CSE 474 Introduction to Machine Learning

Project 1 Report

Haoquan Zhao

50189594

09/17/2018

Instead of using logic operation to solve the FizzBuzz question, we will use machine learning to solve the FizzBuzz problem. If the integer divisible by 3 the output should be *Fizz*, If the integer divisible by 5 the output should be Buzz. If the integer divisible by 15 the output should be *FizzBuzz*. Otherwise, the output should be *Other.*

First, turn each input into a vector of "activations", by converting the input number into binary. The output will be 0, 1, 2, 3 each representing “Other”, “Fizz”, “Buzz”, “FizzBuzz”. Then, we create our model with 1 input layer, 1 hidden layer, 1 output layer. After we train the model with the training data, the machine will be able to predict the result of each input number.

**1: How dropout rate affect accuracy.**

Dropout is a technique where randomly selected neurons are ignored during training. This means that their contribution to the activation of downstream neurons is temporally removed on the forward pass and any weight updates are not applied to the neuron on the backward pass. We expect our model will come up with a higher accuracy if we turn the dropout rate lower, but makes our model more likely to overfit the training data.

I run the model for 5 times and use the mean of 5 result as my final result.

Figure 1: Comparison of Various accuracies with different drop out.

**2: How number of neuron in first-dense-layer affect accuracy.**

(with default hyper-parameter setting)

A neuron in an artificial neural network is a mathematical approximation of a biological neuron. It takes a vector of inputs, performs a transformation on them, and outputs a single scalar value. It can be thought of as a filter. After 5 test I found out the accuracy is getting higher as I add more node in to the first dense layer.

Figure 2. Comparison of Various Accuracies for Different number of node in first dense layer

**3. How number of layer affect the accuracy.**

Layer in an artificial neural network is a layer in between input layers and output layers, where artificial neurons take in a set of weighted inputs and produce an output through an activation function.

I added several layers between first and output layer. They all have 256 nodes with activation function relu. Other setting are remain default. I use the mean of 5 test result. I found the accuracy becomes lower as I add more layer into the model.

**4. How different optimization methods affect accuracy.**

I run the model with each optimization method for 5 times and take the mean of the 5 result as the result.

**5. How different activation function for the output layer affect accuracy.**

The activation function is a function that limits the output signal to a finite value. Neural networks have to implement complex mapping functions hence they need activation functions that are non-linear in order to bring in the much needed non-linearity property that enables them to approximate any function.

I change the activation function on the output layer.

**CONCLUSION**

The software 1.0 using the conventional if else statement to solve this problem. Instead, for the software 2.0 we apply a machine learning model to solve the FizzBuzz problem. Software 2.0 using the training data created by us, and use the training data to train the machine learning model. The big our training data is, the better the model predict the FizzBuzz result.

Base on the result. I found that the more neurons in each layer doesn’t make a significant better accuracy, but it make our model takes much more time to finish the running process. So base on my observation, the best networking setting should be 2 hidden layer, each hidden layer has 2048 neurons. The optimization method adadelta, softmax as activation function. The dropout rate could set as 0.1.