IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 3, March 2021

DOI: 10.17148/IARJSET.2021.8344

# Acute Lymphoblastic Leukemia Classification Using Convolutional Neural Networks and Transfer Learning

**Manas Garg**

Machinaa Technology, Bhilwara, India

**Abstract**: Leukemia is a blood cancer which is the most prevalent childhood cancer type and accounts for approximately 33% of all paediatric cancer. Some cases have occurred previously where zero symptoms are shown by blood until the disease has progresses and reached a dangerous level. This kind of a case mostly causes a misdiagnosis. To address such problem, this paper introduces Convolutional Neural Networks (CNN), which has led to break-through results in computer vision, and is thus a major technique that can be used to solve the Leukemia Classification Challenge. The ability of discovering abstract features with the capability of discrimination of different aspects of interests is possessed by the CNN algorithm. A diagnosis of the disease at early stage leads to an effective treatment. Segmentation from microscopic images has been performed so that the cells represent images in real world. The task is to identify the leukemic blasts at a premature stage. The dataset consists of 15,114 images (Training data = 10,661 images; Validation data = 1,867 images; Testing data = 2,586 images). The proposed method achieves accuracy up to 99% based on the number of epochs and data split.

**Keywords**: Leukemia, Classification, Convolutional Neural Networks (CNN), Transfer Learning

## I. INTRODUCTION

Leukemia is a blood cancer and it is caused by the rise in the quantity of white blood cells in the human body. The extra white blood cells (WBC) crowd out the red blood cells (RBC) and platelets that the human body needs to be healthy. White Blood Cells in excess are not good for the body.

The classification of Leukemia is done generally on the basis of progression speed. On the basis of the progression, the first Leukemia Classification is divided into two groups: acute leukemia and chronic leukemia. The determination of the second type of leukemia is done on the basis of the type of the White Blood Cell (WBC) that has been affected. This consists of Lymphocytic leukemia and myelogenous leukemia. When these two general classifications are combined, leukemia can be classified into four main types on the basis of the level of severity and the infected cells. The four types are: Acute Lymphocytic Leukemia (ALL); Chronic Lymphocytic Leukemia (CLL); Acute Myeloid Leukemia (AML); and Chronic Myeloid Leukemia (CML).

**Table 1: FOUR MAIN TYPES OF LEUKEMIA**

| | Lymphocytic Leukemia | Myelogenous leukemia |
|---|---|---|
| Acute | Acute Lymphoblastic Leukemia (ALL) | Acute Myeloid Leukemia (AML) |
| Chronic | Chronic Lymphocytic Leukemia (CLL) | Chronic Myeloid Leukemia (CML) |

Some researches, about leukemia classification have been done in the recent years. The researches however have been based on simple computer vision techniques. In this approach, the most common algorithm has many rigid steps which consist of: image pre- processing; clustering; morphological filtering; segmentation; feature extraction; classification; and evaluation.

Many of the authors in the research field have used common machine learning techniques such as K-means clustering for the detection and classification of blood cell in images. The conventional statistical features like energy, entropy, contrast, and correlation are extracted and are then given as inputs to machine learning models in most of the cases. It is clear that these “traditional” machine learning algorithms have various disadvantages. The time consumed in order to develop such algorithm is very high. More importantly, the decision of what type of features needs to be utilized for the maximization of the accuracy of the classification algorithm is very time and energy consuming.

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 3, March 2021

DOI: 10.17148/IARJSET.2021.8344

## II. PROPOSED METHOD

In this paper, I propose the use of deep learning to achieve the classification. Deep Learning can be used to extract high level features automatically and learn them while performing classification at the same time. The usage of a novel Convolutional Neural Network (CNN) architecture can be used to discriminate between the normal and abnormal blood cell images. CNN offers various advantages like largely reducing the processing time and also allows leaving majority of the pre-processing steps. Most importantly, it has the very important ability of feature extraction that is way better than the conventional statistical features, which will be shown in the paper further.

Developed from pattern recognition over time and the Artificial Intelligence learning theory, Machine Learning (ML) is a scientific field. Deep Learning, often known as deep neural network or DNN, is a Machine Learning branch and aims for the building of a model with the help of a deep graph which may be organized in single or multiple linear layers and non-linear layer transformations.

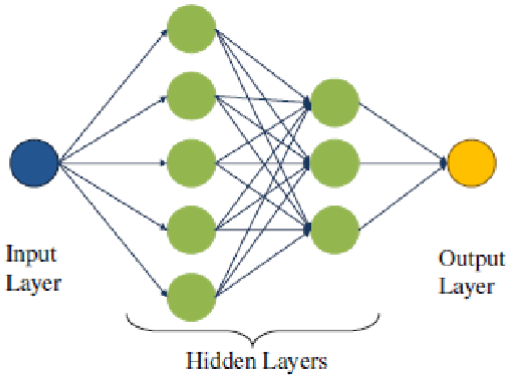


Fig. 1 A sample Neural Network

The figure above shows a small deep neural network consisting of an input layer, two hidden layers, and an output layer. Each circle represents a neuron and each arrow represents a weight. The output of the neuron is also a hidden layer and it is formed from the weighted sum of the inputs which consist of non-linear mapping like sigmoid, tanh, or others. The input instance is fed with $x = (x1, \ldots, xp)\, T$ by the input layer to the network.
When given an input, $x \in Rp$, the Machine Learning algorithm aims to build the prediction function $\theta(x; \omega)$, and is parameterized by $\omega$. The learning problem is to find the best possible instance for the prediction function and the following equation solves the optimization problem [1].

$$min_{\omega} \mathrm{E}_{(\mathrm{X,Y})}[l(\theta(\mathrm{x};\, \mathrm{m}), \mathrm{y})]$$

Fig. 2 Optimization Problem Equation

However, the determination of exact and precise knowledge of the true distribution of data is impossible to know. Hence, the general approach is the sampling of n data points from the unknown distribution, and then minimizing the empirical loss.

$$min_{\omega} \frac{1}{n} \sum_{\mathrm{i}=1}^{\mathrm{n}} l(\theta(\mathrm{x_i};\, \mathrm{m}), \mathrm{y_i})$$

Fig. 3 Empirical Loss Minimization

The CNN is a widely used deep neural network. The introduction of the Graphics Processing Unit (GPU) – a highly parallel programmable unit – visual recognition on a large scale has become easier as the computing power has increased. The CNN architectures have been evolving constantly as the computing power has been developing along with increase in the large-scale hierarchical image database. The usage of fewer parameters and the ability of storing just the temporal features of the input which reduces memory consumption to a great extent has been a specific reason in the success of the CNN algorithm.
Convolutional Neural Networks have three major architectural frameworks known as shared weights, local receipts, and spatial sub-sampling. The extraction of multiple feature maps by sliding around the same unit sets all over the input

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 3, March 2021

DOI: 10.17148/IARJSET.2021.8344

is allowed by local receptive fields. This property of CNN makes it very robust for the distortion and translation of the input. The feature map uses the shared weights, biases which are the reason for the reduced memory needs and learned parameters. The spatial sub-sampling, for the avoidance of output sensitivity under shifts and rotation, reduces the resolution of the feature maps.

The three major types of layer in CNN architecture are:

1) Convolutional layer: The parameters in this layer are learnable weights and biases and are shared in the depth of the input. As the weights slide through the input, feature maps are created and the dot product of the filter and input are calculated. This process is known as "convolution" and further the non-linear activation function activates the feature maps.

2) Pooling layer: To reduce the dimensionality, a down sampling operation is performed by the layer in the feature maps.

3) Fully-connected layers: They get the stacked Convolutional layer outputs and then calculate the weighted sum of the inputs consisting of a non-linear mapping.

## III. DATA PREPARATION AND ARCHITECTURE

The image Fig. 4 is an image of a cell affected with Acute Lymphoblastic Leukemia (ALL). The image Fig. 5 is of a normal cell.

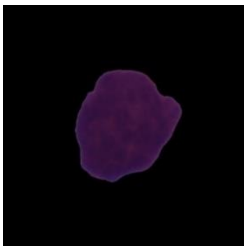

Fig. 4 ALL IMAGE

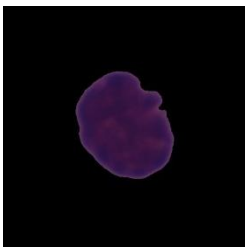

Fig. 5 HEM IMAGE

As clearly evident, the distinguition between a cancerous and non-cancerous cell is not an easy job just on the basis of the viewing from naked eye. The deep neural networks are certainly a very efficient technique to differentiate between the cancerous and non- cancerous cells.

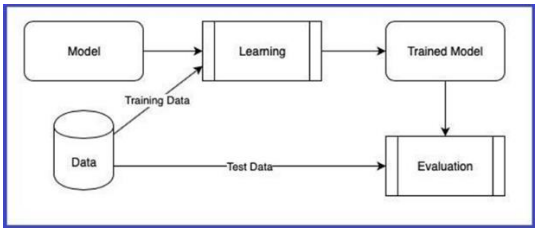


Fig. 6 Data Usage Model

The CNN architecture could be built in various different ways by alternating the sequence of the Convolutional (CONV) layer, pooling layers (POOL), and the fully connected layers (FC). The usage of an optimal CNN architecture is must for the algorithm to run successfully. The architecture of the model:

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 3, March 2021

DOI: 10.17148/IARJSET.2021.8344

```
Model: "functional_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 224, 224, 3)]     0
_________________________________________________________________
block1_conv1 (Conv2D)        (None, 224, 224, 64)      1792
_________________________________________________________________
block1_conv2 (Conv2D)        (None, 224, 224, 64)      36928
_________________________________________________________________
block1_pool (MaxPooling2D)   (None, 112, 112, 64)      0
_________________________________________________________________
block2_conv1 (Conv2D)        (None, 112, 112, 128)     73856
_________________________________________________________________
block2_conv2 (Conv2D)        (None, 112, 112, 128)     147584
_________________________________________________________________
block2_pool (MaxPooling2D)   (None, 56, 56, 128)       0
_________________________________________________________________
block3_conv1 (Conv2D)        (None, 56, 56, 256)       295168
_________________________________________________________________
block3_conv2 (Conv2D)        (None, 56, 56, 256)       590080
_________________________________________________________________
block3_conv3 (Conv2D)        (None, 56, 56, 256)       590080
_________________________________________________________________
block3_conv4 (Conv2D)        (None, 56, 56, 256)       590080
_________________________________________________________________
block3_pool (MaxPooling2D)   (None, 28, 28, 256)       0
_________________________________________________________________
block4_conv1 (Conv2D)        (None, 28, 28, 512)       1180160
_________________________________________________________________
block4_conv2 (Conv2D)        (None, 28, 28, 512)       2359808
_________________________________________________________________
block4_conv3 (Conv2D)        (None, 28, 28, 512)       2359808
_________________________________________________________________
block4_conv4 (Conv2D)        (None, 28, 28, 512)       2359808
_________________________________________________________________
block4_pool (MaxPooling2D)   (None, 14, 14, 512)       0
_________________________________________________________________
block5_conv1 (Conv2D)        (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_conv2 (Conv2D)        (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_conv3 (Conv2D)        (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_conv4 (Conv2D)        (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_pool (MaxPooling2D)   (None, 7, 7, 512)         0
_________________________________________________________________
flatten (Flatten)            (None, 25088)             0
_________________________________________________________________
dense_2 (Dense)              (None, 256)               6422784
_________________________________________________________________
dropout_1 (Dropout)          (None, 256)               0
_________________________________________________________________
dense_3 (Dense)              (None, 2)                 514
=================================================================
Total params: 26,447,682
Trainable params: 6,423,298
Non-trainable params: 20,024,384
_________________________________________________________________
```

Fig. 7 CNN Architecture used

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 3, March 2021

DOI: 10.17148/IARJSET.2021.8344

The validation_accuracy has been chosen as a metric to measure accuracy of the model rather than the training accuracy as it is often misleading.

| | | TABLE 2: PREDICTED CLASS | |
|---|---|---|---|
| | | **ALL (+)** | **HEM (-)** |
| **ACTUAL CLASS** | **ALL (+)** | *2213 (++)* | *14 (-+)* |
| | **HEM (-)** | *22 (+-)* | *2204 (--)* |

The confusion matrix shown above shows that the deep learning model correctly classified 2213 ALL images correctly out of 2227. It also correctly classifies 2204 HEM images out of 2226 images.
The accuracy of the deep learning model is calculated by the function showed in Fig 8 below:

$$Accuracy = \frac{[(+\,+) + (-\,-)]}{[(+\,+) + (-\,-) + (+\,-) + (-\,+)]}$$

Fig 8. Accuracy Metric Function

Accuracy = [(2213) + (2204)] / [(2213) + (2204) + (22) + (14)]
Accuracy ≈ 99.19%

## IV. CONCLUSION

Detecting earlystage Leukemia can be very helpful for an effective successful treatment. The proposed deep neural network model for the classification of leukemia-affected and leukemia-free cells uses Convolutional Neural Networks and Transfer Learning and has an accuracy of 99.19% with respect to various performance metrics. Trained over 10,000 images and tested over 4,000 images, the model is commendable for use in healthcare. It is a more efficient process in terms of both accuracy and speed.

## REFERENCES

[1]. Joseph, T. The Mathematics Behind Deep Learning. 11 Sept. 2020.

## BIOGRAPHY

I am an IBDP (International Baccalaureate Diploma Programme) Year 2 student and hold the post of Head Boy at Sangam School of Excellence. Being a topper since always, I am also a guitarist, basketballer and a passionate programmer.
I am a natural leader and possess the ability to think clearly and maintain composure even in the most stressful times which helps me to carefully resolve any predicament that comes in my way. To make the best of any situation that life throws at me, I handle it with great dexterity.
Along with a strong command of international affairs in my arsenal, I also have a clever thinking and superb oration. Once I put my mind to a task, no impediment can deter me from reaching the goal. From whatever studied, and all the extra courses done, I aim to bring an impact in the world as I believe there is no significance of education until & unless it is used to bring a change in the world and help people across the globe.