92586 Computational Linguistics

15. Convolutions in Text

Alberto Barrón-Cedeño

Alma Mater Studiorum-Università di Bologna
a.barron@unibo.it @_albarron_

5/04/2020

Previously

- doc2vec
- CNN over images

Table of Contents

1 Quick Reminders on CNNs

2 CNNs for NLP

Chapter 7 of Lane et al. (2019)

Quick Reminders on CNNs

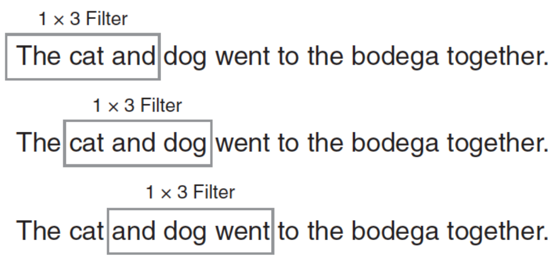
Quick Reminders on CNNs

- Sliding —or convolving— a window over the sample
- Filters (kernels; matrices) slide over fragments of the image
- “Snapshots” of different areas of the image are taken and processed
- Multiple filters produce multiple images
- Multiple convolution layers can be added
- At the end, we can plug a “standard” fully-connected NN

CNNs for NLP

Back to Text

- In images both vertical and horizontal relationships are relevant
- In text only horizontal ones do[1]
- **We need “1D” filters**

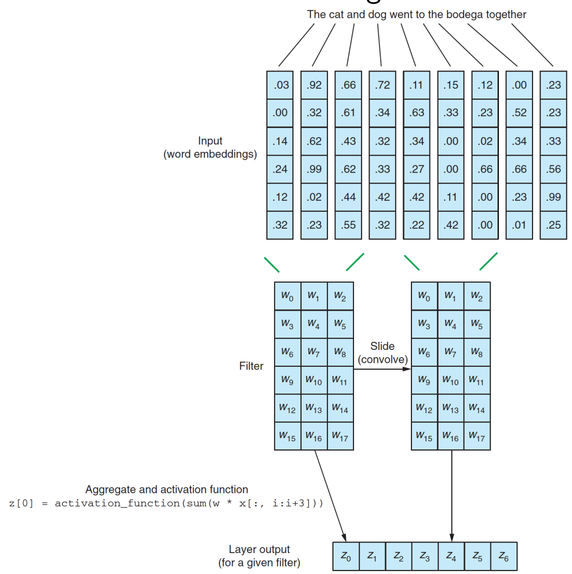


[1] l2r or r2l; for some languages it's the vertical direction that matters (e.g., Japanese)

(Lane et al., 2019, p. 229)

But we do have 2D “filters”

Words are represented with word embeddings: vectors



(Lane et al., 2019, p. 229)

The convolution is (practically) the same as for images

- We now *convolve* in one dimension (not two)
- The computation order is irrelevant, but the outputs have to be fed in the same order
- The filters' weights are fixed for a full sample (parallel computation)
- Their output becomes the features for the classifier

Let us see

Padding

- (In general) in image processing the inputs are of fixed size, regardless of the image (same source!)
- Texts are not fixed length (regardless of their source)

- Instances longer than `maxlen` will be truncated
- Instances shorter than `maxlen` will be **padded**

$x_0, x_1, x_2, x_3, \ldots x_{398} x_{399}$ $x_{400} x_{401}$

$x_0, x_1, x_2, x_3, \ldots x_{397}$ PAD PAD

Let us see

Pooling

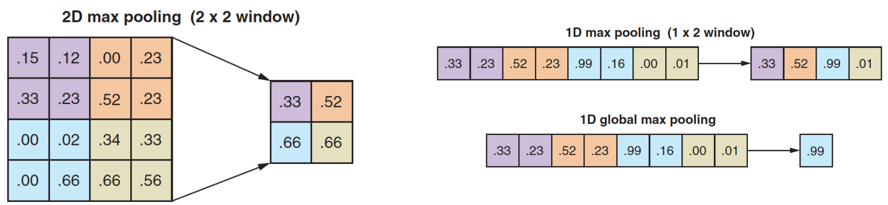
- For each filter one new version of the instance is produced (250 in the example)
- Pooling evenly divides the output of each filter into subsections
- It selects (or computes) a representative value for each subsection

Pooling

Pooling is "the CNN path to dimensionality reduction [. . .] by learning higher-order representations of the source data" (Lane et al., 2019, p. 236)

- The filters job is finding patterns → relationships between words and their neighbours

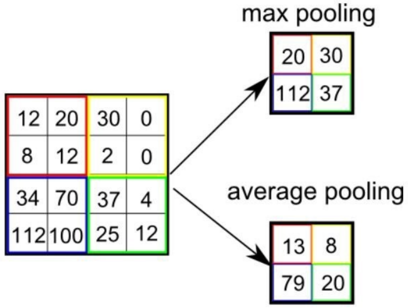
- Pooling in text: a 1D window (e.g., 1×2 or 1×3)



(Lane et al., 2019, p. 237)

Pooling

Max vs Average Pooling



- Average is more intuitive: retaining most of the info
- Max is better: the NN keeps the most prominent feature

Let us see

Image borrowed from `www.quora.com/What-is-max-pooling-in-convolutional-neural-networks`

Recap

- Each filter will produce a 1×398
- For each of the 250 filter outputs , we take the single maximum value for each 1D vector
- Output: one 1×250 vector

This is a crude semantic representation of the text

Dropout: Preventing Overfitting

On each training pass **turn off** a percentage of the input of a layer; it will become 0

- Chosen randomly on each pass
- It will not rely heavily on any feature
- It will be generalize better
- Dropout is applied on training only

Let us see

Photogram of “The Platform” (2019)

Workhorse Loss Functions

Out of the 13+ available loss functions:

binary_crossentropy: the output neuron is either on or off

categorical_crossentropy: the output is one out of many classes

Let us see

Closing Remarks

- Your input is a series of max 400 words; 300 elements each
- Nothing prevents you from stacking other embeddings (think of RGB)
- The output of the convolution layer is tied to the task (in this case, sentiment analysis)
- A CNN is more efficient, thanks to the pooling process and the filters
- You can add many convolution layers

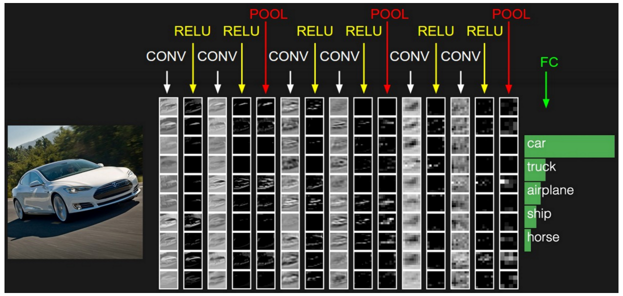


Image borrowed from `https://blog.mapillary.com`

Next time

- Recurrent Neural Networks

References

Lane, H., C. Howard, and H. Hapkem
2019. *Natural Language Processing in Action*. Shelter Island, NY: Manning Publication Co.