

Kuba Kowalski

Neural Networks

Description to the project

1. Introduction

We are going to focus about the specific problem. Considered that having data set called CIFAR10 which contains set of labeled images, we will try to build neural net which recognizes image and gives a label. Our data is 4-dimensional (*batch, channel, width, height*) and the result is 2-dimensional (*batch, index*). Theano library helps me to implement my network.

2. Specifics

I have implemented a few types of layers:

a) Convolutions + ReLU

I connected this two and treated it like one, because it was easier for me. It kind of goes with together. I am using this layer for reducing *height* and *width* in our data and increase *channel* coordinate. Thanks to ReLU we have kind of non-linear function.

Arguments (*outputChannel, size*). First is how many you want to channel on output and the second is the size of „window” which will be looking on our window and each time will give us one point on output.

b) Pooling

Now, this layer reduces *height* and *width* coordinates twice or more by counting a mean adjacent points.

Arguments (*size*). It is the size of „window” of nearest neighbours.

c) Flatten + ReLU

Here again I joined two layers. It flattens our data to two dimensional.

Arguments (*outputChannel*). It is the size of flattened data (second one, because *batch* is „untouchable”).

d) SoftMax

It helps to choose the best option by normalize result.

When we know that, I can show how my neural network which looks like on table 1 (let say our batch size equals 3).

3. Results

In conclusion, my network works pretty good, it is quite simple, but works VERY slow. If there is something that I can work further on my project, this is it. However, I am satisfied. My success rate: 100.0%.

Tablica 1. Buliding neural net.

layer	current shape
Start	(3, 3, 32, 32)
Convolutions(16, 3)	(3, 16, 30, 30)
Pooling(2)	(3, 16, 15, 15)
Convolutions(25, 4)	(3, 25, 12, 12)
Pooling(2)	(3, 25, 6, 6)
Convolutions(32, 3)	(3, 32, 4, 4)
Filter(800)	(3, 800)
Filter(10)	(3, 10)
SoftMax()	(3, 10)