# DSA5202 LEC5 ⟶ Convolutional Neural Network

Recap:

① FCNN { shallow / deep

→ formulation: { T: depth of NN; $d_t$: width of NN

$$\begin{cases} f(x) = v^T f_T(x) \\ f_{t+1}(x) = \sigma(W_t f_t(x) + b_t) \quad t = 0, 1, \dots, T-1. \end{cases}$$

$W_t \in \mathbb{R}^{d_{t+1} \times d_t}$ $b_t \in \mathbb{R}^{d_{t+1}}$

② "Richness" of Hypothesis Space w.r.t { width $M$ / depth $T$

↔ model capacity

☆

→ force NN to learn Hierarchical Feature Extractor

via forcing $f_\theta(x_i)$ matches $y_i$

"learning"

③ Back-propagation Algorithm

→ forward: $x \to x_1 \to \cdots \to x_T \to x_{T+1}$.

→ backward: $x_{T+1} \to p_{T+1} \to \nabla_{W_T} \leftarrow x_T$

$p_T \to \nabla_{W_{T-1}} \leftarrow x_{T-1}$

$\vdots$

$p_1 \to \nabla_{W_0} \leftarrow x_0$

Issue: when we use "mini-batch GD"

$$\Downarrow$$

$$W_{t+1} = W_t - \alpha_t \frac{1}{|B|} \sum_{x \in B} \nabla_W \ell\left(f(x; W), y\right)$$

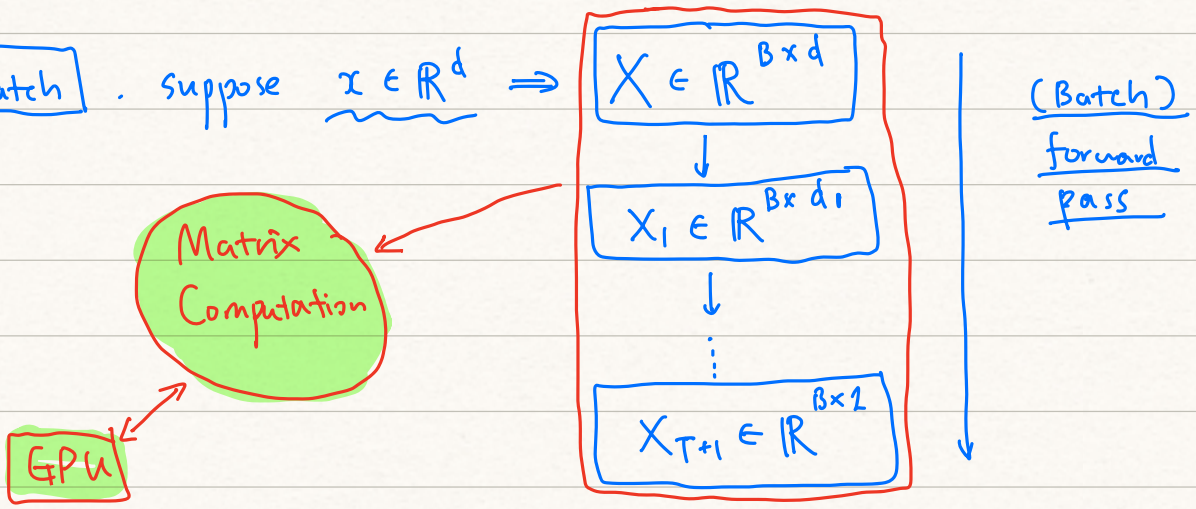
→ for each update, we need to go through $|B|$ times of forward pass and backward propagation.

Big number!

⭐⭐⭐

Remedy: In pratice, we do (Batch) forward pass & (Batch) backward propagation

for 1 batch. suppose $x \in \mathbb{R}^d$ ⇒




Similarly, we also calculate gradient for the total batch

$$X_{T+1} \in \mathbb{R}^{B \times 1} \longrightarrow P_{T+1} \longrightarrow \nabla_{W_T} \in \mathbb{R}^{B \times d_{T+1} \times d_T} \leftarrow X_T$$

$$\downarrow$$

$$P_T \longrightarrow \nabla_{W_{T-1}} \in \mathbb{R}^{B \times d_T \times d_{T-1}} \leftarrow X_{T-1}$$

$$\downarrow$$

$$\vdots$$

$$P_1 \longrightarrow \nabla_{W_0} \in \mathbb{R}^{B \times d_1 \times d} \leftarrow X_0$$

Today's Lecture:

1. Issue of FCNN → structure-agnostic

↓ formulate

idea: for FCNN, there is too much freedom of choosing weight

if $f \in \mathcal{H}_{FCNN}$, then $f_T \in \mathcal{H}_{FCNN}$

Here, $f_T(x) := f(T(x))$, $T(\cdot)$ is translation

for FCNN, it will treat original input and translated input equivalently ⇒ FCNN is structure agnostic, which means it will ignore the natural structure (temporal, spatial) of input.

↓

Dis-advantage if our inputs do have some structure

↳ { image
time series

2. CNN: → Convolutional Neural Network

① Convolution operation on infinitely-long vecs ☆

$$(w * x)(k) = \sum_{i=-\infty}^{+\infty} w(i)\, x(k+i)$$

avg

$$\begin{cases} w = \{w(i) : i \in \mathbb{Z}\} \rightarrow \text{filter} \\ x = \{x(i) : i \in \mathbb{Z}\} \rightarrow \text{signal} \end{cases}$$

filter { smoothing ↔ low-pass
edge ↔ high-pass

② conv. on finite-long vecs ↔ padding

a) circular padding: $x \in \mathbb{R}^n$ $w \in \mathbb{R}^m$ $m < n$

$$(w * x)(k) = \sum_{i=0}^{m-1} w(i) \cdot x(k+i) \qquad k = 0, 1, \ldots, n-1$$

$x \rightarrow$ extend so that $x(j) = x(j-n)$ for $j \geq n$

e.g. $x(n+3) = x(3)$

b) Zero padding : $x \in \mathbb{R}^n$ $w \in \mathbb{R}^m$ $m < n$

center around $x(k)$

$$(w * x)(k) = \sum_{i=0}^{m-1} w(i) \cdot x\left(k+i-\left\lfloor \frac{m}{2} \right\rfloor\right) \qquad k = 0, 1, \ldots n-1$$

$x(j) = 0$ for $j \notin \{0, 1, 2, \ldots, n-1\}$

$\Downarrow$

$x(j) =$ constant $\Longrightarrow$ c) constant padding

③ Compare between MatMul and Convolution

FCNN : $x_{t+1} = \sigma(W_t \cdot x_t + b_t)$ → matrix multiplication

CNN : $x_{t+1} = \sigma(W_t * x_t + b_t)$ → convolution

$W_t$ → include multiple $W$ (kernel)

④ Advantage of CNN

circular padding

$$w * x = \begin{bmatrix} w_0 & w_1 & w_2 & 0 & 0 \\ 0 & w_0 & w_1 & w_2 & 0 \\ 0 & 0 & w_0 & w_1 & w_2 \\ w_2 & 0 & 0 & w_0 & w_1 \\ w_1 & w_2 & 0 & 0 & w_0 \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix}$$

1d case

$C_w$ → less freedom (3 dof)

Recap : In FCNN, dof $= 5 \times 5 = 25$

$f_T \in \mathcal{H}_{FCNN}$ if $f \in \mathcal{H}_{FCNN}$

a) weight sharing → break the invariant of translation for FCNN

b) sparsity → storage

c) translation equivariant

$f_T \in \mathcal{H}_{CNN}$ if $f \in \mathcal{H}_{CNN}$

$\Downarrow$

Structure – inagnostic!

3. Pooling { down-sampling
approximately translation invariant

4. E.g. of Deep CNN architecture

Body { $x_0 = x$
$x_{t+1} = T_{mp} \circ T_{conv} \circ x_t$ $t = 0, 1, \ldots, T-1$

head { $f(x) = T_{fcn} \circ \text{flatten} \circ x_T$