NLP Homework-2

Sathiyajith K S

04/09/2023

1) Detailed implementation of convolution and LSTM layers

Sentiment Analysis Using CNN

1. The CNN layers takes a input of the Word embeddings that are generated with padding using Word2Vec of size 500. Hidden layers of CNN model are created equal to the number of filters used(10). The CNN layers vary based on the window_size for each of the non-square kernel of size window_size*500 . These layers contain padding of (window_size-1,0).
2. For each of the CNN layers, tanh activation function is applied and squeezed for fitting the dimensions for applying maxpooling. Then max_pool1d() is applied over the result with stride equal to its dimensions to downsample the output values.
3. The maxpooling is followed by a linear mapping of values to each of the 3 classes.
4. Then the softmax function is applied to get the probability of output being one of the 3 classes.

Sentiment Analysis Using LSTM

1. The LSTM layer takes input of the Word embeddings that are generated with padding using Word2Vec of size 500. The hidden features size is 256 and the number of layers is 2 with a dropout of 0.5. The inputs and outputs are obtained as batch first which is of dimension (batch,seq, feature).
2. A dropout layer is added of 0.3 to further drop unnecessary parameters.
3. Then the values are linearly mapped to corresponding classes and sigmoid function is applied to get the probabilities over the 3 classes.

2) How many parameters does a CNN or LSTM layer have?

Number of parameters in Conv layer = ((m*n*d)+1)*k

m = width of layer, n = height of layer, d = previous layer's filters, 1 for bias, k = current layer's filters

Number of parameters in Conv1 layer = ((1*500*10)+1)*10 = 5,00,101

Number of parameters in Conv2 layer = ((2*500*10)+1)*10 = 1,00,010

Number of parameters in Conv3 layer = ((3*500*10)+1)*10 = 1,50,010

Number of parameters in Conv4 layer = ((5*500*10)+1)*10 = 2,50,010

Number of parameters in LSTM layer = 4*(n+m+1)*m = 7,75,168

3) Compare their training time for one epoch; feel which one is more time-consuming.

The training time was more for LSTM model than CNN model for every individual epoch and also for the total training period. With GPU, the CNN model took 10 minutes for an epoch, while LSTM model took around 20 minutes for an epoch. With only CPU, the CNN model took 30 minutes for an epoch, while LSTM model took around 50 minutes for an epoch.

4) Try to change the activation functions (e.g., from tanh to relu) to see how performance changes.

If the activation function is changed from tanh to relu, the overall accuracy decreases from 72% to 66%. This might be because tanh provides a nonlinear output whereas ReLU only offers linear output and tanh is generally used for classification because it has a range of -1 to 1, so handles both positive and negative values, like in this sentiment analysis.