAS, Bayesian optimization and reinforcement learning techniques. But, these methods have very high search cost as they take very long time to find the best architecture.

Unlike previous methods which majorly used reinforcement learning, DARTS[3] is a computationally-effective gradient-based neural architecture search algorithm. In the previous methods, they used a discrete search space to search for the architecture from it by using it as a set of states in reinforcement learning, with each state being a possible architecture. Unlike discrete and non-differentiable search space, this algorithm employs a continuous search space. Due to continuous search space being differentiable, they applied gradient descent to find the best possible architecture.

A. Continuous Search Space

The search space consists of a defined number of internal nodes (layers) connected to each other by all the possible operations (in case of CNNs as different type of convolutions, pooling operations, identity and even zero operation signifying no operation between the two nodes) in a directed acyclic graph and it is termed as a *cell*. Cells are stacked along the depth of network to construct the final architecture. All possible operations are used between each two nodes and some weights are assigned to each operation between a pair of nodes which implies the contribution of each operation from one node to another node as weighted output from that node is used as input for the next node. Weights are normalized as probabilities using a simple softmax function. In this way, the discrete space is converted to continuous space.

B. *Bi-level Optimization and final architecture*

The objective is to optimize the weights assigned to each operation o between nodes i and j which we denote as b_{ij}^{o} and the parameters of operation denoted by w. This would require us to minimize both training loss and validation loss. The problem is formularized as:

Say have the initial weights and parameters of operation be b and w respectively. We aim to find optimal parameters by minimizing the training loss that is,

$$w^*(b) = \text{argmin}_w \; L_{train} \,(w, b)$$

and optimal weights can be found by minimizing the validation loss given w* that is,

$$b^* = \text{argmin}_b \; L_{val} \,(w^*(b), b)$$

The equations above can easily be converged to a solution using gradient-based hyperparameter bi-level optimization. Discrete architecture can be achieved by choosing best operations from each of nearest 2 nodes among all available pairs.

IV. Experminents and Results

For the experiments, we used brain tumor image dataset by Cheng, Jun (2017)[1]. 3064 T1-weighted, contrast-enhanced images of three different types of brain tumors: meningioma (708 slices), glioma (1426 slices), and pituitary tumor(930 slices) from 233 patients are included in this brain tumor dataset. We did five experiments using DARTS. Firstly, a search is done to obtain one or more cells, optimizing

There are no sources in the current document. the architecture of a cell. We call this search step. Then in second step, more cells can be stacked and the whole model is trained from scratch and we call step as train step. A key difference in traditional implementation of architectural search and our approach is that we not only take optimized cell and stack them but we use a fine-tuned VGG19 model along with optimized cells to harness transfer learning benefits along with neural architectural search.

A. *Experiment 1*

In this experiment, we used a fine-tuned VGG19 model and extracted the feature maps from pooling layers immediately before 5th layer and 9th layer and used them as inputs to a series of 3 cells. Each cell has 5 internal nodes and the operations available for first 4 nodes were: 3X3 and 5X5 separable convolutions, 3X3 and 5X5 dilated separable convolutions, identity and zero operation. Output of these 4 nodes of cell are concatenated as an input to the 5th node and available operations for this is: 3X3 Max Pooling and 3X3 Average Pooling. Optimal structure of cell is found using this 5 block structure in search step. Then in train step, the same structure with freezed VGG19 layers and 3 optimized cells was trained from scratch.

Observations: Almost all of the final operations (with highest weights) after search step were found to be zero operations and identity operations. After train step, the test accuracy was found to be only 42.27%. This implies that it was not a good model found by our method.

B. *Experiment 2*

Due to recurring zero operations in Experiment 1, we omitted zero operations from set of operations and keeping all the conditions same as Experiment 1, we searched for architecture through 2 steps.

Observations: Most of the final operations (with highest weights) after search step were found to be identity operations. After train step, the test accuracy was found to be only 41%. This clearly implies that our method is biased towards non-convolutional operations.

C. *Experiment 3*

In this experiment, instead of using 3 cells in search step, we used only 1 cell to ensure that the cell is being optimized efficiently, keeping all other conditions same as experiment 2. Then in the train step, we unfreezed fined tuned layers of VGG19 from which feature maps were being extracted so that low level features are tuned further. Also 2 optimized cells from search step were stacked after VGG19 layers and then this network was trained from scratch.

Observations: Most of the final operations (with highest weights) after search step were found to be identity operations. After train step, the test accuracy was found to be only 47%. This implies that our method still doesn't find a good architecture.

D. *Experiment 4*

Even after tuning low level features by unfreezing VGG19 layers, we didn't get a good accuracy. Therefore, we decided to use the output from pooling layer immediately before 13th layer as input to only 1 cell to be used in search step keeping the set of operations same as experiment 3. In train step, all fine-tuned VGG19 layers were used with an optimized cell to train this architecture from scratch.

Observations: Few of the final operations were identity operations. But after train step, still the accuracy was only 35.4%. therefore

E. *Experiment 5*

After all the above 4 experiments, we observed that the training and validation accuracies were quite high in search step. Also being supported by the bias towards non-convolutional layers, it might be possible that the architecture don't want to increase in depth. Hence, we used the setup from search step of 3rd experiment and trained and inferred it while recording test accuracies too.

Observations: We observed that keeping all possible operations intact in search space (with weights), we obtained an accuracy of 91.24%.

Conclusion

Therefore, we conclude that we didn't need deep networks to solve the classification problem. Due to shortage of time, we provide some insight into the future work. As we observed that DARTS was giving most of the operations as zero operation or identity operation while having good training, validation and test accuracies in search step. Therefore, to avoid this problem in DARTS, we can use a new algorithm in which we select top k operations in decreasing order of weights and after that tuning again with train step. The operations between any 2 internal nodes are independent and therefore, they could be parallelized leading to faster computations. Also transfer learning could also be mixed with neural architectural search in many more ways to get some better results.

REFERENCES

[1] Cheng, Jun (2017). brain tumor dataset. figshare. Dataset. https://doi.org/10.6084/m9.figshare.1512427.v5

[2] Chitnis, S., Hosseini, R. & Xie, P. Brain tumor classification based on neural architecture search. *Sci Rep* **12**, 19206 (2022). https://doi.org/10.1038/s41598-022-22172-6

[3] Liu, Hanxiao, Karen Simonyan, and Yiming Yang. "Darts: Differentiable architecture search." *arXiv preprint arXiv:1806.09055* (2018).

[4] Abd El-Wahab, B.S., Nasr, M.E., Khamis, S. et al. BTC-fCNN: Fast Convolution Neural Network for Multi-class Brain Tumor Classification. Health Inf Sci Syst 11, 3 (2023). https://doi.org/10.1007/s13755-022-00203-w.

[5] Saxena, Priyansh, Akshat Maheshwari, and Saumil Maheshwari. "Predictive modeling of brain tumor: a deep learning approach." Innovations in Computational Intelligence and Computer Vision: Proceedings of ICICV 2020. Singapore: Springer Singapore, 2020. 275-285.

[6] Kharrat, Ahmed, et al. "A hybrid approach for automatic classification of brain MRI using genetic algorithm and support vector machine." Leonardo journal of sciences 17.1 (2010): 71-82.

[7] Swati, Zar Nawab Khan, et al. "Brain tumor classification for MR images using transfer learning and fine-tuning." Computerized Medical Imaging and Graphics 75 (2019): 34-46..