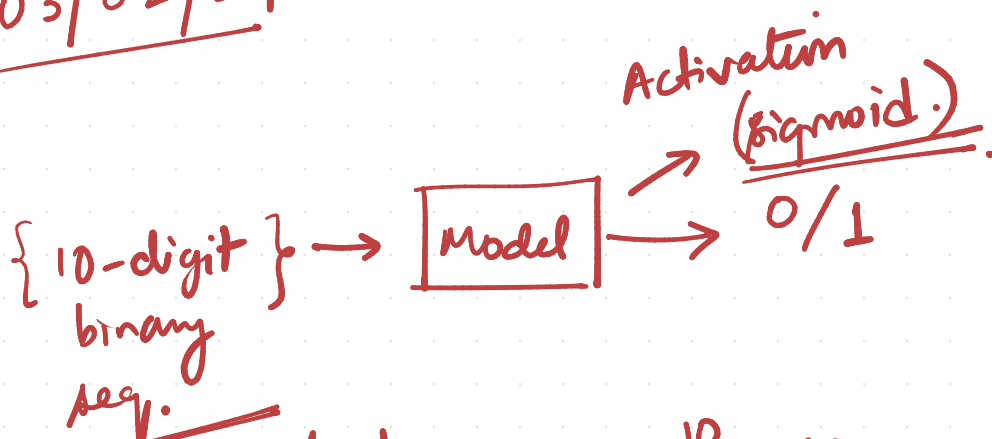


03/02/24



2-digit

$x_2$	$x_1$	(#1 > #0)?
1	1	1
1	0	0
0	1	0
0	0	0

$2^{10} = 1024$  input-o/p pairs.

Training data = 1024

& Data Generator.

- Data loader ⇒
- Training pipeline
- Validation pipeline
- Testing pipeline.

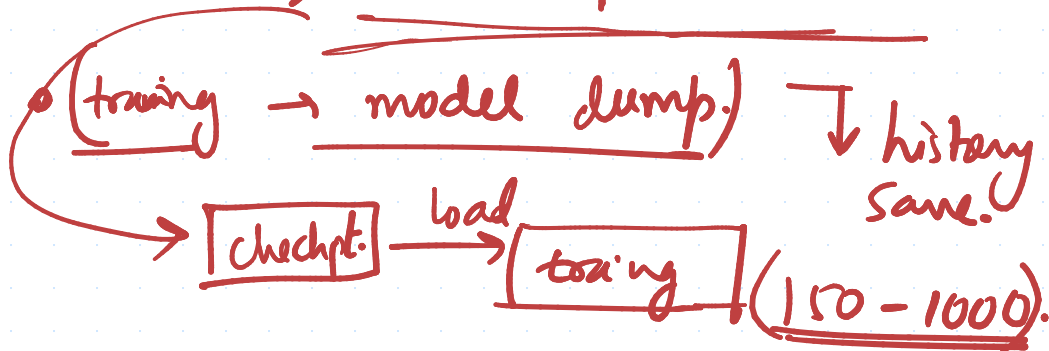
Pytorch lightning → (rigid)  
↓  
rapid prototyping  
(lr-search) ↓  
optimal.

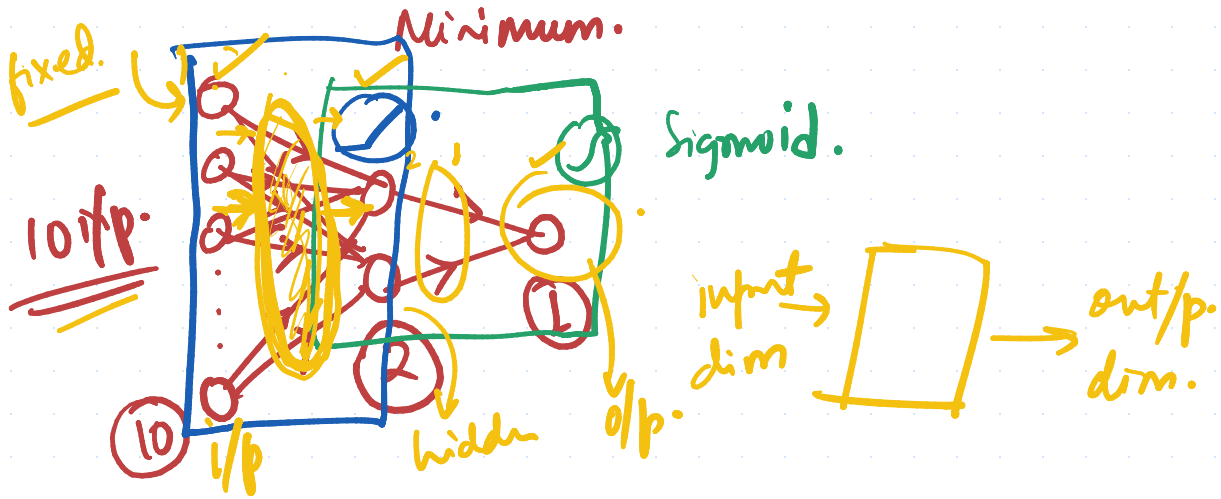
• Model.

• Resume.

1000 epochs.

⇒ 150 → power shutdown.





## Model Defn: Class Based.

init → op size, Act, inp size. } defined.

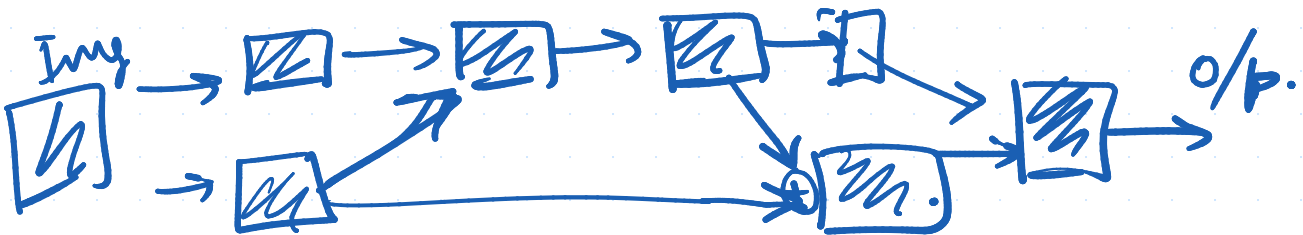
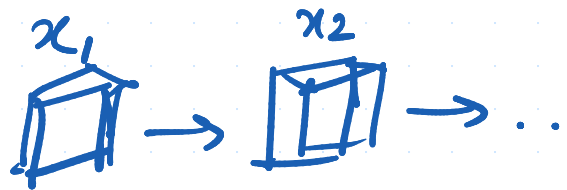
forward → Use defn. to construct model.

conv = nn.Conv2d(:)

conv2d = nn.Conv2d(inp, out, kernel size, padding, stride)

$$x_2 = \text{conv2d}(x_1)$$

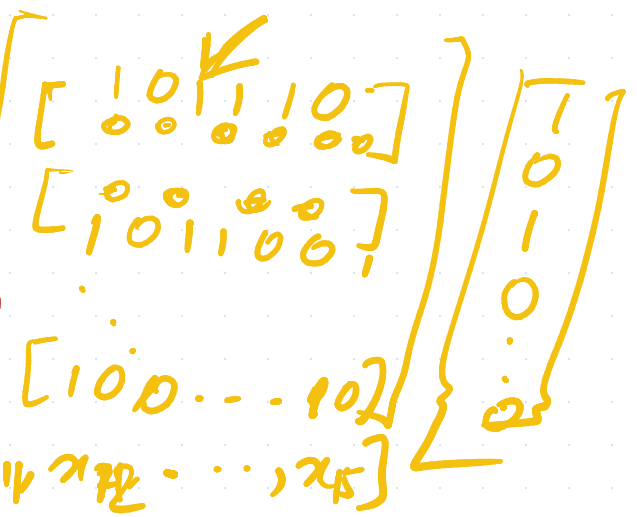
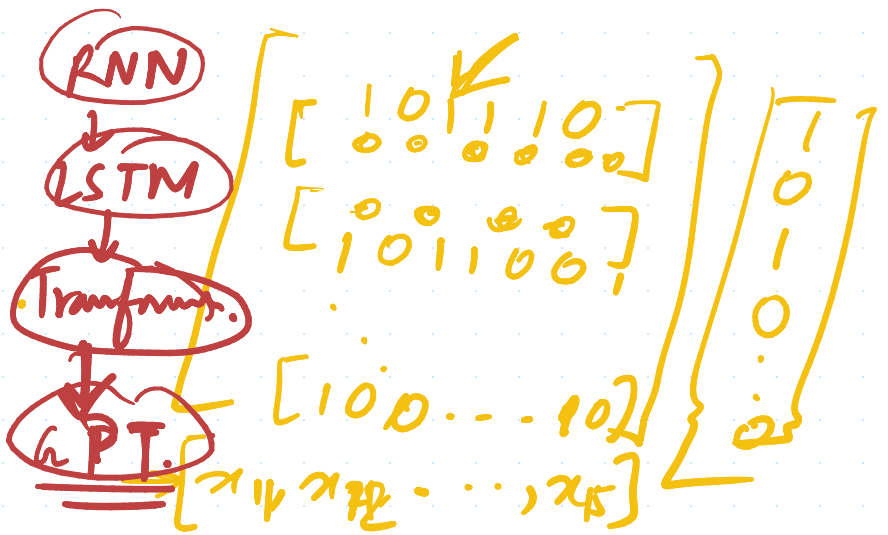
$$x_2 = \text{conv2d}(x_1)$$



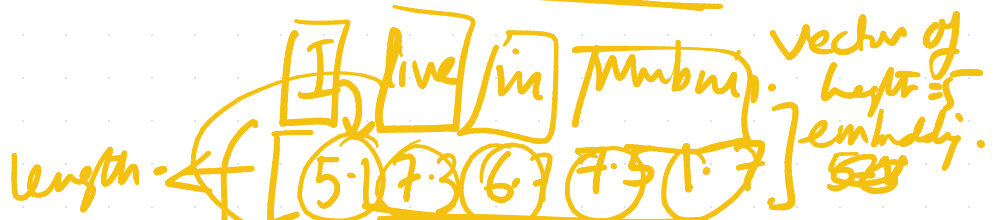
Data generator

→ init

→ -- get\_item()



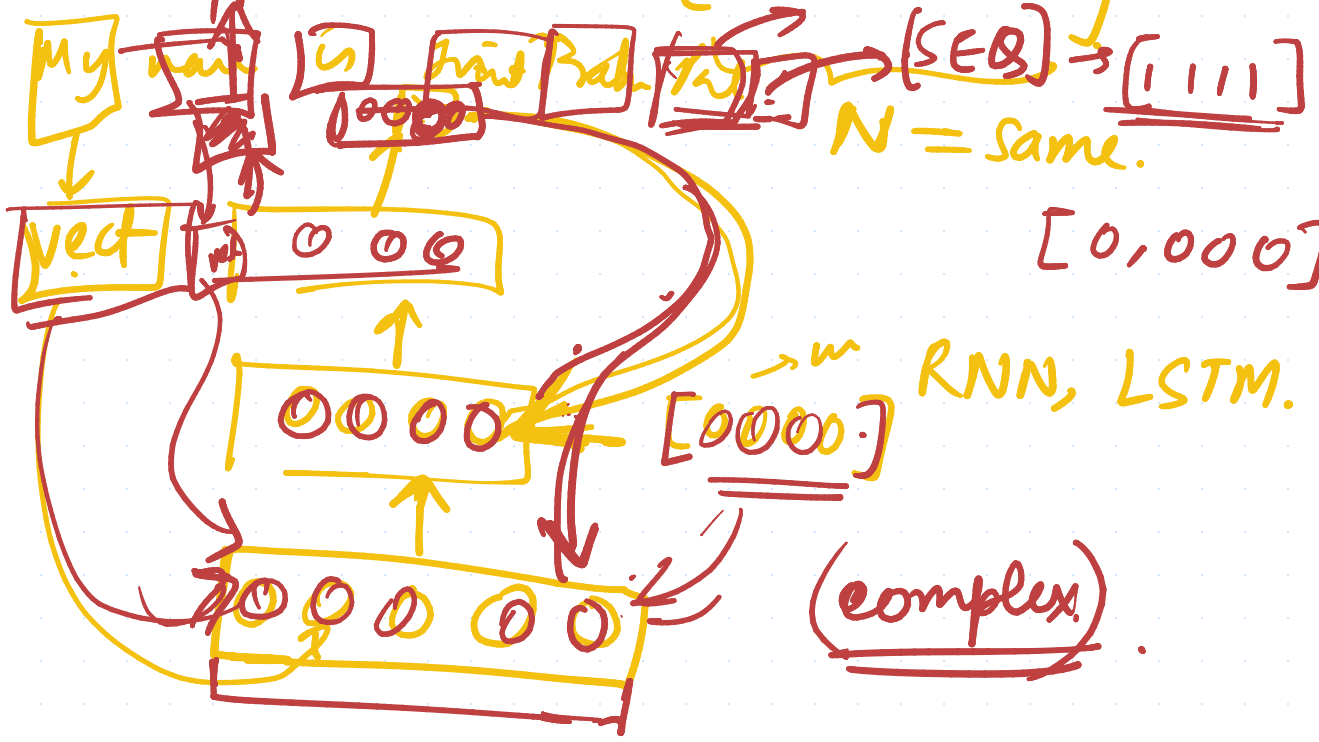
→  $[x_{21}, x_{22}, \dots, x_{20}]$



M = differ. →  $[6, 2, 7, 3, \dots]$

Max sequence  $[- \dots -]$

t=1 M<sub>1</sub> t=2 o/p M<sub>2</sub> t=3 M<sub>2</sub>



1000 length  $\rightarrow$  LSTM.

[ Ram is a good He - - - ]  
- - - - - Shayam - - - ]

Somish - - - - - LSTM.  
Mr. - - - - -

Trans. (GPT  $\rightarrow$  4000 words) 2 page.

Pytorch Model load  $\rightarrow$  LSTM Defn is reqd. 100.

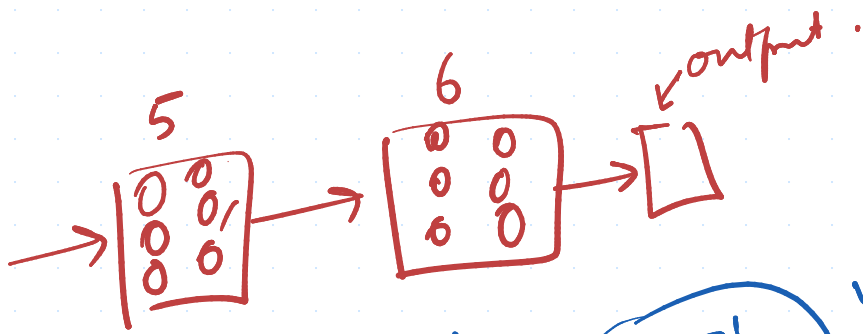
Defn. of model required.

Keras  
model.load(:)

Class Model()  
nt - - -  
- - -  
func.()

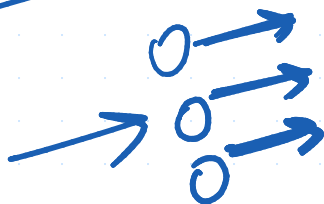
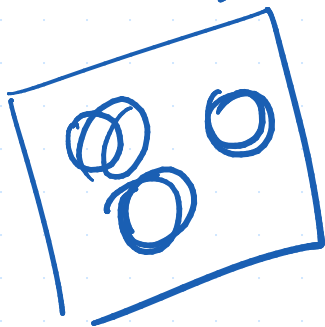
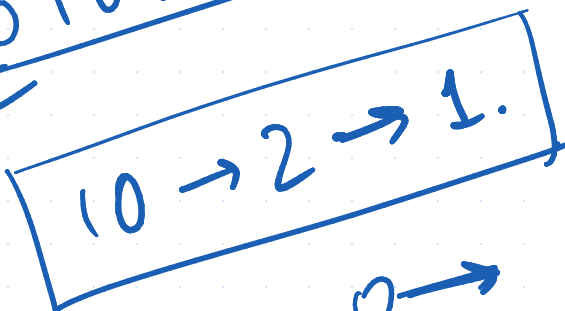
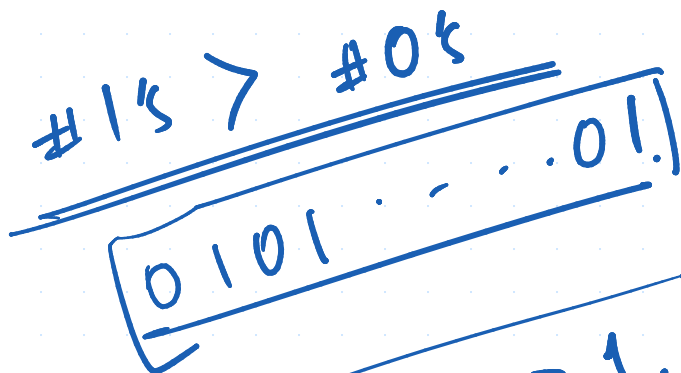
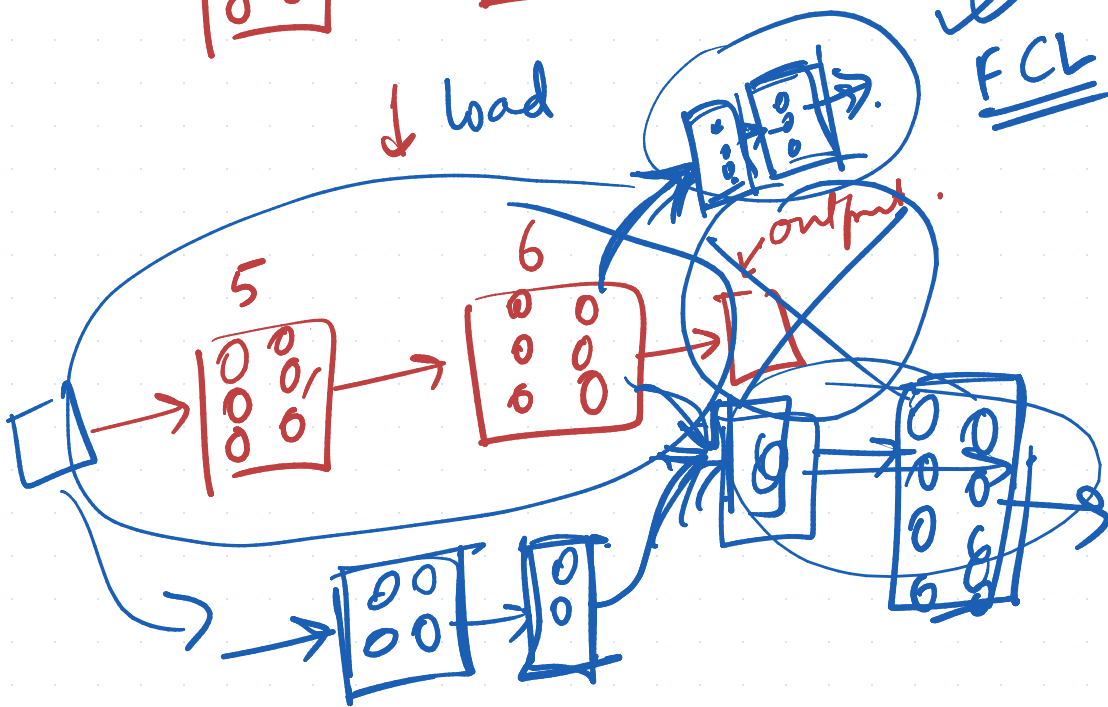
Dump Keras.  $\rightarrow$  Dump defn. as binary.  
(file = h5)





↓ load

← FCL.

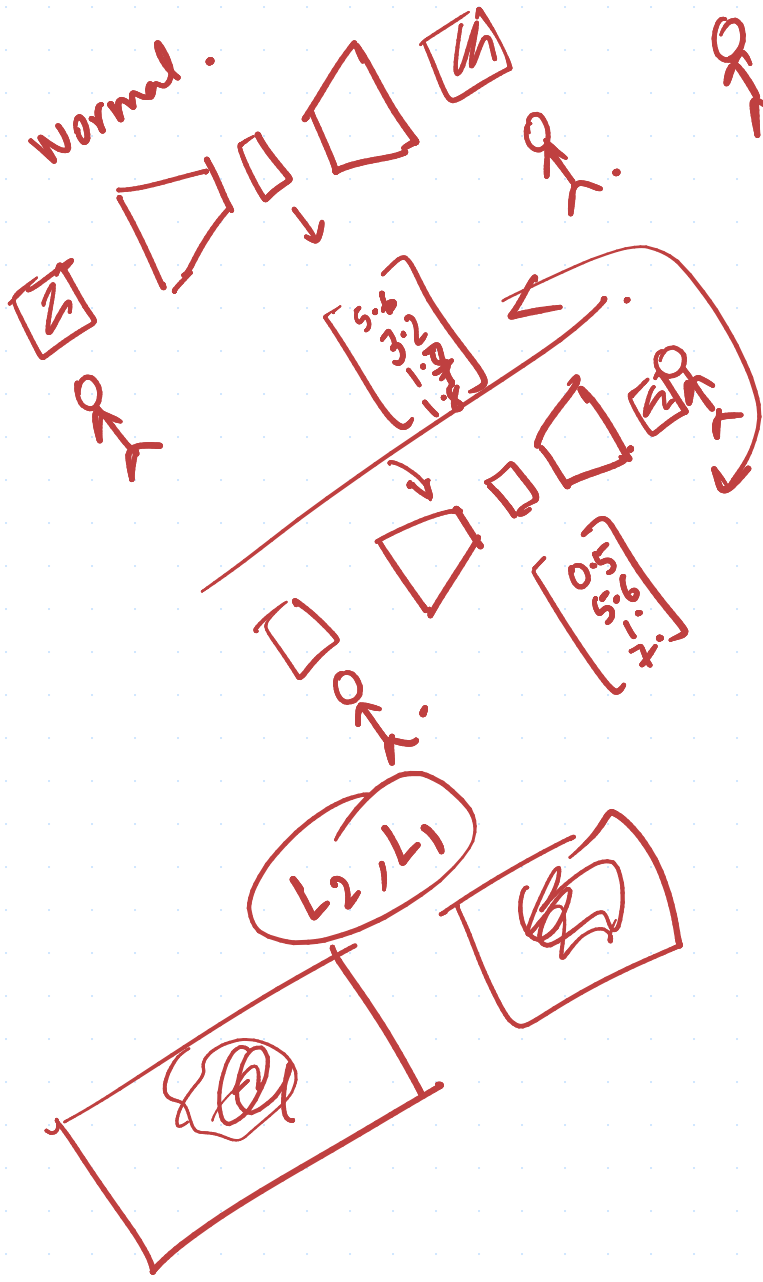


Dataloader  
 (softmax)

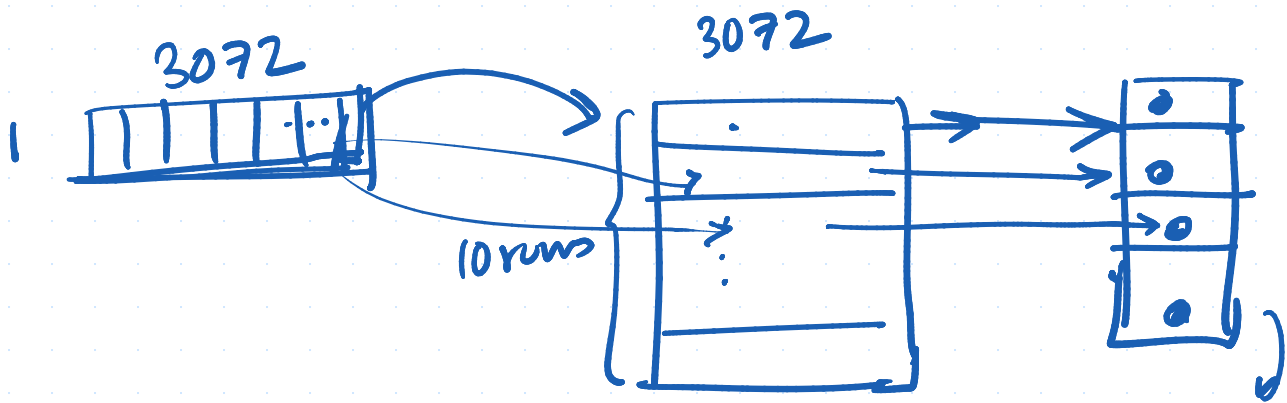
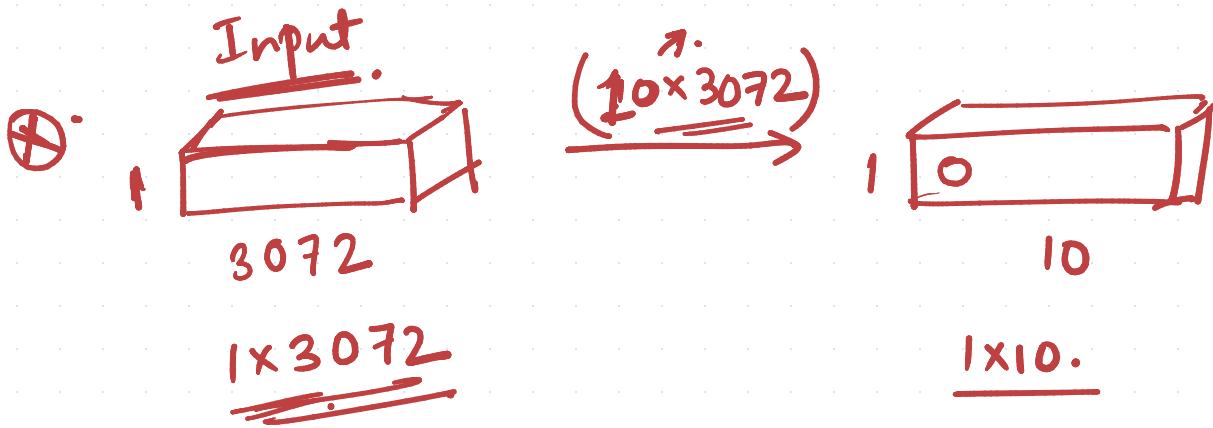
cat/dog/other class.

(Multiclass classification)

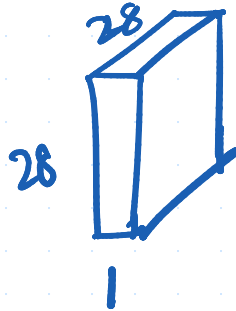
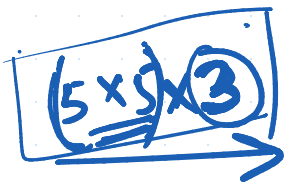
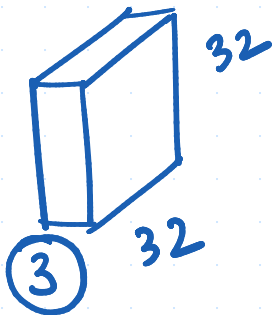
- ~~Skipline~~ →
- ~~Convolution / parameter~~ → (code/theory) 93
- Theory completion →
- Project / Recitation
- Assignment discussion



# Convolution & Parameter Calculation



Input / Image

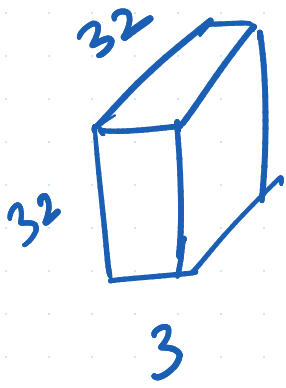


→ Activation map.

# parameters ⇒  
# (weights + biases)

$$5 \times 5 \times 3 + 1$$

$$\Rightarrow 75 + 1 = 76.$$



$$3 \times (5 \times 5)$$

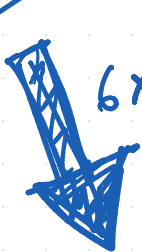
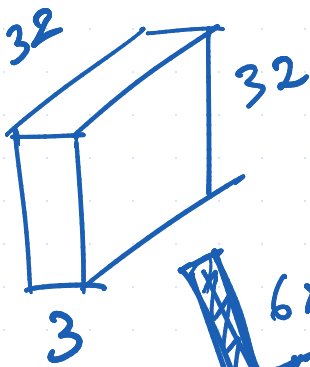
Try to reduce the no. of parameters.

$$(1 \times 5 \times 5) \times 3 + 3 = 78$$

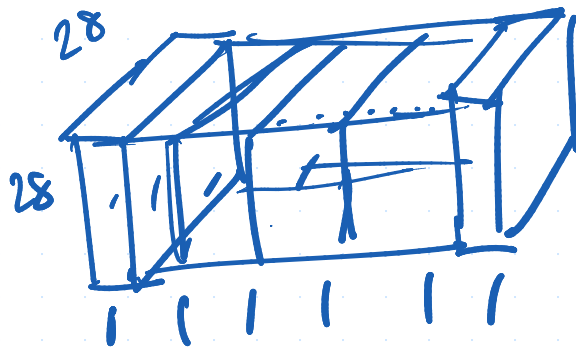
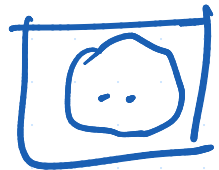
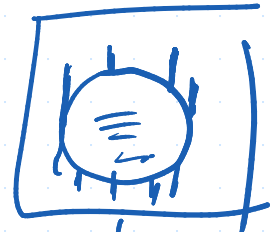
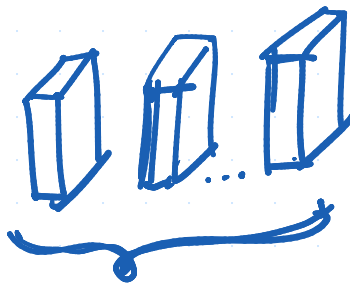
$$\Rightarrow 75 + 3 = 78$$

$$76$$

⚡



$$6 \times (3 \times 5 \times 5)$$



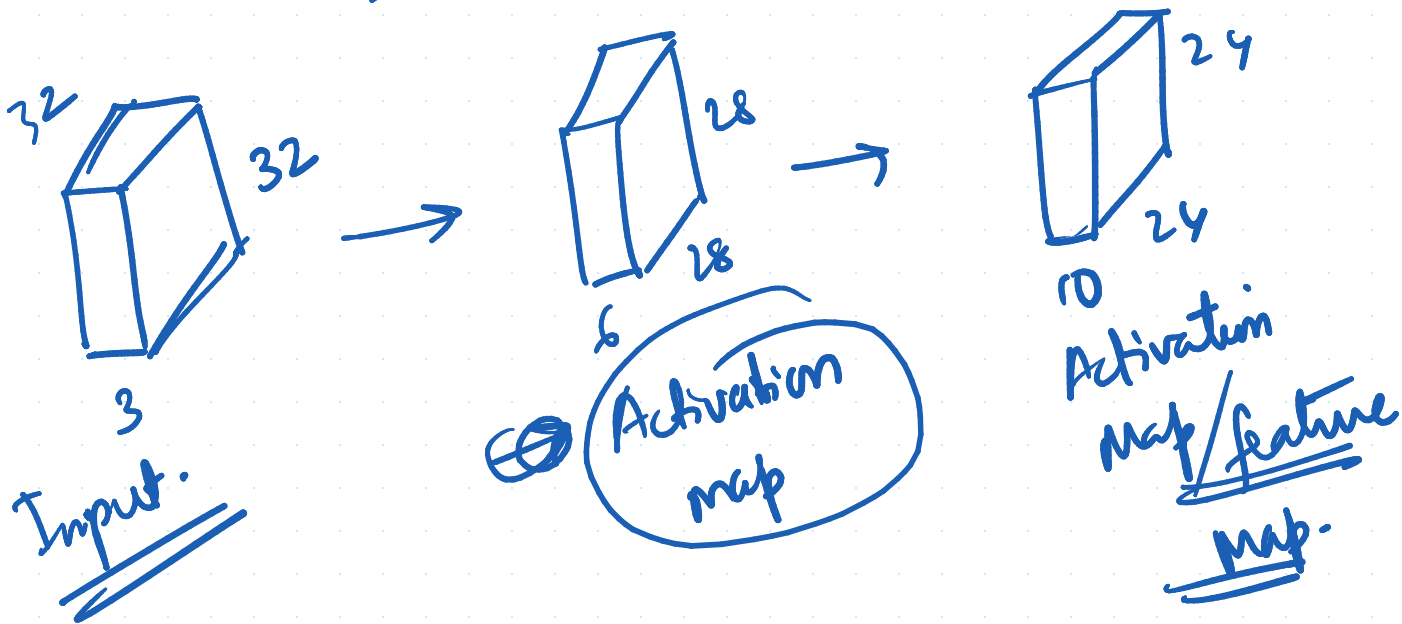
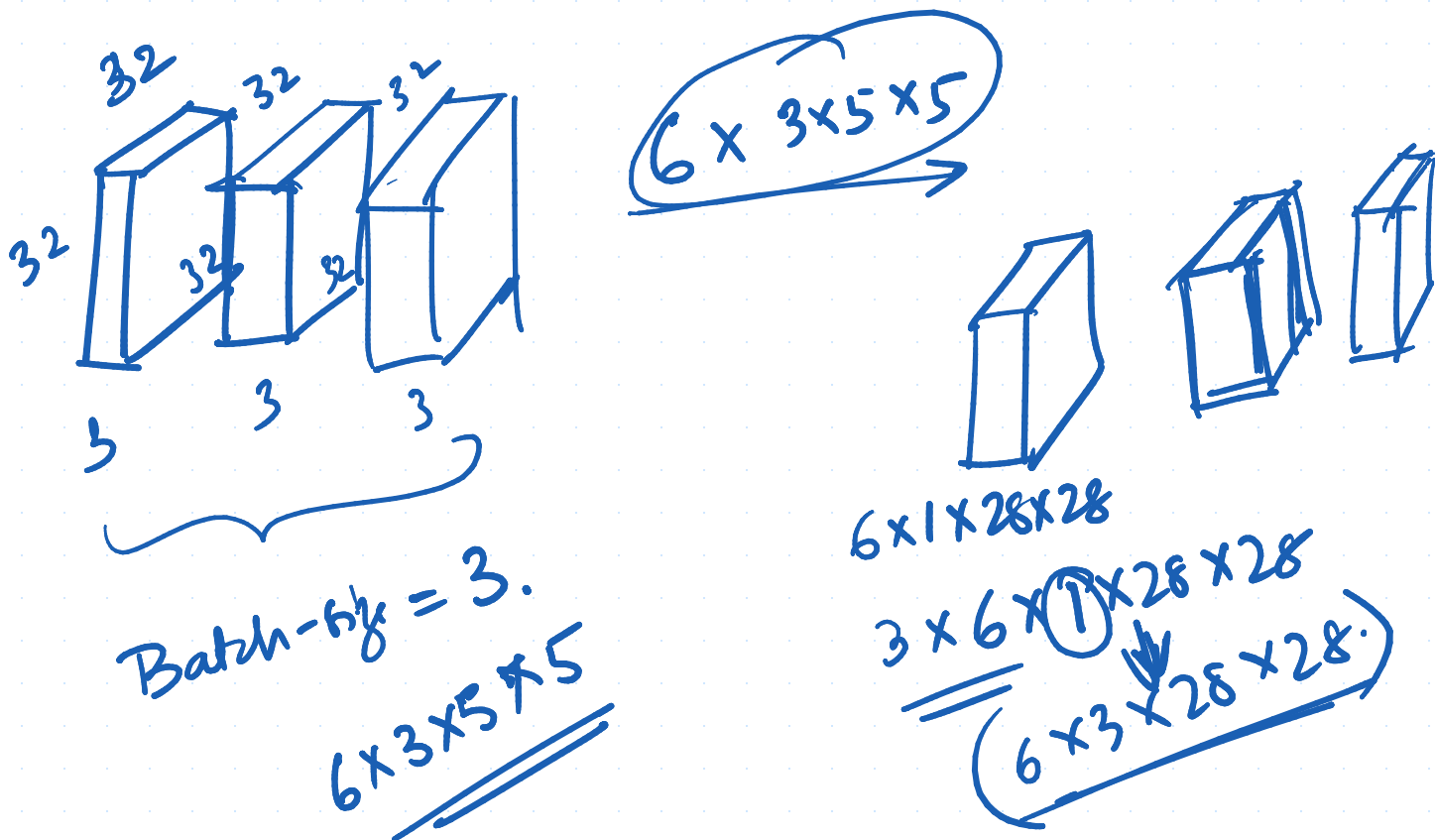
$$6 \times 1 \times 28 \times 28$$

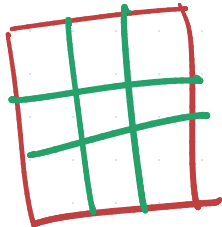
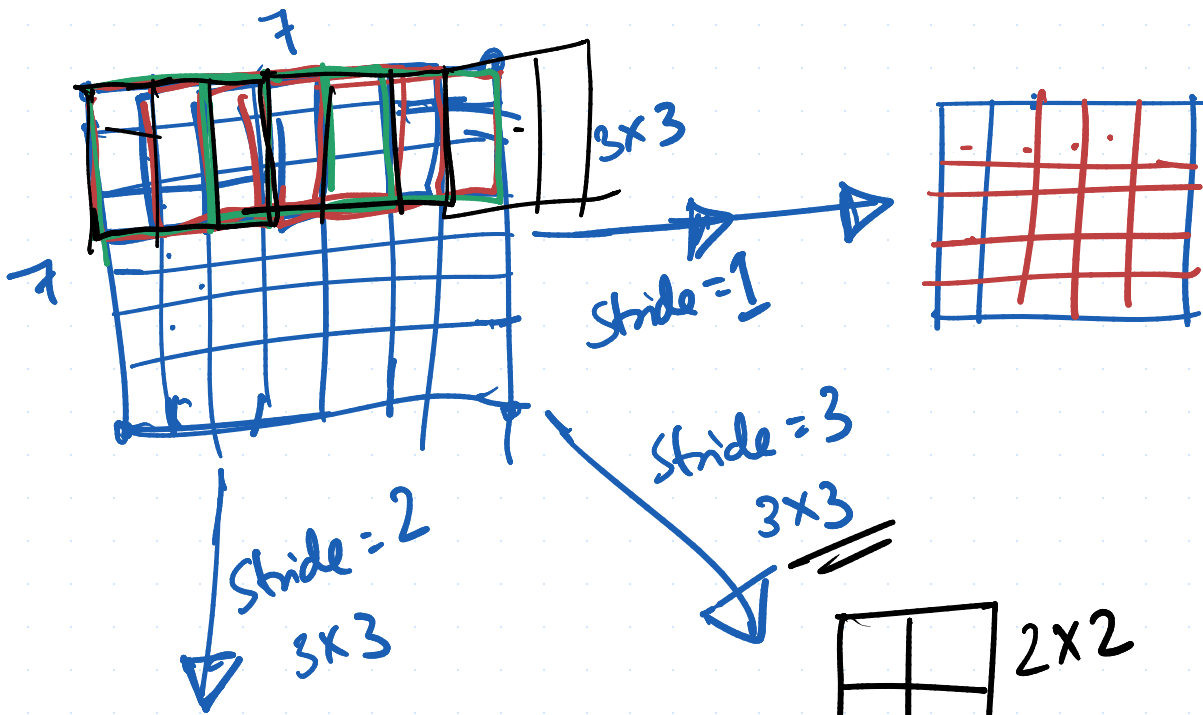
$$\downarrow$$

$$(6 \times 28 \times 28)$$



# parameters  $\Rightarrow \underline{\underline{(6 \times 3 \times 5 \times 5 + 6)}}$

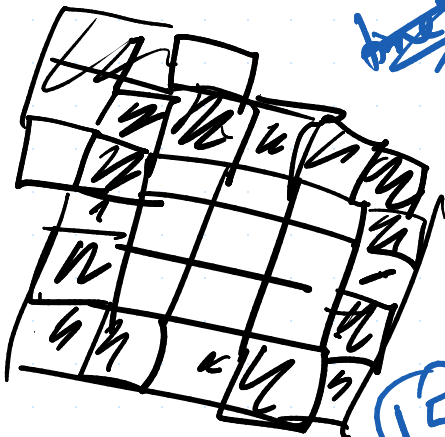




C++ → fast,

Python 3 - Competitive Programming

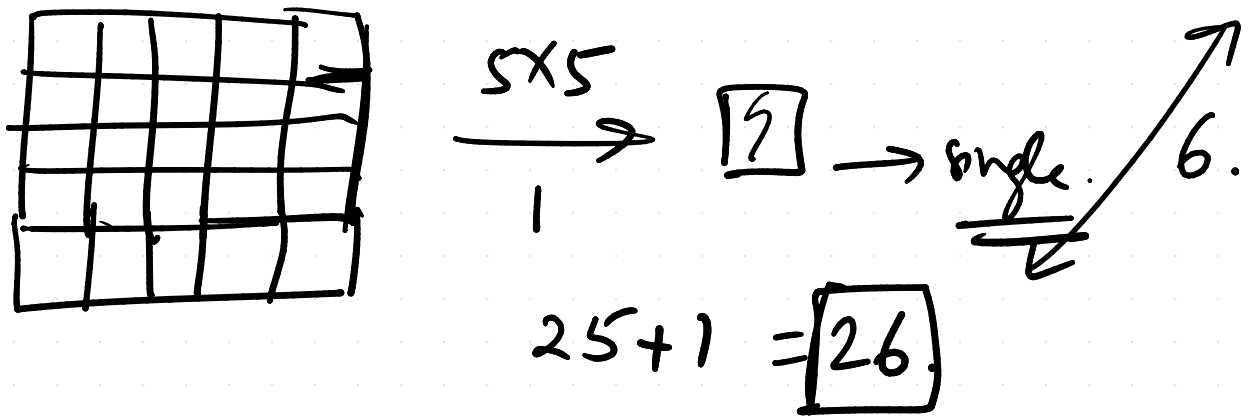
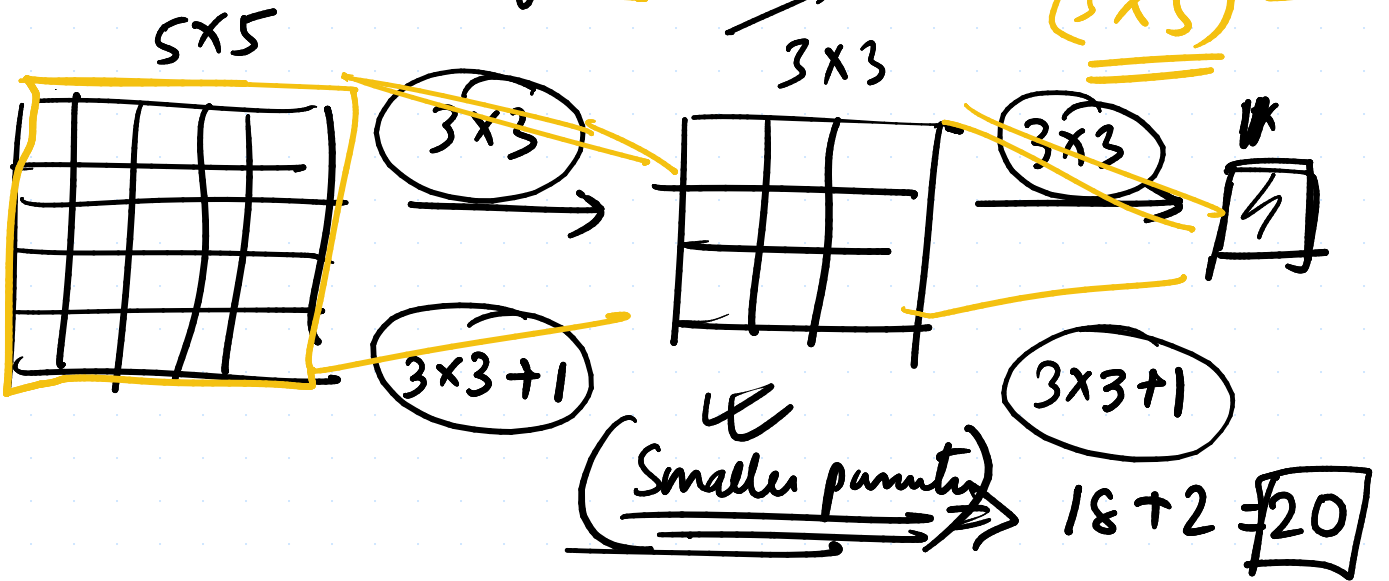
leetcode ②



ICPC 4hrs → ①5  
Google Code Jam 2  
 ← 4 → hard → problem.  
 $\frac{25}{2} = 12 \rightarrow 12 \times 4 \Rightarrow$  ④8  
DSA → Google/Microsoft

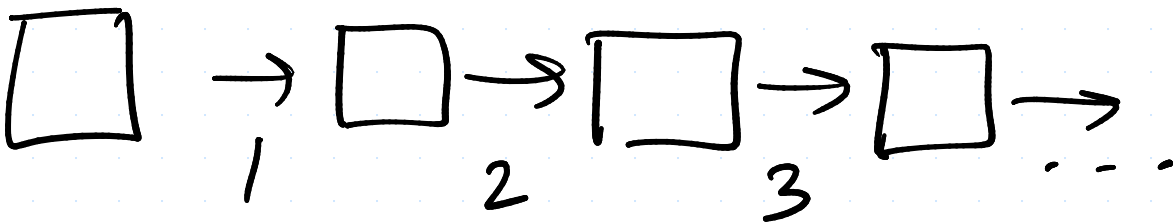
Receptive field.

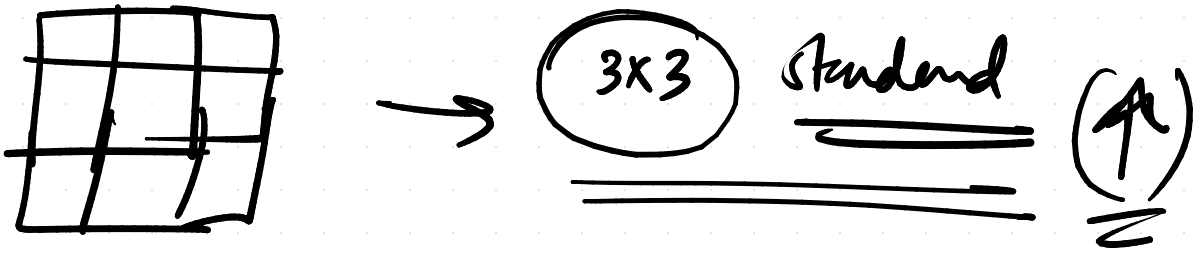
2 (3x3) kernels in series - (5x5)



computation ↓

memory / storage · overhead





AlexNet  $\Rightarrow$  (1x1)      (5x5)

