Redes neuronales avanzadas: convolucionales

Diplomatura en Ciencia de Datos, Aprendizaje Automático y sus Aplicaciones

1. Intuición en imágenes

Recursos

Curso Stanford

Tutorial towardsdatascience.com

Filminas originales: Jorge Sánchez, inspiradas en A. Vedaldi - cs231n (Stanford)

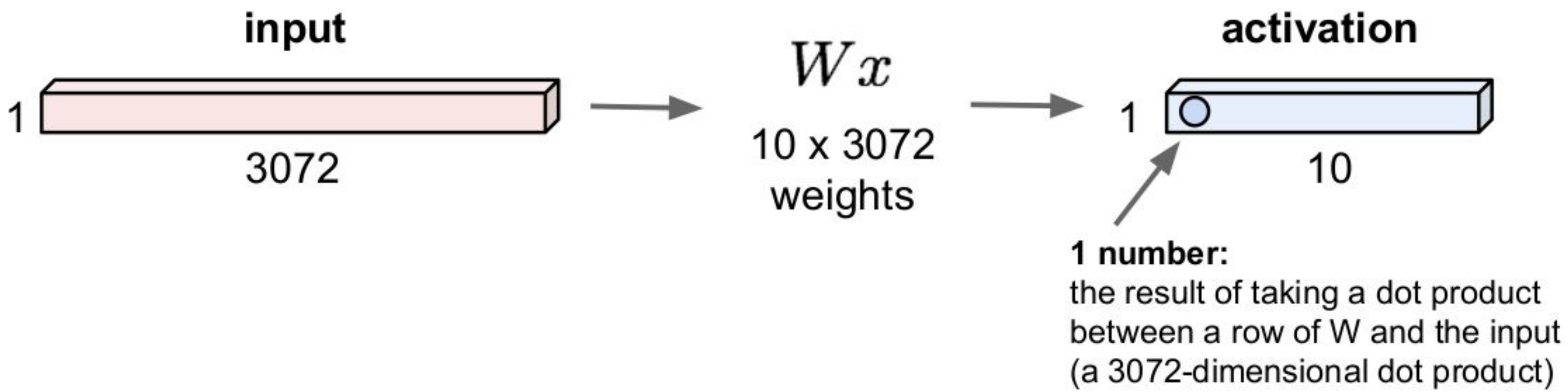
Imágenes = 2D

Las imágenes tienen propiedades espaciales:

- ➔ En orden de los datos es en dos (o tres) dimensiones
- ➔ Los mismos patrones se repiten en distintas posiciones: bordes, esquinas

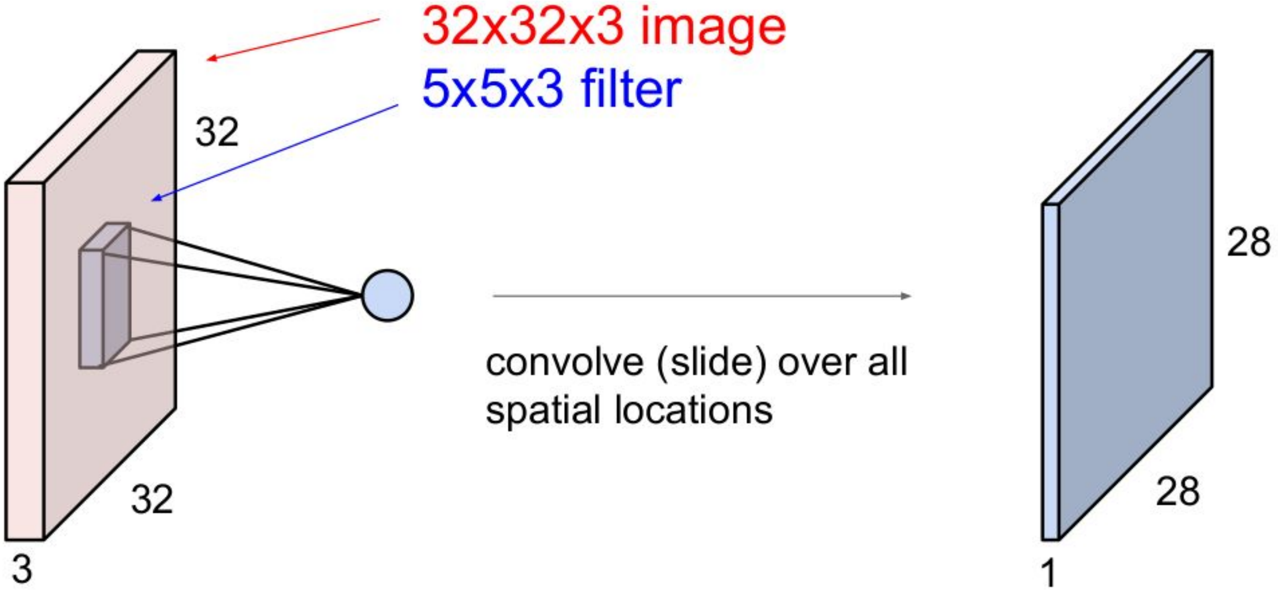
Multilayer perceptron

32x32x3 image -> stretch to 3072 x 1

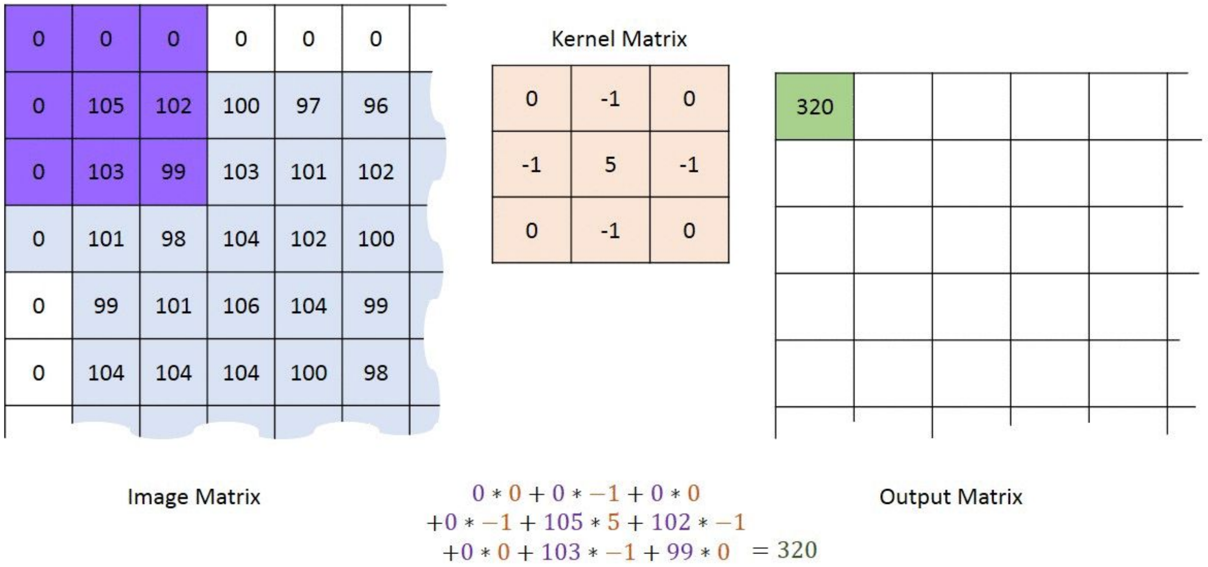


Convolución

Tenemos que procesar una entrada 2D o 3D



Convolución



Convolution with horizontal and vertical strides = 1

Convolución

La convolución es la combinación de dos matrices (funciones):

- ➔ La imagen original
- ➔ El filtro de la convolución

El resultado es otra matriz (función)

Filtro de convolución

Los filtros capturan distintos aspectos de una imagen: detección de bordes, desenfoque, [más ejemplos]

Filtro de convolución

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

-1	-1	-1
-1	8	-1
-1	-1	-1

6				

Filtro de convolución

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

-1	-1	-1
-1	8	-1
-1	-1	-1

6	6			

Filtro de convolución

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

-1	-1	-1
-1	8	-1
-1	-1	-1

6	6	-2		

Filtro de convolución

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

-1	-1	-1
-1	8	-1
-1	-1	-1

6	6	-2	-3	6
6	-4	-4	3	4
-2	-3	5	2	3
6	-4	-4	3	4
6	6	-2	-3	6

Filtro de convolución

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

-1	-1	-1
-1	8	-1
-1	-1	-1

6	6	-2	-3	6
6	-4	-4	3	4
-2	-3	5	2	3
6	-4	-4	3	4
6	6	-2	-3	6

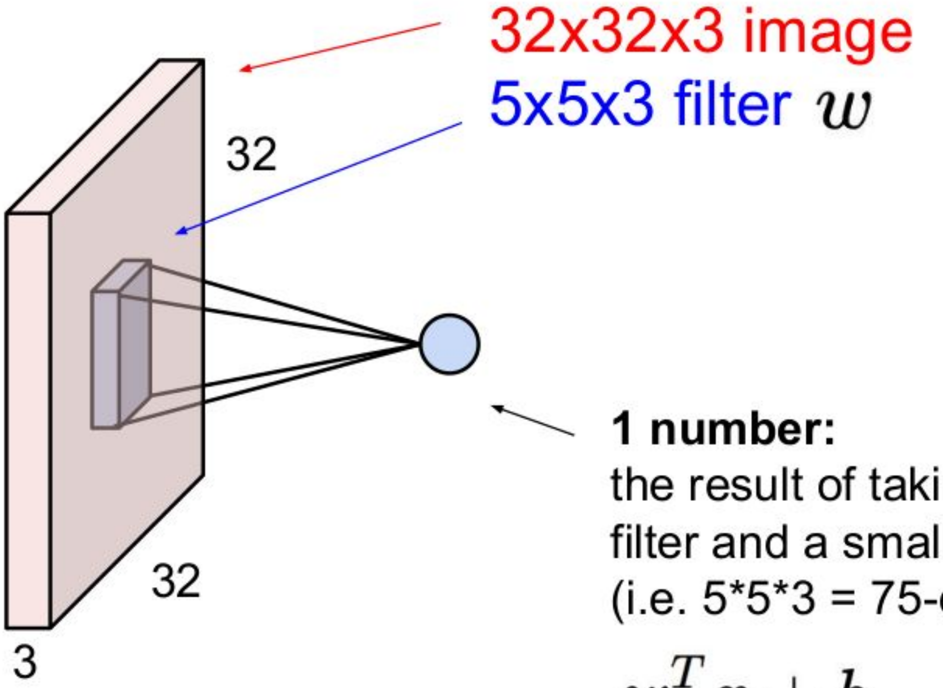
Para jugar un rato

2. CNN

Los filtros pueden ser vistos como extractores de información.

Las redes convolucionales APRENDEN numerosos filtros (feature extractors)

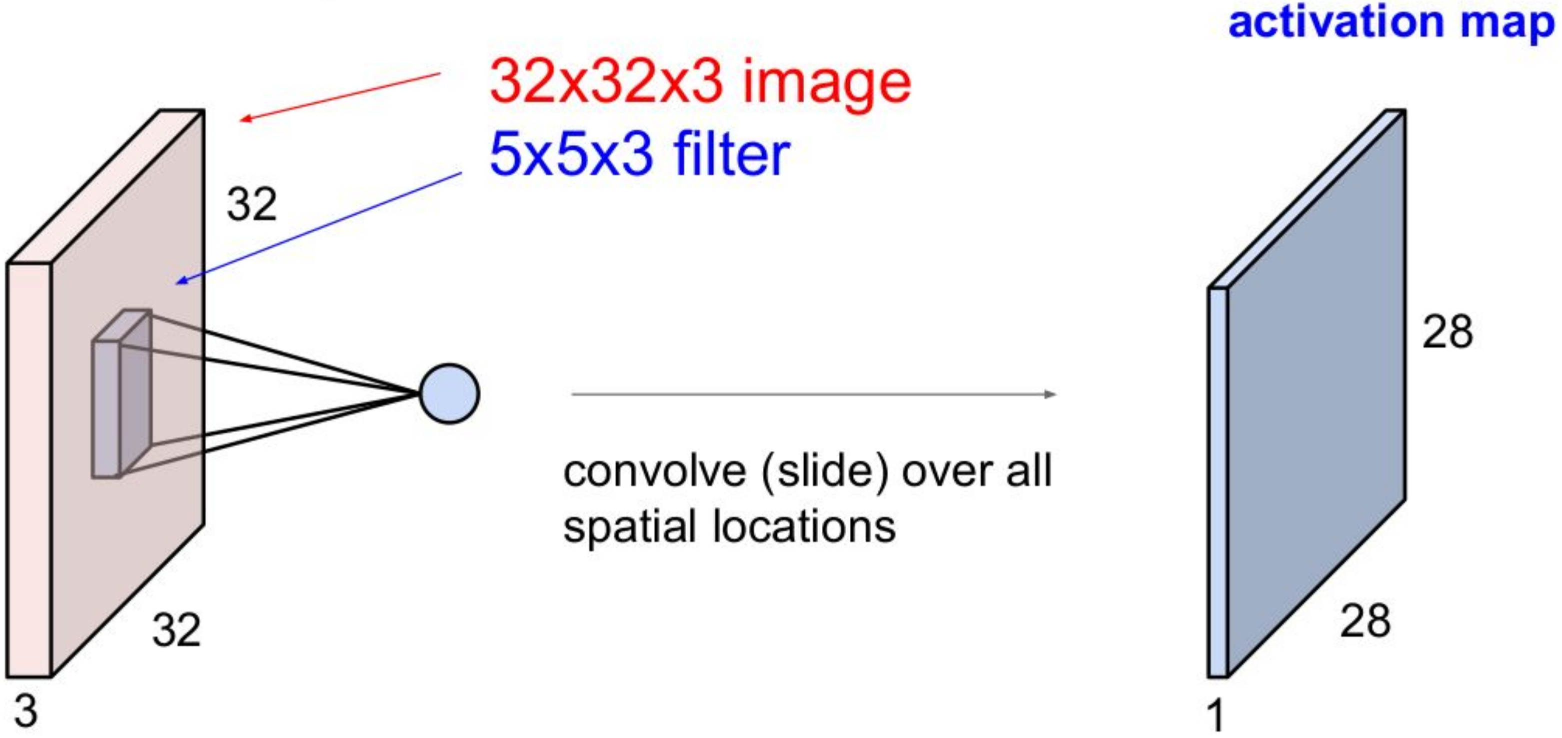
Convolution Layer



1 number:
the result of taking a dot product between the filter and a small 5x5x3 chunk of the image
(i.e. 5*5*3 = 75-dimensional dot product + bias)

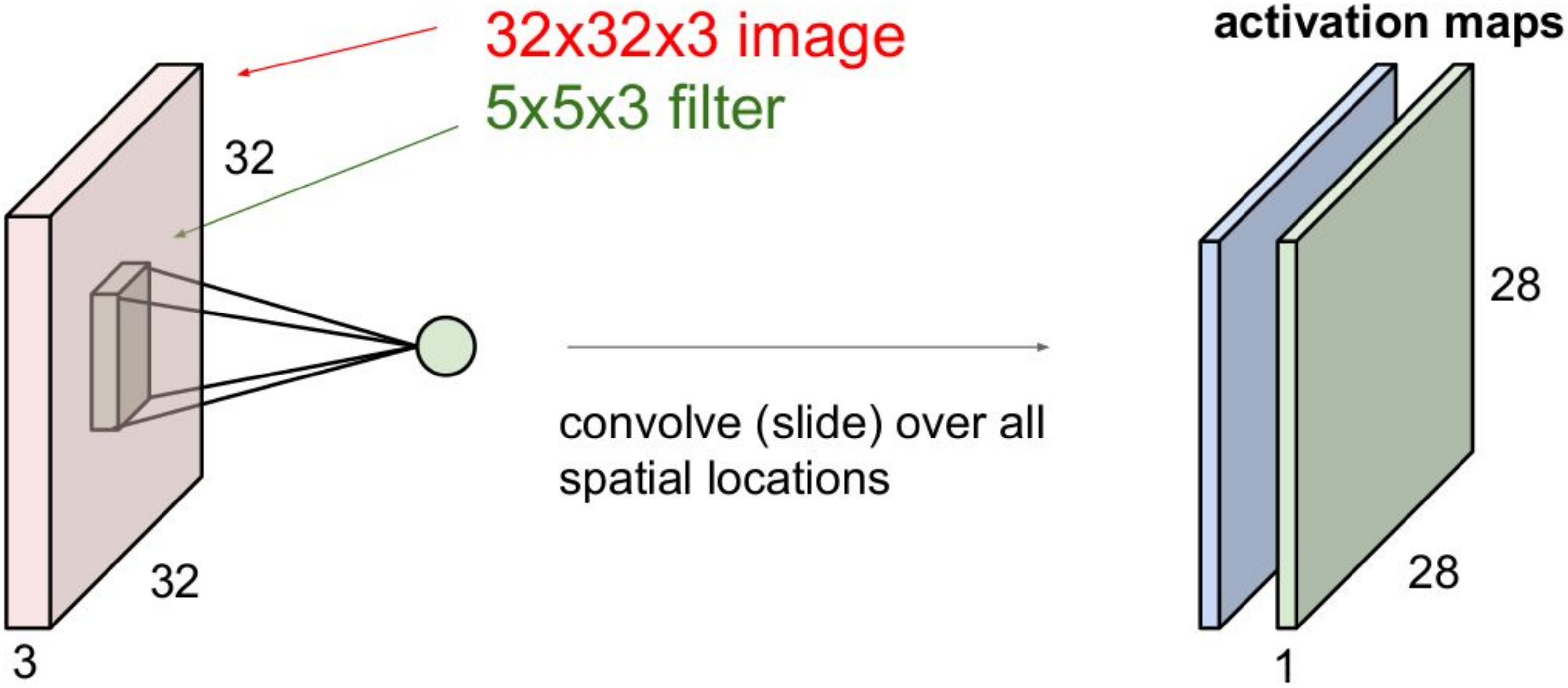
$$w^T x + b$$

Convolution Layer

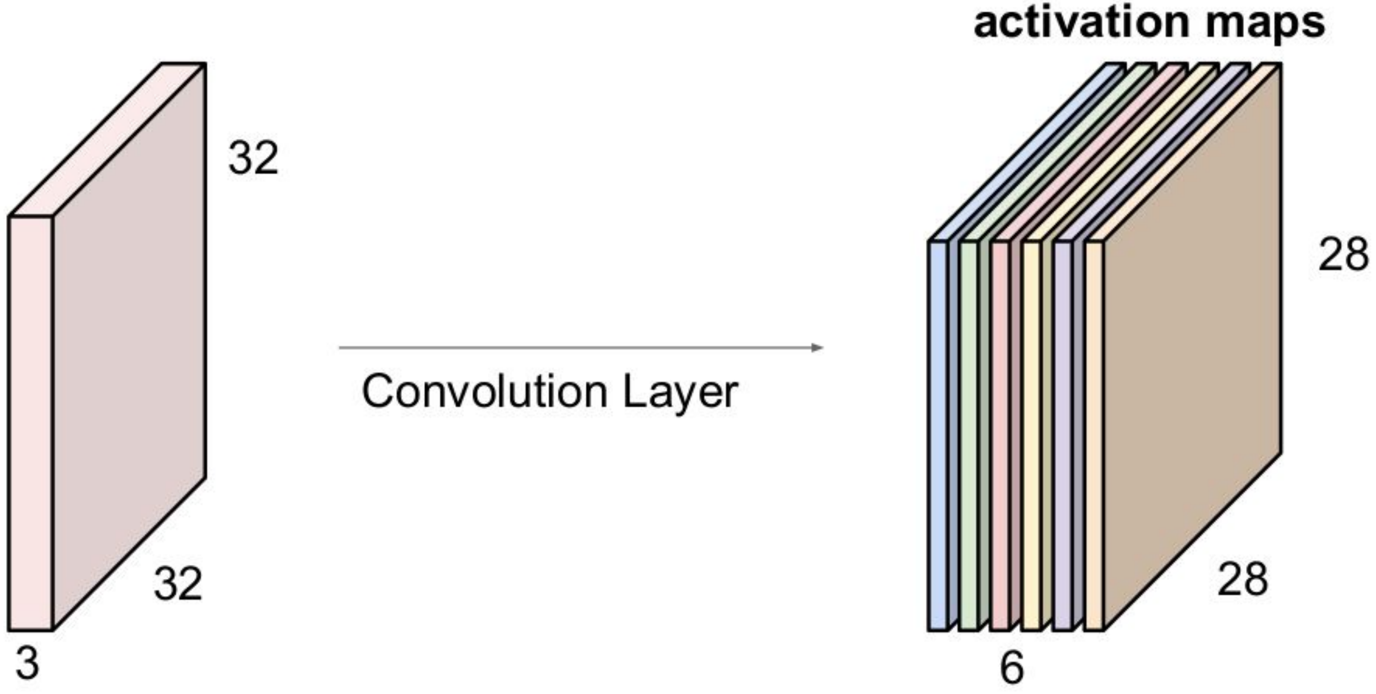


Convolution Layer

consider a second, green filter



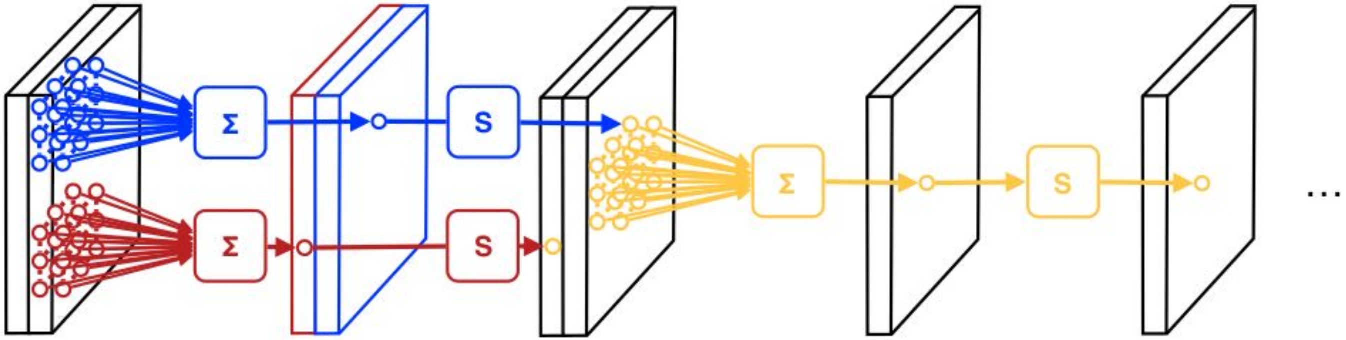
For example, if we had 6 5x5 filters, we'll get 6 separate activation maps:



We stack these up to get a "new image" of size 28x28x6!

Multiple layers

Convolution, activation, convolution, activation, …

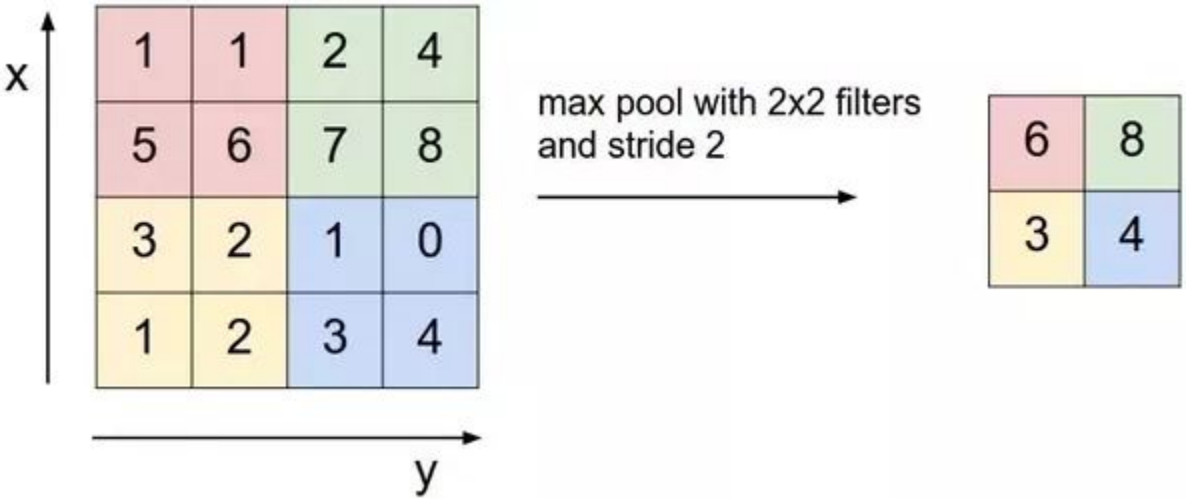


A deep convolutional neural networks chains several filtering & non-linear activation function sequences.

The non-linear activation functions are essential. Why?

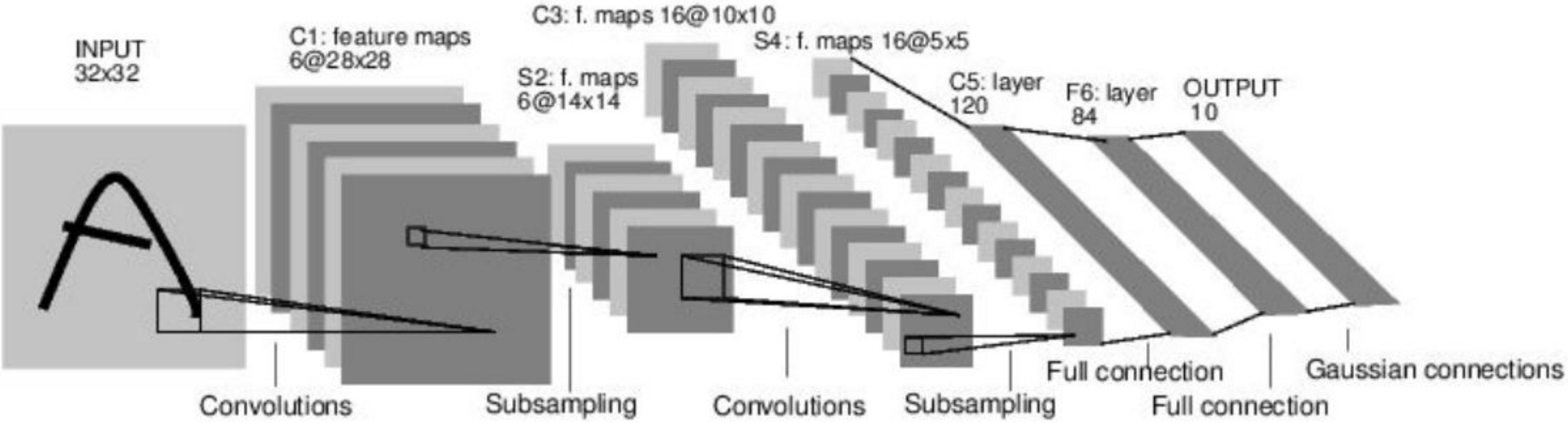
Pooling

El pooling es un mecanismo para reducir la dimensionalidad de las capas.



Case Study: LeNet-5

[LeCun et al., 1998]

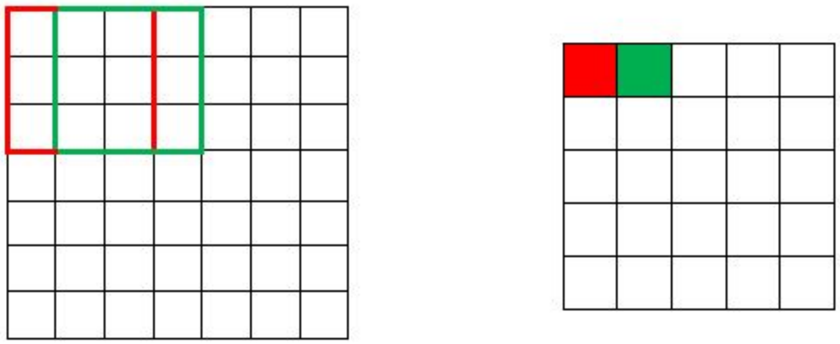


Conv filters were 5x5, applied at stride 1
Subsampling (Pooling) layers were 2x2 applied at stride 2
i.e. architecture is [CONV-POOL-CONV-POOL-CONV-FC]

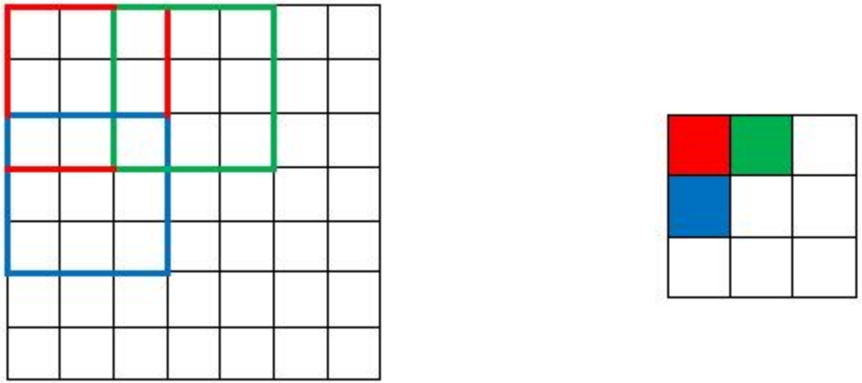
Stride

El stride controla la distancia entre la aplicación de los filtros, y el tamaño del output

Stride 1



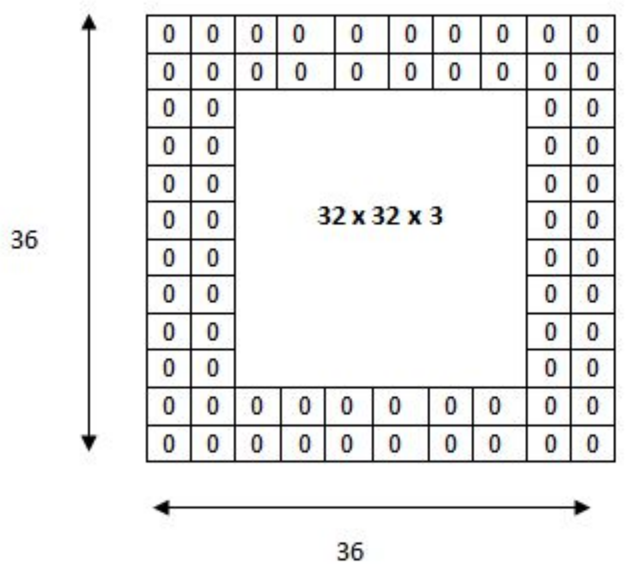
Stride 3



https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks-Part-2/

Padding

Padding 2

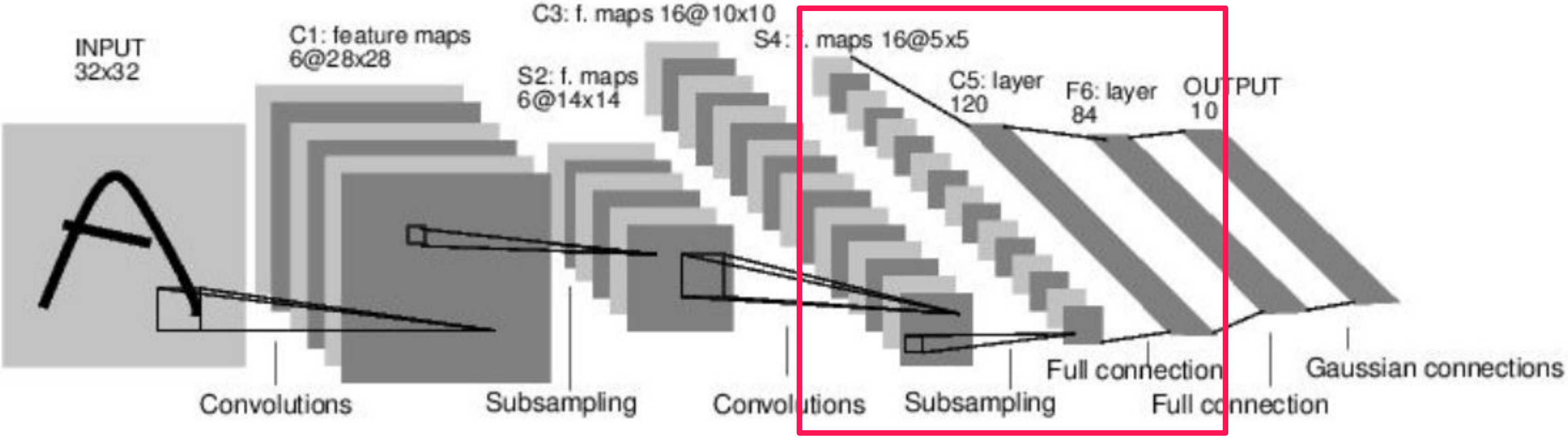


El padding permite la aplicación de los filtros a los elementos del borde de la imagen, y afecta el tamaño del output

https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks-Part-2/

Flatten

Para poder aplicar la capa de clasificación, tenemos que convertir la salida de la convolución a 1D

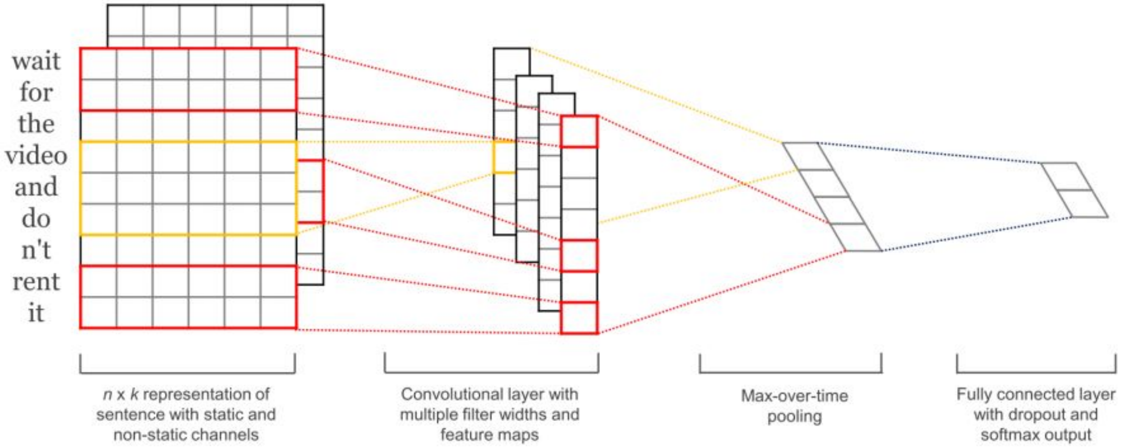


3. Práctico [con notebook]

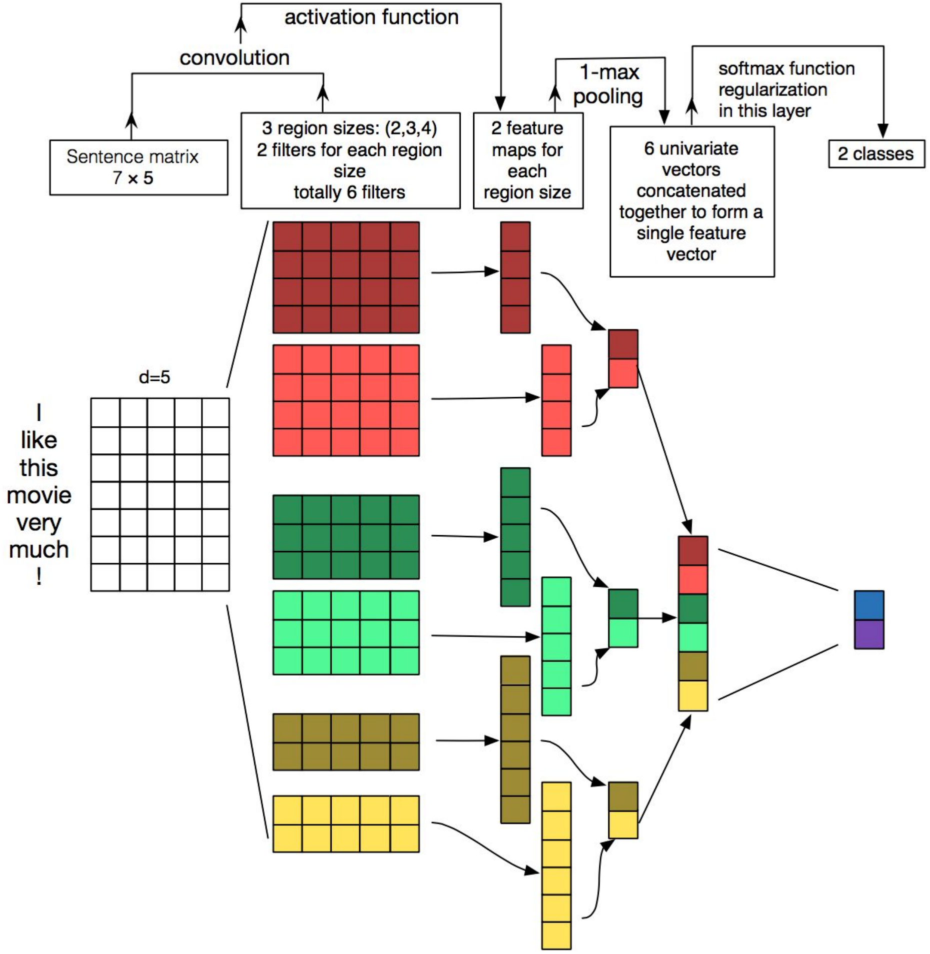
4. CNNs para texto?

Convolución 1D

Aplicamos la convolución a cada palabra (Conv1D)



http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/



http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/

References

+ Chris Piech, et al. 2015. *Deep Knowledge Tracing*. In Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS’15)

+ Siddharth Reddy, Igor Labutov, and Thorsten Joachims. 2016. *Learning Student and Content Embeddings for Personalized Lesson Sequence Recommendation*. In Proceedings of the Third (2016) ACM Conference on Learning @ Scale (L@S ’16).