Convolutional Neural Networks for Text Classification

Sebastian Sierra

MindLab Research Group

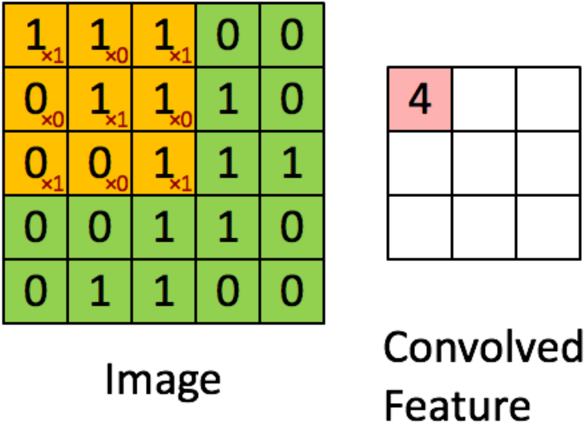
July 1, 2016

Outline

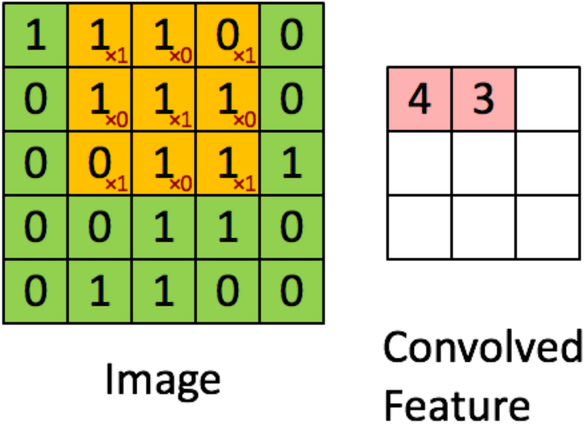
Outline

What is a Convolution?

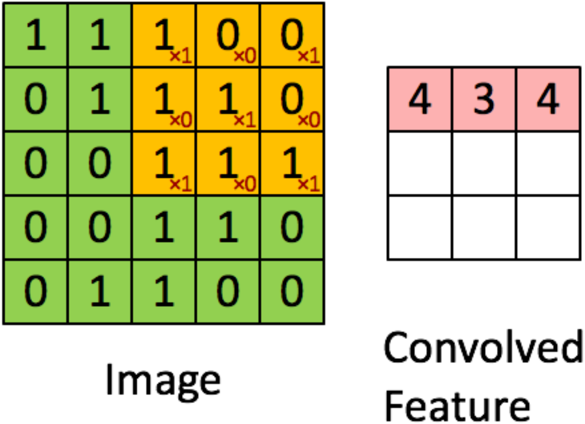
- Convolutions are great for extracting features from Images.
- Convolutional Neural Networks (CNN) are biologically-inspired variants of MLPs



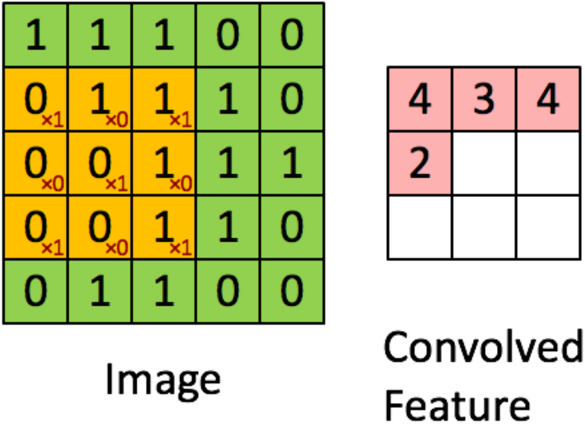
Green: Input, Yellow: Convolutional Filter, Red: Output



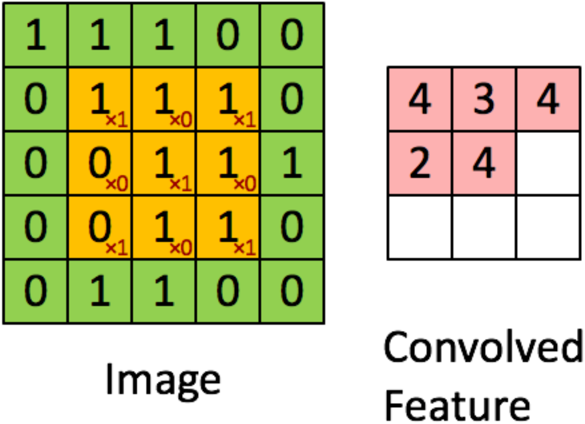
Green: Input, Yellow: Convolutional Filter, Red: Output



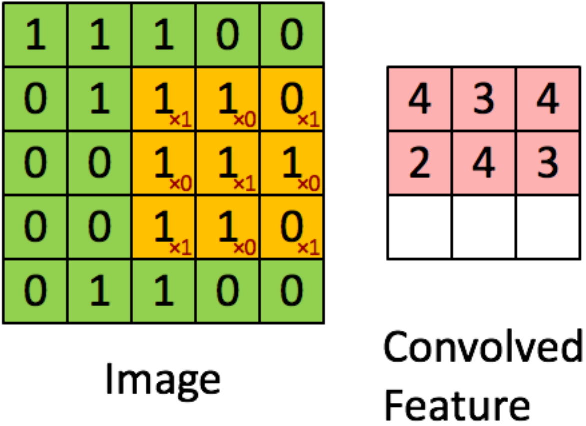
Green: Input, Yellow: Convolutional Filter, Red: Output



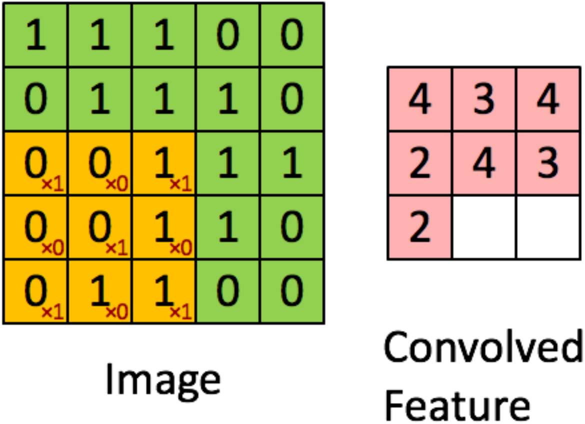
Green: Input, Yellow: Convolutional Filter, Red: Output



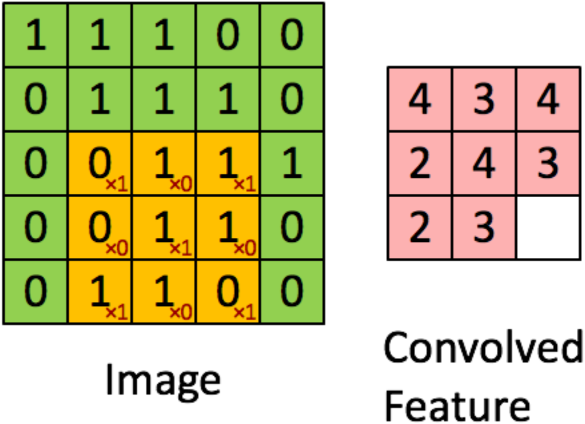
Green: Input, Yellow: Convolutional Filter, Red: Output



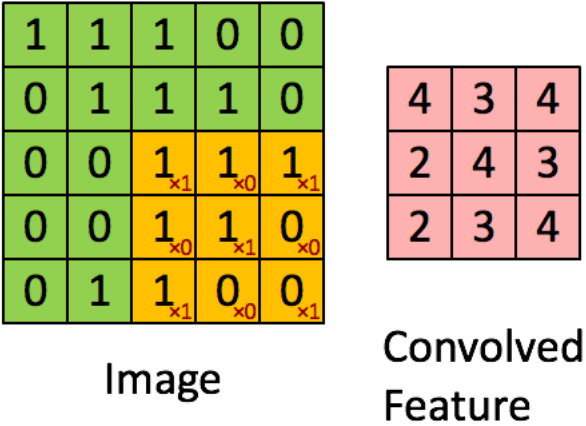
Green: Input, Yellow: Convolutional Filter, Red: Output



Green: Input, Yellow: Convolutional Filter, Red: Output



Green: Input, Yellow: Convolutional Filter, Red: Output



Green: Input, Yellow: Convolutional Filter, Red: Output

Outline

Convolutional Neural Network

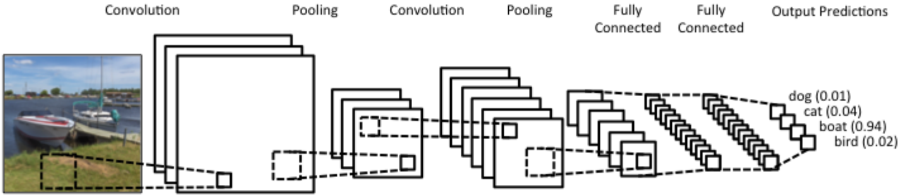


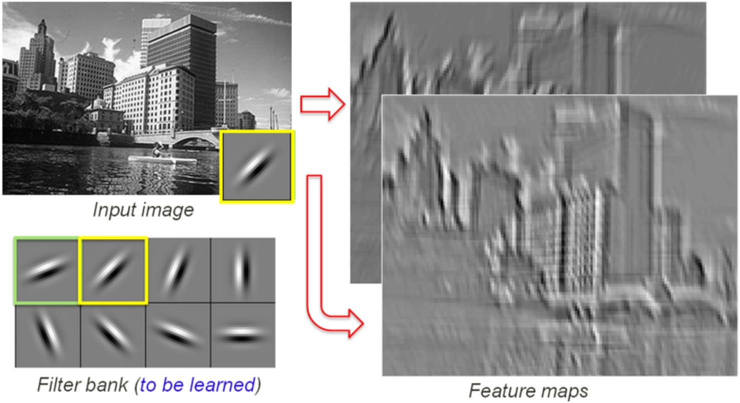
Figure 1: Close up of Convolutional Neural Network

Convolutional Neural Network

- CNNs are networks composed of several layers of convolutions with nonlinear activation functions like ReLU or tanh applied to the results.
- Traditional Layers are fully connected, instead CNN use local connections.
- Each layer applies different filters (thousands) like the ones showed above, and combines their results.

Properties of Convolutional Neural Networks

- Local Invariance
- Compositionality



Outline

CNN in NLP

Do they make sense in NLP?

Perhaps Recurrent Neural Networks would make more sense trying to learn patterns extracted from a text sequence. They are not cognitively or linguistically plausible.

Advantage

There are fast GPU implementations for CNNs

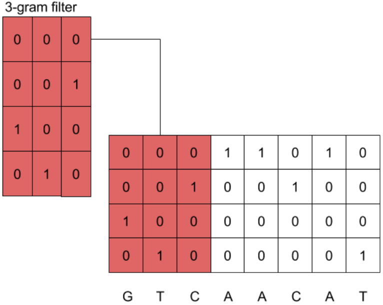
An example of how CNN work

3-gram filter

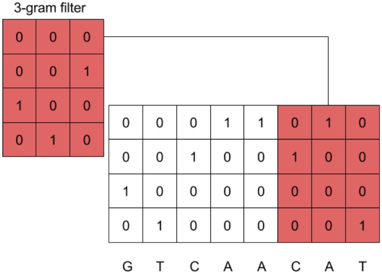
0	0	0
0	0	1
1	0	0
0	1	0

G	T	C	A	A	C	A	T
0	0	0	1	1	0	1	0
0	0	1	0	0	1	0	0
1	0	0	0	0	0	0	0
0	1	0	0	0	0	0	1

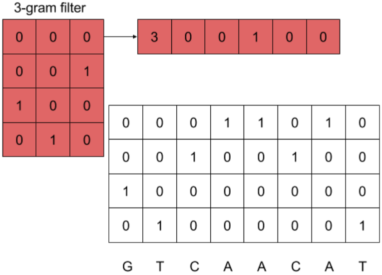
An example of how CNN work



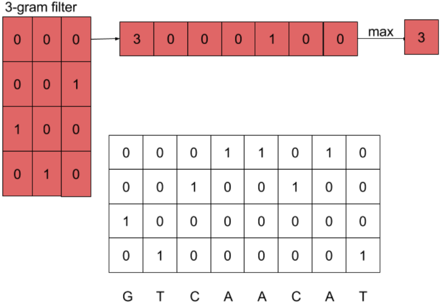
An example of how CNN work



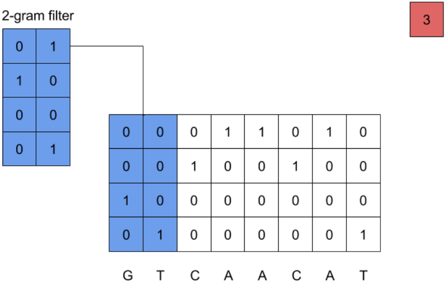
An example of how CNN work



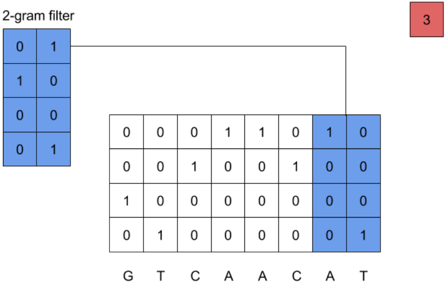
An example of how CNN work



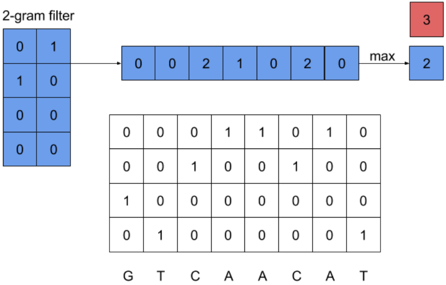
An example of how CNN work



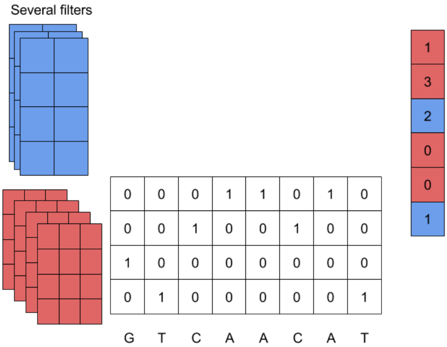
An example of how CNN work



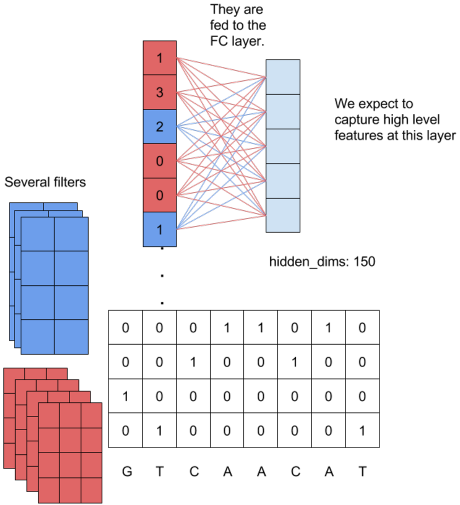
An example of how CNN work



An example of how CNN work



An example of how CNN work



Char-CNN

Let's try this model

Char-CNN

Let's try this model

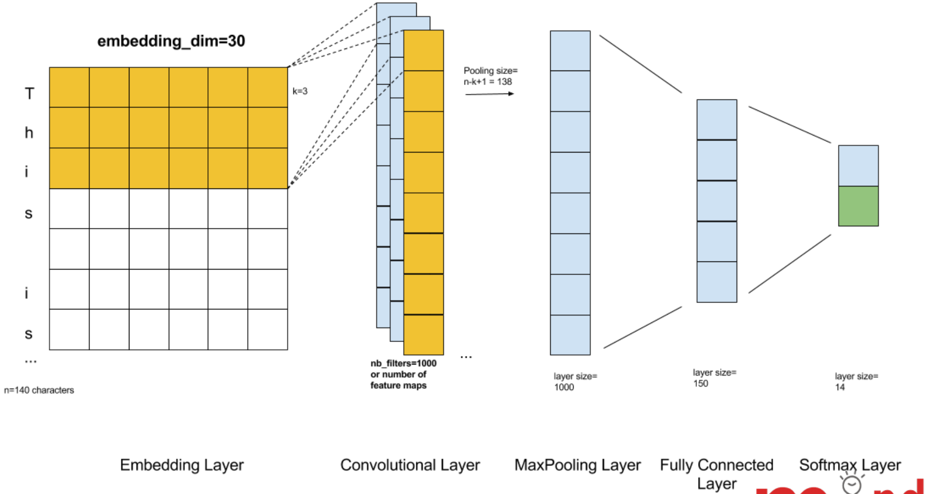
$C \in \mathbb{R}^{d \times l}$: Matrix representation of a sequence of length l(140, 300, ?).

$H \in \mathbb{R}^{d \times w}$: Convolutional filter matrix where,

d : Dimensionality of character embeddings (used 30)

w : Width of convolution filter (3, 4, 5)

A simple architecture



Steps for applying a CNN

1. Apply a convolution between C and H to obtain a vector $f \in \mathbb{R}^{l-w+1}$

$$f[i] = \langle C[*, i : i + w - 1], H \rangle$$

$\langle A, B \rangle$ is the Frobenius inner product. $Tr(AB^T)$

2. This vector f is also known as feature map.
3. Take the maximum value over time as the feature that represents filter H. (K-max pooling)

$$\widehat{f} = relu(\max_i \{f[i]\} + b)$$

4. Then we do the same for all m filters.

$$z = [\widehat{f_i}, ..., \widehat{f_m}]$$

Intution behind

- Why *ReLU* and not *tanh*?
- Should I use multiple filter weights *H*?
- Should I use variable filter widths *w*?
- Can I add another channel as in Computer Vision domain?
- Is Max-Pooling capturing the most important activation?
- Would they capture morphological relations?

Outline

Regularization tricks

- Use dropout (Gradients are only backpropagated through certain inputs of z).
- Constrain L_2 norms of weight vectors of each class(rows in Softmax matrx $W^{(S)}$) to a fixed number: If $\|W_c^{(S)}\| > s$, the rescale it so $\|W_c^{(S)}\| = s$.
- Early Stopping

CNN in NLP - Previous Work

- Previous works: NLP from scratch (Collobert et al. 2011).
- Sentence or paragraph modelling using words as input (Kim 2014; Kalchbrenner, Grefenstette, and Blunsom 2014; Johnson and T. Zhang 2015a; Johnson and T. Zhang 2015b).
- Text classification using characters as input (Kim et al. 2016; X. Zhang, Zhao, and LeCun 2015)

Outline

Outline

Collobert, Ronan et al. (2011). "Natural language processing (almost) from scratch". In: *Journal of Machine Learning Research* 12.Aug, pp. 2493–2537.

Johnson, Rie and Tong Zhang (2015a). "Effective Use of Word Order for Text Categorization with Convolutional Neural Networks". In: *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Denver, Colorado: Association for Computational Linguistics, pp. 103–112. URL: http://www.aclweb.org/anthology/N15-1011.

– (2015b). "Semi-supervised convolutional neural networks for text categorization via region embedding". In: *Advances in neural information processing systems*, pp. 919–927.

Kalchbrenner, Nal, Edward Grefenstette, and Phil Blunsom (2014). "A Convolutional Neural Network for Modelling Sentences". In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Baltimore, Maryland: Association for Computational Linguistics, pp. 655–665. URL: http://www.aclweb.org/anthology/P14-1062.

Kim, Yoon (2014). "Convolutional Neural Networks for Sentence Classification". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, pp. 1746–1751. URL: http://www.aclweb.org/anthology/D14-1181.

Kim, Yoon et al. (2016). "Character-Aware Neural Language Models". In: *Thirtieth AAAI Conference on Artificial Intelligence*.

Zhang, Xiang, Junbo Zhao, and Yann LeCun (2015). "Character-level Convolutional Networks for Text Classification". In: *Advanced in Neural Information Processing Systems (NIPS 2015)*. Vol. 28.