IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
IJARSCT
Impact Factor: 6.252
Volume 2, Issue 6, June 2022

Review on Case Study of Image Classification using CNN

Pratik Yadav[1], Naveen Madur[2], Yogesh Dongare[3], Vishwajeet Rajopadhye[4], Mr.Shrikant Salatogi[5]
UG Scholars, Department of Computer Science and Engineering[1,2,3,4]
Guide, Department of Computer Science and Engineering[5]
SVERIs College of Engineering, Pandharpur, India

Abstract: *Traditional neural networks though have achieved appreciable performance at image classification, they have been characterized by feature engineering, a tedious process that results in poor generalization to test data. In this report, we present a convolutional neural network (CNN) approach for classifying CIFAR-10 datasets. This approach has been shown in previous works to achieve improved performances without feature engineering. Learnable filters and pooling layers were used to extract underlying image features.*

Keywords: Classification using CNN, Feature engineering, CIFAR-10, Pooling

I. INTRODUCTION

The ability to classify things correctly requires many hours of training. People get things wrong many times, until eventually, they get it right. The same structure applies to machine learning. By using a high-quality set of data, deep learning can classify objects comparatively well or even better than humans can. With achieving utterly accurate image classifier, some of the monotonous jobs could be replaced by machines, so that humanity could focus on the most enjoyable activities.Using the improved model in which 2D convolutional layer replacing the maxpooling and dense function of existing model. The classification of 10 objects from CIFAR-10 dataset, with convolutional layer we can achieve maximum accuracy with increased classification rate.

II. OBJECTIVE

One limitation of traditional Artificial Neural Networks (ANN), particularly those with increasing depth, is they can become stuck in local minima, where training ceases to improve performance even though better solutions are available. Neural networks can give very precise solutions; however, this depends on the network being initialized in a suitable area of the search space. RBM techniques address this problem, using unsupervised learning to initialize the network according to significant features of the environment, followed by a method to fine-tune performance according to the task, which can be done using back-propagation. This allows initialization of a deep network that is not likely to give adequate behavior from random initialization of weights. However, RBM networks have shown limitations in terms of reliability, and a number of attempts have been made to improve discovery of features. Evolutionary Computation provides an alternative approach to machine learning, usually based on genetic algorithms and reinforcement techniques. Evolutionary systems tend to be very reliable at finding a good solution, however the use of random variation, rather than gradient descent used in ANNs, often does not provide the same precision found with neural networks or kernel methods. Learning Classifier Systems (LCS) are an evolutionary technique that combine evolutionary processes with reinforcement learning, to maintain a population of classifiers that collectively model the observed system. The Genetic Algorithm used by many LCS approaches follows an evolutionary analogy, however the process of capturing a population of rules based on reinforcement can be viewed as an analogy of cognitive learning processes, with a greater degree of generalization than Reinforcement Learning.

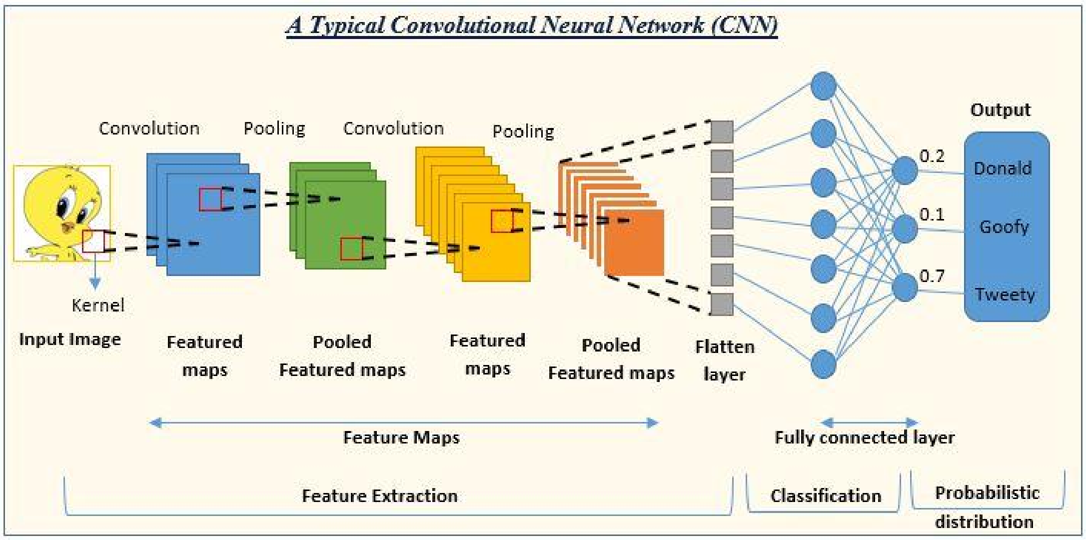
III. LITERATURE REVIEW

Before network training, the traditional convolutional neural network model weighs all layers at one time and then updates weights by back-propagation. This process improves accuracy during training. If we increase the depth of neural network for higher computation the higher computational power is required this leads to increase in cost as well.

Minimizing the loss during learning process we have trained model by supervised learning in which model is given with data samples and respective outcomes. When a model generates output, it compares it with the desired outputs the try to bring output close to desired output.

Accuracy of model is improved by optimization algorithm through several cycles until convergence. For optimization in previous network we used adam with 100 epochs which gives the lower accuracy than improved convolutional neural network. Adam optimizer performs better than others like RMSProp, adagrad, adadelta, etc.

CNNs use convolutional layers that, for each kernel, replicate that same kernel's weights across the entire input volume and then output a 2D matrix, where each number is the output of that kernel's convolution with a portion of the input volume. So we can look at that 2D matrix as output of replicated feature detector. Then all kernel's 2D matrices are stacked on top of each other to produce output of a convolutional layer.



IV. METHODOLOGY

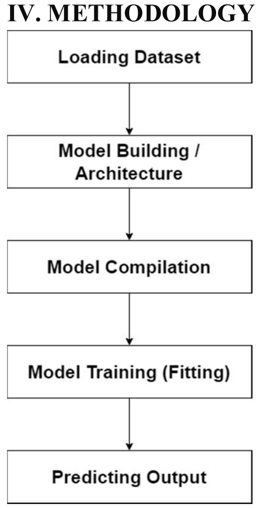


Figure 1: Outline of Project model

4.1 Loading Training Datasets

Manually checking and classifying images could be a tedious task especially when they are massive in number and therefore it will be very useful if we could automate this entire process using computer vision. The advancements in the field of autonomous driving also serve as a great example of the use of image classification in the real-world. The applications include automated image organization, stock photography and video websites, visual search for improved product discoverability, large visual databases, image and face recognition on social networks, and many more; which is why, we need classifiers to achieve maximum possible accuracy. The training dataset consists of 50,000 32X32 colour images of 10 different objects. These are the different classes of images in the dataset.

4.2 Building Simple Neural Network Model

A. Artificial Neural Network

We have built an Artificial neural network using sequential model in which we have used flatten and dense layers with activation functions like 'relu' and 'sigmoid'. From which we can able to get less than 50% of accuracy so we have decided to make a new Convolutional model with accuracy greater than the traditional artificial neural network model.

B. Convolutional Neural Network (Baseline Model)

For achieving accuracy greater than artificial neural network we have created a new model using convolutional neural network using sequential model by adding Conv2D , MaxPooling2D layers into the model. Conv2D is a 2D Convolution Layer, this layer creates a convolution kernel that is wind with layers input which helps produce a tensor of outputs.This layer creates a convolution kernel that is convolved with the layer input to produce a tensor of outputs. If use_bias is True, a bias vector is created and added to the outputs. Finally, if activation is not None, it is applied to the outputs as well.When using this layer as the first layer in a model, provide the keyword argument input_shape (tuple of integers or None, does not include the sample axis), e.g. input_shape=(32, 32, 3) for 32x32 RGB pictures.

4.3 Model Building (CNN Architecture)

- **Sequential:** The sequential API allows you to create models layer-by-layer for most problems. It is limited in that it does not allow you to create models that share layers or have multiple inputs or outputs
- **Dropout:** The term "dropout" is used for a technique which drops out some nodes of the network. Dropping out can be seen as temporarily deactivating or ignoring neurons of the network. This technique is applied in the training phase to reduce overfitting effects.
- **Dense:** A Dense layer feeds all outputs from the previous layer to all its neurons, each neuron providing one output to the next layer. It's the most basic layer in neural networks. A Dense (10) has ten neurons
- **Flatten:** This Python package provide a function flatten () for flattening dict-like objects. It also provides some key joining methods (reducer), and you can choose the reducer you want or even implement your own reducer.
- **Conv2D:** Keras Conv2D is a 2D Convolution Layer, this layer creates a convolution kernel that is wind with layers input which helps produce a tensor of outputs.
- **MaxPooling2D:** Max pooling is a sample-based discretization process. The objective is to down-sample an input representation (image, hidden-layer output matrix, etc.), reducing its dimensionality and allowing for assumptions to be made about features contained in the sub-regions binned

4.4 Model Compilation

As we have defined our model architecture and then assign loss function, optimizer, and metrics in our model by using Sequential().compile(…) or in this case model.compile() method.We need to keep a check on the loss and accuracy of our MLP withCNN so that we can be sure that our defined MLP with CNN is working good and going towards minima.This can be done using tensorboard, to access tensorboard from thecolab environment.

4.5 Model Training (Fitting)

Model fitting is the measure of how well a machine learning model generalizes data similar to that with which it was trained. A good model fit refers to a model that accurately approximates the output when it is provided with unseen inputs. Fitting refers to adjusting theparameters in the model to improve accuracy.

IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

IJARSCT
Impact Factor: 6.252

Volume 2, Issue 6, June 2022

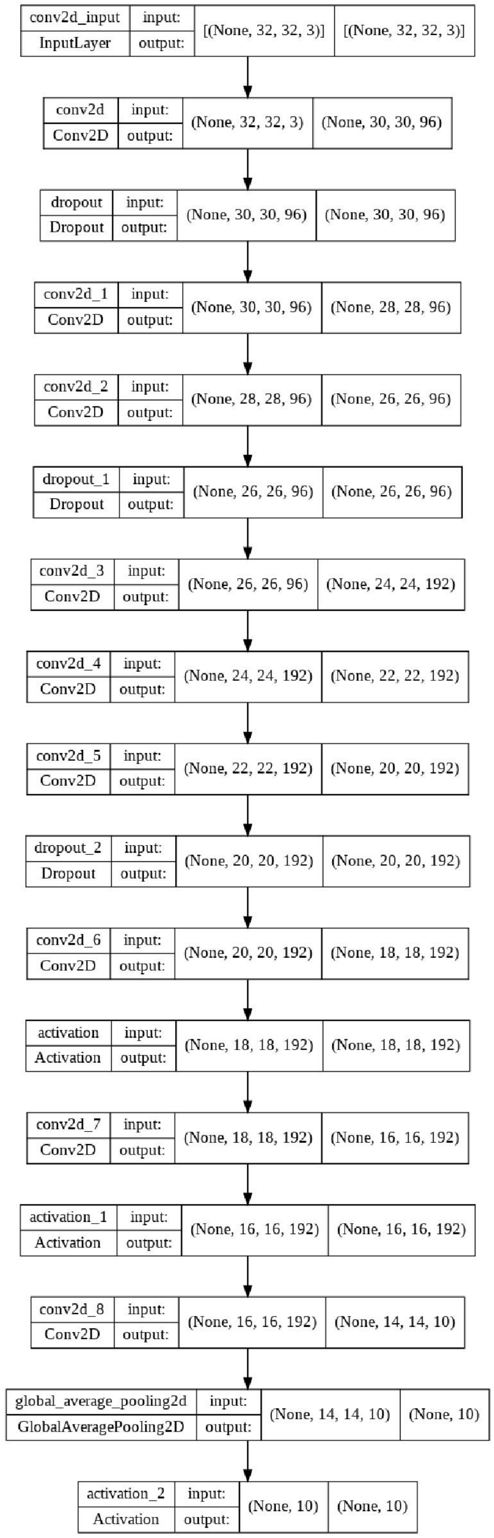


Figure 2 : Layers of CNN Model

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Impact Factor: 6.252
Volume 2, Issue 6, June 2022

4.6 Making Prediction / Experimental Outcomes

The results obtained in this research are important as they indicate that it is possible to increase the accuracy of CNN models by easily running and modifying its traditional structure with the use of programming language We have shown the correct output obtained from model predictions but few of them are Incorrect

Correct Predictions

Incorrect Prediction

V. CONCLUSION

Our research compared various models that used different deep learning techniques and architectures. Feature extraction using the Convolutional Neural network showed the best performance of all the models. We utilized the best deep learning practices to successfully accomplish our goal to classify images present in smart devices. This classifier opens up the opportunity to better manage and understand the plethora of images that make way into our smart devices. Moreover, it paves the way to efficiently analyze the images that are shared across social media platforms. Our approach also provides easy flexibility to accommodate other broad categories of images that might emerge in the future or if a particular category is required to conquer a specific problem.

ACKNOWLEDGMENT

The assistance provided by Mr. Shrikant C.Salotagi throughout the project to our group was greatly appreciated.

REFERENCES

[1]. Tensorflow : Official website
[2]. YouTube : Official website