CLASSIFICATION OF PNEUMONIA BY MODIFIED DEEPLY SUPERVISED RESNET AND SENET USING CHEST X-RAY IMAGES

M A Ramitha
Department of Computer Science and Engineering,
Karpagam Academy of Higher Education,
Coimbatore, Tamil Nadu, India.
ramithaa@gmail.com

N Mohanasundaram
Department of Computer Science and Engineering,
Karpagam Academy of Higher Education,
Coimbatore, Tamil Nadu, India.
itismemohan@gmail.com

Submitted: *Feb, 02, 2021* ***Revised:*** *May, 10, 2021* ***Accepted:*** *May, 21, 2021*

Abstract: In Deep Learning, a Convolutional Neural Network (CNN) extracts the features from the visual imagery. These features can be used for various complex tasks such as image classification and segmentation and detection of different objects. The convolutional layers are stacked over each other to form the state-of-the-art models. A modified SENet architecture is introduced in this study to classify pneumonia from chest x-ray images. Six ResNet blocks are connected back to back. The output from the sixth ResNet and the side outputs from the last three ResNets are fused together. This output is fed as input to the SENet block. The validation accuracy of this fusion architecture is 91.84% on chest x-ray images.

Keywords: CNN, Deep Learning, AlexNet, VGGNet, InceptionNet, ResNet, DenseNet, SENet, ILSVRC.

I. INTRODUCTION

The neurons in the human brain are the stimulus behind neural network. The fundamental block in the neural network is termed as neuron. Each neuron performs a mathematical function and provides output to a particular input. The mathematical function is termed as convolution. These neurons are arranged as layers. Convolutional neural network (CNN) is a neural network model [1]. CNN composed of stacked convolutional layers. It comprised of an input layer, several hidden layers and output layer [2] the input is provided to the first layer called input layer. It extracts adequate information from the input data. The hidden unit (convolution layers), receives the information from the input layer and extracts the features and transforms it into something that the output unit (fully connected layers) can use. To make the network deeper, it is necessary to combine more hidden layers in the existing network.

The information collected by a node is passed to every node in the very next layer. Each node which receives the data makes changes in it and again sent it the

next layer. A software contest named ImageNet Large Scale Visual Recognition Challenge (ILSVRC) has been conducted by ImageNet project [3]. It analyses different algorithms for detection and classification of images. There are several CNN architectures like AlexNet, VGGNet, ResNet InceptionNet, DenseNet, XceptionNet, and SENet [4]. All these architectures are ILSVRC winners. Some other architecture is formed by merging the above said architectures.

Figure 1 shows the architecture of CNN. Proposed system discussed about an architecture that combines deeply supervised ResNet and SENet.

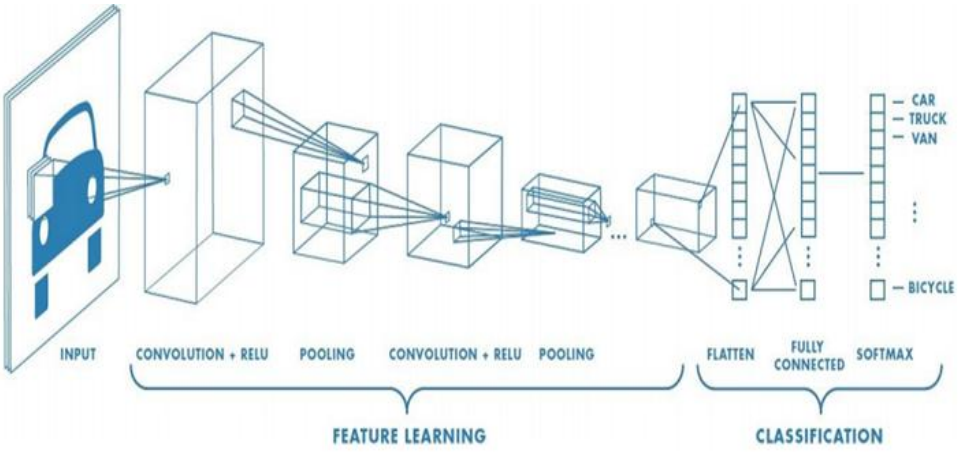

Fig .1 Architecture of CNN

II. RELATED WORKS

To enhance the computational efficiency and to diminish the error rate, architectural developments have been introduced [5]. LeNet was the basic and simplest CNN architecture. It brought a drastic development in the performance on the recognition of hand digits [6]. AlexNet was the initial deep CNN architecture that offered remarkable performance in image-recognition and classification [7]. AlexNet holds eight learned layers, which comprise convolutional and fully connected layers. It uses 650,000 neurons and incorporates 60 M parameters.

VGGNet has introduced a successful design principle for deep learning networks**.** Instead of11x11 and 5x5filters, VGG introduced 3x3 filters [8]. The 2x2 and 3x3 filters are connected back-to-back. These back-to-back filters replaced the large sized filters. InceptionNet presents a new concept called Inception block [9]. It uses three filters of different sizes. These filters are capsulated into one inception block. It exploits the idea termed as 'split, transform, and merge'.

A 152-layered deep CNN is the idea behind ResNet [10]. It is deeper and more accurate than the deep nets proposed before. In CNN, the original mapping is represented as F (x). But in ResNet, it is F(x)+x, where x is an input to the layer.
Shortcut connection is used to attain this result. The shortcut connections help to perform identity mapping. Outputs from the shortcut connections are added to the outputs from stacked layers.

Accuracy gain of deep residual network, result in an enhanced Inception-ResNet, because of an improved version of InceptionNet. Inception-V4 and Inception-ResNet [11] are inspired by InceptionNet. By making the Inception layer deeper and wider, more efficient Inception-V4 is introduced. The inception-V4 has more number of inception modules in its architecture than in inception-V3 architecture. Inception-ResNet is a modified version of ResNet. It comprises residual learning and inception block. Inception ResNet and plain Inception-V4 have the same power.

Instead of using the inception modules, XceptionNet introduces depth-wise separable convolution [12]. Cross- channel correlation is obtained from 1x1 Convolutions and the spatial correlation of every output channel is obtained by mapping the spatial correlation of each channel. Each channel of an input goes through spatial convolution independently. Then a point-wise convolution is performed on it. This makes depth-wise separable convolutions. There are 36 convolution layers of 14 modules. All these modules have linear residual connections around them.

ResNeXt [13] is the combined version of VGG and GoogleNet architecture. One 3x3 filter is used inside a 'split, transform and merge block'. It also adopts residual learning from ResNet. There are 14 modules to incorporate all 36 convolution layers. These 14 modules have linear residual connections. This is the simple architecture which Shows VGG/ResNet strategy. ResNeXt carries out a series of operations on the input provided. The operations provided by ResNeXt are:

- Split: Vector x undergoes a split operation.
- Transform: The low-dimensional representation is transformed into wixi.
- Aggregate: The wixi in all representations are added together.

The Densely Connected feed-forward network [14] collects all feature maps from its previous layer. These feature maps, along with its own feature maps, are provided to the inputs of the next. The model obtained after training is highly parameter-compliant.

Feature maps of different layers are concatenated together. So, each layer receives input from subsequent layers and improves efficiency. One advantages of using DenseNet is that it resolves the problem associated with the vanishing gradient. It ensures the reuse of features and also strengthens its feature-propagation.

SE blocks models interdependencies between channels [15]. These inter dependencies are used to recalibrate channel-wise feature maps. For any transformation consider a function maps the input X to the feature maps U where $U \epsilon R^{H \times W \times C}$. A squeeze operation is performed over the features U. The aggregated feature maps are aggregated with their spatial representation (HXW). The result is a channel descriptor.

With the help of the channel-descriptor excitation operation produces pre-channel modulation weights. These weights are applied to feature map U, and the output received is considered as the output of the SE block. This output is directly fed into the subsequent layers of the network.

In Deeply supervised ResNet architecture, there are three sets of additional side-output layers to intermediate hidden layers of 11-layer ResNet [16]. It can perform multi-scale learning by predictions of intermediate supervised layers. In order to improve the final performance, it also introduced "companion" objective function.

This Network architecture consists of five pre-activation residual units. Each of these units exist two convolutional layers. Three additional sets of side-output layers are

inserted after the third, fourth, and fifth residual units, respectively. Fusion layer used to combine outputs from the multi-scale side-output layers.

Deeply supervised ResNet did not add side-output layers after the first two residual units, because the size of feature maps produced by these units is quite large and the layers in these units mainly learns low-level visual features that are not comprehensive enough for classification [17]. All the side-output layers consist of an average pooling layer, a fully connected layer with five units and a softmax layer. The average sum of outputs of the three fully connected layers is fed in to softmax layer of the fusion layer.

III. PROPOSED SYSTEM

The inspiration behind this fused deeply supervised and SeNet is deeply supervised ResNet. The Proposed system concatenates the side outputs taken from deeply supervised ResNet with SENet block. This architecture enjoys the advantages of Deeply Supervised ResNet and SENet. The proposed system consists of six ResNet blocks and one SENet.It connects six ResNet blocks back to backSide outputs are taken from the last 4 ResNet blocks. The reasons behind this are:

- The size of the feature map obtained from the first two blocks will be large.
- First two blocks learn low-level visual features. That is not enough for classification.

The objective function implemented in the proposed system is termed as "companion" objective function. These are implemented at the hidden layers of RenNet. It resolves robustness of learned features and "vanishing" gradients obstacles present in CNN architectures.The side output from last four ResNet block is fused together and is fed to the SENet block. The input to the SENet obtained from the fusion layer is first gone through a squeeze operation. Figure 2 depicts the architectural details of the proposed system.

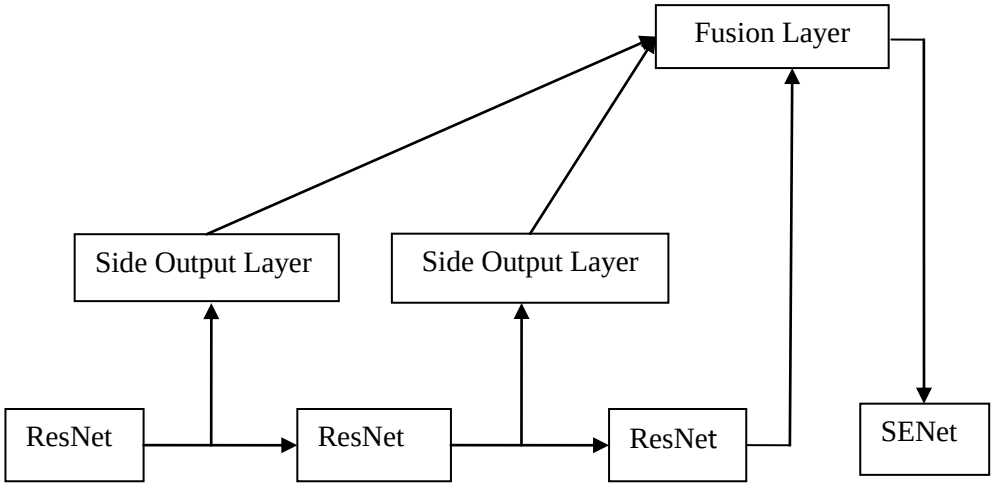

Fig. 2 Architecture of fused deeply supervised ResNet

The global spatial information is squeezed into a channel descriptor [17]. This is obtained by using global average pooling. It generates channel-wise statistics. This architecture used sigmoid activation, σ function. Channel wise dependencies are captured by using excitation operation. The final output of the block is obtained by rescaling the transformation output U with the activations [18]. Figure 3 describes Squeeze and excitation operation.

IV. RESULT AND DISCUSSION

The proposed system uses Kaggle chest x-ray pneumonia dataset. *Kaggle* is the world's largest data science community with powerful tools and resources to help you achieve data science goals [17]. In Pneumonia chest x-ray data set there are 1341 normal x-ray images and 3875 diseased images. The data set is split into training, and test sets.75% of the data is used for training and the remaining 25% is used for testing.

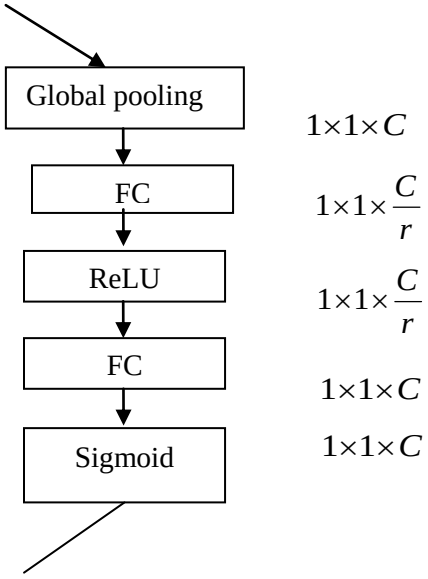

Fig. 3 Squeeze and excitation operation

The data imported is preprocessed. As the information is collected from different sources, the format of the data will be different. This raw data cannot be used for the analysis. So data preprocessing technique convert the raw data into an understandable data set. When training an image, the image is resized into 512 x 512. The proposed system evaluated 4 metrics– Accuracy, Precision, Recall and F1Score.

Accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition

$$Accuracy = \frac{Number\ of\ \ correct\ \ predictions}{Total\ Number\ of\ \ predictions} \quad (1)$$

The proposed system is a binary classification system and thus the binary classification accuracy is calculated as

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

Precision is the amount of positive predictions that were correct. Precision can be calculated as

$$precision = \frac{TP}{Total\ Number\ of\ \ Positive\ predictions} \tag{3}$$

Recall is the percentage of positive cases in the all positive predictions.

$$recall = \frac{TP}{Number\ of\ \ Acual\ Positive\ predictions} \tag{4}$$

F1-score is a measure of a model's accuracy on a dataset. The F-score is a way of combining the precision and recall of the model, and it is defined as the harmonic mean of the model's precision and recall.

$$F1Score = 2 \times \frac{precision \times recall}{precision\ + recall} \tag{5}$$

This project used co lab to implement the project. Co lab is a cloud based platform that allows writing and executing python code [19].This platform is well suited for machine learning, data analysis and education. It is a free Jupiter notebook environment and runs in the cloud. Google Colab supports free GPU and libraries such as Keras, PyTorch, Tensor Flow and Open CV. Table 1 Shows the comparison of different architectures.

TABLE 1 Comparison of different architectures

Method	**Accuracy**	**Precision**	**Recall**	**F1 Score**
Combined Dense- SENet	79.5%	93.9%	71.8%	80.3%
Combination of Deeply Supervised ResNet and SENet with 6 ResNet blocks and five SENet blocks.	88.06%	88.14%	97.18%	92.44%
Fusion of Deeply Supervised ResNet and SENet with six ResNet blocks and one SENet blocks.	91.84%	92.98%	91.79%	92.39%
Fusion of Deeply Supervised ResNet and SENet with six ResNet blocks and two SENet blocks.	91.66%	90.43%	96.92%	93.56%
Fusion of Deeply Supervised ResNet and SENet with six ResNet blocks and three SENet blocks.	89.58%	87.01%	97.9%	92.15%
Deeply Supervised ResNet	88.30%	89.52%	92.05%	90.77%

The proposed system implement on top of the implementations of deeply supervised ResNet and SENet. On 8GB Ram dual core Intel core i5 8th generation processor takes 15 days to train the dataset. As it takes long time for training, the program is upgraded to implement it in Colab. In Colab, training takes only about 2.5 days.

During the training time, one model is implemented by combining DenseNet and SENet called as Combined Dense- SENet. The performance of the network was very poor and hence moved on to the combination of deeply supervised ResNet and SENet. Then the next experiment consists of six deeply supervised ResNet blocks and five SENet block. The accuracy obtained is 90%.Then by reducing the number of SENet bocks to one, the accuracy obtained is 93%. Then the next experiment consists of six deeply supervised ResNet blocks and Two SENet results a drop in accuracy to 91.66%. Accuracy drops to 89.5% by adding one more layer to SENet.

The accuracy obtained for six deeply supervised ResNet alone was 83.30%. So by trial and error method it is clear that six deeply supervised ResNet along with one SENet blocks shows better accuracy than that of other combinations. The accuracy obtained by Combined Dense- SENet is 79.48%. The accuracy obtained by Fusion of Deeply Supervised ResNet and SENet with six ResNet blocks and five SENet blocks is 90.06% and the accuracy improved to 90.55% as the number of SENet blocks become one. The comparison shows that the fusion of deeply supervised ResNet and SENet with six ResNet blocks and one SENet blocks shows better accuracy than the other two networks.

V. CONCLUSIONS

At present CNN plays a vital role in the classification and prediction of images. With the advancement in technology different CNN architecture has evolved over time. A combination of deep ResNet and SENet improves the accuracy of the system. The output produced a validation accuracy of 91.84% on chest x-ray images from Kaggle. But still it cannot address the problem related to the poor performance the image fed is noisy or if the resolution of the image is low.

REFERENCES

[1]. T. Guo, J. Dong, H. Li and Y. Gao, “Simple convolutional neural network on image classification”, IEEE 2nd International Conference on Big Data Analysis, 2017, pp. 721-724.

[2]. I. Aniemeka, “A Friendly Introduction to Convolutional Neural Networks”, Hashrocket, 2017. (Accessed from https://hashrocket.com/blog/posts/a-friendly-introduction-to-convolutional-neural-networks)

[3]. O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma and L. Fei-Fei, “Imagenet large scale visual recognition challenge”, International journal of computer vision, Vol. 115, No. 3, 2015, pp. 211-252.

[4]. S. Albawi, T.A. Mohammed and S. Al-Zawi, “Understanding of a convolutional neural network,” International Conference on Engineering and Technology, 2017, pp. 1-6.

[5]. R. Karim, “Illustrated: 10 CNN architectures. *towardsdatascience.com”, 2019, https://towardsdatascience.com/illustrated-10-cnnarchitectures95d78ace614d*

[6]. I. T. Job, "Image Classification with Deep Learning: A theoretical introduction to machine learning and deep learning". (Accessed from

https://medium.com/analytics-vidhya/image-classification-with-deep-learning-a-theoretical-introduction-to-machine-learning-and-deep-d118905c6d3a)

[7]. A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet classification with deep convolutional neural networks", Communications of the ACM, Vol. 60, No. 6, 2017, pp. 84-90.

[8]. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", arXiv preprint arXiv: 1409.1556, 2014.

[9]. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov and A. Rabinovich, "Going deeper with convolutions", In Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1-9.

[10]. K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition", In Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770-778.

[11]. C. Szegedy, S. Ioffe, V. Vanhoucke and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning", In Proceedings of the AAAI Conference on Artificial Intelligence, 2017, Vol. 31, No. 1.

[12]. F. Chollet, "Xception: Deep learning with depth wise separable convolutions", In Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 1251-1258.

[13]. S. Xie, R. Girshick, P. Dollár, Z. Tu and K. He, "Aggregated residual transformations for deep neural networks," In Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 1492-1500.

[14]. D.M. Blei, A.Y. Ng and M. I. Jordan, "Latent dirichlet allocation", The Journal of machine Learning research, No. 3, 2003, pp. 993-1022.

[15]. J. Hu, L. Shen and G. Sun, "Squeeze-and-excitation networks", In Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 7132-7141.

[16]. D. Zhang, W. Bu and X. Wu, "Diabetic retinopathy classification using deeply supervised ResNet", IEEE Smart World, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation, 2017, pp. 1-6.

[17]. J. Hu, L. Shen and G. Sun, "Squeeze-and-excitation networks", In Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 7132-7141.

[18]. https://www.pinterest.com/pin/210472982574094564/

[19]. https://heartbeat.fritz.ai/getting-started-withgoogle-colab-notebooks-117e2bb0c220