Blake Castleman

bc3029

COMS 4701

HW5 README File

Keras & Model Questions

Architecture Choices:

I chose to use two convolutional layers with max pooling layers existing after each convolution. Lastly, I included a fully connected layer with a softmax function as the output. For both convolutional layers, I used a 3x3 kernel (first with 64 filters then with 32 filters) and both max pooling used 2x2 poolings. This was because I was mainly concerned on capturing high frequency data and was less concerned about low frequency features. I chose not to use a dropout layer as I did not believe my model required its implementation to reach the desired testing accuracy.

1. What is one-hot encoding? Why is this important and how do you implement it in keras?

One-hot encoding is a sparse representation of labeling and is used when no relationship exists between categorical data (a binary vector exists denoting if the categories exist for every label present). This is important in machine learning when predictions may be split between two classes (leading to performance limitations) and can be implemented manually by either creating a list of one-hot encoded numpy vectors with binary values or by using built in keras functions OneHotEncoder() and encoder.fit_transform().

2. What is dropout and how does it help overfitting?

Dropout is the technique of randomly setting some input features to zero at a specified frequency. This prevents overfitting by making the model non-reliant on specific features and having it focus on other features to make classification decisions during training (dropout does not occur during evaluation/prediction).

3. How does ReLU differ from the sigmoid activation function?

ReLU is a simple max(0,input) function that cleans negative values while the sigmoid function is squashing function that forces numbers asymptotically between 0 to 1. ReLU tends to have a higher performance during network training, prevent vanishing gradients, and are more sparse as values can directly become zero (sigmoid will always have a nonzero value outputted); sigmoids however prevent training values exploding (as values are always squashed).

4. Why is the softmax function necessary in the output layer?

A softmax normalizes output values into a probabilistic distribution. This becomes necessary in the output layer of multiclass problems in order to have direct comparisons on the probabilities of a sample belonging to certain classes.

5. This is a more practical calculation. Consider the following convolution network:

(a) Input image dimensions = 100x100x1

(b) Convolution layer with filters=16 and kernel size=(5,5)

(c) MaxPooling layer with pool size = (2,2)

What are the dimensions of the outputs of the convolution and max pooling layers?

The convolution layer outputs a 96x96x16 image (assuming no padding is used on the borders) and the max pooling layer outputs a 95x95x16 image (assuming the stride = 1).