Convolution Neural Network for Image classification

Zhihao Shi& Zigang Geng

University of Science and Technology of China

szh52793@mail.ustc.edu.cn&aa397601@mail.ustc.edu.cn

Introduction

Neural network is a powerful tool to deal with image classification. In image classification, a simple CNN can gain a surprising result.

Stochastic gradient descent algorithm is a popular method to optimize the loss function in machine learning problems.

Materials and Methods

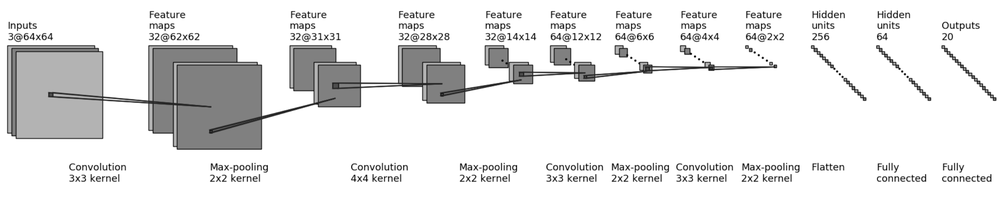
Material

The given dataset is cultured from https://tiny-imagenet.herokuapp.com/the tiny ImageNet dataset. This dataset contains 10,000 images in total, which are divided into 20 classes. All training images are color images, and each of them has 64 * 64 pixels.The class labels are as follows:

label index	1	2	3	4	5
text description	goldfish	frog	koala	jellyfish	penguin
label index	6	7	8	9	10
text description	dog	yak	house	bucket	instrument
label index	11	12	13	14	15
text description	nail	fence	cauliflower	bell paper	mushroom
label index	16	17	18	19	20
text description	orange	lemen	banana	coffee	beach

Network Framework

The model architecture, shown in figure 1, consists of four convolution Layers, four maxpooling layers and two fully connection layers.



The feature maps are as follows. (table1)

Table 1: The feature maps

Input	layer	Output	Parameters
64*64*3	conv3-32	62*62*32	3*3*3*32
62*62*32	maxpooling2	31*31*32	0
31*31*32	conv4-32	28*28*32	4*4*32*32
28*28*32	maxpooling2	14*14*32	0
14*14*32	conv3-64	12*12*64	3*3*32*64
12*12*64	maxpooling2	6*6*64	0
6*6*64	conv3-64	4*4*64	3*3*64*64
4*4*64	maxpooling2	2*2*64	0
2*2*64	flatten	256*1	0
256*1	fc-256*64	64*1	256*64
64*1	fc-64*20	20*1	64*20
20*1	softmax	20*1	0

Optimization

Acceleration of convolution

Before the convolution, we preprocessed the input data with "im2col". In this step, the 2-D input features can be concatenated into one long row (1-D) of the input matrix. And then, we can calculate the convolution by the matrix multiplication instead of the Hadamard product.

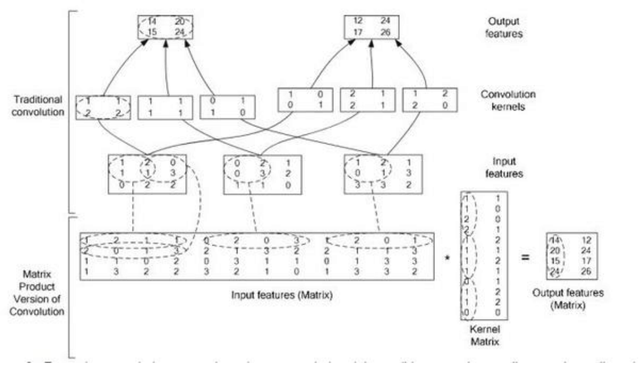


Figure 1: im2col

Dropout

To reduce overfitting, we used the "Dropout" when training.Using "dropout" reduced overfitting significantly. The method improved two percentage points.

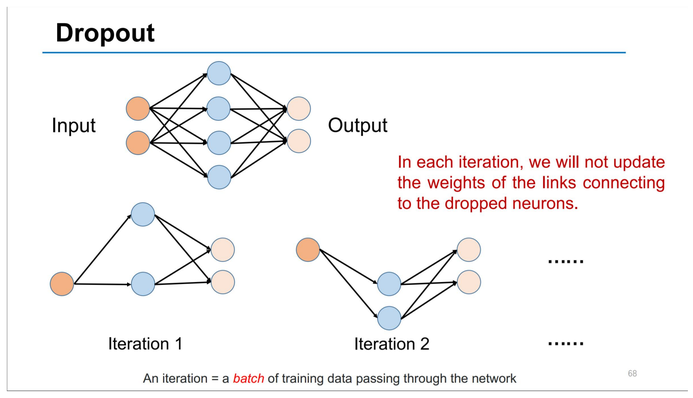


Figure 2: dropout

Self-ensemble

Ensemble is a common technology in training neural networks. In this project, we trained five neural networks. Every network will predict independently. Finally, we will use the mode of the prediction as the output. The method improved two percentage points.

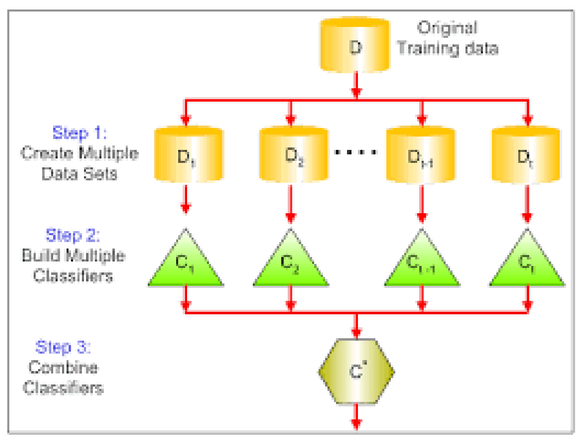


Figure 3: self-ensemble

Results

We randomly split the 10000 samples into two parts, the training set (8000 samples) and the validation set (2000 samples). We trained the model in the training set and validate its performance in the validation set.

We set learning rate as 0.0001. The training process is shown in the figure 4 and 5.

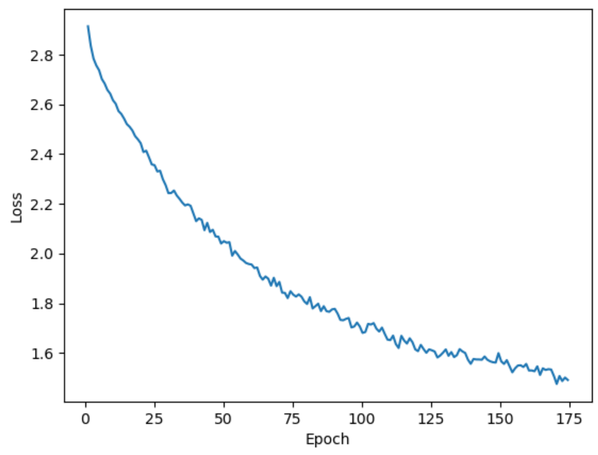


Figure 4: The training loss

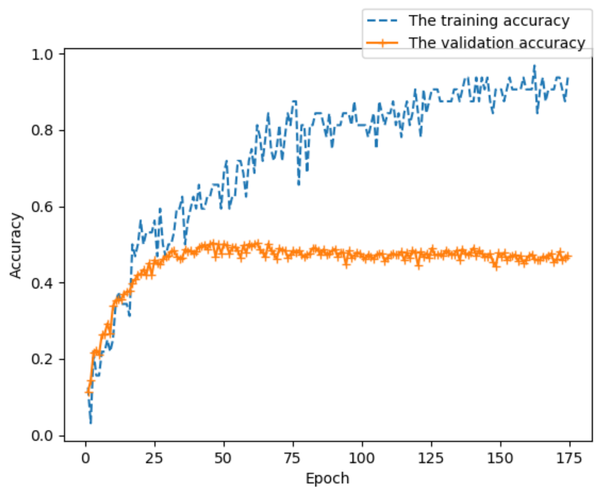


Figure 5: The accuracy

After training, the top-1 accuracy on the validation set was 50%, but the top-3 accuracy on the validation set was 65%.

Conclusions

In this project, we implemented a simple CNN to classify the given dataset. To reduce overfitting, we used "dropout". In order to speed up the training rate, we used SGD when training and accelerate the convolution operation. At last, the accuracy on the validation set was 65%.