

DEEP LEARNING BASED PREDICTING IMAGES USING CONVNET

Abstract

In this paper, we will be learn about how to predict cats and dogs using convolutional neural networks (CNN) model. In this project we will be using tensorflow and keras python modules. Import Image Generator from the keras module. Along with these there will be need of the dataset which contains dogs and cats which must be created on the requirement of user. Firstly, import the required libraries which are mentioned above along with matplotlib library which will be useful in plotting the loss, accuracy, val_loss, val_accuracy. Secondly, build a CNN model in which the building of CNN model contain initialization, convolution, pooling, flatten, full connection, output layer. After completion of building the model, compile the model, and train the model. Fourthly, write the code to make the prediction. And the model's accuracy will be depended on the number of epochs provided while training the model. Lastly plot the accuracy provided by the epochs.

Keywords: convolutional layer, pooling layer, fully connected layers

Authors

Chandra Sekhar Koppireddy

Assistant Professor
Department of Computer Science and Engineering
Pragati Engineering College
Kakinada, India
chandrasekhar.koppireddy@gmail.com

Veeravalli Sohan Venkata Satvik

Student of Computer Science and Engineering –DS
Pragati Engineering College
Kakinada, India
veeravallisohan@gmail.com

I. INTRODUCTION

As we can see that this paper is about “Predicting Cats vs Dogs using CNN”. We are hearing a lot about CNN in deep learning problems. Some of them know about CNN and some others they don't. In this paper also there is a constant repeating which is CNN. Everyone may think “What is meant by CNN?”, “Why do we use CNN?”, “What is the purpose of CNN?” etc. Like these there will be number of questions raising in the minds of people who are reading this paper.

Let's see the answers for the above questions:

CNN: General abbreviation of CNN is stated as “CONVOLUTIONAL NEURAL NETWORKS”. Let's look at the definition: These are specialized type of artificial neural networks which use a mathematical operation called as convolution instead of general matrix multiplication in at least one of their layers.

As the title suggests predictions of two animals which will be done by providing number of images to predict, so CNN is specifically designed to process pixel data and are used in image processing and recognition.

The architecture consists of CONVOLUTIONAL LAYERS, POOLING LAYERS, FULLY CONNECTED LAYERS, RECEPTIVE FIELD, WEIGHTS. This model also follows the above mentioned architecture to get the desired output of our own. As we can also see the building blocks of CNN, some are mentioned in architecture, they are CONVOLUTIONAL LAYER, POOLING LAYER, ReLU LAYER, FULLY CONNECTED LAYER, LOSS LAYER. Convolutional Neural Networks is a part of deep learning.

As mentioned earlier, CNN is used for the purpose of image processing and recognition, because, CNN does not lead to change of pixels from the images provided by the user.

The purpose of CNN image classification is “The system uses supervised machine learning classification algorithm for the purpose of deconstructing of an image and identifying its distinct features”.

The general applications of CNN are Image recognition, Natural Language Processing, Video analysis, Anomaly detection, Drug discovery etc. These are few answers for the questions which are mentioned earlier.

II. RELATED WORK

As mentioned earlier, CNN is mainly used as image processing and recognition. Not only processing, this can also be used for predictions like stock predictions, image predictions etc. Majority of this is used in stock prediction. There are many works related to this CNN model, not only in image processing but also along with object or image prediction, stock predictions, health predictions like heart disease predictions.

III. BACKGROUND

Present working model is about learning how to perform image prediction by providing image datasets to train a CNN model. In this section let's review convolution neural networks which is the main element of our framework.

- 1. Convolutional layer:** For CNN model, convolutional layer is the basic block. The convolutional layer perform convolution operation on the data. The *parameters* of this layer consists of a set of learnable filters, which have a small receptive field, but extend through the full depth of the input volume. Generally, convolution operation is an algorithm used to measure the changes caused by applying filter on the input. The coverage of a filter is shown by the size of that filter.
- 2. Pooling layer:** Followed by Convolutional Layer, Pooling layer is the next important concept in CNN which is a form of non-linear down-sampling. MAX POOLING is the most common non-linear function which is used for implementing pooling. Max pooling partitions the input image into set of rectangles and for each sub-region the maximum output is printed. This operation is useful for reducing the computational cost of learning process. Along with this, this operation can be used for fixing overfitting problem in CNN. Pooling layer helps to reduce the risk of overfitting.
- 3. Full connected layer:** After several convolutional and max pooling layers, the last classification is done via fully connected layer. This layer is responsible for converting extracted features in the previous layers to find the final output.
- 4. Output layer:** Earlier mentioned layers, convolutional and pooling layers are only used to extract the features and reduce the parameters from the original images. The output layer has a loss function like categorical cross-entropy, to compute the error in prediction.

IV. FRAME WORK

In this section, we will look into the algorithm. Starting of this, we will be needing to import the required libraries like Numerical python (Numpy), matplotlib, mainly keras and tensorflow. Followed by this we have to load the data from user's system. From keras module import Image Generator function and fill the required parameters to perform this operation on the dataset provided. We can create any dataset of our own requirement. The dataset which I have used for this is "Cats vs Dogs" which consists of 8000(4000 cats and 4000 dogs images) images of training set and 2000 images of test set. Along with these sets we will be having images to perform predictions. After loading the datasets, we have to start building a CNN model. Firstly we need to initialise the CNN model. Secondly we have to perform convolution operation and add a convolutional layer using Conv2D() function. Along with it add a pooling layer using MaxPool() function. Add one more layer of these two layers. Then perform flattening using Flatten() function on the layers. Followed by use Dense() function to perform full connection and for the output layer. After completion of building the model compile the model using compile() function. Then we have to train the model by fit() function and provide epochs as one of the parameter in fit() function. More the number of epochs, more the value of accuracy and the value of error will be low. After training the model write a piece of code to perform predictions and provide the model with the image the user wants to predict.

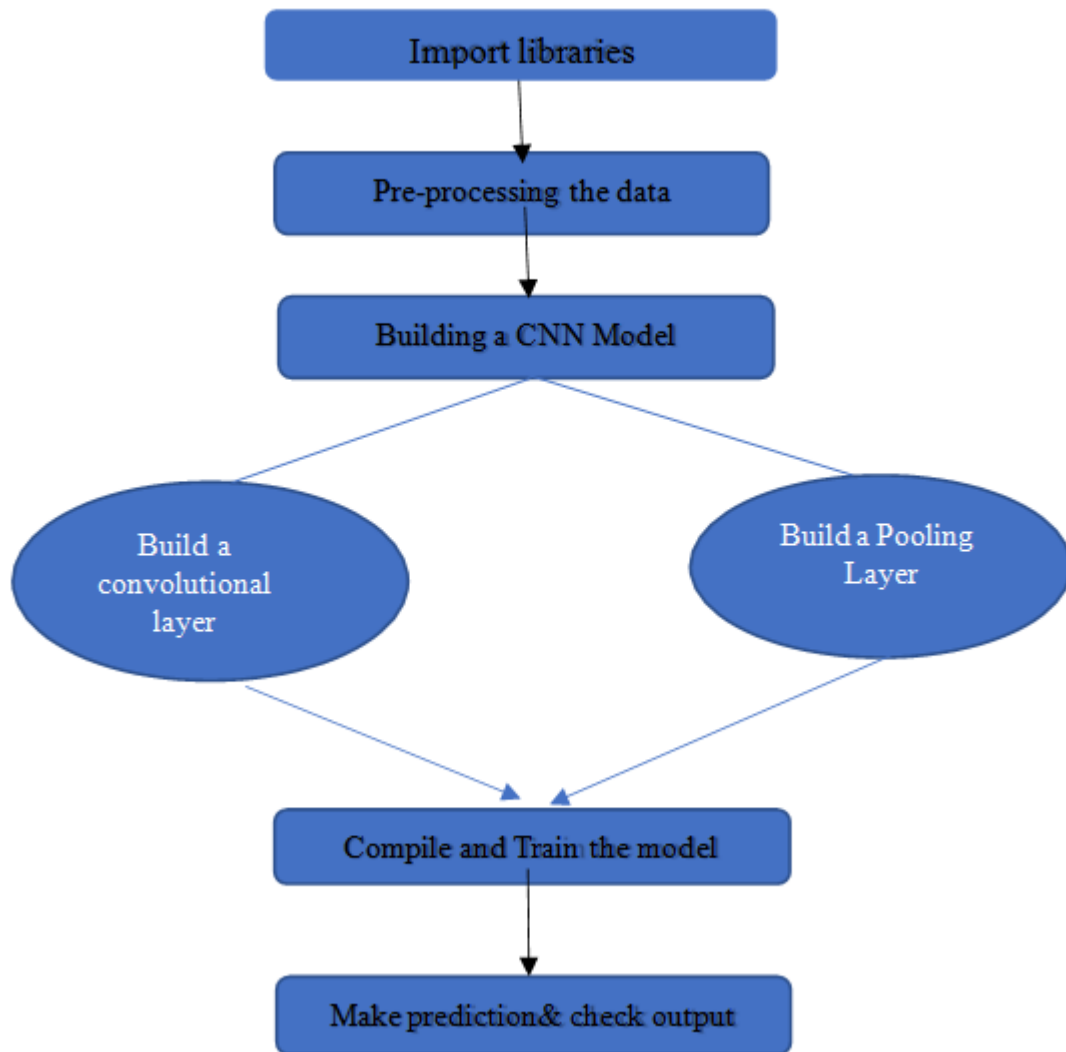


Figure 1: Flow Chart about the Model Algorithm.

Let us consider an example for the above framework.

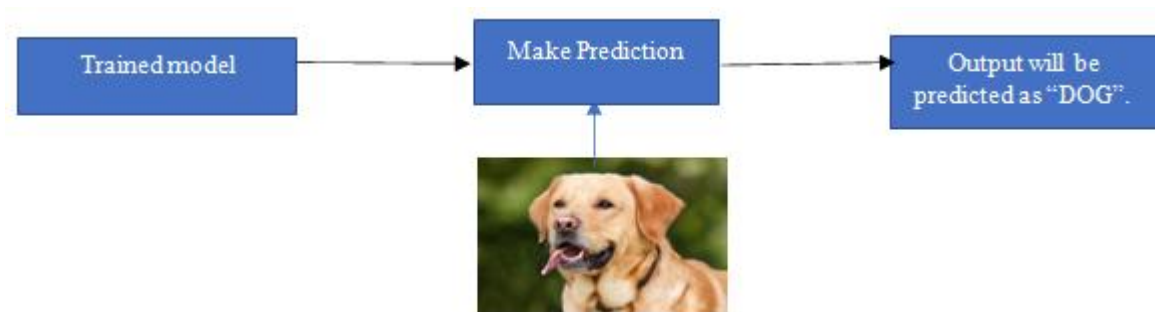


Figure 2(a): Representation of the Model Output (Dog)

Like above we can perform the prediction for “CAT” image also.

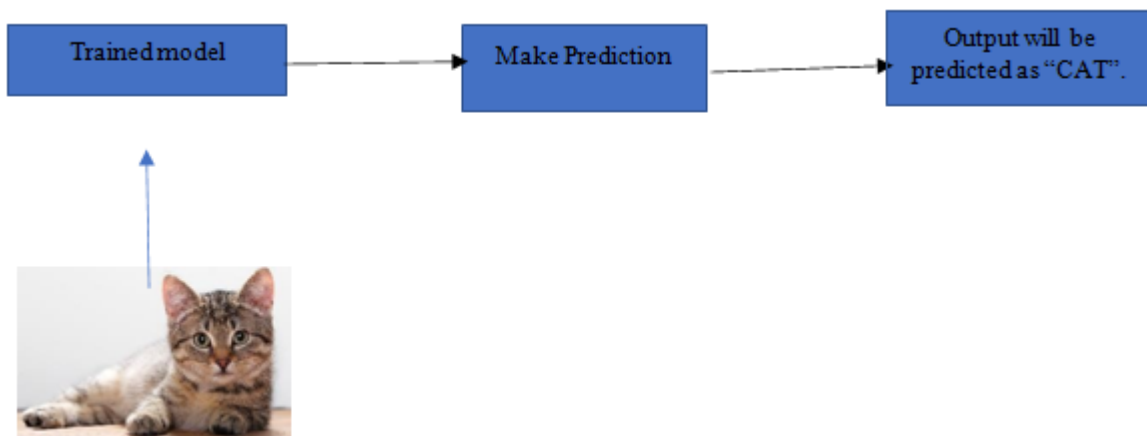


Figure 2(b): Representation of the Model Output (Cat)

These are the pictorial presentation of algorithm as mentioned above in the framework section.

After performing the above mentioned steps, plot the loss, accuracy, val_loss and val_accuracy. On plotting these values we will see the plot something like this:

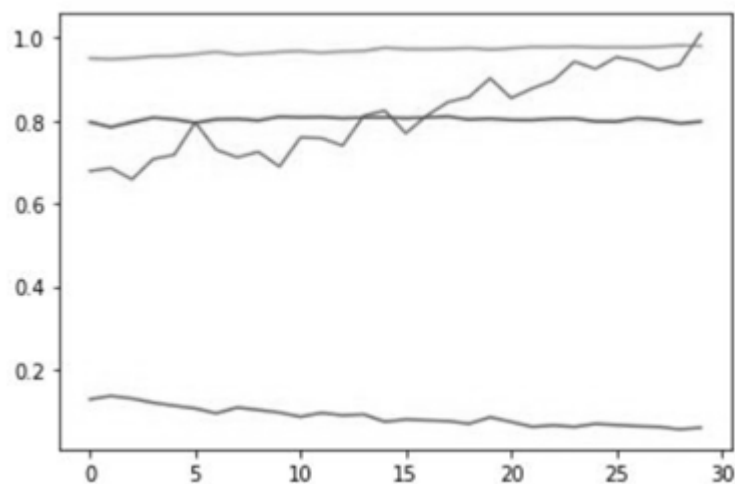


Figure 3: Plotting of the Given Values.

The above shown figure is the plot of the values loss, accuracy, val_loss, val_accuracy. The above shown plot is for 30 epochs and these values are given by loss: 0.0552, accuracy: 0.9811, val_loss: 0.9346, val_accuracy: 0.7930. The blue line indicates loss, yellow indicates accuracy, red indicates val_accuracy, green indicates val_loss.

V. CONCLUSION

In this paper, there is detail explanation of how to perform image prediction using CNN model. This is the general approach to perform prediction of an image provided by the user. In this way we can perform many other image prediction not only cats and dogs but also any other images of the users requirement.

REFERENCES

- [1] Dave Steinkraus; Patrice Simard; Ian Buck (2005). "Using GPUs for Machine Learning Algorithms". 12th International Conference on Document Analysis and Recognition (ICDAR 2005)
- [2] Habibi, Aghdam, Hamed (2017-05-30). Guide to convolutional neural networks: a practical application to traffic-sign detection and classification. Heravi, Elnaz Jahani. Cham, Switzerland.
- [3] He, Kaiming; Zhang, Xiangyu; Ren, Shaoqing; Sun, Jian (2016). "Deep Residual Learning for Image Recognition" (PDF). 2016 IEEE Conference on Computer Vision and Pattern Recognition
- [4] van den Oord, Aaron; Dieleman, Sander; Schrauwen, Benjamin (2013-01-01). Burges, C. J. C.; Bottou, L.; Welling, M.; Ghahramani, Z.; Weinberger, K. Q. (eds.). Deep content-based music recommendation (PDF). Curran Associates, Inc. pp. 2643–2651. - -
- [5] Bengio, Yoshua; Lamblin, Pascal; Popovici, Dan; Larochelle, Hugo (2007). "Greedy Layer-Wise Training of Deep Networks" (PDF). Advances in Neural Information Processing Systems: 153–160