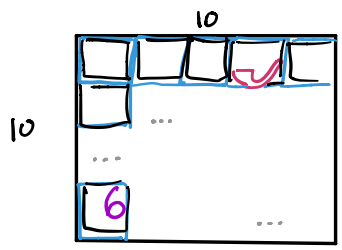
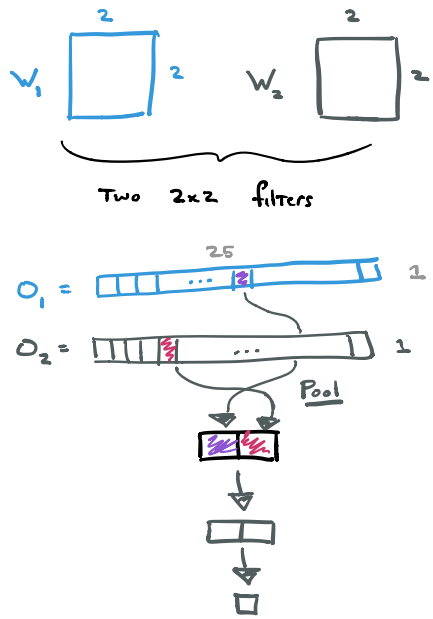
DS 4440 ConvNets II

Last Time:





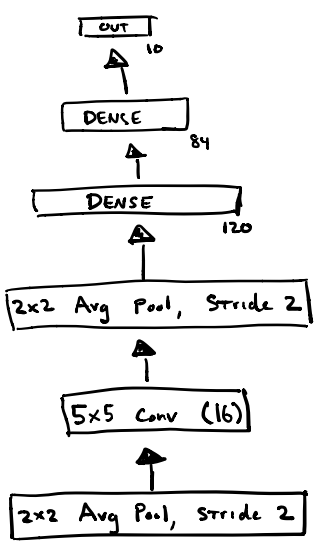
Design decisions:

- Filter/window size
- Stride size
- Padding

In practice, deeper models have been successful in image classification.

Idea: Stack (many) conv layers on top of each other, usually in blocks.

LeNet (LeCun, '98)

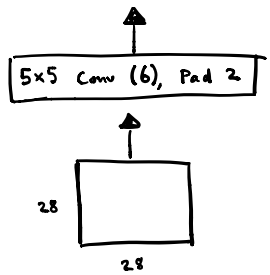


This had some success but did not enjoy wide adoption: Hardware & data needed to catch up.

In 2012 the time came: AlexNet (2012) — a deeper successor to LeNet won the ImageNet challenge.

Two key design changes:

- Sigmoid → Relu
- Deeper (8 layers)

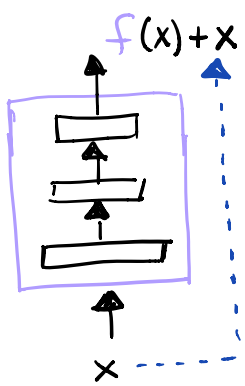


(See notebook.)

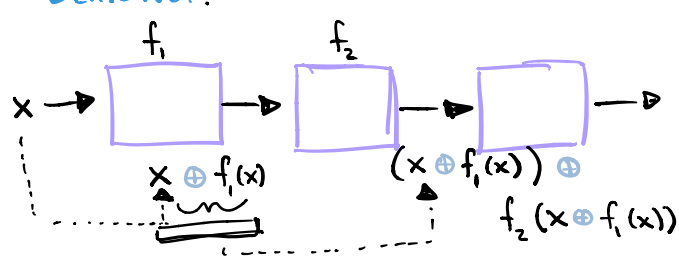
Residual Connections

As we create deeper networks, they are increasingly tricky to fit. We saw in the notebook that, e.g., using ReLU is key to allow ∇ to flow backwards.

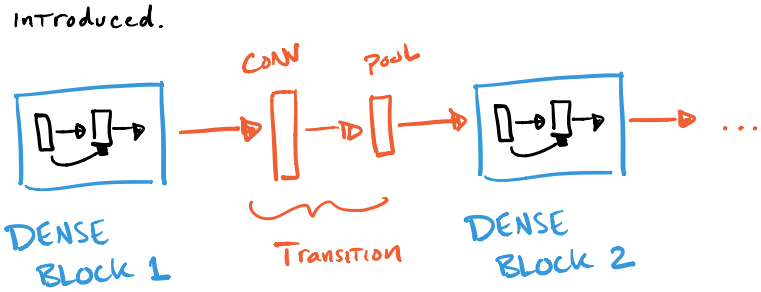
Another trick for this: Residual Connections that bypass blocks.



Here we <u>add</u> but could also concatenate $\oplus$ as in DenseNet.

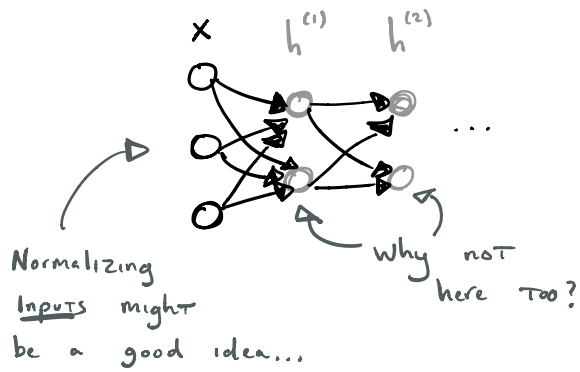


To keep dimensions sane, usually transition layers are introduced.



Batch Norm

When we covered optimizers, we discussed normalizing inputs for SGD.



The idea is to normalize <u>per-batch</u>.

Assume pre-activation outputs z^l at layer l (so $h^l = a(z^l)$).

$$\mu^l = \frac{1}{m}\sum_{j=1}^{m} z_m^l \qquad \sigma^{l^2} = \frac{1}{m}\sum_{j=1}^{m}(\mu_j^l - z_j^l)^2$$

$$z_{Norm}^l = \frac{z^l - \mu^l}{\sqrt{\sigma^{l^2}+\epsilon}}$$

for numerical reasons

$\emptyset$ mean

unit variance

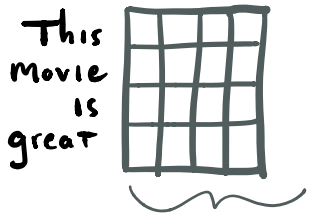
But this assumes all layers have $\emptyset$ mean and variance 1! BN is more flexible, allowing this to be shifted.

$$z_{Norm}^l = \boxed{\hat{\gamma}^l} \odot \frac{z^l - \hat{\mu}}{\sigma^{l^2}} + \boxed{\hat{\beta}^l}$$

To be <u>learned</u>.

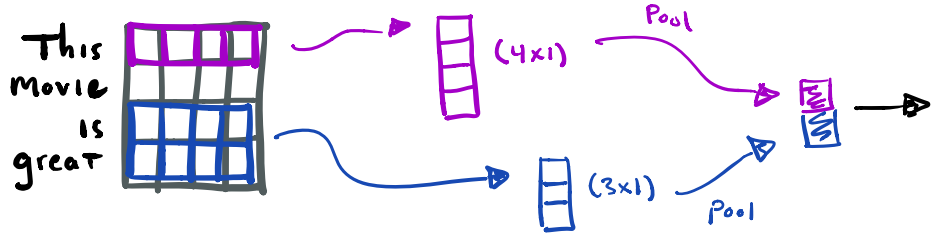
CNNs for Text

ConvNets have been most successful for images, but have also been used in NLP!



d-dimensional word embeddings (lecture 8!)

Q. What should our filter size be?



As for images, we usually use multiple filters.