Chapter Seven

Sequence Learning

Zhang Yi, *IEEE Fellow*
Autumn 2020

Outline

- A Sequence Recognizing Example
- Review of BP for Mono-target Output NNs
- BP Method for Multi-target Outputs NNs
- BP Algorithm for Multi-target Outputs NNs
- Illustrative Examples
- Assignment

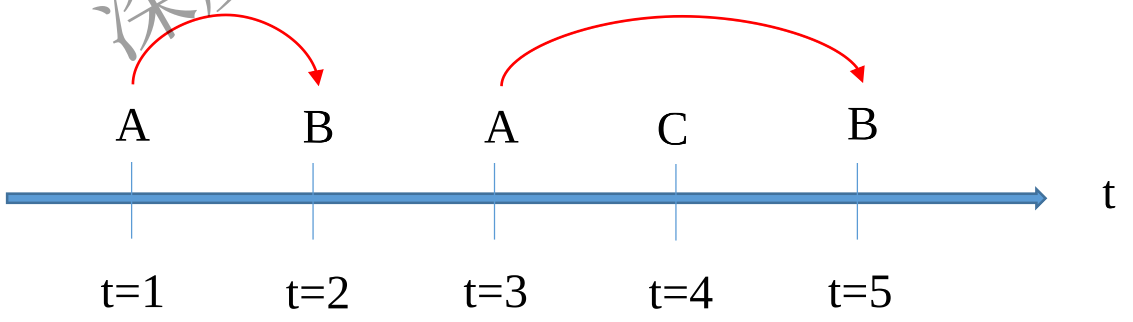
A Sequence Recognizing Example

Generated Sequences

1. ABACB
2. CCBBA
3. CACCB
4. ACCCB
5. CACBC
6. AAACB
7. BAACB
8. CCBAB
9. BCCAB
10. CABAC

……………

Problem: Recognize A followed by B.



A Sequence Recognizing Example

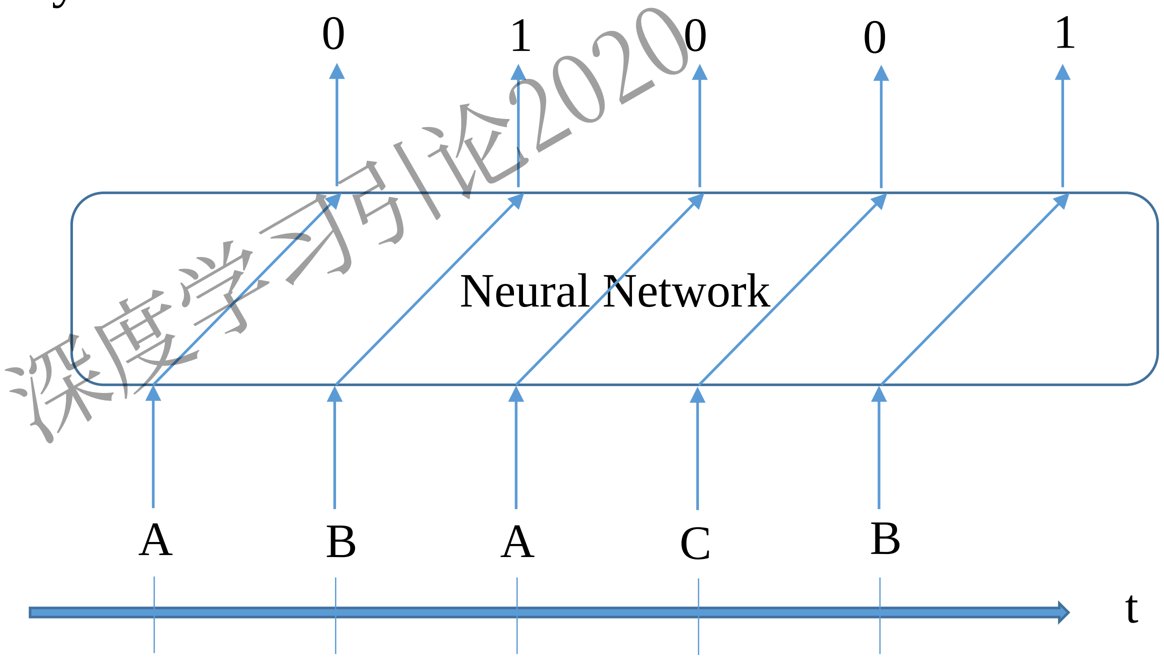
Recognize A followed by B Problem

The task is to recognize A followed by B.

Generated Sequences

1. ABCAB
2. CCBBA
3. CACCB
4. ACCCB
5. CACBC
6. AAACB
7. BAACB
8. CCBAB
9. BCCAB
10. CABAC

……………......



Problem: Can we use neural network to solve this problem?

A Sequence Recognizing Example

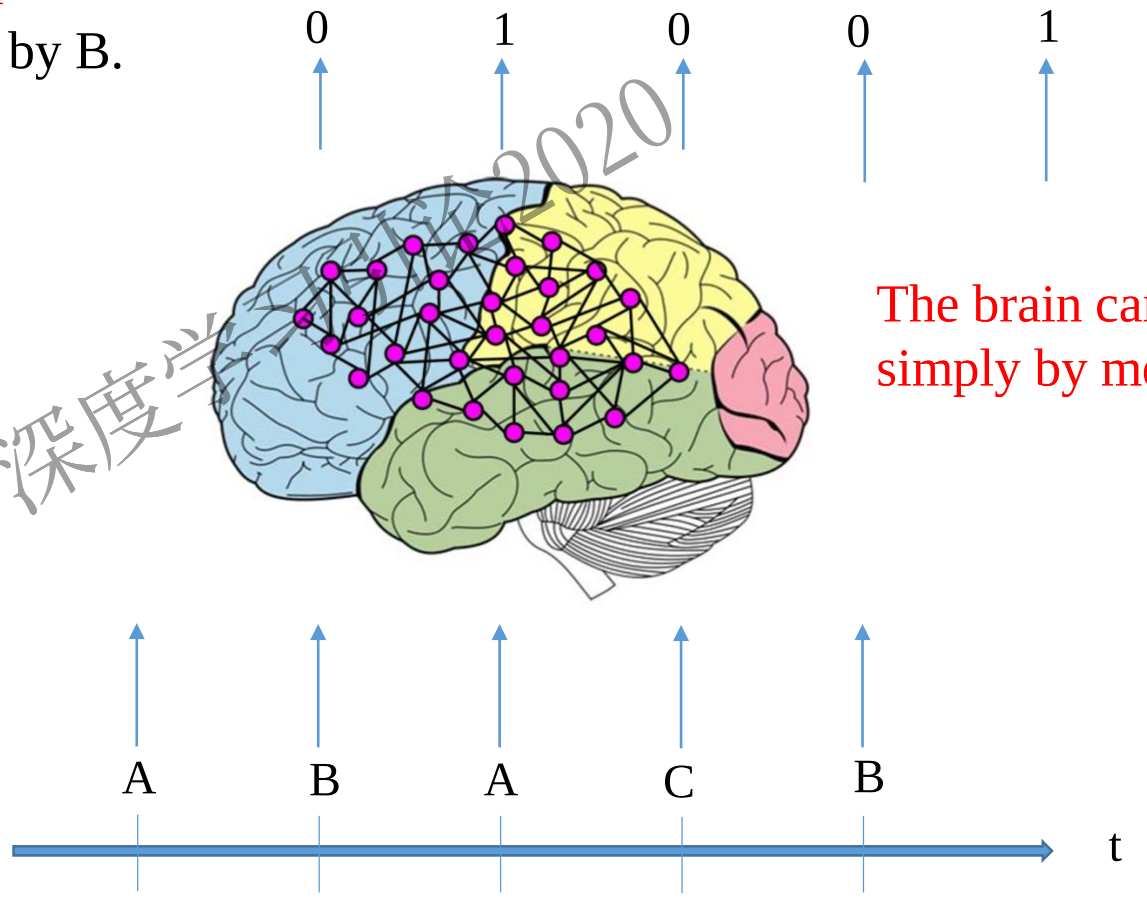
Recognize A followed by B Problem

The task is to recognize A followed by B.

Generated Sequences

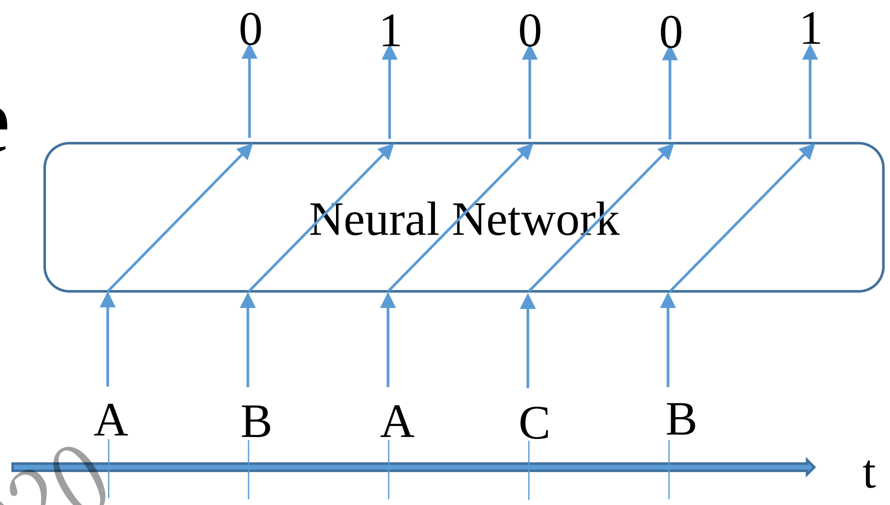
1. ABCAB
2. CCBBA
3. CACCB
4. ACCCB
5. CACBC
6. AAACB
7. BAACB
8. CCBAB
9. BCCAB
10. CABAC

……………......



The brain can solve this problem simply by memorizing the last A.

A Sequence Recognizing Example



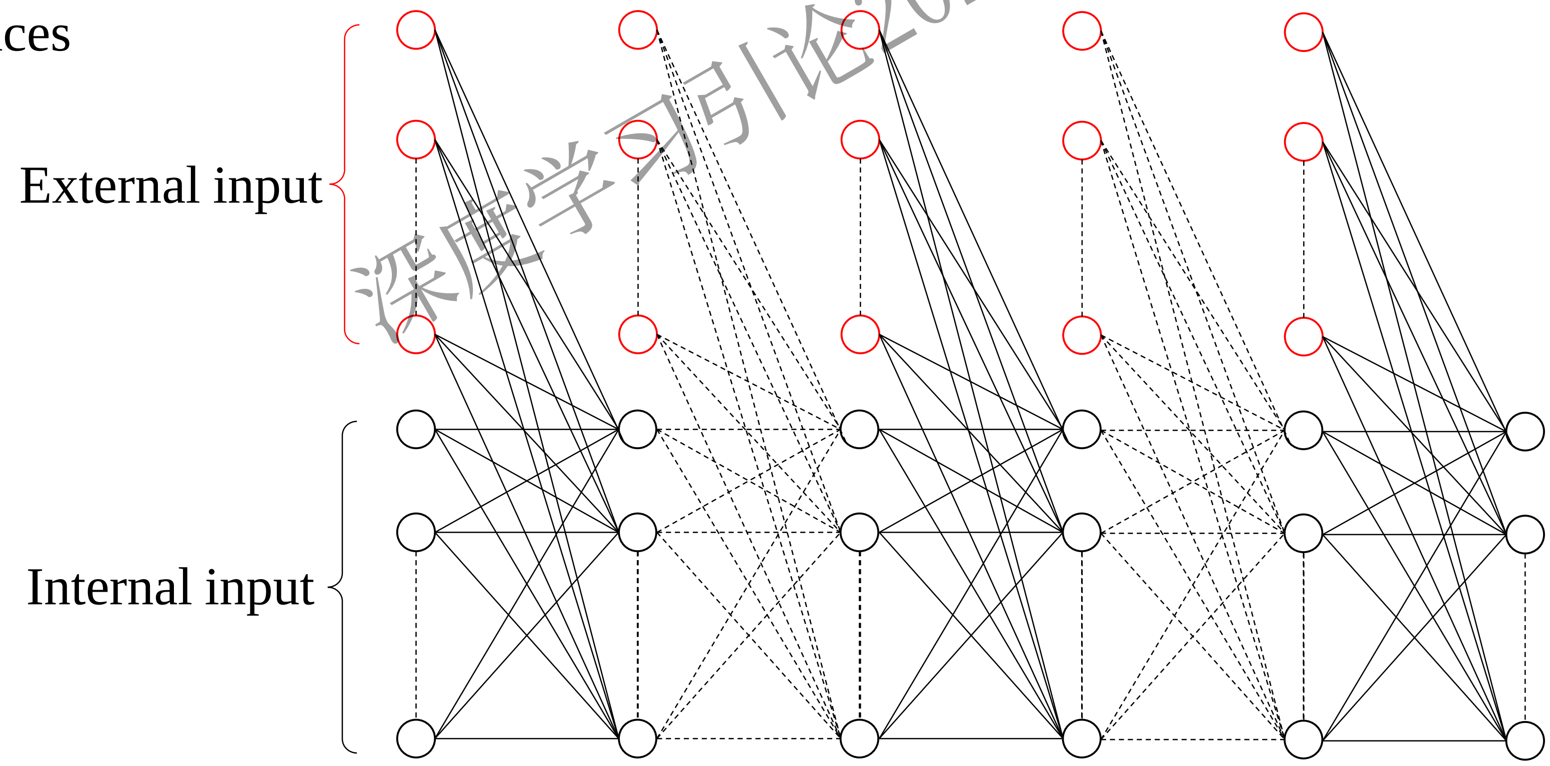
Recognize A followed by B Problem

The task is to recognize A followed by B.

Generated Sequences

1. ABCAB
2. CCBBA
3. CACCB
4. ACCCB
5. CACBC
6. AAACB
7. BAACB
8. CCBAB
9. BCCAB
10. CABAC

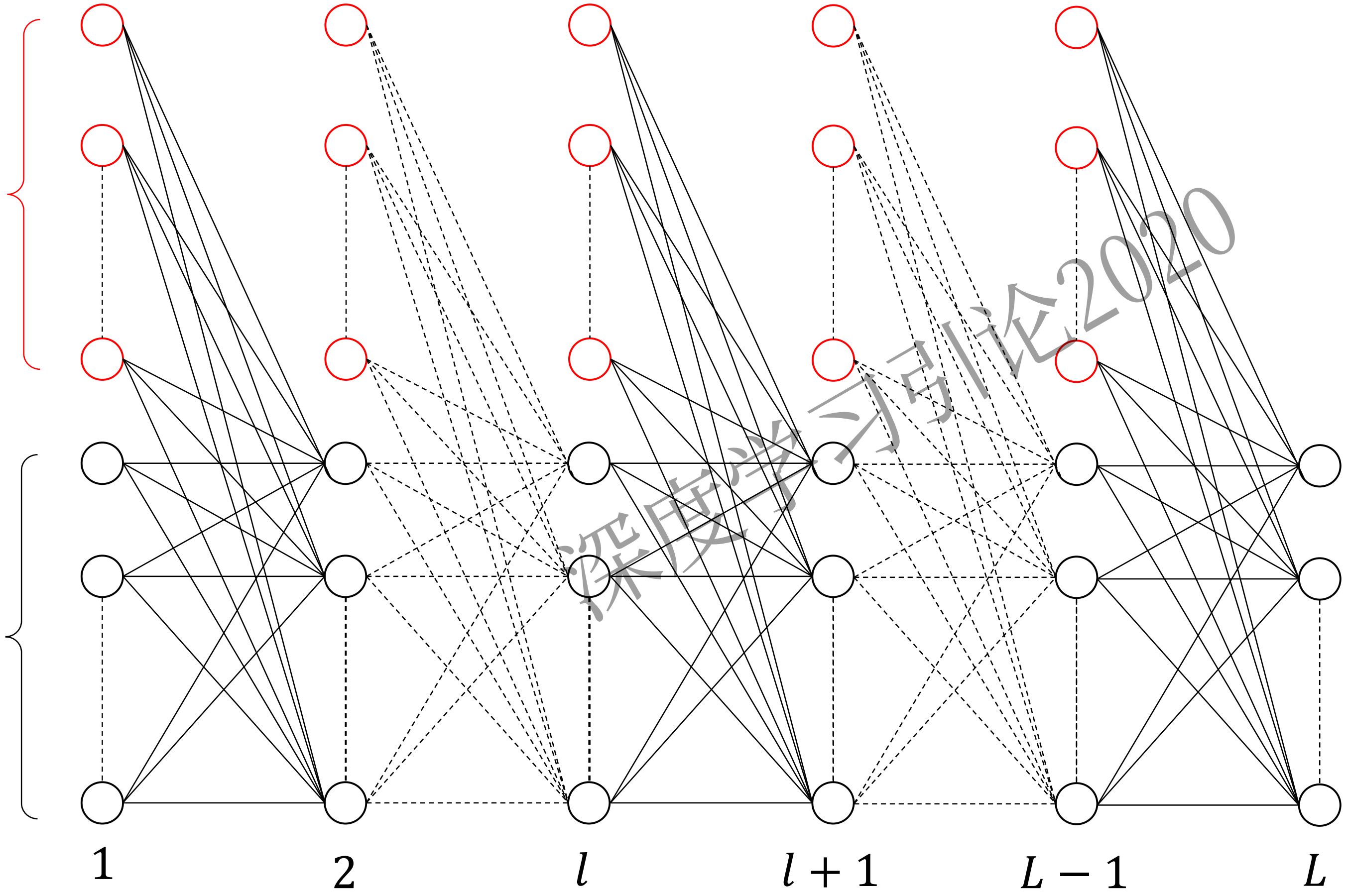
……………..



A Sequence Recognizing Example

External input

Internal input



Mono-target output

$$y^L = \begin{bmatrix} y_1^L \\ \vdots \\ y_{n_L}^L \end{bmatrix}$$

In the last layer only

Mono-target output network cannot solve the sequence recognizing problem.

A Sequence Recognizing Example

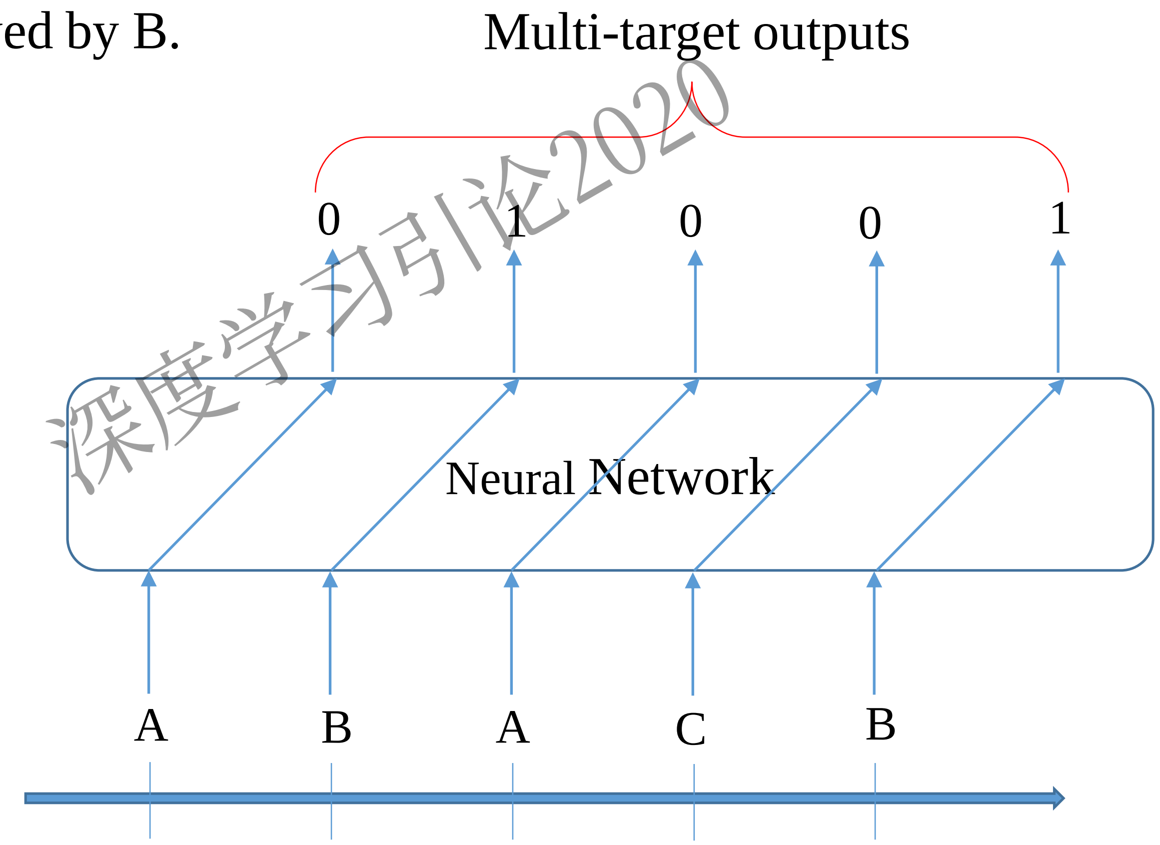
Recognize A followed by B Problem

The task is to recognize A followed by B.

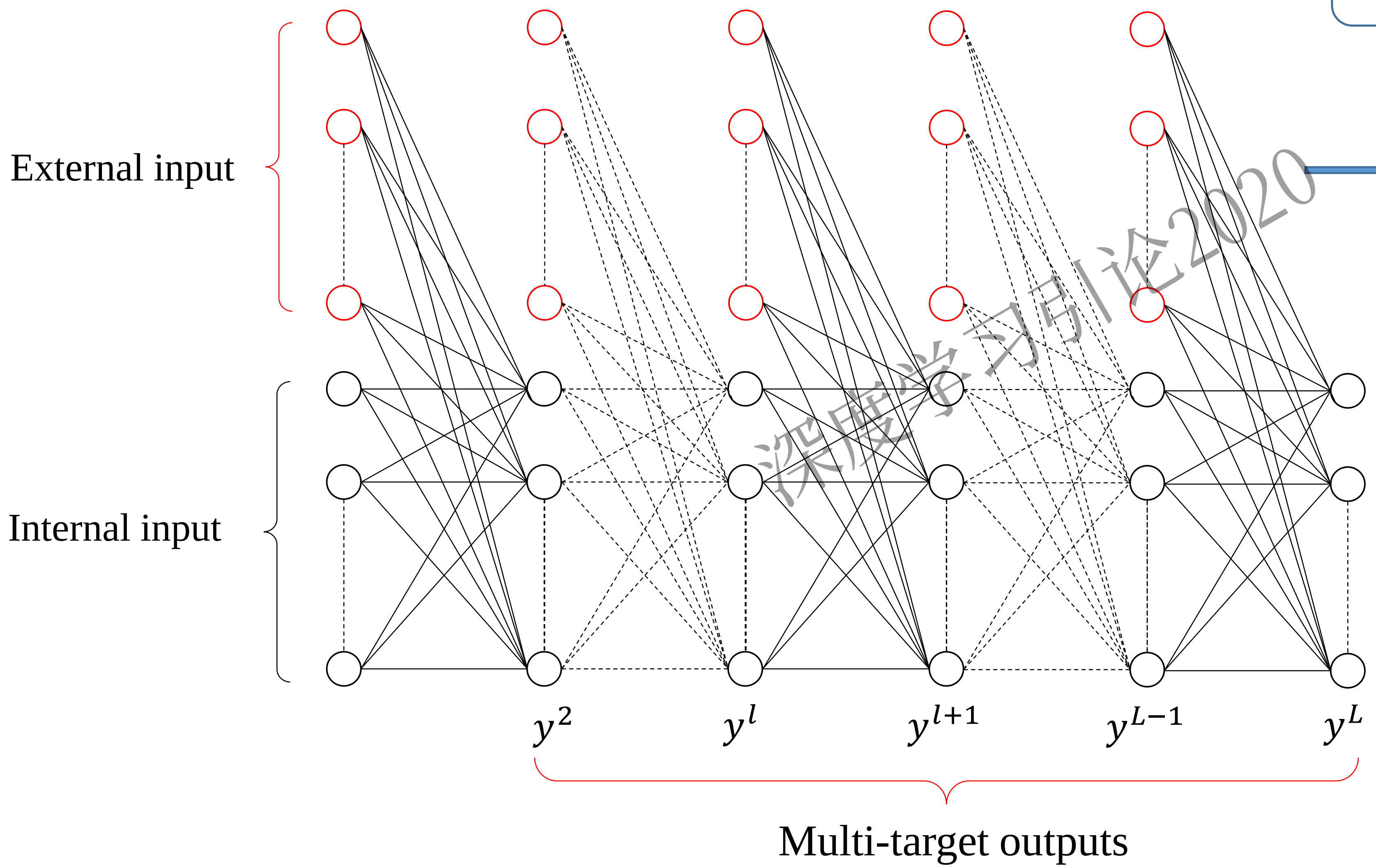
Generated Sequences

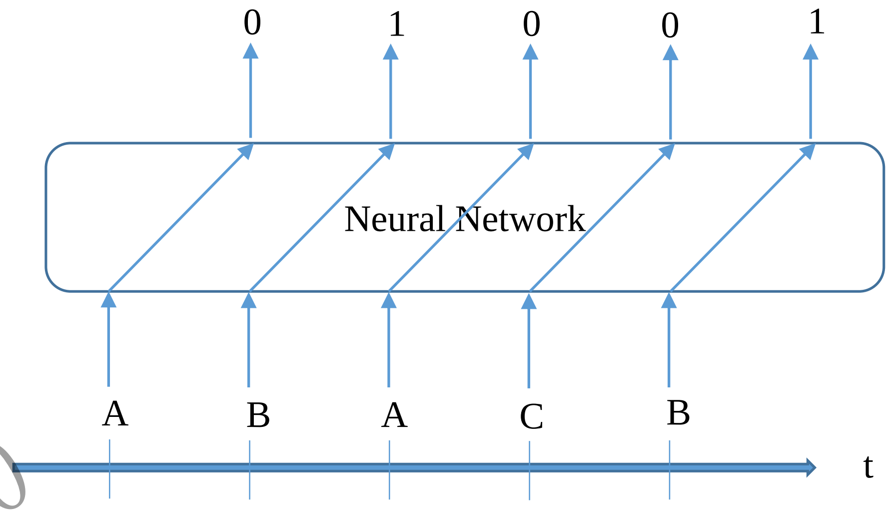
1. ABCAB
2. CCBBA
3. CACCB
4. ACCCB
5. CACBC
6. AAACB
7. BAACB
8. CCBAB
9. BCCAB
10. CABAC

……………..



A Sequence Recognizing Example



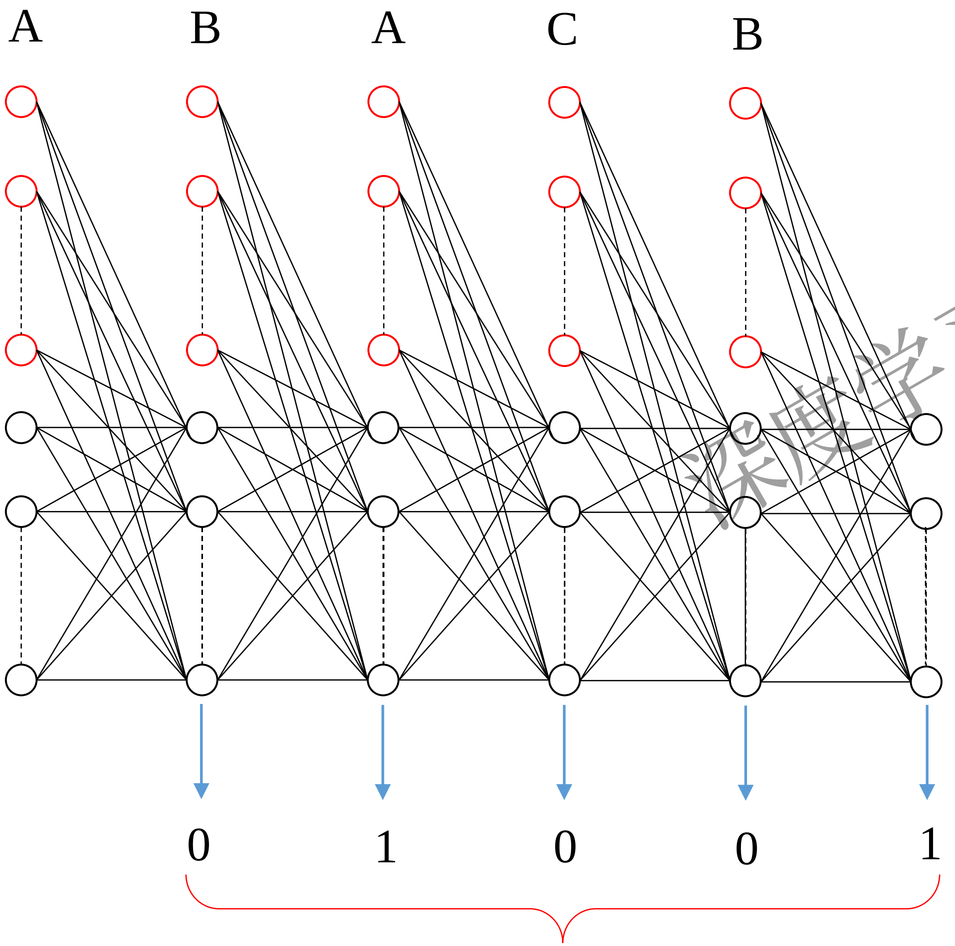


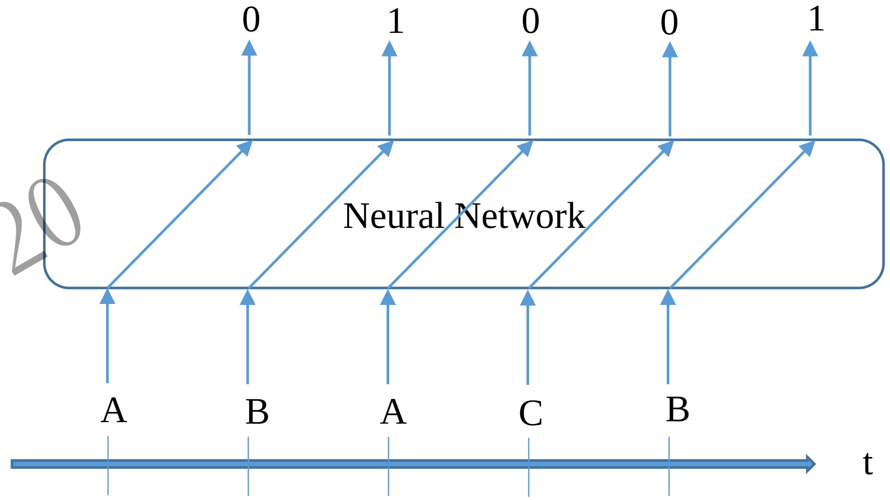
Multi-target outputs

$$y^l = \begin{bmatrix} y_1^l \\ \vdots \\ y_{n_L}^l \end{bmatrix}$$

$$(l = 2, \cdots, L)$$

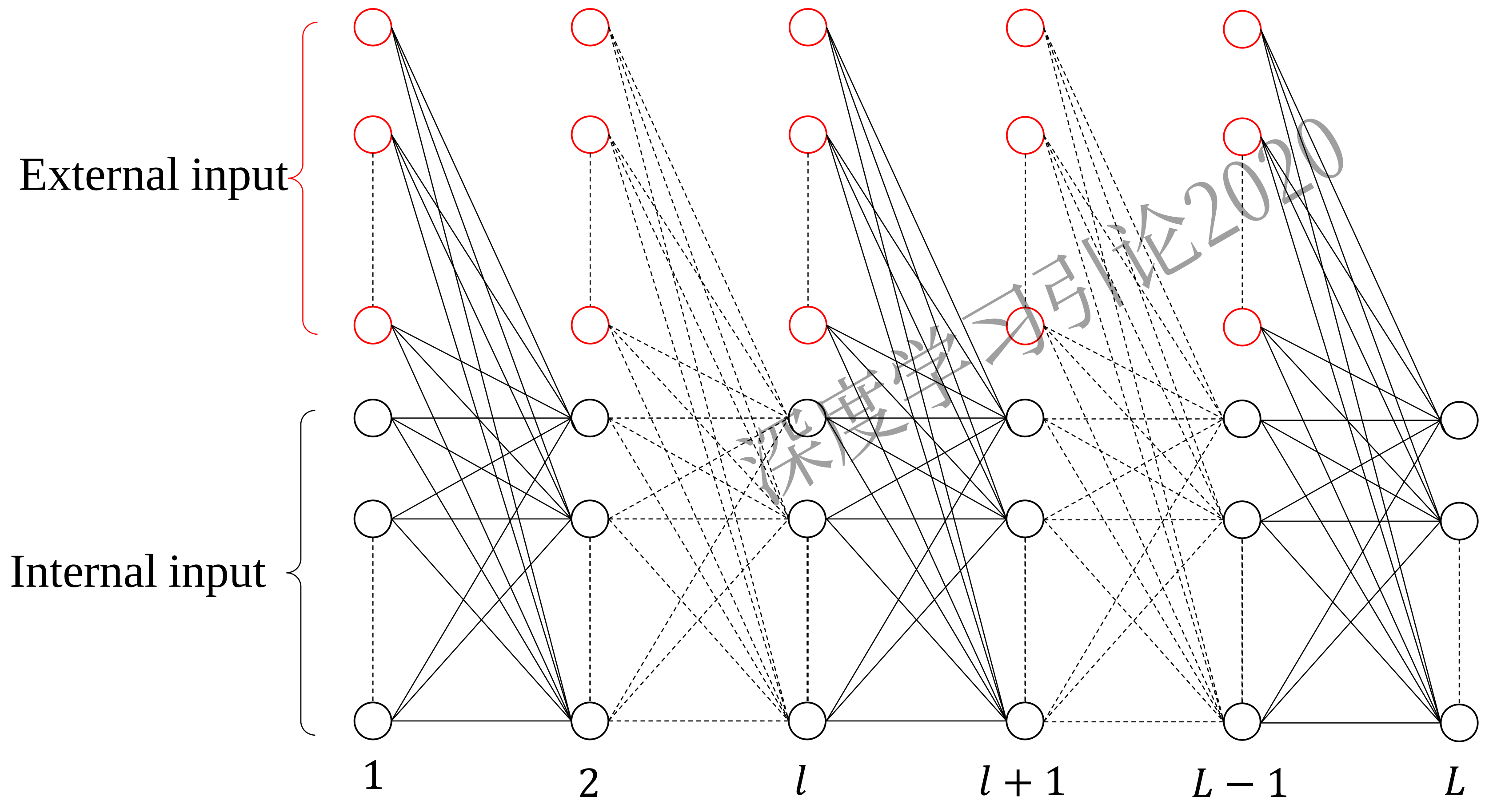
A Sequence Recognizing Example





Problem:
How to develop algorithm to train the network?

A Sequence Recognizing Example



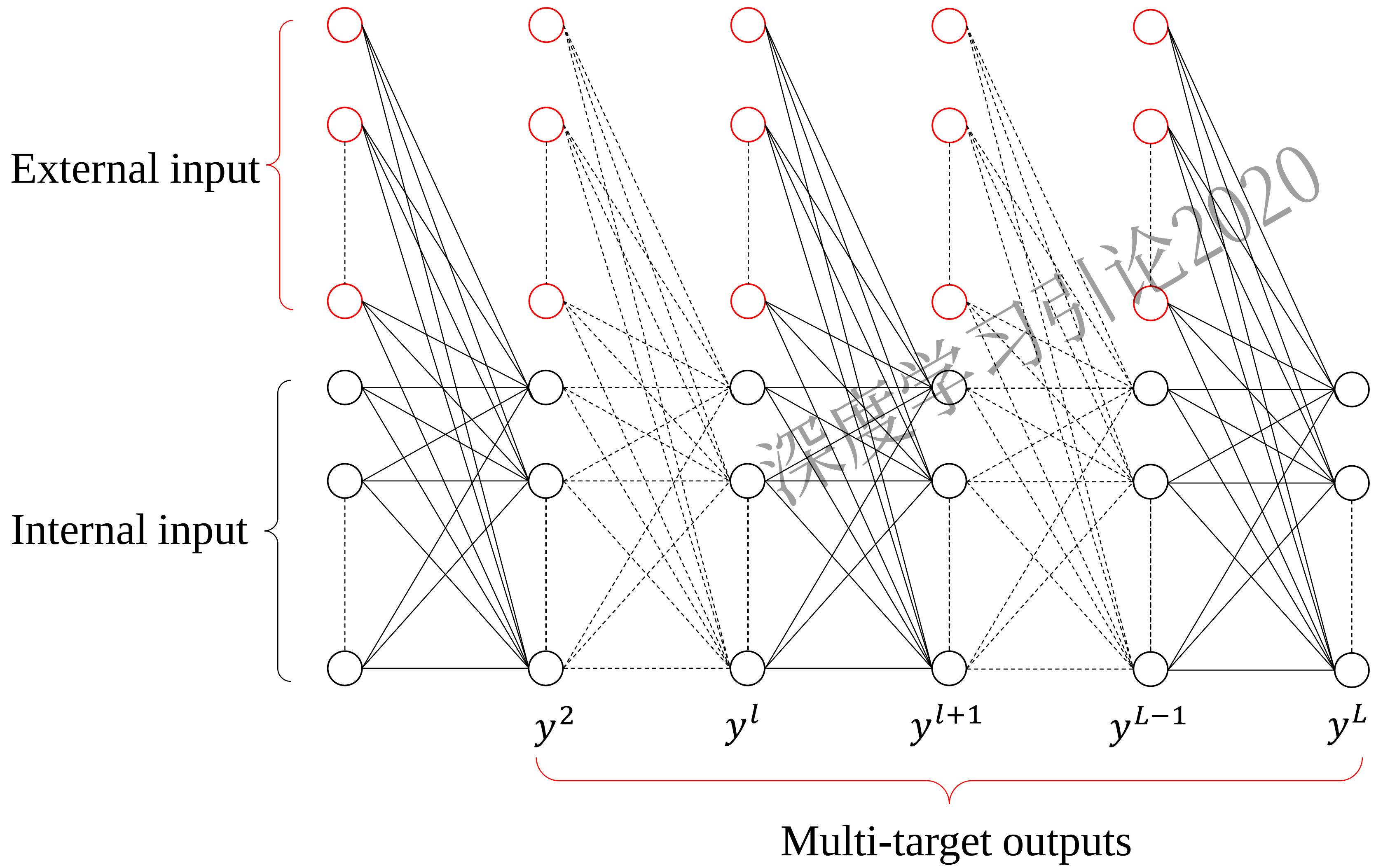
Mono-target output

$$y^L = \begin{bmatrix} y_1^L \\ \vdots \\ y_{n_L}^L \end{bmatrix}$$

In the last layer only

We already developed BP Algorithm for mono-target output.

A Sequence Recognizing Example



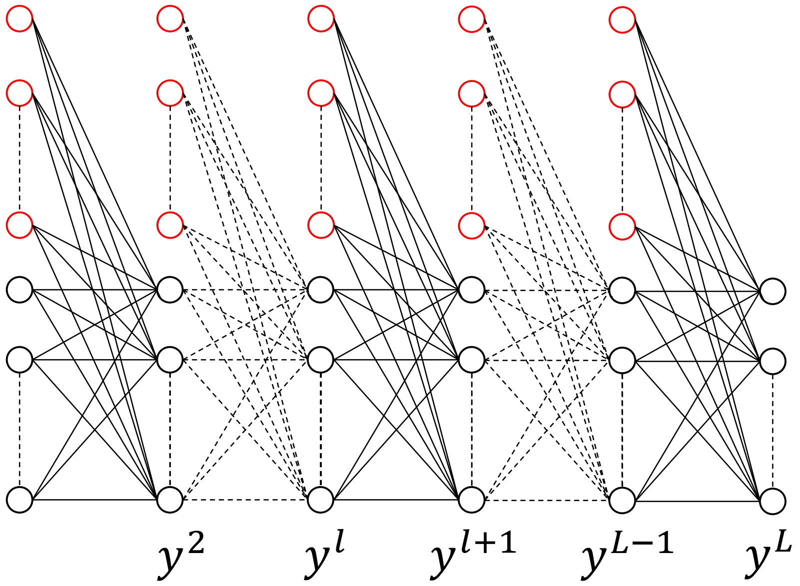
Problem:
Can we develop learning algorithms for multi-target output similar to BP for mono-target output?

Multi-target outputs

$$y^l = \begin{bmatrix} y_1^l \\ \vdots \\ y_{n_L}^l \end{bmatrix}$$

$$(l = 2, \cdots, L)$$

A Sequence Recognizing Example



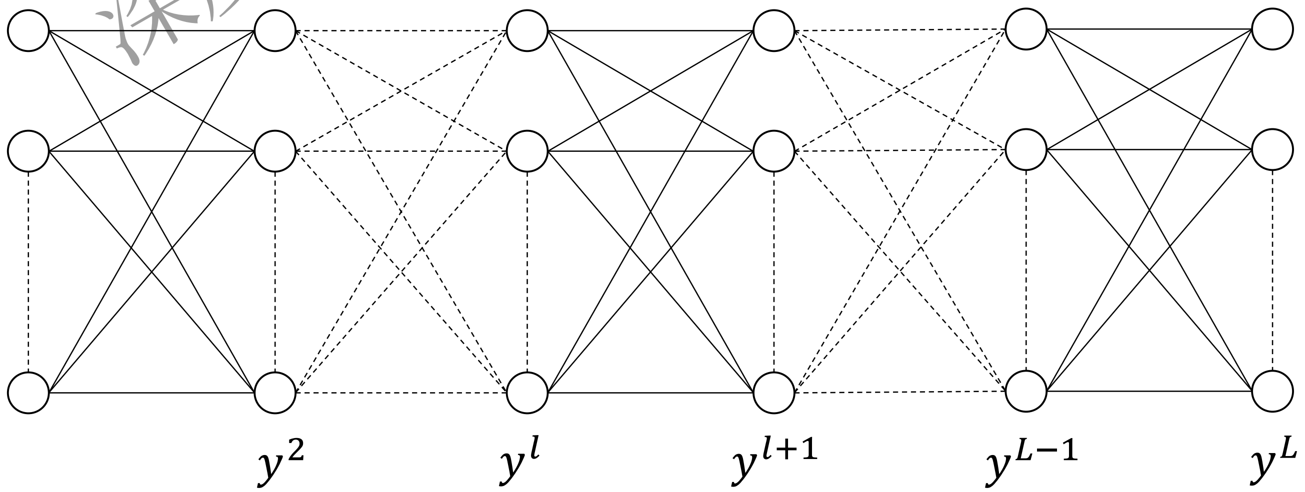
Multiple Target Outputs

$$y^l = \begin{bmatrix} y_1^l \\ \vdots \\ y_{n_L}^l \end{bmatrix}$$

$$(l = 2, \cdots, L)$$

Combine the external and internal inputs in first layer

Equivalent



Problem:

Can we develop learning algorithms similar to BP?

Outline

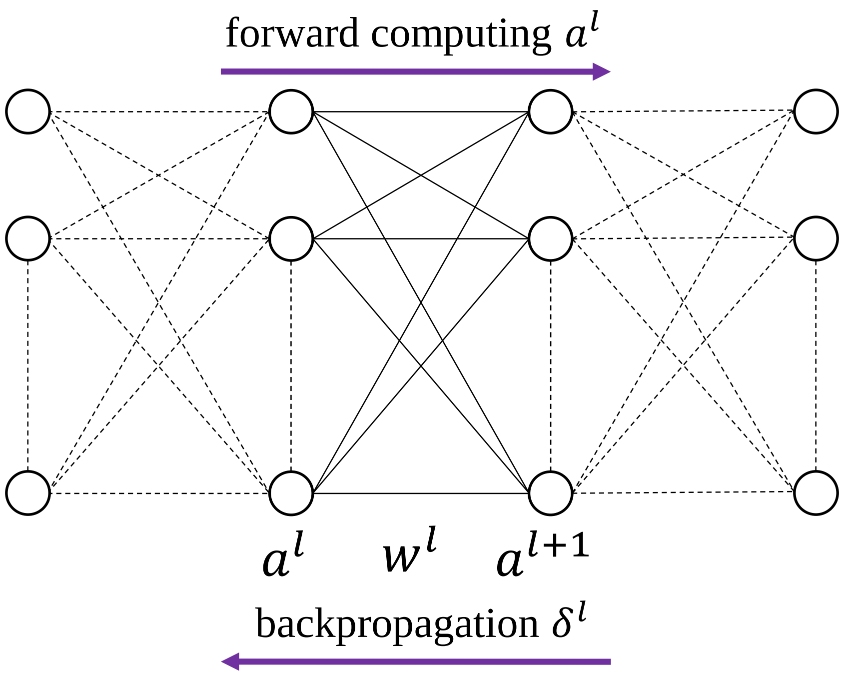
- A Sequence Recognizing Example
- Review of BP for Mono-target Output NNs
- BP Method for Multi-target Outputs NNs
- BP Algorithm for Multi-target Outputs NNs
- Illustrative Examples
- Assignment

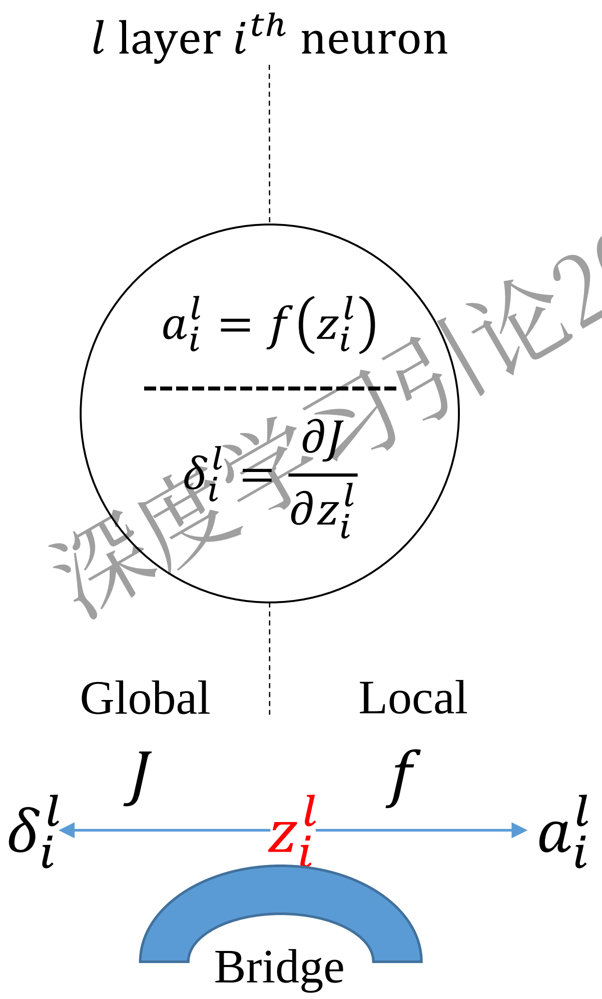
Only One Page to Understand BP

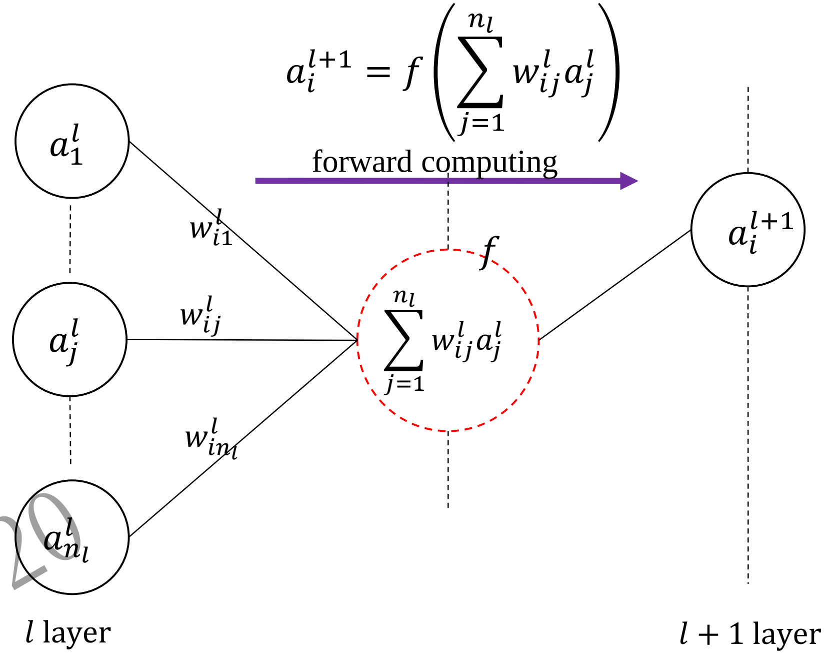
Cost function: $J(w^1, \cdots, w^{L-1})$

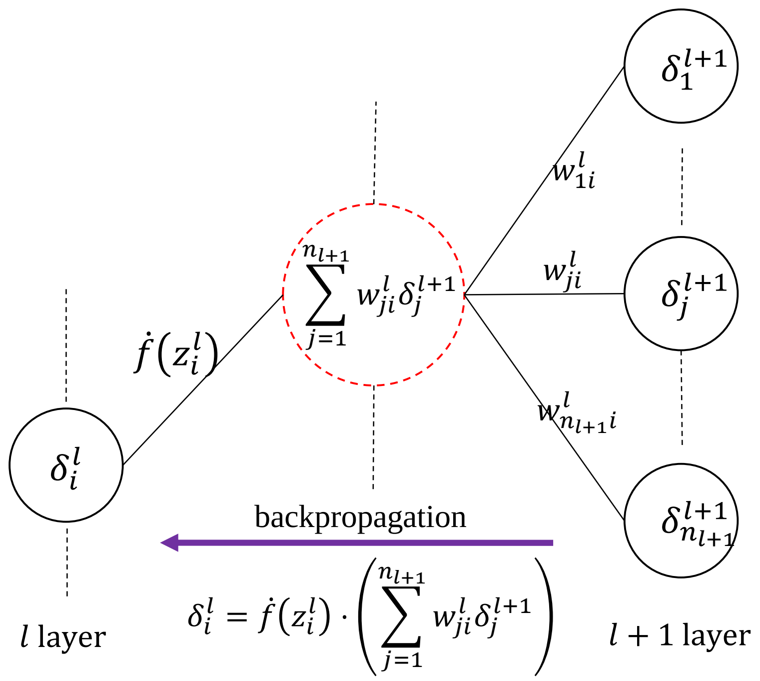
Updating rule: $w_{ji}^l \leftarrow w_{ji}^l - \alpha \cdot \frac{\partial J}{\partial w_{ji}^l}$

Relationship: $\frac{\partial J}{\partial w_{ji}^l} = \delta_j^{l+1} \cdot a_i^l$





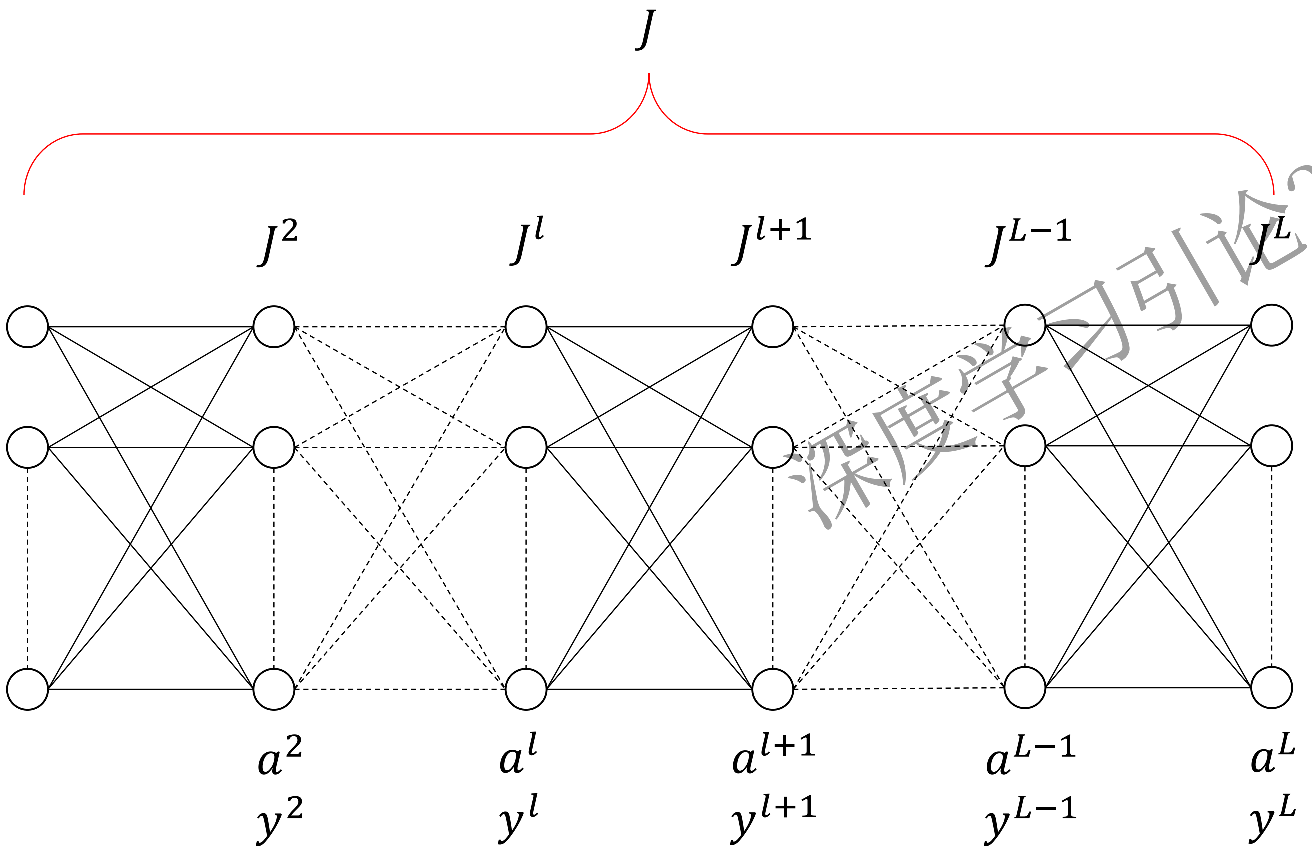




Outline

- A Sequence Recognizing Example
- Review of BP for Mono-target Output NNs
- BP Method for Multi-target Outputs NNs
- BP Algorithm for Multi-target Outputs NNs
- Illustrative Examples
- Assignment

BP Method for Multi-target Outputs NNs



Cost function

$$J^l = \frac{1}{2}\sum_{i=1}^{n_l}\left(a_i^l - y_i^l\right)^2, (l = 2, \cdots, L)$$

$$J = \sum_{l=2}^{L} J^l = \frac{1}{2}\sum_{l=2}^{L}\sum_{i=1}^{n_l}\left(a_i^l - y_i^l\right)^2$$

Network Outputs

$$a^l = \begin{bmatrix} a_1^l \\ \vdots \\ a_{n_l}^l \end{bmatrix}$$

$(l = 2, \cdots, L)$

Multi-target outputs

$$y^l = \begin{bmatrix} y_1^l \\ \vdots \\ y_{n_l}^l \end{bmatrix}$$

$(l = 2, \cdots, L)$

BP Method for Multi-target Outputs NNs

Steepest Descent Method

$$J^l = \frac{1}{2}\sum_{i=1}^{n_l}\left(a_i^l - y_i^l\right)^2, (l = 2, \cdots, L)$$

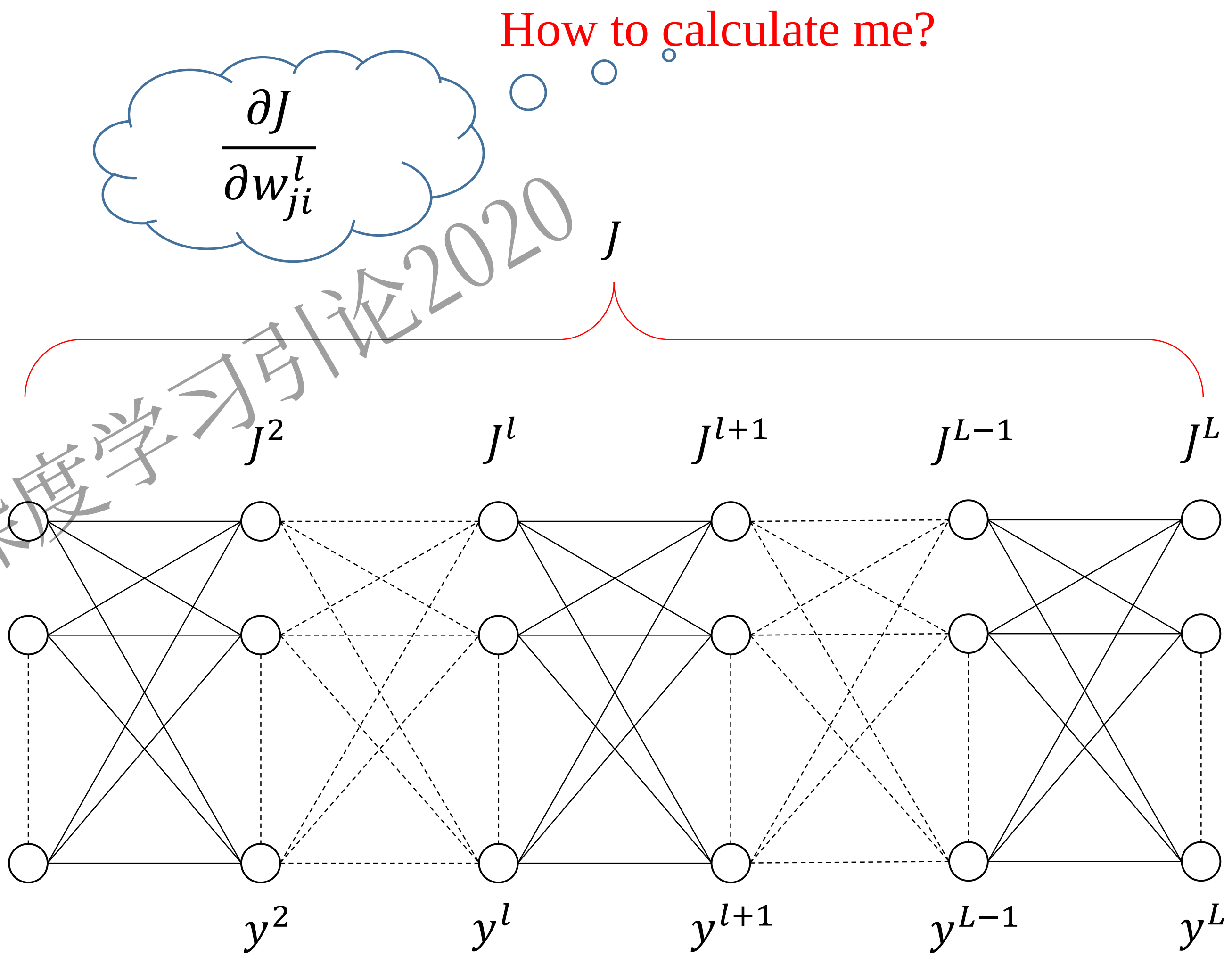
$$J = \sum_{l=2}^{L} J^l = \frac{1}{2}\sum_{l=2}^{L}\sum_{i=1}^{n_l}\left(a_i^l - y_i^l\right)^2$$

1. Computing

$$\frac{\partial J}{\partial w_{ji}^l}$$

2. Iterating

$$w_{ji}^l \leftarrow w_{ji}^l - \alpha \cdot \frac{\partial J}{\partial w_{ji}^l}$$



l layer

$$a_i^l = f\left(z_i^l\right)$$

$$\text{define } \delta_i^l = \frac{\partial J}{\partial z_i^l}$$

Relation between δ_i^l and $\frac{\partial J}{\partial w_{ji}^l}$

$$\frac{\partial J}{\partial w_{ji}^l} = \delta_j^{l+1} \cdot a_i^l$$

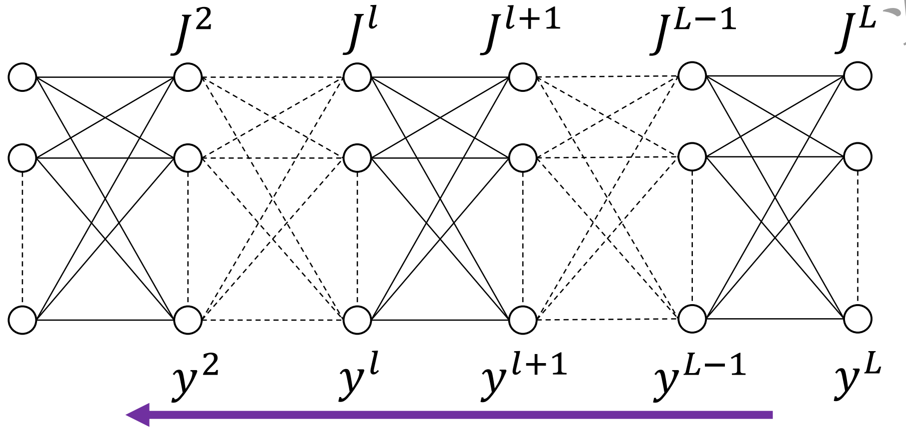
Why?

$$\frac{\partial J}{\partial w_{ji}^l} = \frac{\partial J}{\partial z_j^{l+1}} \cdot \frac{\partial z_j^{l+1}}{\partial w_{ji}^l} = \delta_j^{l+1} \cdot a_i^l$$

$$J^l = \frac{1}{2}\sum_{i=1}^{n_l}\left(a_i^l - y_i^l\right)^2, (l = 2, \cdots, L)$$

$$J = \sum_{l=2}^{L} J^l = \frac{1}{2}\sum_{l=2}^{L}\sum_{i=1}^{n_l}\left(a_i^l - y_i^l\right)^2$$

J



Problem: Can we back propagate δ^l?

l layer

$$a_i^l = f\left(z_i^l\right)$$

$$\delta_i^l = \frac{\partial J}{\partial z_i^l}$$

$$\frac{\partial J}{\partial w_{ji}^l} = \delta_j^{l+1} \cdot a_i^l$$

w_{ji}^l

$l + 1$ layer

$$\delta_j^{l+1} = \frac{\partial J}{\partial z_j^{l+1}}$$

$$z_j^{l+1} = \sum_{i=1}^{n_l} w_{ji}^l a_i^l$$

Step 1: Calculating δ^L in Last Layer

L layer

$$a_i^L = f(z_i^L)$$

$$\delta_i^L = \frac{\partial J}{\partial z_i^L}$$

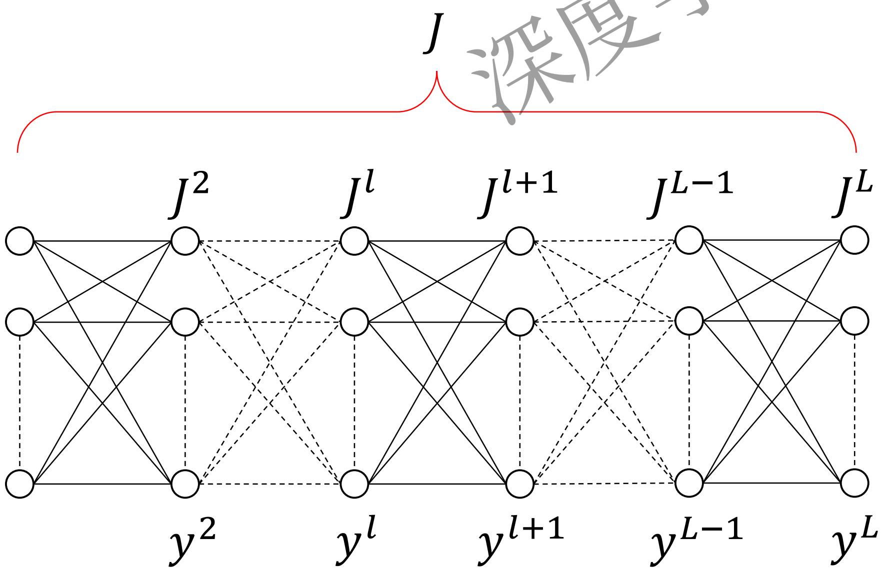
$$J^l = \frac{1}{2}\sum_{i=1}^{n_l}\left(a_i^l - y_i^l\right)^2, (l = 2, \cdots, L)$$

$$J = \sum_{l=2}^{L} J^l = \frac{1}{2}\sum_{l=2}^{L}\sum_{i=1}^{n_l}\left(a_i^l - y_i^l\right)^2$$

It holds that,

$$a_i^L = f(z_i^L)$$

$$\delta_i^L = \frac{\partial J}{\partial z_i^L} = \frac{\partial J^L}{\partial z_i^L} = \frac{1}{2}\cdot\frac{\partial\left(a_i^l - y_i^l\right)^2}{\partial z_i^L} = \left(a_i^L - y_i^L\right)\cdot\frac{\partial a_i^L}{\partial z_i^L} = \left(a_i^L - y_i^L\right)\cdot\dot{f}(z_i^L)$$

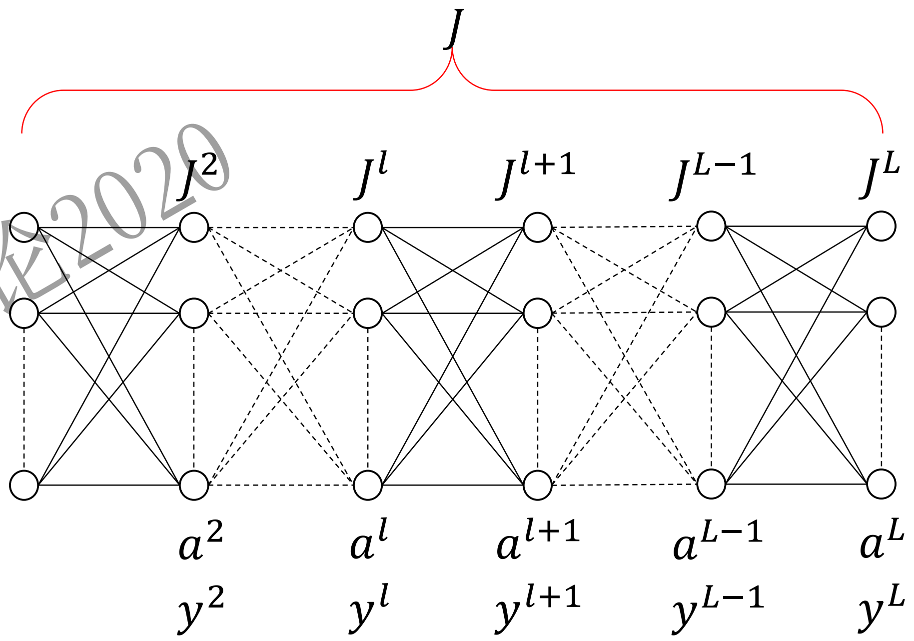


Step 2: Relation Between δ^l and δ^{l+1}

$$J^l = \frac{1}{2}\sum_{i=1}^{n_l}\left(a_i^l - y_i^l\right)^2, (l = 2, \cdots, L)$$

$$J = \sum_{l=2}^{L} J^l = \frac{1}{2}\sum_{l=2}^{L}\sum_{i=1}^{n_l}\left(a_i^l - y_i^l\right)^2$$

J may have an explicit dependence on z_i^l , it may also have an implicit dependence on z_i^l through later output values. To avoid ambiguity in interpreting partial derivatives, define $z_i^l(*) = z_i^l$.



$$\delta_i^l = \frac{\partial J}{\partial z_i^l} = \frac{\partial J}{\partial z_i^l(*)} \cdot \frac{\partial z_i^l(*)}{\partial z_i^l} + \sum_{j=1}^{n_{l+1}} \frac{\partial J}{\partial z_j^{l+1}} \cdot \frac{\partial z_j^{l+1}}{\partial z_i^l}$$

Step 2: Relation Between δ^l and δ^{l+1}

$$J^l = \frac{1}{2}\sum_{i=1}^{n_l}\left(a_i^l - y_i^l\right)^2, (l = 2, \cdots, L)$$

$$J = \sum_{l=2}^{L} J^l = \frac{1}{2}\sum_{l=2}^{L}\sum_{i=1}^{n_l}\left(a_i^l - y_i^l\right)^2$$

J may have an explicit dependence on z_i^l , it may also have an implicit dependence on z_i^l through later output values. To avoid ambiguity in interpreting partial derivatives, define $z_i^l(*) = z_i^l$.

$$\delta_i^l = \frac{\partial J}{\partial z_i^l} = \frac{\partial J}{\partial z_i^l(*)} \cdot \frac{\partial z_i^l(*)}{\partial z_i^l} + \sum_{j=1}^{n_{l+1}} \frac{\partial J}{\partial z_j^{l+1}} \cdot \frac{\partial z_j^{l+1}}{\partial z_i^l}$$

An Illustrate Example

$$J = x + y, y = \exp(x)$$

$$\frac{\partial J}{\partial x} = \frac{\partial J}{\partial x} + \frac{\partial J}{\partial y} \cdot \frac{\partial y}{\partial x}$$

$$x^* = x$$

$$\frac{\partial J}{\partial x} = \frac{\partial J}{\partial x^*} \cdot \frac{\partial x^*}{\partial x} + \frac{\partial J}{\partial y} \cdot \frac{\partial y}{\partial x}$$

Step 2: Relation Between δ^l and δ^{l+1}

$$\delta_i^l = \frac{\partial J}{\partial z_i^l} = \frac{\partial J}{\partial z_i^l(*)} \cdot \frac{\partial z_i^l(*)}{\partial z_i^l} + \sum_{j=1}^{n_{l+1}} \frac{\partial J}{\partial z_j^{l+1}} \cdot \frac{\partial z_j^{l+1}}{\partial z_i^l}$$

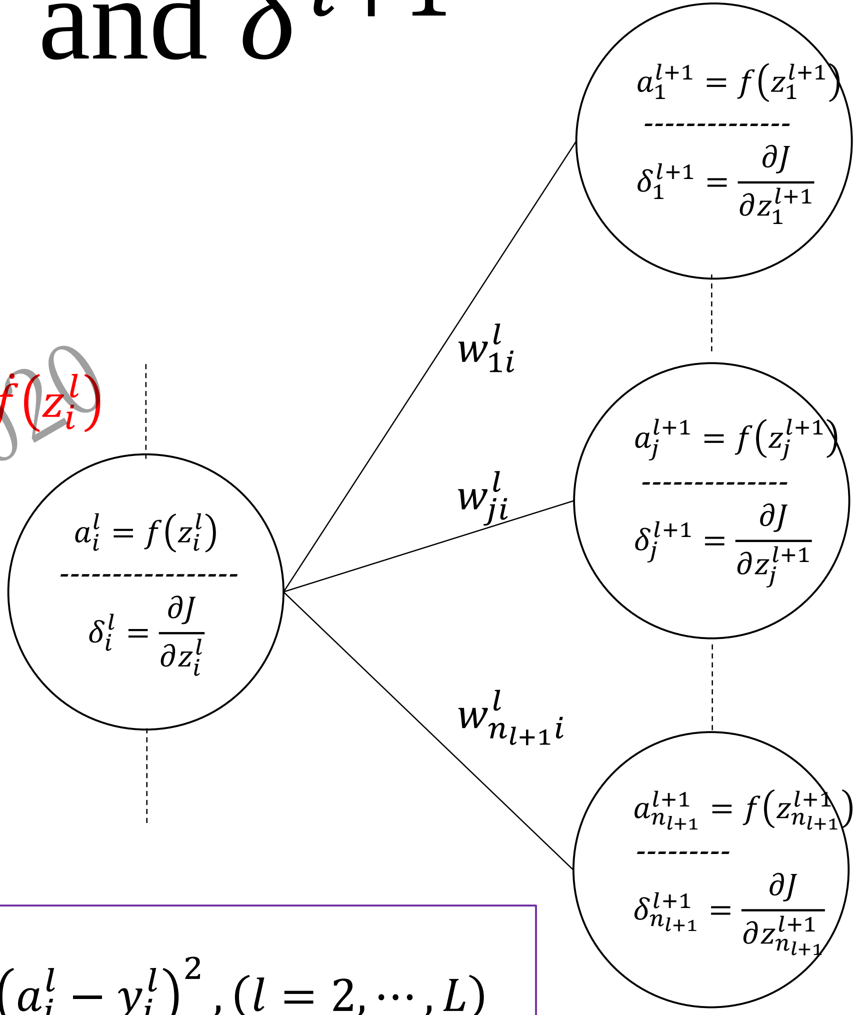
$$\frac{\partial J}{\partial z_i^l(*)} \cdot \frac{\partial z_i^l(*)}{\partial z_i^l} = \frac{\partial J^l}{\partial z_i^l} = \frac{1}{2} \cdot \frac{\partial \left(a_i^l - y_i^l\right)^2}{\partial z_i^l} = \left(a_i^l - y_i^l\right) \cdot \frac{\partial a_i^l}{\partial z_i^l} = \left(a_i^l - y_i^l\right) \cdot \dot{f}\left(z_i^l\right)$$

$$\sum_{j=1}^{n_{l+1}} \frac{\partial J}{\partial z_j^{l+1}} \cdot \frac{\partial z_j^{l+1}}{\partial z_i^l} = \sum_{j=1}^{n_{l+1}} \delta_j^{l+1} \cdot \frac{\partial z_j^{l+1}}{\partial z_i^l} = \dot{f}\left(z_i^l\right) \cdot \left(\sum_{j=1}^{n_{l+1}} \delta_j^{l+1} \cdot w_{ji}^l\right)$$

$$z_j^{l+1} = \sum_{i=1}^{n_l} w_{ji}^l a_j^l$$

$$a_i^{l+1} = f\left(z_i^{l+1}\right)$$

$$\frac{\partial z_j^{l+1}}{\partial z_i^l} = w_{ji}^l \cdot \frac{\partial a_j^l}{\partial z_i^l} = w_{ji}^l \cdot \dot{f}\left(z_i^l\right)$$



$$J^l = \frac{1}{2} \sum_{i=1}^{n_l} \left(a_i^l - y_i^l\right)^2, (l = 2, \cdots, L)$$

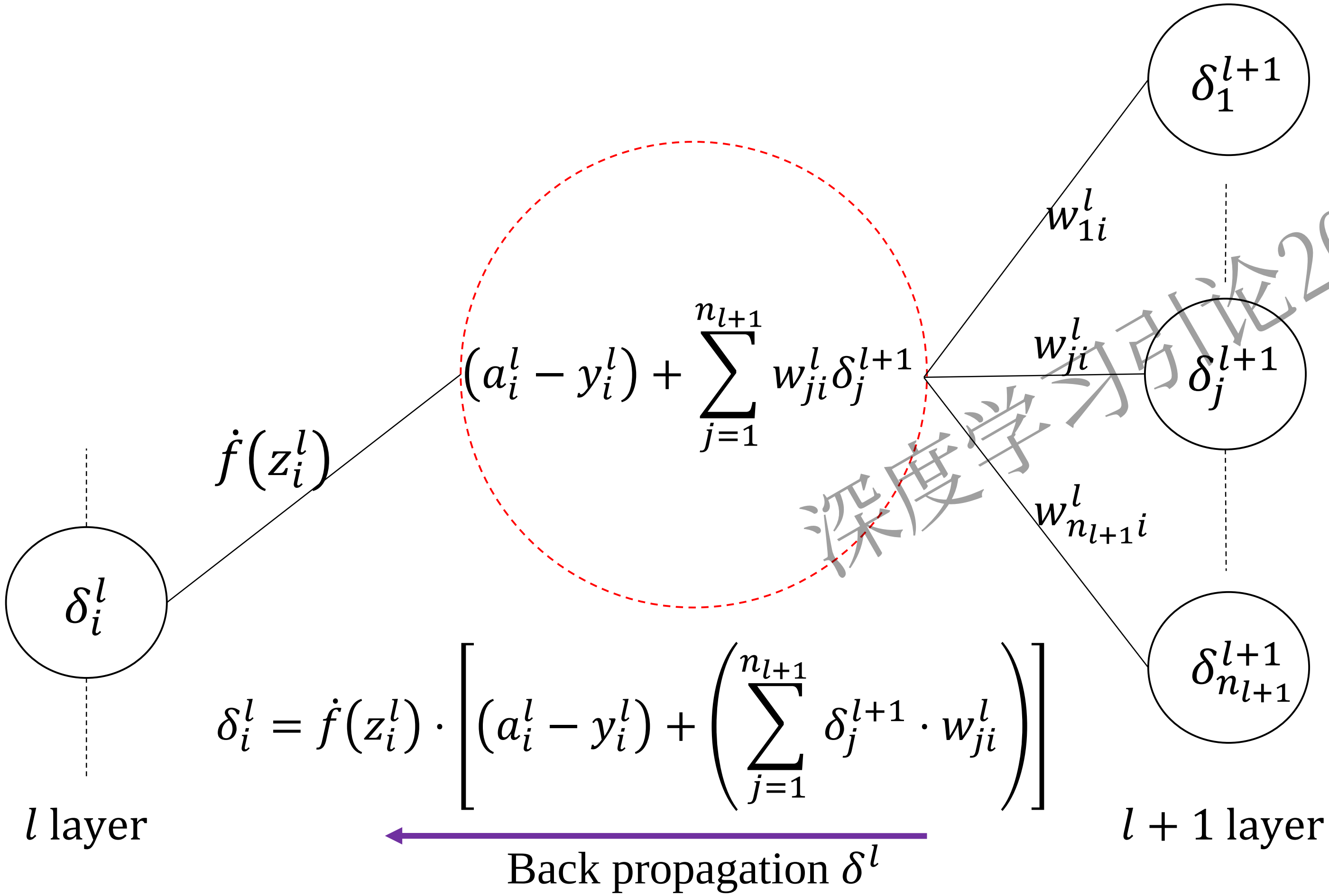
$$J = \sum_{l=2}^{L} J^l = \frac{1}{2} \sum_{l=2}^{L} \sum_{i=1}^{n_l} \left(a_i^l - y_i^l\right)^2$$

$$\delta_i^l = \dot{f}\left(z_i^l\right) \cdot \left[\left(a_i^l - y_i^l\right) + \left(\sum_{j=1}^{n_{l+1}} \delta_j^{l+1} \cdot w_{ji}^l\right)\right]$$

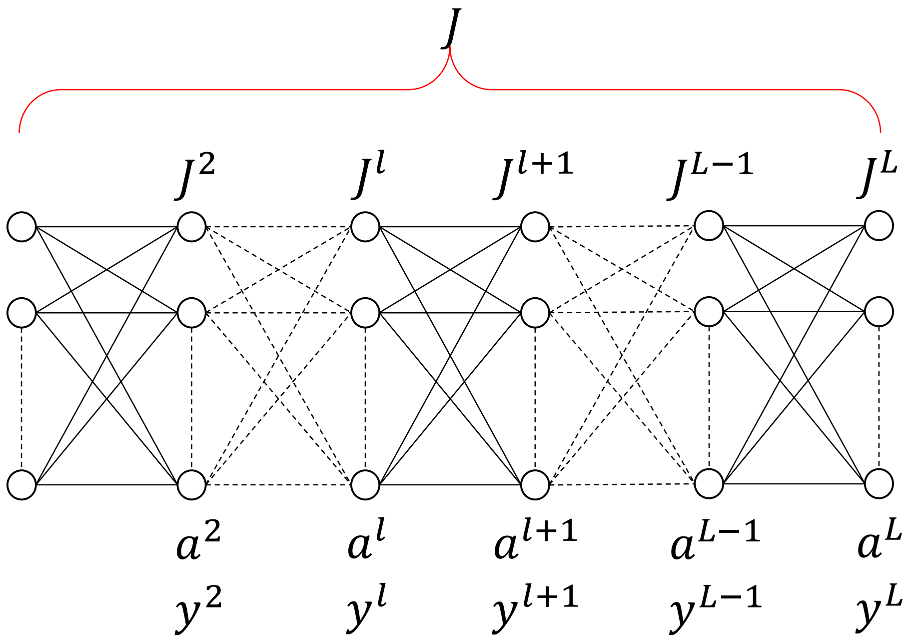
Step 3: Backpropagation δ^l

$$J^l = \frac{1}{2}\sum_{i=1}^{n_l}\left(a_i^l - y_i^l\right)^2, (l = 2, \cdots, L)$$

$$J = \sum_{l=2}^{L} J^l = \frac{1}{2}\sum_{l=2}^{L}\sum_{i=1}^{n_l}\left(a_i^l - y_i^l\right)^2$$



$$\delta_i^l = \dot{f}(z_i^l) \cdot \left[\left(a_i^l - y_i^l\right) + \left(\sum_{j=1}^{n_{l+1}} \delta_j^{l+1} \cdot w_{ji}^l\right)\right]$$

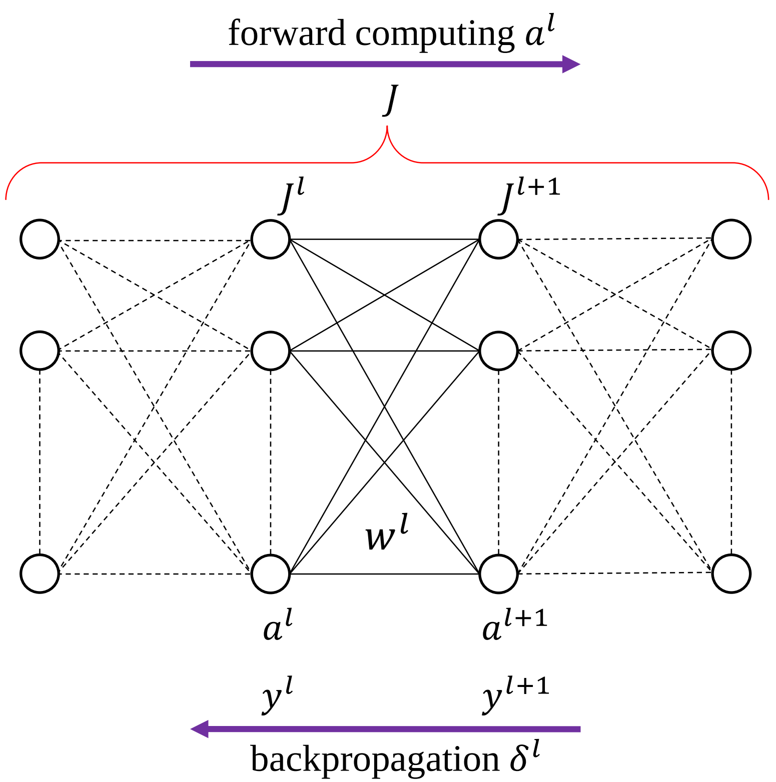


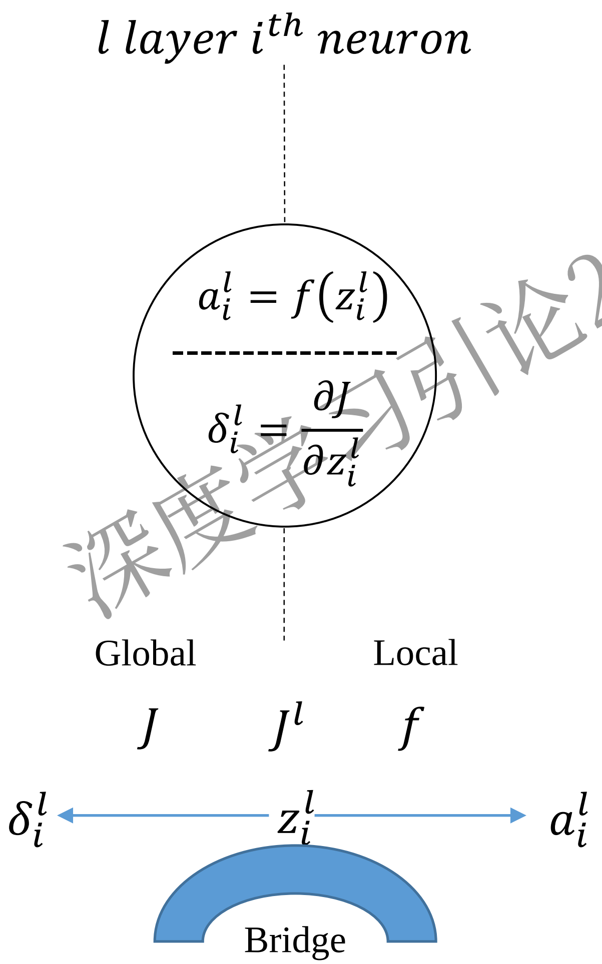
Only One Page to Understand BP

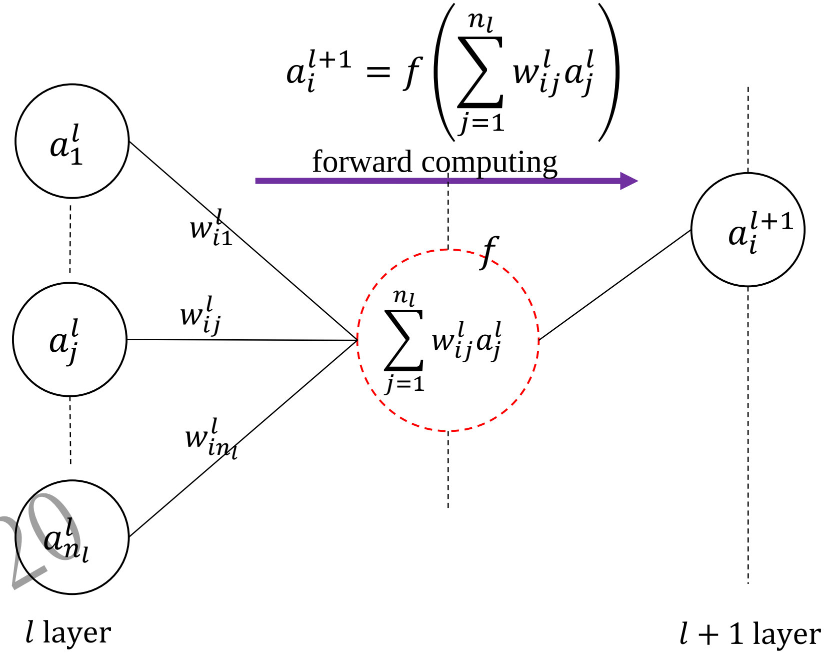
Cost function: $J(w^1, \cdots, w^{L-1})$

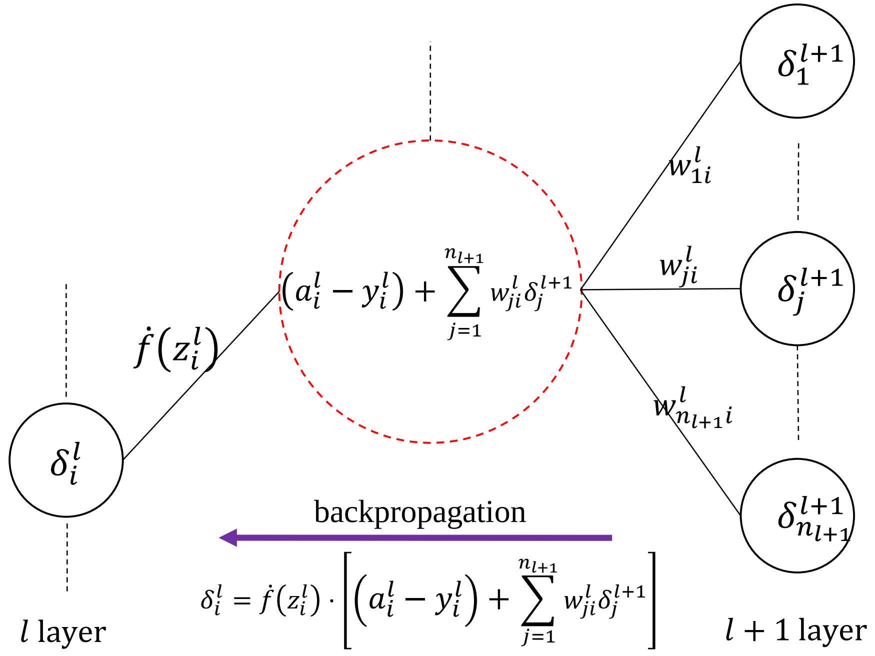
Updating rule: $w_{ji}^l \leftarrow w_{ji}^l - \alpha \cdot \frac{\partial J}{\partial w_{ji}^l}$

Relationship: $\frac{\partial J}{\partial w_{ji}^l} = \delta_j^{l+1} \cdot a_i^l$





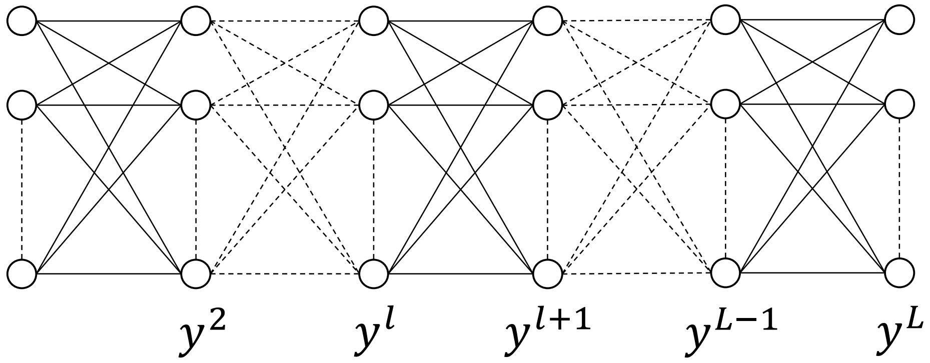




Outline

- A Sequence Recognizing Example
- Review of BP for Mono-target Output NNs
- BP Method for Multi-target Outputs NNs
- BP Algorithm for Multi-target Outputs NNs
- Illustrative Examples
- Assignment

The BP Algorithm



Step 1. Input the training data set $D = \{(x, y)\}$

Step 2. Initial each w_{ij}^l, and choose a learning rate α.

Step 3. For each mini-batch sample $D_m \subseteq D$

for each $x \in D_m$

$a^1 \leftarrow x$;

for $l = 2{:}\,L$

$a^l \leftarrow fc(w^l, a^l)$;

end

$\delta^L = \frac{\partial J}{\partial z^L}$;

for $l = L - 1{:}\,2$

$\delta^l \leftarrow bc(w^l, \delta^{l+1})$;

end

$\frac{\partial J}{\partial w_{ji}^l} \leftarrow \frac{\partial J}{\partial w_{ji}^l} + \delta_j^{l+1} \cdot a_i^l$;

end

Step 4. Updating

$$w_{ji}^l \leftarrow w_{ji}^l - \alpha \cdot \frac{\partial J}{\partial w_{ji}^l};$$

Step 5. Return to Step 3 until each w^lconverge.

function $fc(w^l, a^l)$

$for\ i = 1{:}\,n_{l+1}$

$$z_i^{l+1} = \sum_{j=1}^{n_l} w_{ij}^l a_j^l$$

$$a_i^{l+1} = f(z_i^{l+1})$$

end

Relationship:

$$\frac{\partial J}{\partial w_{ji}^l} = \delta_j^{l+1} \cdot a_i^l$$

function $bc(w^l, \delta^{l+1})$

$for\ i = 1{:}\,n_l$

$$\delta_i^l = \dot{f}(z_i^l) \cdot \left[(a_i^l - y_i^l) + \left(\sum_{j=1}^{n_{l+1}} \delta_j^{l+1} \cdot w_{ji}^l \right) \right]$$

end

Outline

- A Sequence Recognizing Example
- Review of BP for Mono-target Output NNs
- BP Method for Multi-target Outputs NNs
- BP Algorithm for Multi-target Outputs NNs
- Illustrative Examples
- Assignment

Illustrative Example: Sequence Recognition

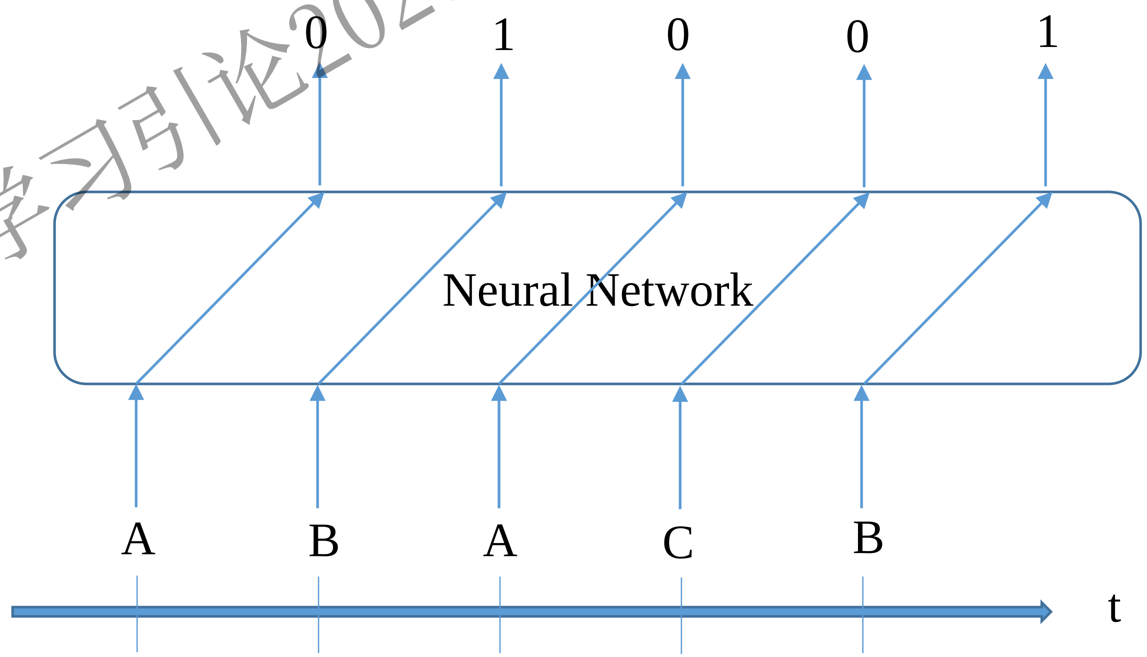
Recognize A followed by B Problem

The task is to recognize A followed by B.

Generated Sequences

1. ABACB
2. CCBBA
3. CACCB
4. ACCCB
5. CACBC
6. AAACB
7. BAACB
8. CCBAB
9. BCCAB
10. CABAC

……………......



Illustrative Example: Sequence Recognition

Coding the Inputs：

$$A = \begin{bmatrix}1\\0\\0\end{bmatrix} \quad B = \begin{bmatrix}0\\1\\0\end{bmatrix} \quad C = \begin{bmatrix}0\\0\\1\end{bmatrix}$$

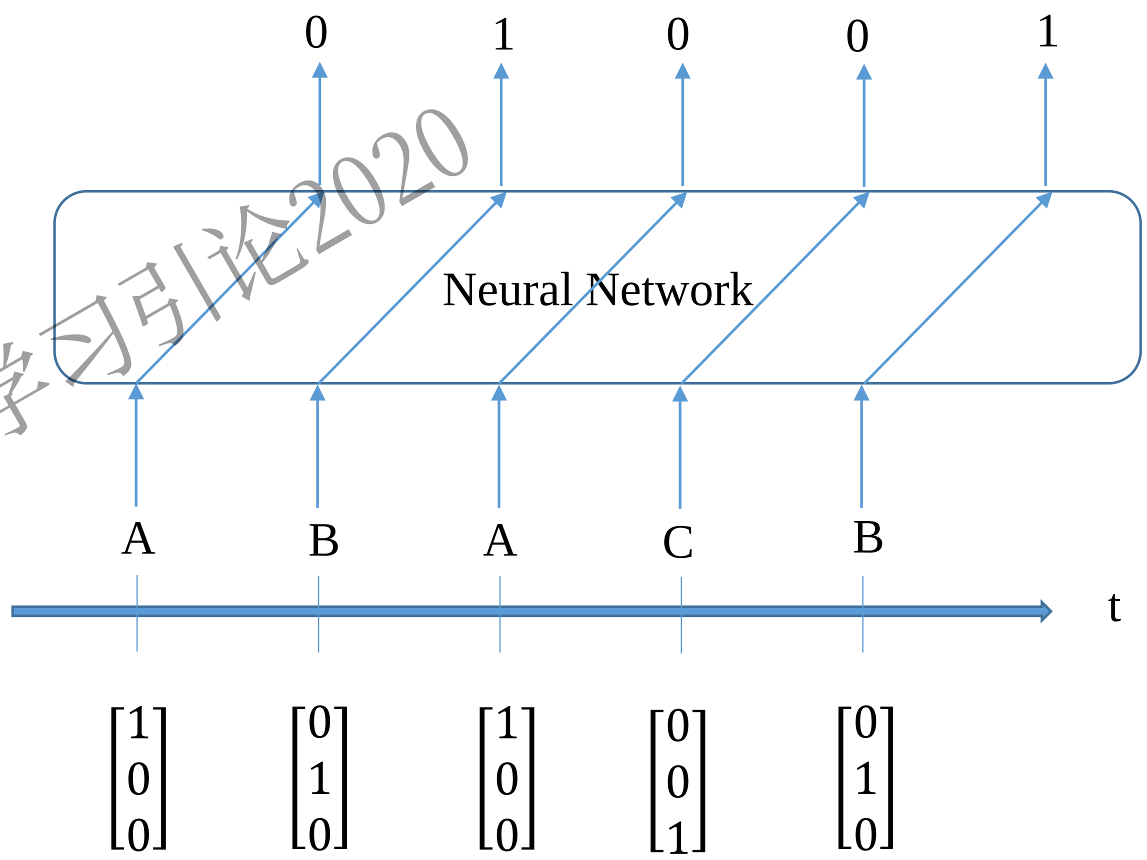
Generated Sequences

1. A B C A B

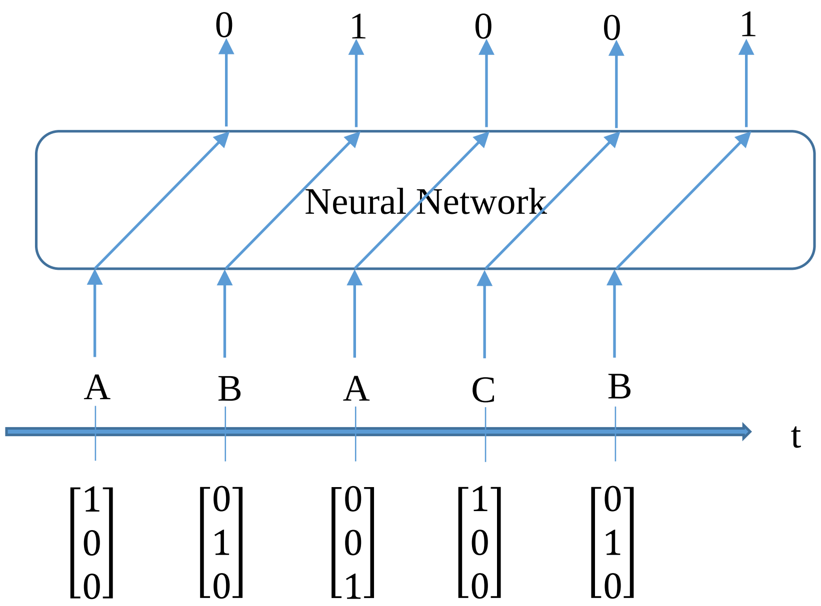
$$\begin{bmatrix}1\\0\\0\end{bmatrix} \begin{bmatrix}0\\1\\0\end{bmatrix} \begin{bmatrix}0\\0\\1\end{bmatrix} \begin{bmatrix}1\\0\\0\end{bmatrix} \begin{bmatrix}0\\1\\0\end{bmatrix}$$

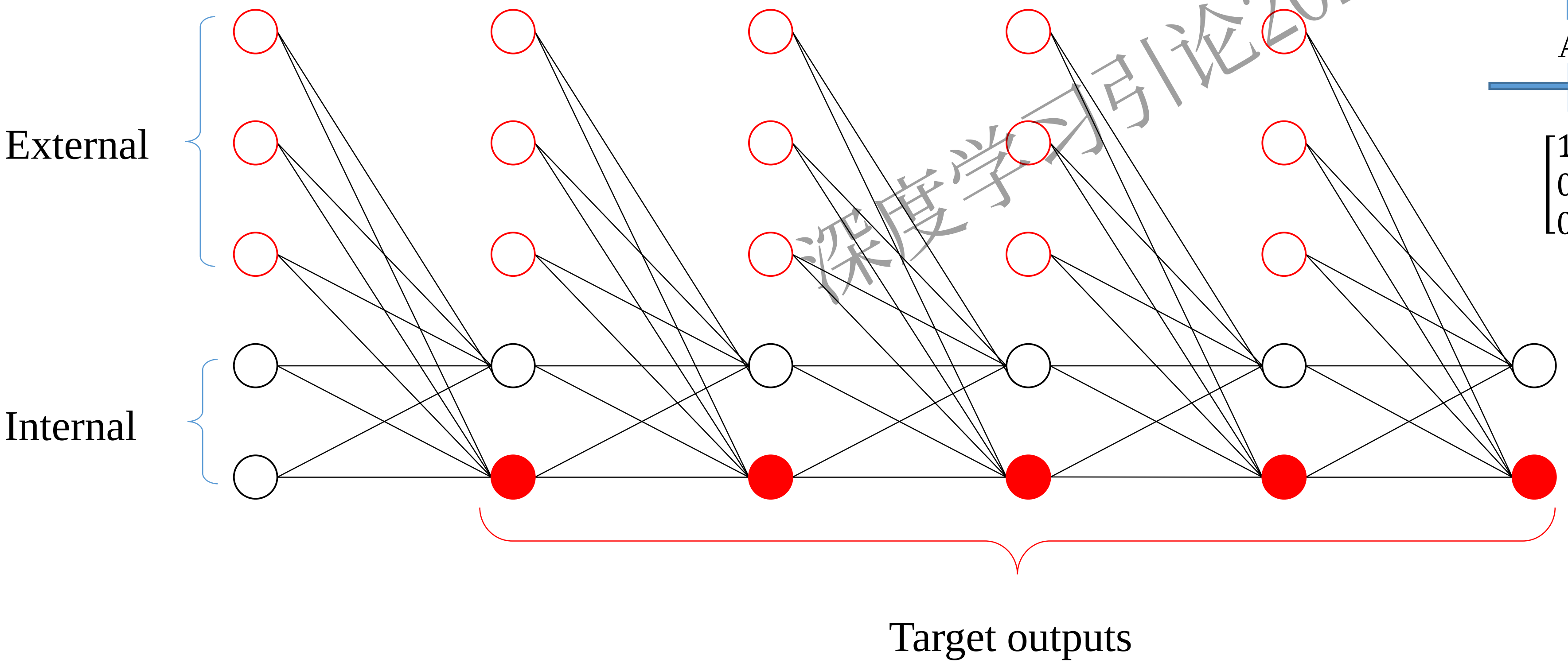
2. C A C C B

$$\begin{bmatrix}0\\0\\1\end{bmatrix} \begin{bmatrix}1\\0\\0\end{bmatrix} \begin{bmatrix}0\\0\\1\end{bmatrix} \begin{bmatrix}0\\0\\1\end{bmatrix} \begin{bmatrix}0\\1\\0\end{bmatrix}$$

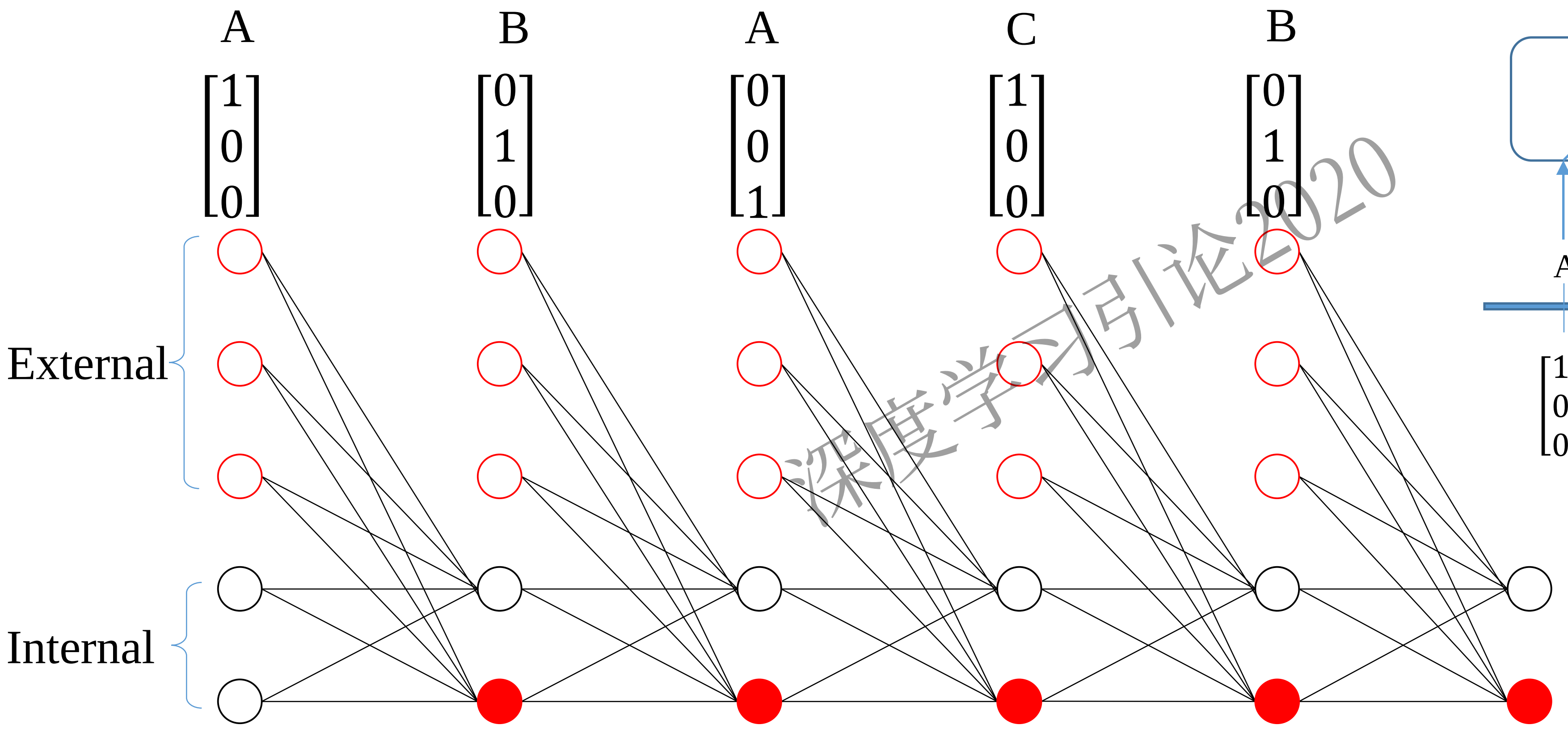


Illustrative Example: Sequence Recognition

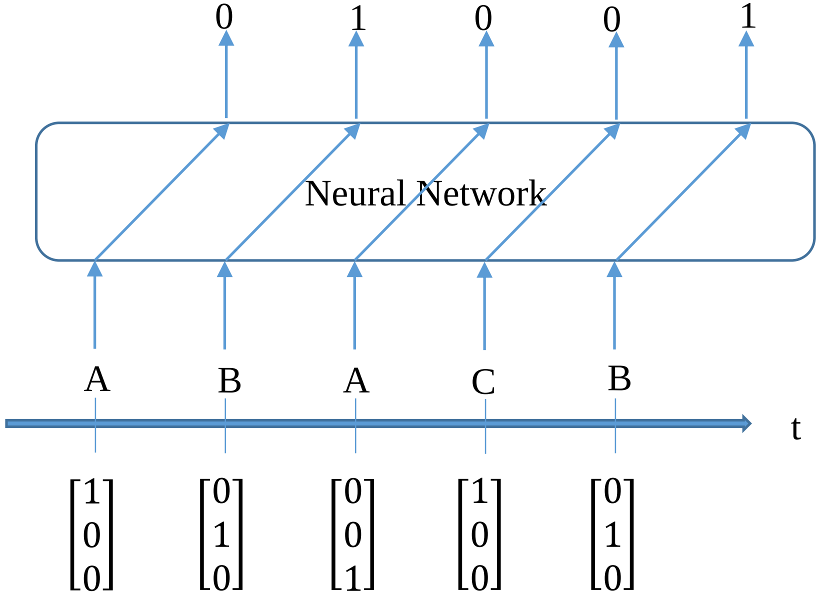




Illustrative Example: Sequence Recognition



$$y^2 = \begin{bmatrix}0\\0\end{bmatrix} \quad y^3 = \begin{bmatrix}0\\1\end{bmatrix} \quad y^4 = \begin{bmatrix}0\\0\end{bmatrix} \quad y^5 = \begin{bmatrix}0\\1\end{bmatrix} \quad y^6 = \begin{bmatrix}0\\1\end{bmatrix}$$

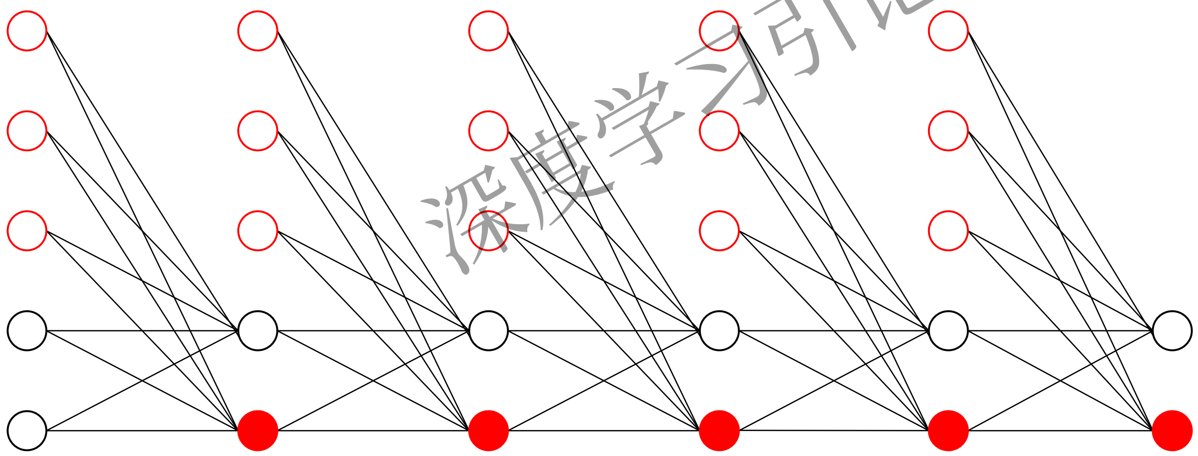


Illustrative Example: Sequence Recognition

Generated Training Sequences

1. ABCAB
2. CCBBA
3. CACCB
4. ACCCB
5. CACBC
6. AAACB
7. BAACB
8. CCBAB
9. BCCAB
10. CABAC

…………......



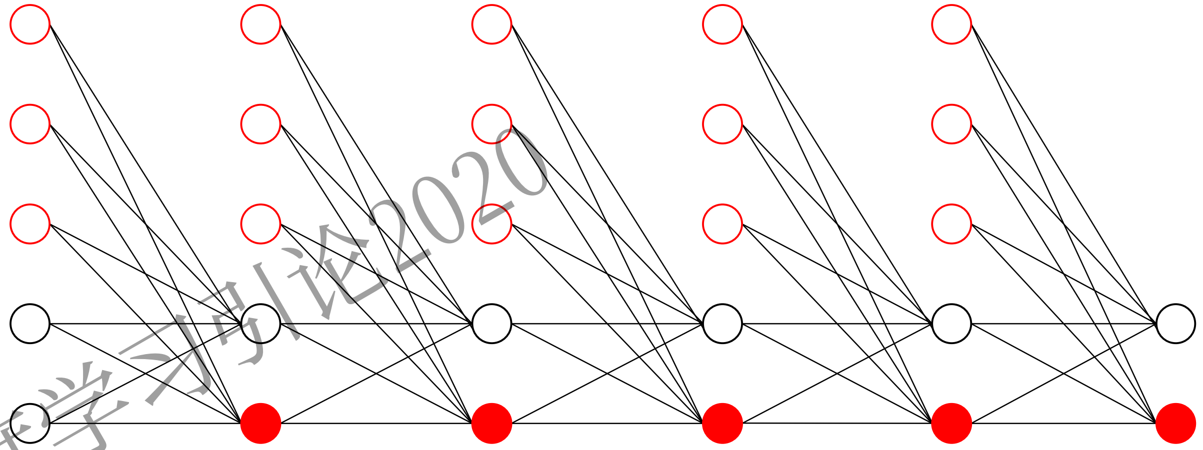
Generated Testing Sequences

1. CBCAC
2. ACBBA
3. BACCB
4. ACBCB
5. AACBC
6. BAACB
7. AAACB
8. CCBAB
9. BBCAB
10. AABAC

…………......

Illustrative Example: Sequence Recognition

$$A = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad C = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$



Example outputs:

CAACB	→	0.0184	0.0001	0.0211	0.0801	0.9928
ABBCA	→	0.0179	0.9375	0.0267	0.0012	0.0000
AACBA	→	0.0179	0.0336	0.0286	0.8722	0.0000
CACBB	→	0.0184	0.0001	0.0170	0.8494	0.0013
BCAAA	→	0.0182	0.0001	0.0001	0.0622	0.0018

Illustrative Example: Sequence Recognition

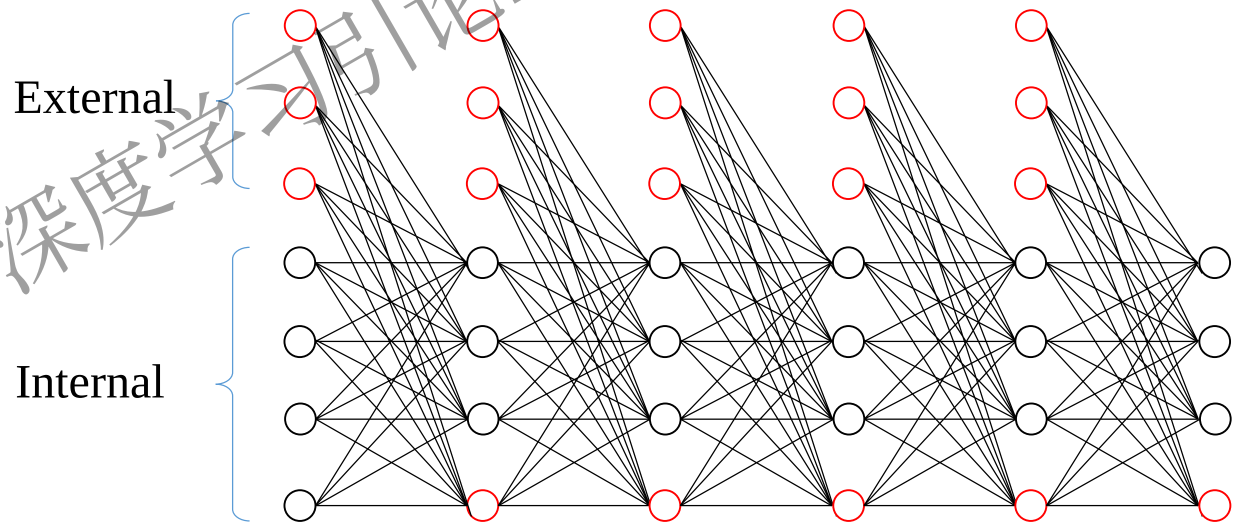
Recognize A followed by B Problem

The task is to recognize A followed by B.

Generated Sequences

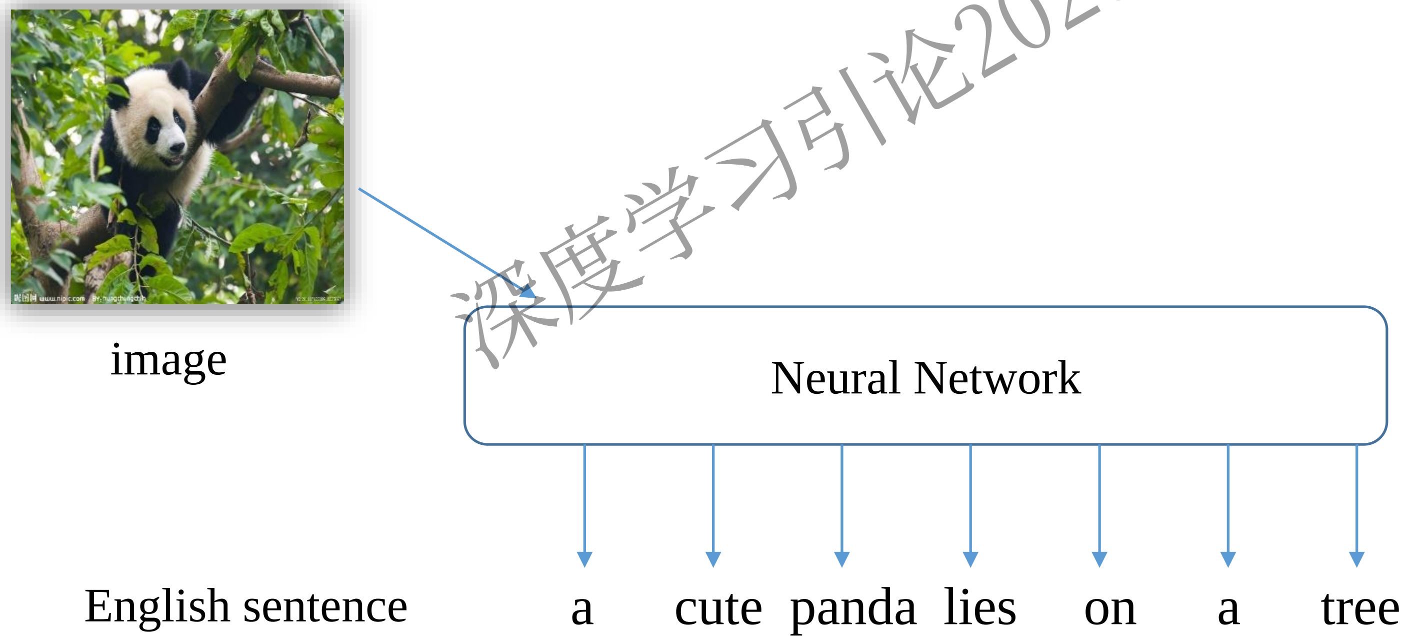
1. ABCAB
2. CCBBA
3. CACCB
4. ACCCB
5. CACBC
6. AAACB
7. BAACB
8. CCBAB
9. BCCAB
10. CABAC

……………......



Illustrative Example: Image Caption

Image Caption:
The task is to describe the content of an image using properly formed English sentence.



Illustrative Example: Image Caption

Dataset：COCO

COCO is a new image recognition, segmentation, and captioning dataset sponsored by Microsoft.
http://mscoco.org/dataset/#download
There are：

- 80,000 training samples
- 40,000 validation samples
- 40,000 test samples

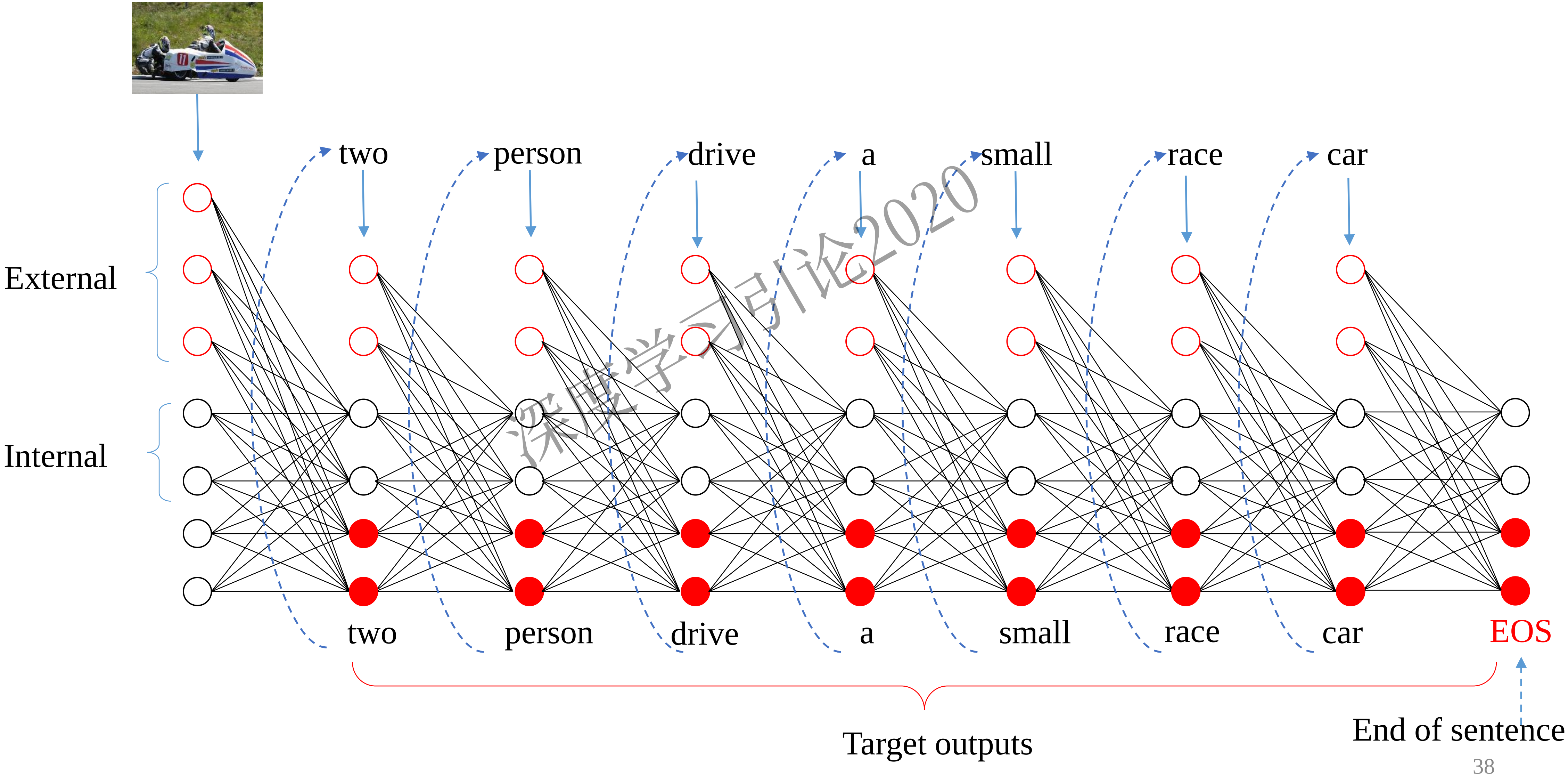
Microsoft

Dataset

$$D = \{(x, \underbrace{s_1, s_2, s_3, s_4, s_5}))\}$$

1.Two person drive a small race car .
2.Two racer drive a white bike down a road .
3.Two motorist be ride along on their vehicle that be oddly design and color .
4.Two person be in a small race car drive by a green hill .
5.Two person in race uniform in a street car .

Illustrative Example: Image Caption



Illustrative Example: Image Caption

Coding the Inputs：

1. building the vocabulary

End of sentence

a	an	two	car	EOS	⋯

$$\begin{bmatrix}1\\0\\0\\0\\0\\\vdots\\0\\0\\0\\0\end{bmatrix}_{23000}\quad \begin{bmatrix}0\\1\\0\\0\\0\\\vdots\\0\\0\\0\\0\end{bmatrix}_{23000}\quad \begin{bmatrix}0\\0\\1\\0\\0\\\vdots\\0\\0\\0\\0\end{bmatrix}_{23000}\quad \begin{bmatrix}0\\0\\0\\1\\0\\\vdots\\0\\0\\0\\0\end{bmatrix}_{23000}\quad \begin{bmatrix}0\\0\\0\\0\\1\\\vdots\\0\\0\\0\\0\end{bmatrix}_{23000}\quad \cdots$$

One-hot word vector

2. coding the sentence

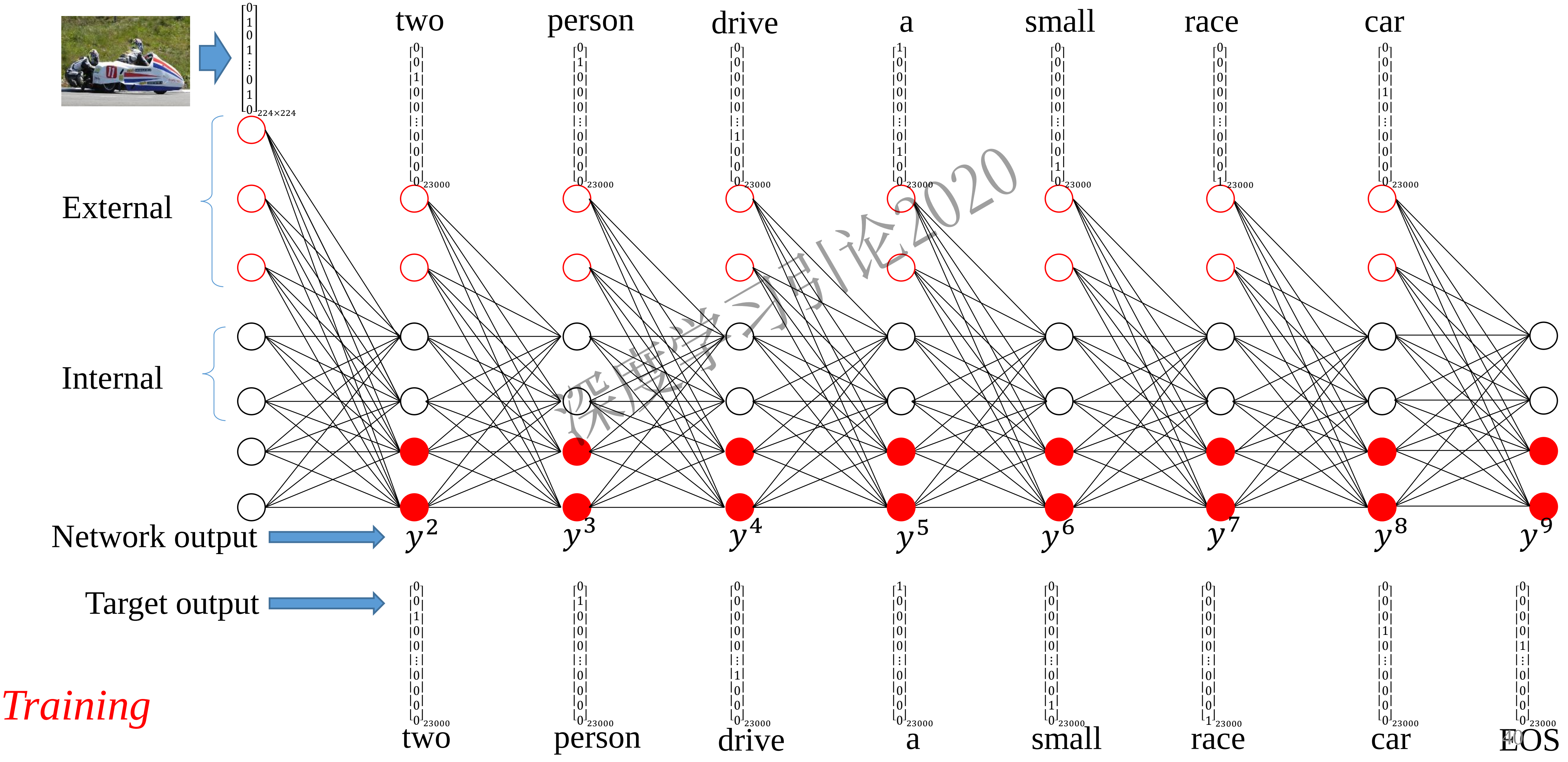
$s =$ two person drive a small race car EOS

$$\begin{bmatrix}0\\0\\1\\0\\0\\\vdots\\0\\0\\0\\0\end{bmatrix}_{23000}\quad \begin{bmatrix}0\\1\\0\\0\\0\\\vdots\\0\\0\\0\\0\end{bmatrix}_{23000}\quad \begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\1\\0\\0\\0\end{bmatrix}_{23000}\quad \begin{bmatrix}1\\0\\0\\0\\0\\\vdots\\0\\0\\0\\0\end{bmatrix}_{23000}\quad \begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\0\\0\\1\\0\end{bmatrix}_{23000}\quad \begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\0\\0\\0\\1\end{bmatrix}_{23000}\quad \begin{bmatrix}0\\0\\0\\1\\0\\\vdots\\0\\0\\0\\0\end{bmatrix}_{23000}\quad \begin{bmatrix}0\\0\\0\\0\\1\\\vdots\\0\\0\\0\\0\end{bmatrix}_{23000}$$

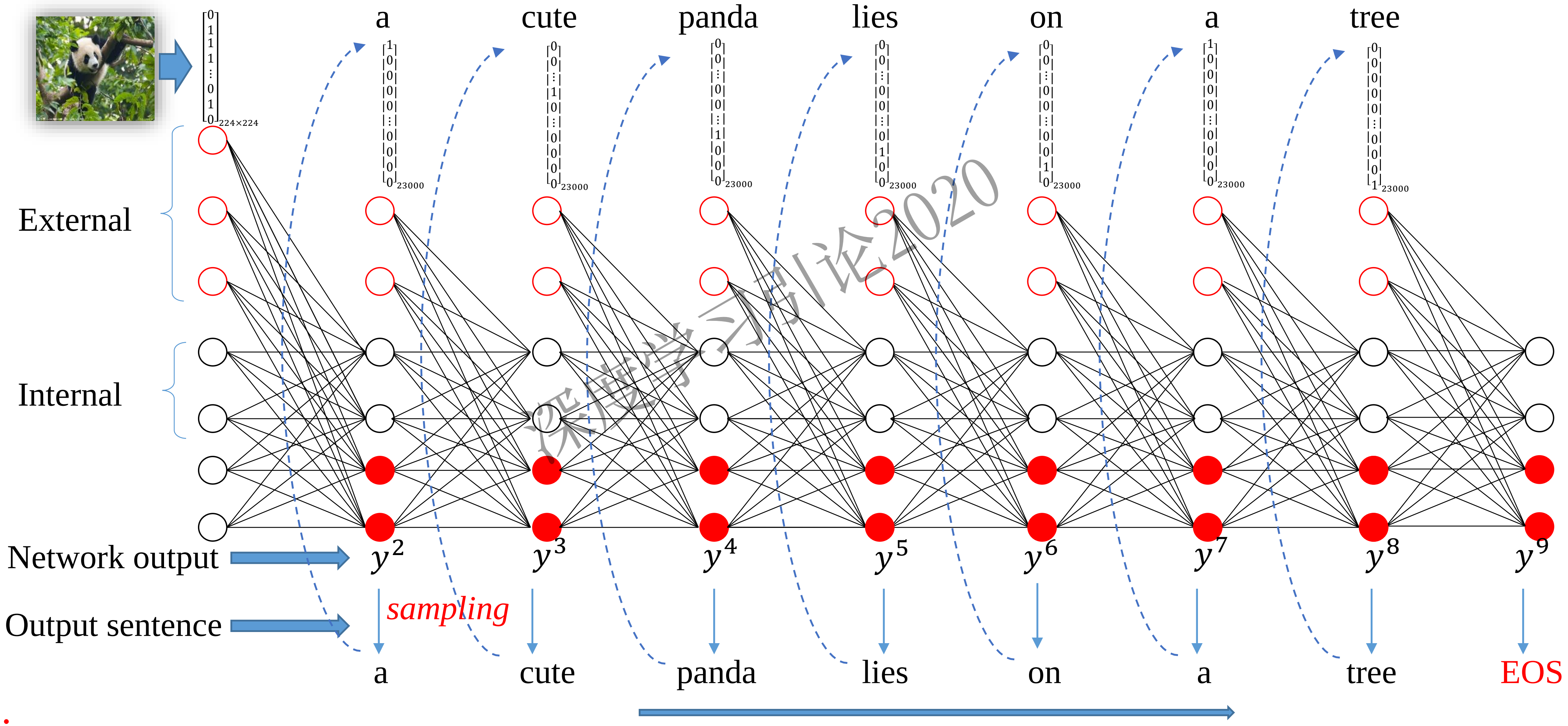
3. digitizing the image

$$\begin{bmatrix}0\\1\\0\\1\\\vdots\\0\\1\\0\end{bmatrix}_{224\times 224}$$

Illustrative Example: Image Caption



Illustrative Example: Image Caption



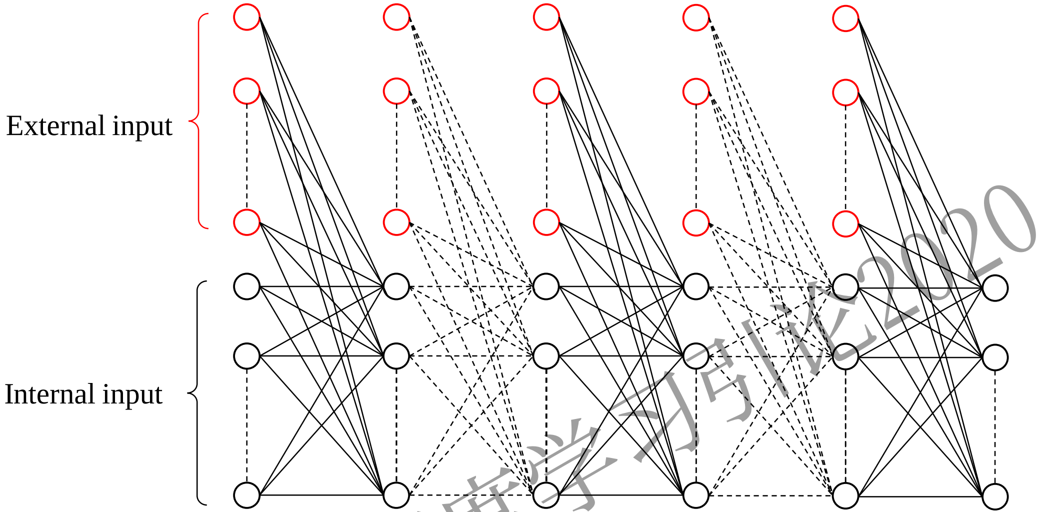
Testing

Generate the word one by one, until get the word 'EOS'

Illustrative Example: Poem Creating

陆游
卜算子・咏梅

驿外断桥边，
寂寞开无主。
已是黄昏独自愁，
更著风和雨。
无意苦争春，
一任群芳妒。
零落成泥碾作尘，
只有香如故。

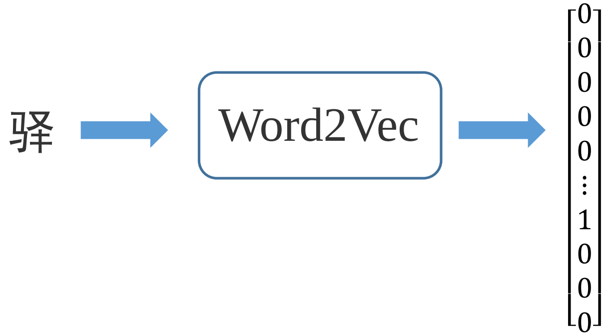


Can artificial neural networks create poem?

毛泽东
卜算子・咏梅

风雨送春归，
飞雪迎春到。
已是悬崖百丈冰，
犹有花枝俏。
俏也不争春，
只把春来报。
待到山花烂漫时，
她在丛中笑。

Illustrative Example: Poem Creating

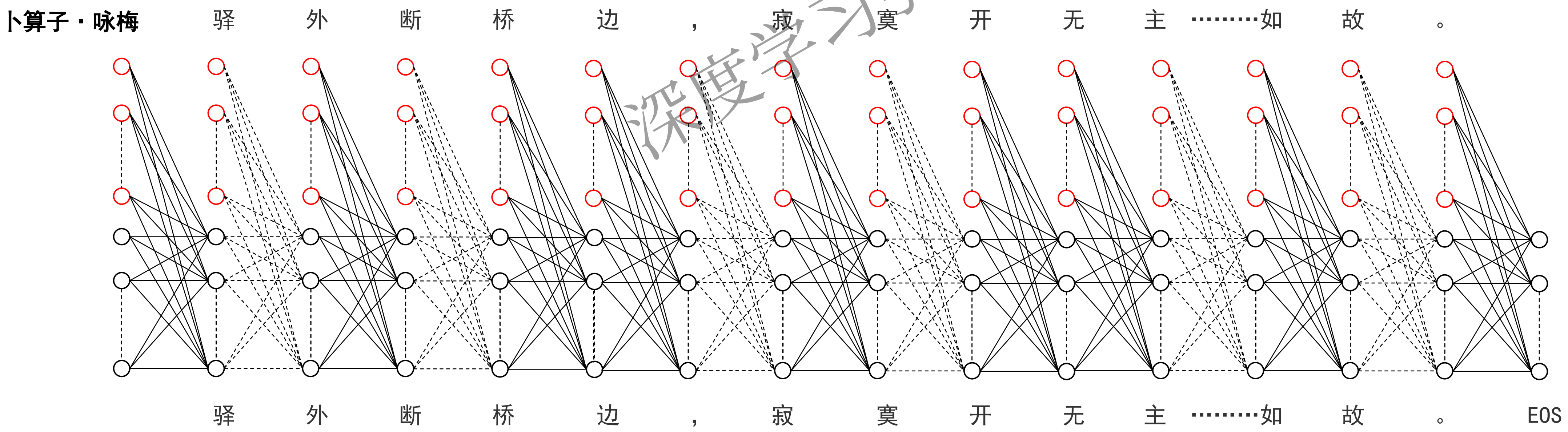


卜算子・咏梅

驿外断桥边，
寂寞开无主。
已是黄昏独自愁，
更著风和雨。
无意苦争春，
一任群芳妒。
零落成泥碾作尘，
只有香如故。

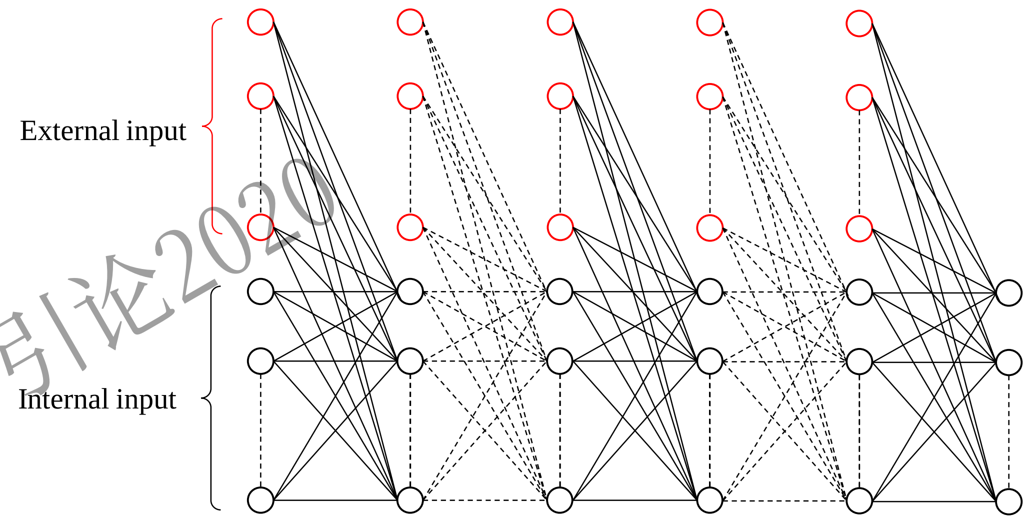
卜算子・咏梅

风雨送春归，
飞雪迎春到。
已是悬崖百丈冰，
犹有花枝俏。
俏也不争春，
只把春来报。
待到山花烂漫时，
她在丛中笑。



Illustrative Example: Poem Creating

Machineilab



卜算子・咏梅

花谢早疏篱，
几度陶潜里。
永日梅花昔底寒，
比向梅花妒。
荣悴幻非凡，
谓是娇芳伴。
后著金陵几日时，
中酒争先理。

卜算子・咏梅

朱阁见幽芳，
露叶梅花里。
玉屑琼台地屑琼，
瑞鹊惊飞尾。
便倚彩毫归，
枝上簪盐谱。
别作千秋一笑随，
好趁伊家笑。

卜算子・咏梅

小试买梅花，
并蒂栖香粉。
肯向红蕖似竹姿，
一叶清风许。
心思寺炉高，
深院松间曲。
万纸参差故与黎，
效我何知道。

Outline

- A Sequence Recognizing Example
- Review of BP for Single Target Output NNs
- BP Method for Multiple Target Outputs NNs
- BP Algorithm for Multiple Target Outputs NNs
- Illustrative Examples
- Assignment

Assignment

Assignment:
Design a multi-target outputs neural network to learn to complete sequence.
The first two items of a sequence uniquely determine the remaining four.

Training Dataset

AA1212	AC1231	AD1221	AE1213
BA2312	BB2323	BC2331	BE2313
CB3123	CC3131	CD3121	CE3113
DA2112	DB2123	DC2131	DD2121
EA1312	EB1323	ED1321	EE1313

Testing Dataset

AB1223	BD2321	CA3112	DE2113	EC1331

The End

A Sequence Recognizing Example