E0-250 DL: Assignment 1

Magare Aadesh (Sr. No. 15605)

Jan 27, 2020

1 FizzBuzz

We consider the task of FizzBuzz. In this task an integer divisible by 3 is printed as fizz, and integer divisible by 5 is printed as buzz. An integer divisible by both 3 and 5 is printed as fizzbuzz. We use deep learning approach to solve the problem.

2 Dataset

We create a training data set of numbers ranging from 101 to 1000, a simple python code is used to generate the correct labels. Numbers from 1 to 100 are used as test set to evaluate the model.

3 Experiments

The experiments involved trying out different architectures and hyper parameters for the task to maximize the resulting accuracy.

The input numbers are first converted into binary representation of fixed size i.e. 10 bits. which are then given as input to the network, thus the input layer consists of 10 neurons. Output layer has 4 output neurons where each represent one output class i.e. fizzbuzz, fizz, buzz and original number respectively.

I started with just one hidden layer with accuracy of 68 % which goes upto 75% with a softmax layer at the end. Adding an extra hidden layer shoots the accuracy to 98% and interestingly a softmax layer in this architecture reduces accuracy to 60%. Thus the final model does not have softmax layer at the end. Adding any more layers to network does not give any performance gain, infact it reduces the accuracy. I think this is due to limited training data, additional layers makes the network too big to learn from just 900 training instances.

It was observed that Tanh as activation function in hidden layers give the best performance. Other activations such as Sigmoid, ReLU, LeakyReLU were tried but they give significant decrease in accuracy.

Training the network for more no of epocs also does not give any accuracy gains thus training was stopped at 10,000 epocs.

4 Best Model

Final model giving the best performance (**Accuracy: 98%**) after all experiments is described bellow.

Hyperparameter	Value
Hidden Layers	2
Neurons in hidden layers	10, 8
Output layer	4 neurons, no activation
Activation function	Tanh
Loss	CrossEntropy
Optimizer	Adam
Learning rate	1e-3
Epocs	10000