AI for Medical Science

Rathod, Kishansinh (SRH Hochschule Heidelberg Student) - 11013344

Submitted to : Dr. Simon Ziegler

Index

1. Introduction

1.1 Description

In this my research module I focused on the application of Artificial Intelligence in the field of Medical Science. To make it narrower, I focused on Medical Imaging(MRI) using Deep Learning and what king of problem may arise during this with experimental approach.

Machine learning is a process that can be applied to medical images for the identification of patterns. While it is an effective instrument that can assist in medical diagnosis, it can be misused. Usually, machine learning starts with the algorithm computing the image characteristics that are believed to be significant in the prediction or diagnosis of interest. To classify the image or compute any metric for the given image area, the machine learning algorithm framework then identifies the best combination of these image features.

What has occurred recently in machine learning, and what does it mean for medical imaging in the future? Over the past few years, machine learning has experienced a huge amount of publicity. It all started around 2009 when, upon several significant benchmarks, so-called deep artificial neural networks began outperforming other existing models. Deep neural networks are the state-of-the-art machine learning models in a number of fields at present, from image recognition to the processing of natural language, and are commonly used in academia and industry. These advances have great potential, slowly being recognized, for medical imaging technology, medical data processing, medical diagnostics, and healthcare in general [1].

In my research project, I provide a brief overview of recent developments and some related challenges in medical image processing applied to machine learning. Since this has become a very large and rapidly expanding field, we will not explore the entire application landscape, but will concentrate primarily on deep learning with MRI.

1.2 What is MRI ?

Magnetic resonance imaging (MRI) is a medical imaging technique that produces detailed pictures of the organs and tissues in your body using a magnetic field and computer-generated radio waves. Big, tube-shaped magnets are the majority of MRI machines. The magnetic field temporarily realigns water molecules in your body while you lie inside an MRI unit. Radio waves cause faint signals to be created by these aligned atoms, which are used to create cross-sectional MRI images, such as slices in a bread loaf.
Neuroimaging techniques such as magnetic resonance imaging (MRI) have been shown to provide biological proof that neurodegenerative cognitive impairment is neurodegenerative, as they provide extensive information on subcortical structures, strong gray matter comparison, and brain tissue integrity. In particular, it is known that changes in the occipital, parietal, prefrontal and temporal lobes can be understood using MRI, including cortical injury, focal lesions and gray matter loss. As complex, unstructured data structures, brain MRI scans are characterized and therefore require sophisticated means to perform efficient, quantitative analysis. Although neuropsychological tests and MRI scans have been tried and checked for their diagnostic and psychometric strengths and

shortcomings, there is a small body of work that has attempted to understand the combined effect of integrating these data sets for MCI diagnosis. [2].

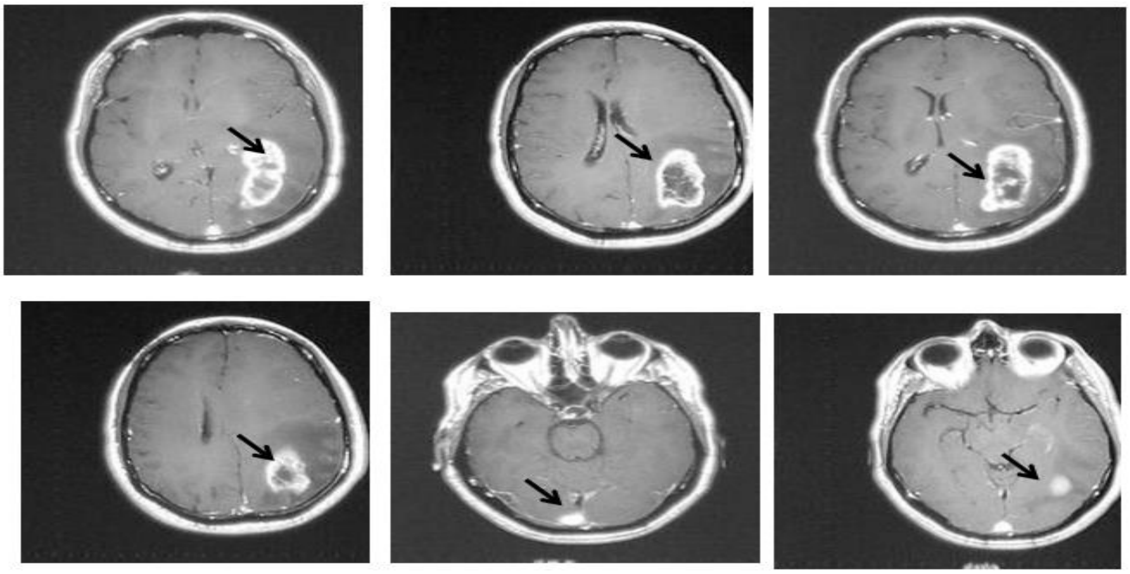

Image 1 : Sample MRI images of brain (https://www.researchgate.net/figure/Sample-MRI-images-of-brain-The-arrows-show-the-presence-of-tumors_fig1_228761309)

2. Research objective

During this research project I was intended to explore the field of Deep learning techniques for Medical Imaging. Hereby, I identify some of the research objectives

- **Identify the best model for medical image classification**:
  Medical images classification is one of major Deep Learning application which can be helpful in diagnosis. Your AI model must be correct when working on this sort of serious field since a mistake in this can be very important. For example, if we have a model with an accuracy of more than 90 percent during training and testing, however, model accuracy during training is not enough, then it is not only necessary. Model behavior and accuracy can not be established until unseen data is tested. Often the model performs very well during training but behaves differently on unseen knowledge. So one of the objective is to identify such Deep Learning model which perform best on Medical Images like MRI, Xray.

- **Working with small amount of data**
  The lack of data is one of the major challenges of applying machine learning techniques in the medical industry. If we operate on small datasets, it is very common that either overfitting or underfitting can be problematic for the model. So what is it lets see :
  Overfitting is a modeling error that occurs when a function is too closely fit to a limited set of data points.

Underfitting refers to a model that can neither model the training data nor generalize to new data. An underfit machine learning model is not a suitable model and will be obvious as it will have poor performance on the training data.

If we are concerned with the classification of brain tumors using MRI image, more data can be available from patients with a negative report compared to patients with a positive report. So, when we use this form of data set to train the model, our AI model may face underfitting and overfitting problems. So, this is one the major objective that needs to be addressed.

3. CNN models

For the study of biomedical imaging data and nonimaging data, machine learning methods have been commonly used in the past few years. More recently, because of its impressive success in predicting various clinical outcomes of interest, a machine learning system known as deep learning, which is focused on artificial neural networks, has gained increased attention. Researchers have developed convolutionary neural network (CNN) models in sub-specialties based on imaging data, which are powerful deep learning techniques for object recognition and classification. For common tasks such as classification with superior precision, CNN models have proven the ability to detect patterns within image data sets and predict corresponding outputs. A series of filters, equivalent to a small image patch, automatically scan through the entire image within the CNN to find similar spatial characteristics of the image. These filters can be trained and modified separately, so that important information for a particular task and data set can be identified by a collection of them. In a reasonably straightforward way, such methods can be used to train deep learning models on 2D MRI scans.

Here I present some of the and it's technical details which I studied through this period and some of them I used for implementation

Table : Short technical details of studied models

VGG	GG16 is a convolutional neural network model proposed by K. Simonyan and A. Zisserman from the University of Oxford in the paper "Very Deep Convolutional Networks for Large-Scale Image Recognition". The model achieves 92.7% top-5 test accuracy in ImageNet, which is a dataset of over 14 million images belonging to 1000 classes. It was one of the famous model submitted to ILSVRC-2014.The idea of using smaller filter kernels and therefore deeper networks (up to 19 layers for VGG19, compared to 7 for AlexNet and ZFNet), and training the deeper networks using pre-training on shallower versions.
AlexNet	AlexNet has had a large impact on the field of machine learning, specifically in the application of deep learning to machine vision. It famously won the 2012 ImageNet LSVRC-2012 competition by a large margin (15.3% VS 26.2% (second place) error rates). Notable features include the use of RELUs, dropout regularization, splitting the computations on multiple GPUs, and using data augmentation

	during training. ZFNet [67], a relatively minor modification of AlexNet, won the 2013 ILSVRC competition.
ResNet	Introduced skip connections, which makes it possible to train much deeper networks. A 152 layer deep ResNet won the 2015 ILSVRC competition, and the authors also successfully trained a version with 1001 layers. Having skip connections in addition to the standard pathway gives the network the option to simply copy the activations from layer to layer (more precisely, from ResNet block to ResNet block), preserving information as data goes through the layers. Some features are best constructed in shallow networks, while others require more depth. The skip connections facilitate both at the same time, increasing the network’s flexibility when fed input data. As the skip connections make the network learn residuals, ResNets perform a kind of boosting
DenseNet	Builds on the ideas of ResNet, but instead of adding the activations produced by one layer to later layers, they are simply concatenated together. The original inputs in addition to the activations from previous layers are therefore kept at each layer (again, more precisely, between blocks of layers), preserving some kind of global state. This encourages feature reuse and lowers the number of parameters for a given depth. DenseNets are therefore particularly well-suited for smaller data sets (outperforming others on e.g. Cifar-10 and Cifar-100).
U-net	A widely common and successful 2D image segmentation network. It is first downsampled via a "traditional CNN when fed an input image, before being upsampled by transposing convolutions until it reaches its original scale. In addition, there are skip connections based on the ResNet ideas that concatenate features from downsampling to upsampling routes. It is a fully-convolutionary network.

4. Experiment

To perform my experiment, I used MRI images and worked on Brain Tumor Detection. A brain tumor is a collection, or mass, of abnormal cells in your brain. Your skull, which encloses your brain, is very rigid. Any growth inside such a restricted space can cause problems.

As explained earlier, we have list of different models. To check the performance, I choose VGG16 for my experiment as well as based VGG16 and my theoretical Learning I designed a new model. I used both models for Brain Tumor Detection using MRI images of brain.

4.1 Dataset

As my main objective here is to check working on small dataset then I choose a small dataset from www.kaggle.com. The dataset contains 253 images. The dataset contain two label: no and yes. One of them is with brain tumor one is without.

For the experiment purpose the whole dataset divided in 80-20%. 80 percent for training purpose with both label and 20 for testing dataset.

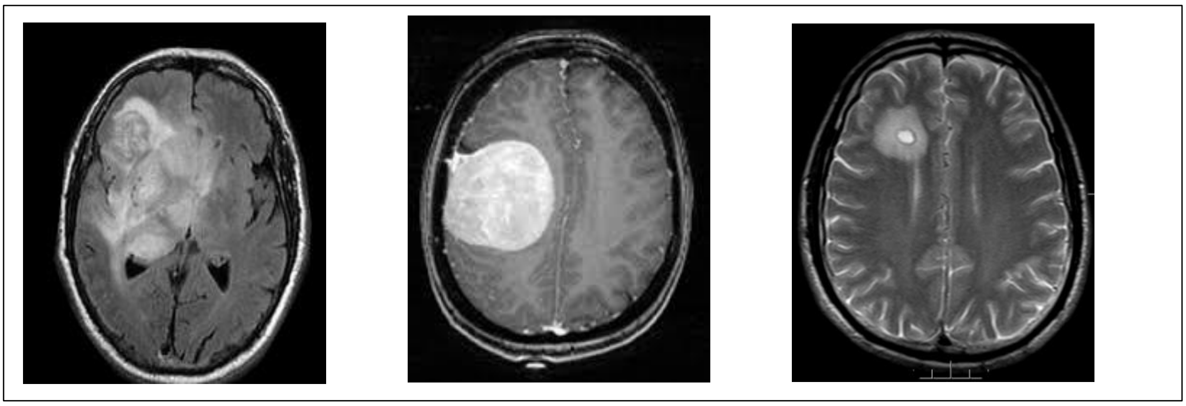

Fig 4.1.1 : Sample Image with label ***Yes***

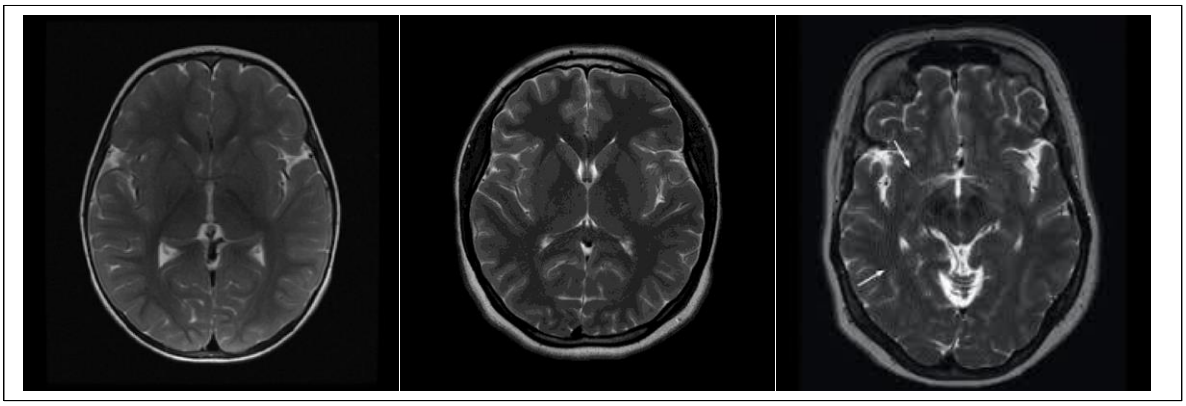

Fig 4.1.2 : Sample Image with label ***No***

4.2 Definitions

CNN architecture made of convolution. Convolutional layers are the major building blocks used in convolutional neural networks.

The simple application of a filter to an input that results in an activation is a convolution. When repeatedly applying the same filter to an input, an activation map called a feature map shows the position and intensity of the detected feature in an input, such as an image. Let's see one by one term use in CNN.

- **Activation Function (Relu)**

  Rectified Linear Unit (ReLU) transform function only activates a node if the input is above a certain quantity, while the input is below zero, the output is zero, but when the input rises above a certain threshold, it has a linear relationship with the dependent variable.

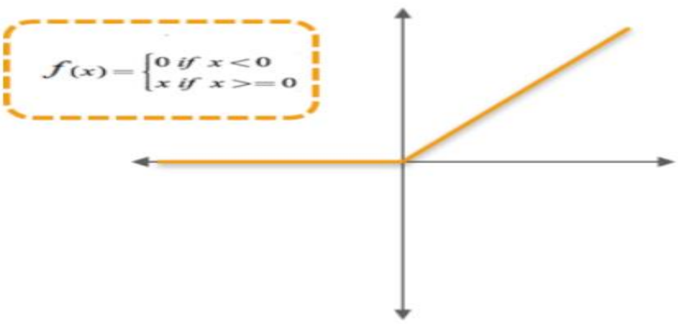

Image 4.2.1 : Relu Activation Function

- Maxpooling

  A sample-based method of discretization is Max pooling. The aim is to down-sample an input representation (image, output matrix of the hidden layer, etc.), decrease its dimensionality and allow assumptions to be made about features found in the binned sub-regions.

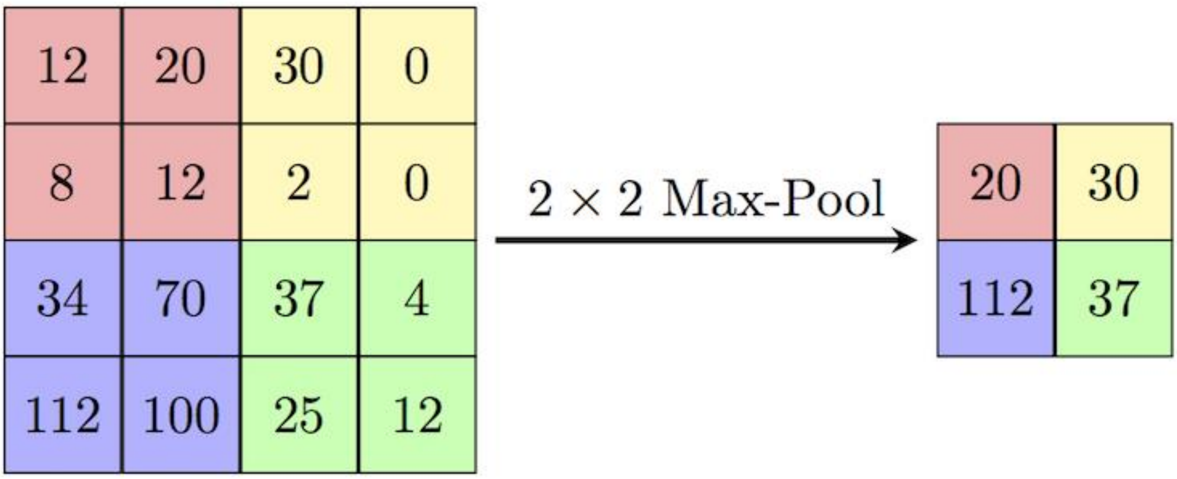

Image 4.2.2 : Maxpooling (https://computersciencewiki.org/index.php/Max-pooling_/_Pooling#:~:text=Max%20pooling%20is%20a%20sample,in%20the%20sub%2Dregions%20binned.)

- Softmax
  Softmax is exponential and enlarges differences - push one result closer to 1 while another closer to 0. It turns scores aka logits into probabilities. Cross entropy (cost function) is often computed for output of softmax and true labels (encoded in one hot encoding).

- Dense Layer
  The standard, deeply connected neural network layer is a dense layer. It is the layer that is most common and used frequently. The thick layer performs the procedure below on the input and returns the output.

4.3 Experiment 1

For my first experiment I choose VGG16. Following is the architecture of VGG16

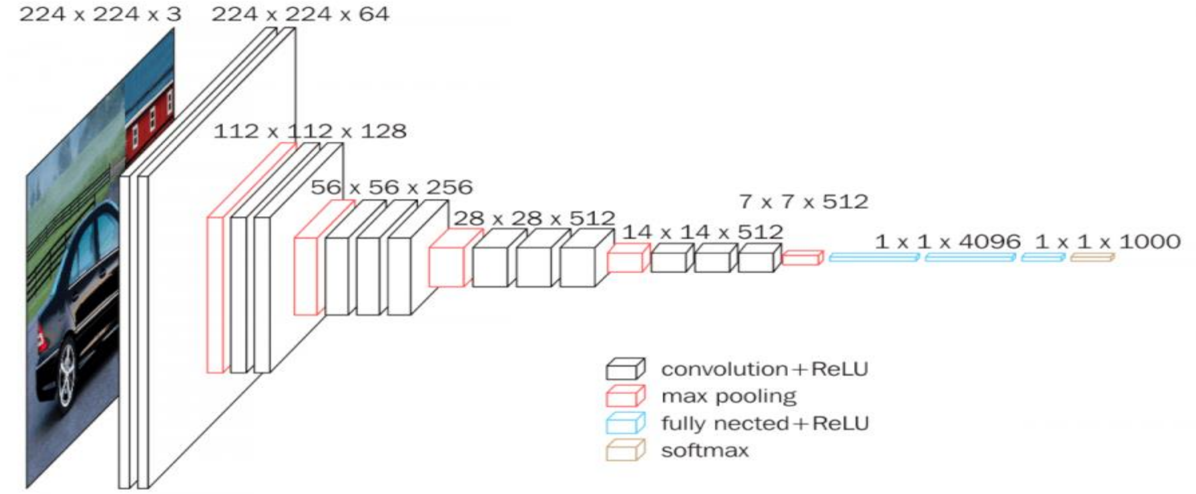

Image 4.3.1 : VGG16 architecture. [*https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c*](https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c)

Experiment parameter :

Augmentation : Yes (flip)
Color channel : RGB
Batch size : 6
Training epoch : 15

Experiment results

```
Epoch 00013: saving model to vgg16.h5
6/6 [==============================] - 206s 40s/step - loss: 0.6863 - accuracy: 0.6739 - val_loss: 0.6862 - val_accuracy: 0.6250
Epoch 14/15
accuracy
	training             (min:    0.407, max:    0.669, cur:    0.663)
	validation           (min:    0.469, max:    0.719, cur:    0.562)
Loss
	training             (min:    0.658, max: 1368.759, cur:    0.685)
	validation           (min:    0.642, max:    0.695, cur:    0.689)

Epoch 00014: saving model to vgg16.h5
6/6 [==============================] - 207s 40s/step - loss: 0.6861 - accuracy: 0.6580 - val_loss: 0.6891 - val_accuracy: 0.5625
Epoch 15/15

	training             (min:    0.407, max:    0.669, cur:    0.663)
	validation           (min:    0.469, max:    0.719, cur:    0.656)
Loss
	training             (min:    0.658, max: 1368.759, cur:    0.684)
	validation           (min:    0.642, max:    0.695, cur:    0.686)

Epoch 00015: saving model to vgg16.h5
6/6 [==============================] - 171s 30s/step - loss: 0.6843 - accuracy: 0.6710 - val_loss: 0.6861 - val_accuracy: 0.6562
```

Image 4.3.2 : VGG16 accuracy matrix

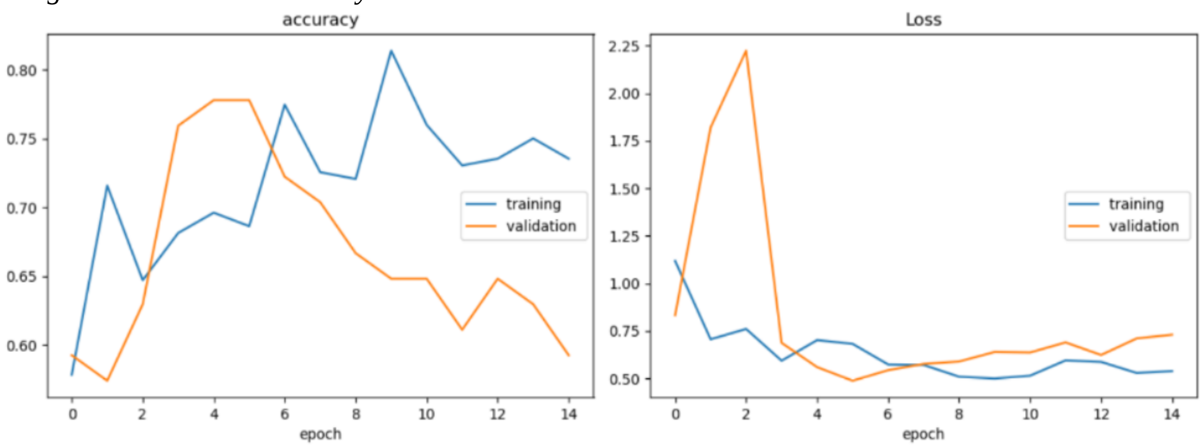

Image 4.3.3 : VGG16 Accuracy vs Loss graph

Conclusion

At the beginning of training model performed well but while epoch going on the testing accuracy constantly decreased. Which is clear sign that model is overfitting here.

4.4 Experiment 2

For my next experiment I again tried VGG16 with different parameter.

Experiment parameter :

Data Augmentation : Yes
Color channel : RGB
Batch size : 12
Training epoch : 20

Experiment results

```
Epoch 00018: saving model to vgg16.h5
17/17 [==============================] - 190s 11s/step - loss: 0.6779 - accuracy: 0.6348 - val_loss: 0.6855 - val_accuracy: 0.5625
Epoch 19/20
accuracy
	training           (min:    0.564, max:    0.618, cur:    0.608)
	validation         (min:    0.458, max:    0.583, cur:    0.521)
Loss
	training           (min:    0.681, max: 1471.357, cur:    0.681)
	validation         (min:    0.683, max:    0.725, cur:    0.690)

Epoch 00019: saving model to vgg16.h5
17/17 [==============================] - 200s 12s/step - loss: 0.6803 - accuracy: 0.6110 - val_loss: 0.6903 - val_accuracy: 0.5208
Epoch 20/20

	training           (min:    0.564, max:    0.618, cur:    0.608)
	validation         (min:    0.458, max:    0.583, cur:    0.542)
Loss
	training           (min:    0.680, max: 1471.357, cur:    0.680)
	validation         (min:    0.683, max:    0.725, cur:    0.687)

Epoch 00020: saving model to vgg16.h5
17/17 [==============================] - 193s 12s/step - loss: 0.6778 - accuracy: 0.6284 - val_loss: 0.6872 - val_accuracy: 0.5417
```

Image 4.4.1 : VGG16 accuracy matrix

Conclusion

Here also we can see the validation accuracy is lower than training accuracy which means model is Overfitted.

4.5 Experiment 3

For my next experiment I again tried VGG16 with different parameter here I use grayscale image as input and trained model with grayscale image only because MRI images contain only grayscale channel.

Experiment parameter :

Data Augmentation : Yes (Hflip)
Color channel : Grayscale
Batch size : 12
Training epoch : 15

Experiment results

```
Epoch 00013: saving model to vgg16.h5
34/34 [==============================] - 218s 6s/step - loss: 0.5455 - accuracy: 0.7293 - val_loss: 0.5514 - val_accuracy: 0.8148
Epoch 14/15
accuracy
	training           (min:    0.569, max:    0.755, cur:    0.745)
	validation         (min:    0.537, max:    0.870, cur:    0.870)
Loss
	training           (min:    0.516, max:  235.967, cur:    0.516)
	validation         (min:    0.460, max:    0.847, cur:    0.513)

Epoch 00014: saving model to vgg16.h5
34/34 [==============================] - 212s 6s/step - loss: 0.4797 - accuracy: 0.7920 - val_loss: 0.5127 - val_accuracy: 0.8704
Epoch 15/15

	training           (min:    0.569, max:    0.755, cur:    0.750)
	validation         (min:    0.537, max:    0.870, cur:    0.870)
Loss
	training           (min:    0.501, max:  235.967, cur:    0.501)
	validation         (min:    0.460, max:    0.847, cur:    0.503)

Epoch 00015: saving model to vgg16.h5
34/34 [==============================] - 263s 8s/step - loss: 0.4822 - accuracy: 0.7572 - val_loss: 0.5029 - val_accuracy: 0.8704
```

Image 4.5.1 : VGG16 accuracy matrix

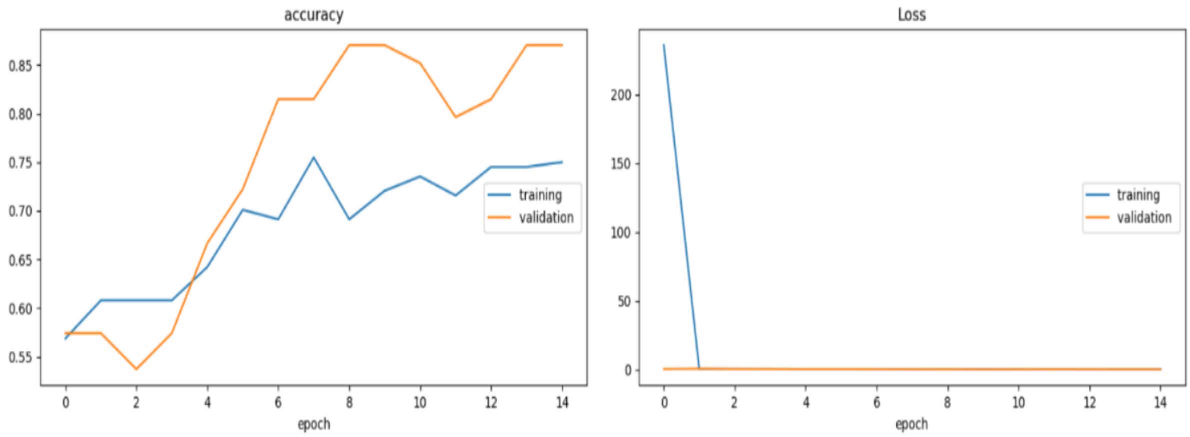

Image 4.5.2 : VGG16 Accuracy vs Loss graph

Conclusion

Here the model having good training as well as testing accuracy. At the beginning, the model has very high data loss, but this is not a concern at all because it has good accuracy at the end. But one issue here is that the model does not perform well on unseen data and performance errors.

4.6 Experiment 4 (New proposed model)

Till now I checked with VGG and it's performance so now applied my theoretical knowledge and then I design new model based on that. I use grayscale image as input and trained model with grayscale image. Given below is architecture of my model.

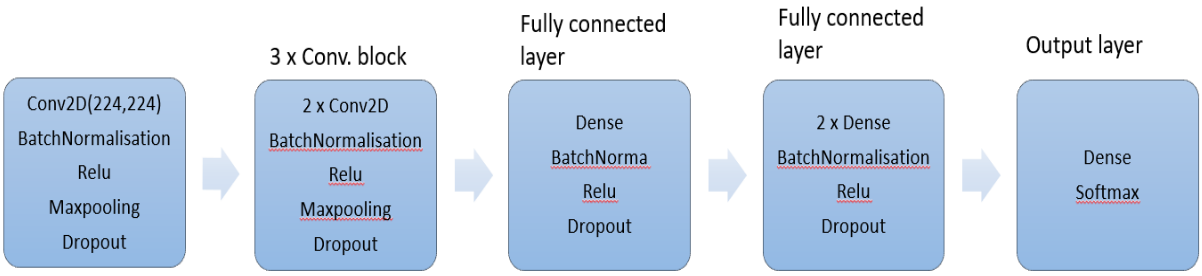

Figure 4.6.1 : Architecture of proposed model

Experiment parameter :

Data Augmentation : Yes (Horizontal flip)
Color channel : Grayscale
Batch size : 12
Training epoch : 15

Experiment results

```
Epoch 00018: saving model to my_model_weights.h5
17/17 [==============================] - 236s 14s/step - loss: 0.1823 - accuracy: 0.9486 - val_loss: 0.6843 - val_accuracy: 0.7500
Epoch 19/20
accuracy
        training                (min:    0.618, max:    0.931, cur:    0.926)
        validation              (min:    0.521, max:    0.854, cur:    0.729)
Loss
        training                (min:    0.177, max:    0.836, cur:    0.198)
        validation              (min:    0.401, max:  436.661, cur:    0.590)

Epoch 00019: saving model to my_model_weights.h5
17/17 [==============================] - 245s 15s/step - loss: 0.1894 - accuracy: 0.9386 - val_loss: 0.5900 - val_accuracy: 0.7292
Epoch 20/20

        training                (min:    0.618, max:    0.931, cur:    0.917)
        validation              (min:    0.521, max:    0.854, cur:    0.688)
Loss
        training                (min:    0.177, max:    0.836, cur:    0.214)
        validation              (min:    0.401, max:  436.661, cur:    0.793)

Epoch 00020: saving model to my_model_weights.h5
17/17 [==============================] - 245s 15s/step - loss: 0.2181 - accuracy: 0.9125 - val_loss: 0.7932 - val_accuracy: 0.6875
```

Image 4.6.2 : Proposed model accuracy matrix

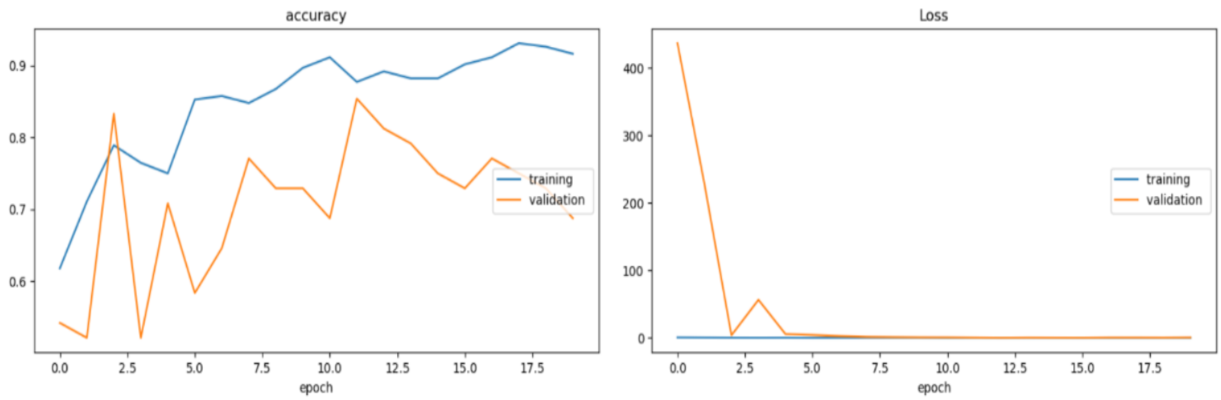

Image 4.6.3 : Proposed model Accuracy vs Loss graph

Conclusion

The new proposed model having good training and testing accuracy. At the beginning, the model has very high data loss, but this is not a concern at all because it has good accuracy at the end. The model also performed well on unseen data. Six unseen images given to the model from that model predicted five with the right label when doing manual testing.

5. Further enhancement

Accuracy is the primary concern for fields such as medical imaging, but there is no model that provides you with very close to accurate prediction accuracy. There is still possibility of future enhancement in Deep Learning model for medical imaging which I mentioned here.

5.1 GANs for augmentation

A generative adversarial network is a class of machine learning systems founded in 2014 by Ian Goodfellow and his peers. Two neural networks in a game compete with each other. This technique learns how to produce new data with the same statistics as the training set, given a training set.

A generative adversarial network consists of two neural networks pitted against each other. The generative network G is tasked with creating samples that the discriminative network D is supposed to classify as coming from the generative network or the training data. The networks are trained simultaneously, where G aims to maximize the probability that D makes a mistake while D aims for high classification accuracy [1].
There are some recent research in the field which can be helpful in this area for example MedGAN. MedGAN builds upon recent advances in the field of generative adversarial networks (GANs) by merging the adversarial framework with a new combination of non-adversarial losses. It utilize a discriminator network as a trainable feature extractor which penalizes the discrepancy between the translated medical images and the desired modalities. Moreover, style-transfer losses are utilized to match the textures and fine-structures of the desired target images to the translated images. Additionally, it present a new generator architecture, titled CasNet, which enhances the sharpness of the translated medical outputs through progressive refinement via encoder-decoder pairs [3].

5.2 YOLO

YOLO is a state-of-the-art object detection algorithm that is incredibly fast and accurate. Introduced a new, simplified way to do simultaneous object detection and classification in images. It uses a single CNN operating directly on the image and outputting bounding boxes and class probabilities. It incorporates several elements from the above networks, including inception modules and pretraining a smaller version of the network. It's fast enough to enable real-time processing. YOLO makes it easy to trade accuracy for speed by reducing the model size. YOLOv3-tiny was able to process images at over 200 frames per second on a standard benchmark data set, while still producing reasonable predictions [1]

6. Conclusion

This research project described the classification of migraine MRI data using a CNN. Research proposed new model for medical image classification especially MRI images. Model gave very good accuracy but still not up to the mark. There is still and always possibility of enhancement in model.
Additionally, In the medical domain there are many problems to which neural network methods can theoretically contribute. It is necessary for researchers to acquire interdisciplinary knowledge and to design effective processes of data acquisition that contribute to neural network performance. More medical-related issues may theoretically be addressed by integrating expertise from both parties to promote cooperation between experts from both computer science and medical fields.

7. References

1. Lundervold, Alexander Selvikvåg, and Arvid Lundervold. "An overview of deep learning in medical imaging focusing on MRI." *Zeitschrift für Medizinische Physik* 29.2 (2019): 102-127.
2. Qiu, Shangran, et al. "Fusion of deep learning models of MRI scans, Mini–Mental State Examination, and logical memory test enhances diagnosis of mild cognitive impairment." *Alzheimer's & Dementia: Diagnosis, Assessment & Disease Monitoring* 10 (2018): 737-749
3. Karim Armanious, Chenming Jiang, Marc Fischer, Thomas Küstner, Tobias Hepp, Konstantin Nikolaou, Sergios Gatidis, Bin Yang, MedGAN: Medical image translation using GANs, Computerized Medical Imaging and Graphics, Volume 79, 2020, 101684, ISSN 0895-6111,
4. Ng, Hwei Geok, et al. "Classification of MRI Migraine Medical Data Using 3D Convolutional Neural Network." International Conference on Artificial Neural Networks. Springer, Cham, 2018.
5. Jain, Rachna, et al. "Deep learning based detection and analysis of COVID-19 on chest X-ray images." *Applied Intelligence* (2020): 1-11.
6. Gu, Shanqing, Manisha Pednekar, and Robert Slater. "Improve Image Classification Using Data Augmentation and Neural Networks." *SMU Data Science Review* 2.2 (2019): 1.
7. Wang, Guotai, et al. "Interactive medical image segmentation using deep learning with image-specific fine tuning." *IEEE transactions on medical imaging* 37.7 (2018): 1562-1573.
8. Zhou, Tongxue, Su Ruan, and Stéphane Canu. "A review: Deep learning for medical image segmentation using multi-modality fusion." *Array* 3 (2019): 100004.
9. Loey, Mohamed, et al. "Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-50 for medical face mask detection." *Sustainable Cities and Society* (2020): 102600.
10. Miranda, Eka, Mediana Aryuni, and E. Irwansyah. "A survey of medical image classification techniques." *2016 International Conference on Information Management and Technology (ICIMTech)*. IEEE, 2016.
11. Pravitasari, Anindya & Iriawan, Nur & Almuhayar, Mawanda & Azmi, Taufik & Irhamah, Irhamah & Fithriasari, Kartika & Purnami, Santi & Ferriastuti, Widiana. (2020). UNet-VGG16 with transfer learning for MRI-based brain tumor segmentation. TELKOMNIKA (Telecommunication Computing Electronics and Control). 18. 1310. 10.12928/telkomnika.v18i3.14753.
12. https://pjreddie.com/darknet/yolo/