Deep Learning for NLP 2020
Exercise 09 Solution

July 12, 2020

1 Pingo

- Which of the following statements about CNNs are correct?
  - ☑ The convolution operation creates an activation map (aka feature map) for every filter.
  - ☐ In CNNs, one usually applies a pooling layer first, followed by a convolutional layer.
  - ☐ A stride of 2 means that every convolution operation folds two neighboring input values.
  - ☐ When applying CNNs on NLP tasks, the features learned by every filter matrix can be easily visualized and interpreted.
- Which facts about regularization strategies are true?
  - ☑ Dropout can be applied for CNNs.
  - ☑ The dropout probability p is a hyperparameter.
  - ☑ Early stopping requires a validation set.

2 Theoretic Background of CNNs

1. Is a convolution over individual words (window size $k = 0$, i.e. one word per convolution window) useful? Explain in up to three sentences.

   Answer: Convolution over single words may not make sense at first glance. However, the dimensions of such a filter matrix will still be $1 \times 1 \times d$ where d is the dimension of the word representation. This means that convolution with $k = 0$ can still perform a somewhat useful transformation of individual words.

3 Dimensions and Parameters

We use a convolutional neural network for sentence classification.

Each input sentence consists of 197 words. Each word is represented by a vector from a 300-dimensional embedding space. There are 5581 unique words. The network consists of a trainable embedding layer followed by a convolutional layer, a global max-pooling layer and another fully-connected hidden layer. 111 filters, each with a window size of $k = 2$ convolve over the input. The stride is 1. The convolution is narrow. The hidden layer has 42 neurons and uses dropout with keep probability $p = 80\%$. The output layer is a single neuron with sigmoid activation function.

Using pen and paper...

1. Compute the output shape of each layer.
   - Embedding Layer: $(197, 300)$
   - Convolution Layer: $(196, 111)$
   - Max Pooling Layer: (111)
   - Fully-Connected Layer: (42)
   - Output Layer: (1)
2. Compute the number of trainable parameters of each layer. Don't forget the bias for the filters and the hidden layers!
   - Embedding Layer: $5581 \cdot 300 = 1674300$
   - Convolution Layer: $111 \cdot (300 \cdot 2 + 1) = 166611$
   - Max Pooling Layer: 0
   - Fully-Connected Layer: $42 \cdot (111 + 1) = 4704$
   - Output Layer: $42 + 1 = 43$