Lungs Cancer Detection

Ningthoujam Dhanachandra Singh & Rimjhim

17 April 2017

1 Abstract of the project

The count of lung cancer patient is increasing day by day. It would be a boon if cancer can be detected earlier. Researchers are looking forward to use CT images for detecting lung cancer. Some areas of lung become abnormal for a cancerous patient. Abnormal areas can be both cancerous or non-cancerous. We are planning to design a deep learning network that can detect whether abnormal areas are cancerous or not. The data set consists of several CT scan images. The basic aim is to analyze the pattern of abnormal cancerous area in lung so that given any CT image, it can be accurately detected whether the patient is cancerous or not. The images are in format of DICOM images. We are using a deep convolutional network for analyzing the pixel pattern of the images. This requires several stages of preprocessing of images followed by a deep convolutional network.

2 Introduction

2.1 Literature survey

CT images are commonly used for early detection of Lungs Cancer. Deep neural networks have become very popular for detecting lungs cancer. Many researchers have used deep neural network for detecting cancer. In most of the works, researchers analyze 'lungs nodule' and not the complete image. The preprocessing of images are equally important as architecture of the network. Jinsa Kuruvilla et. al [1] has used several statistical feature to classify the images. They have used Feed Forward neural network and feed forward back propagation neural network . They have tried several training functions for back-propagation and got maximum of specificity of 100% . Their network in not convolutional network but have used image segmentation for reducing the dimension of the images.

Rotem Golan et al. [2] has designed a Deep Convolutional Network for cancer detection. They have used back propagation for training. They have used volumetric convolution, max pooling and fully connected layers in their network. They have used ReLu activation for convolutinal layers, and softmax for output layer. They have achieved sensitivity of around 79%.
Bram van Ginneken et al. [3] has also used deep convolutional networks to identify cancerous nodules. They have attained a sensitivity of around 71%.The state of art results are summarized in Figure 1

- Data Source : https://www.kaggle.com/c/data-science-bowl-2017/data The data is from a competition on 'Kaggle'.

AuthorName	Year	Approach	Mean Squared Error	Results
Jinsa Kuruvilla,K.Gunavathi		Feed Forward, Back propagation(gradient descent)	0.112	91.11(Accuracy)
Rotem Golan		Convulational Neural Network,Back propagation(lung nodule)	-	78.9(Sensitivity)
Bram van Ginneken	2015	Convulational Neural Network	-	71%(Sensitivity)

S.No.	Experiments	Model Number	Dataset size	Number of images	Cancer images in test data	Non cancer images in test data	Accuracy
1	Exp1	1	2gb	3408	192	486	71.26%
2	Exp2	1	8gb	14245	594	2068	80.43%
3	Exp3	2	10gb	19200	704	2894	98.61%
4	Exp4	2	16gb	30407	1224	4406	96.13%
5	Exp3	1	10gb	19200	704	2894	97.72%
6	Exp4	1	16gb	30407	1224	4406	93.00%

Figure 1: Summary of different experiments with different architecture

- Tools : For preprocessing of images we are using, two famous python tools i.e. 'pydicom' and 'opencv'. And no doubt basic tools of python are also used such as numpy, sklearn, pandas,etc.
- Tool for Deep Learning : We are using python supported tool called as 'Keras' for implementing our deep neural network.

3 Resources

3.1 Work done

3.1.1 Description of the data

The input are the image files in 'dicom' format. The format and configuration of images are different since the images are taken at different time and from different types of camera. So, we converted all images into similar size and format. In order to reduce the size of the input data, we have segmented the image. We will use pixel as input to the neural network. There are only two classes of images i.e. cancerous or non-cancerous.The total size of the input data is 150GB.There is a small subset of data of size around 2GB, which can be used for various testing purposes.

3.1.2 Exploration of different neural networks and observation from the same

We are basically working on convolutional networks. The fundamental architecture of such network contains convolutional layers followed a fully-connected layer. We have experimented on two architectures which are described as follows:

- a.Model 1: 4 2-D convolutional layers + 2 Fully connected layers + output layer
- b.Model 2: 2 2-D convolutional layers + 1 Fully connected layers + output layer

The observation from the different experiments are summarized in Figure 3. We can see that our accuracy varies in range starting from 71% to 98%. On increasing data size the accuracy increases.

3.1.3 Error vs epoch plot

The loss versus epoch plot for different observations in shown in Figure 2. For every experiment, we observed that error decreases with epochs.

3.1.4 Final architecture

The results are better with Model-2. So, this will be our final architecture. Model 2: 2 2-D convolutional layers + 1 Fully connected layers + output layer

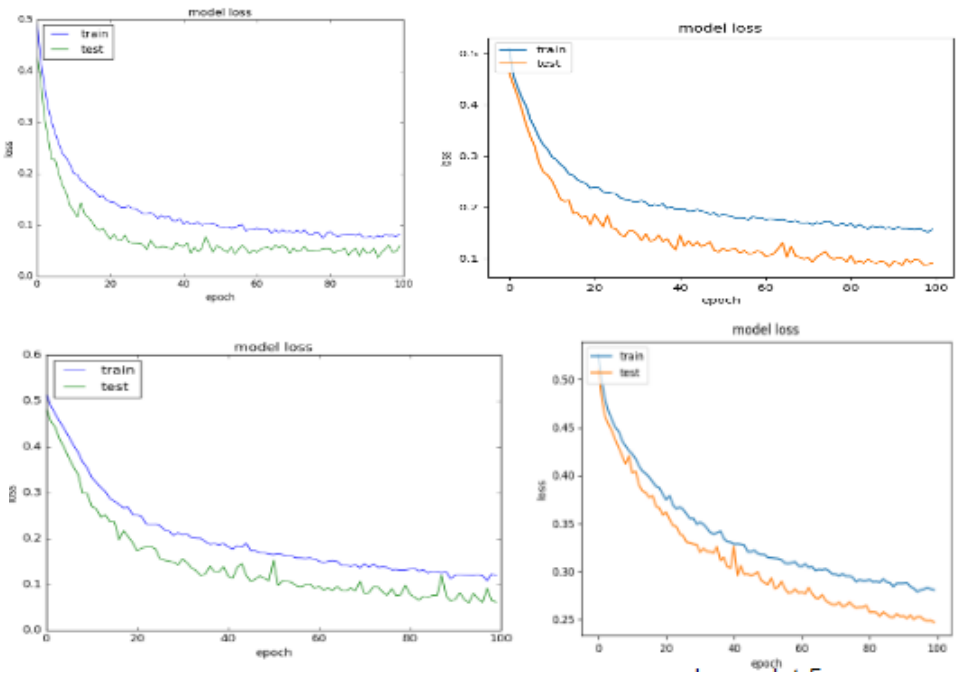

Figure 2: Loss vs Epoch for different experiments

	SGD(Accuracy)	RMSprop(Accuracy)
2Gb	71%	91%
10Gb	98.61%	96%
16GB	96.13%	91.13%

Figure 3: Optimizer 'SGD' vs 'RMSprop'

3.1.5 Results from different optimization techniques

We tried on two popular optimizers i.e 'sgd' and 'rmsprop'. After various experiments, we got to know that 'sgd' was performing better. The outcome of various experiments are summarized in Figure 3.2, we can see that for 'sgd' the results vary with change in data size but for 'mse' it is not appearing the same.

3.1.6 Github Link

we have uploaded the code and the link is : https://github.com/Dhanachandra/DeepLearning_project/blob/master/lungcancerdetection.py

3.2 Future work

Primarily there are two major challenges for this project. First one is to detect where an image is cancerous or non-cancerous. Second is to deal with such a huge data. First challenge is accomplished in this project using a small part of complete data. In future, we will try to perform all experiments on complete data.