

# Deep Learning for Speech and Language Processing

## Exercise Sheet 5: CNNs

Maximilian Schmidt, Pascal Tilli, Dirk V  th, Ngoc Thang Vu

December 7th, 2020

### Deadline

Please submit your solutions until Monday, December 14th, 2020, noon. You may also see the deadline at any time on the website under *Exercises*  $\rightarrow$  *Upload*.

### Submission

#### Calculation Tasks

Upload (only solutions!) to <https://dlcourse.ims.uni-stuttgart.de/> using your account.

#### Theory Tasks

You don't have to submit this part, but it will help you with the contents of this course.

### Theory Task: 2D CNN Warm-Up

#### Exercise 1.

- (1) Derive a general formula on how to compute the output dimensions  $h, w$  of a convolution layer given the following parameters: input has  $o \times p$  dimensions, the filter height is  $m$ , the filter width is  $n$ , the stride is  $s$  and the symmetrical padding (i.e. same padding on both ends within same dimension) in the first and second dimension of the input is  $p_1$  and  $p_2$ , respectively.
- (2) By how many % is the number of outputs of a convolution layer reduced using  $2 \times 2$  pooling with a stride of 2 (assume no padding is needed)?
- (3) How many cells of the input matrix are covered by a single cell in the output of stacking a convolution layer with filter size  $f_2 = 2 \times 3$  on top of a convolution layer with filter size  $f_1 = 3 \times 3$ ? Assume no pooling layer between the two convolution layers and a stride of 1 for both filters.

### Calculation task: 2D CNN Forward Pass Computation

Consider the CNN shown in Figure 1 given the following parameters:

$$W_1 = \begin{bmatrix} 0 & 0 & 0 & -1 \\ -1 & 1 & 0 & -1 \end{bmatrix}, W_2 = \begin{bmatrix} 1 & 0 & -1 & 1 \end{bmatrix}, W_3 = \begin{bmatrix} -1 & 1 & 0 & 1 \end{bmatrix}, W_O = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & -1 \end{bmatrix}.$$

The activation function applied after the convolution filters is the logistic sigmoid  $\sigma$ . The high-level features vector  $a$  is fed through a fully connected layer with weight matrix  $W_O$  and softmax activation function to obtain  $y$ . Neither the filters nor the weight matrix of the output layer have biases.

The sentence matrix representing the input sentence is as follows:  $X = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix}$ .

## Exercise 2.

- (1) Calculate the convolution maps that result from the convolution operations with the given input sentence. A convolution map  $f^W = X \star W$  here is a convolution operation with filter  $W$  of size  $m \times n$  applied to input matrix  $X$  resulting in the convolution map  $f^W$ , where  $f_{hw}^W = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} w_{ij} x_{h+i, w+j}$ . Note that 0-based indexing is used here, i.e. the first element in the first row of  $W$  is denoted by  $w_{00}$  instead of  $w_{11}$ .
  - (1) Calculate  $f^{W_1}$
  - (2) Calculate  $f^{W_2}$
  - (3) Calculate  $f^{W_3}$
- (2) State the concatenated feature vector (vector  $a$  in figure 1 - the three concatenated results of the 1-max-pooling across the three feature maps)
- (3) Finally, calculate the output  $y$  for the given input sentence.

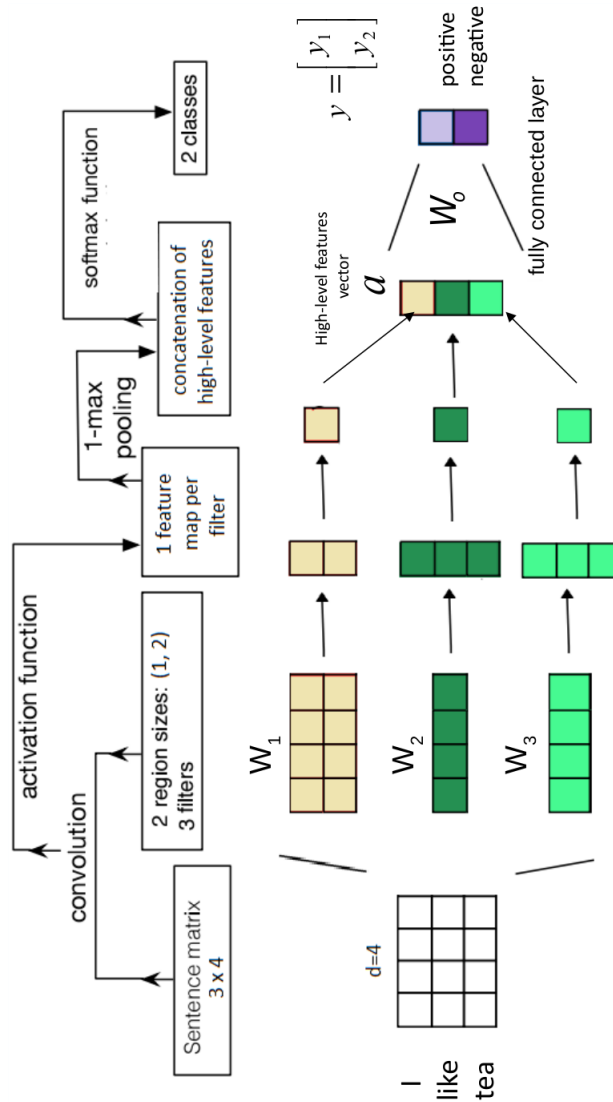


Figure 1: CNN architecture for Exercise 2. Thanks to Daniel Ortega for designing this exercise!