Recurrent Neural Network

VietAI teaching team

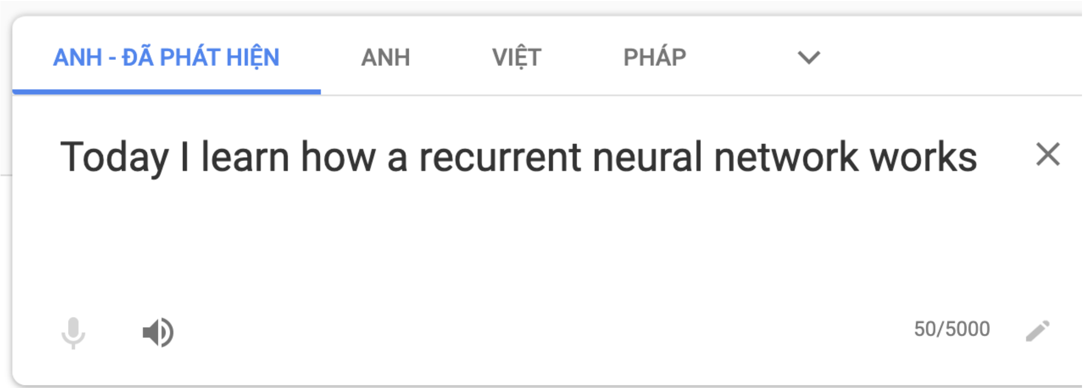
1 Giới thiệu bài toán

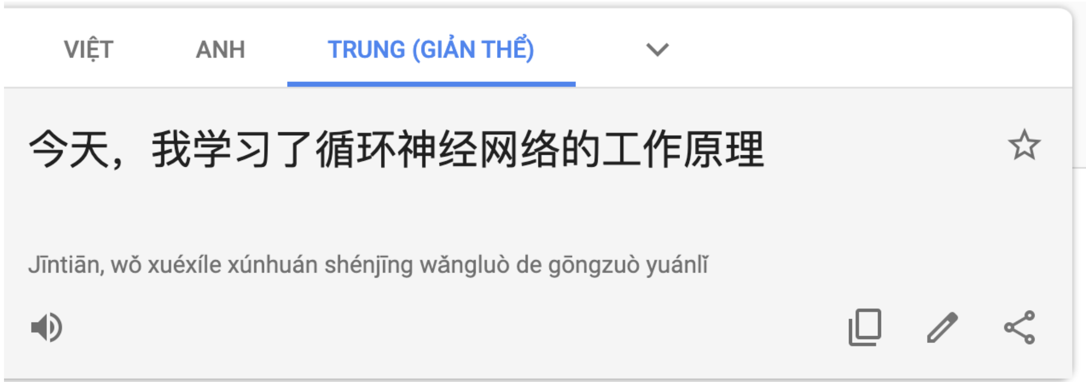
This is good!

Not really bad

Awful

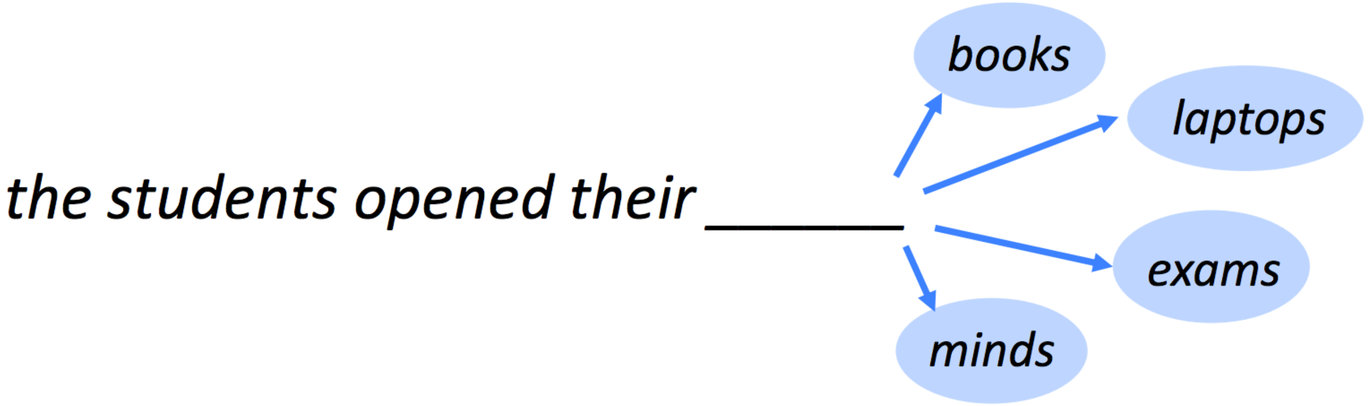
1 Giới thiệu bài toán





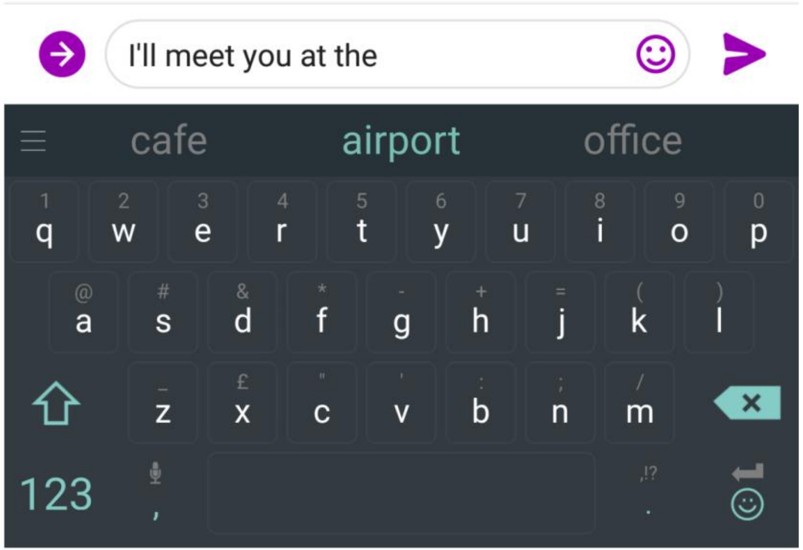
1 Giới thiệu bài toán

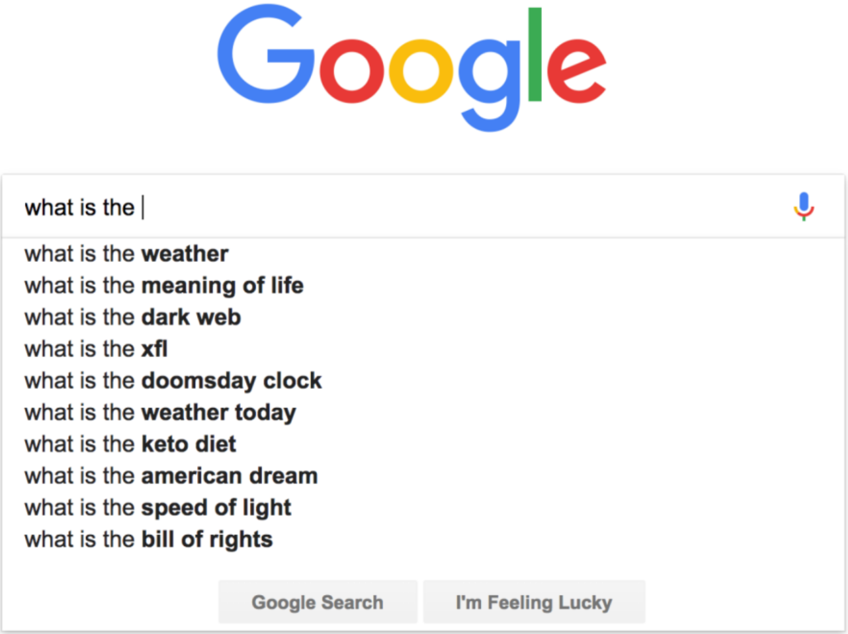
Language model



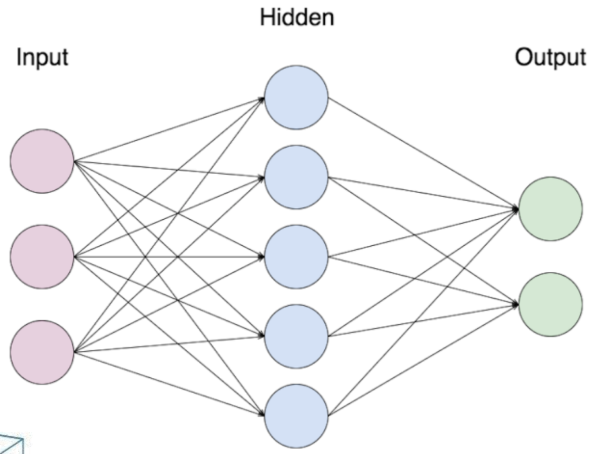
$$P(w_m|w_1, ..., w_{m-1})$$

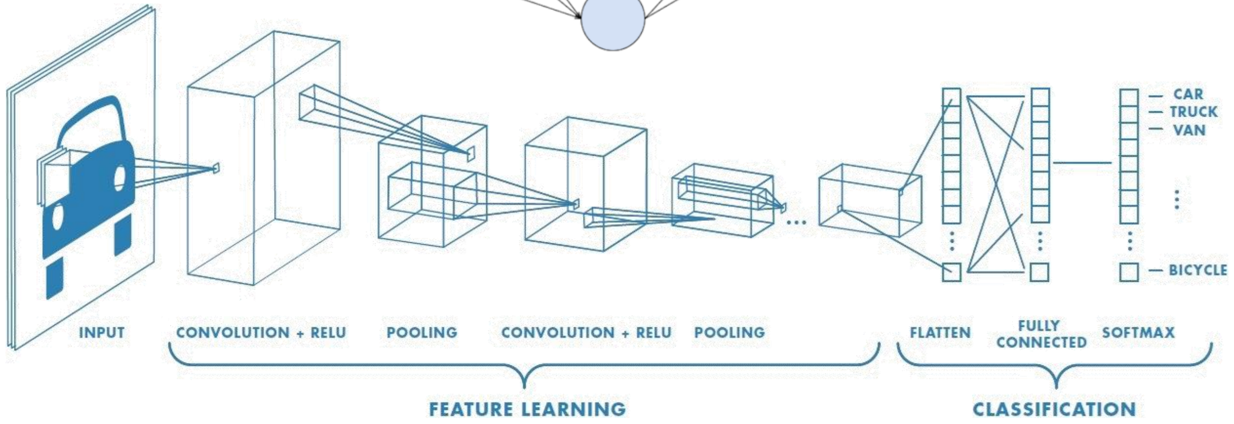
1 Giới thiệu bài toán





1 Giới thiệu bài toán





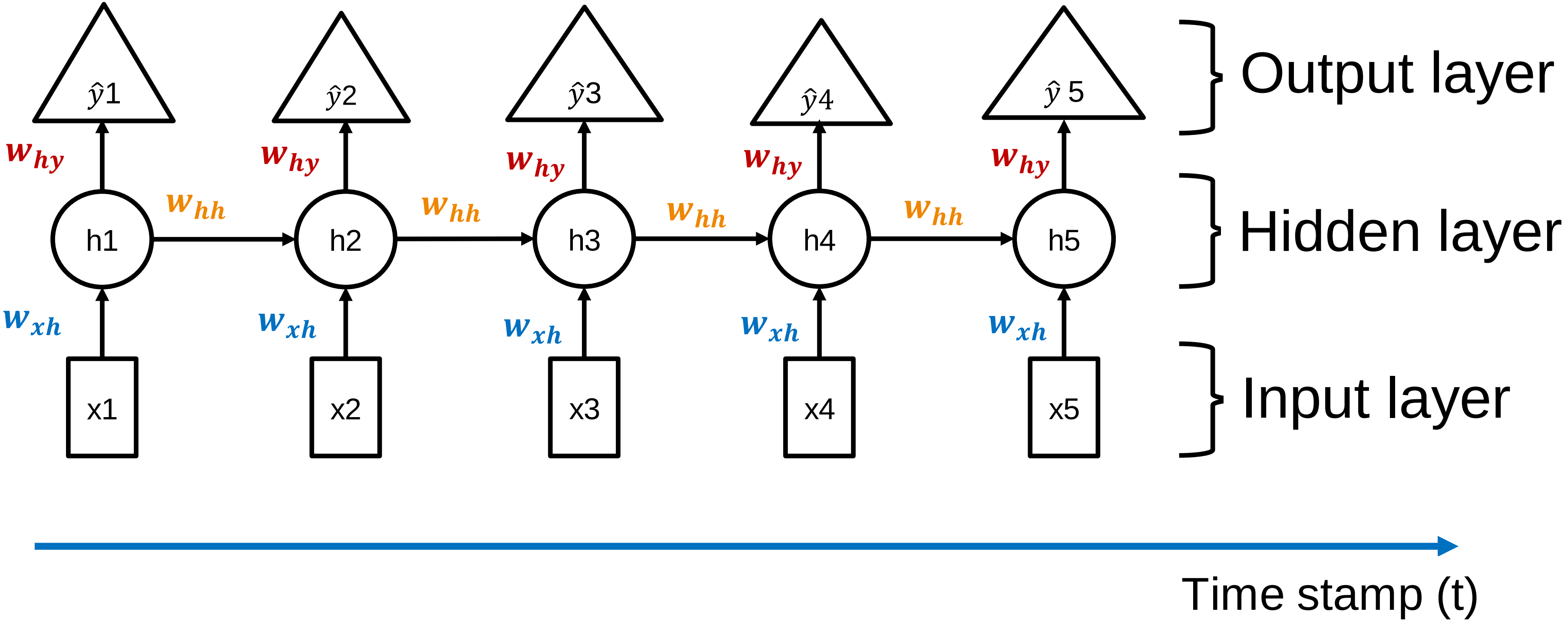
Nội dung

1. Cấu trúc Recurrent Neural Network (RNN)
2. Cách hoạt động
3. Backpropagation through time
4. Vanishing/Exploding gradients

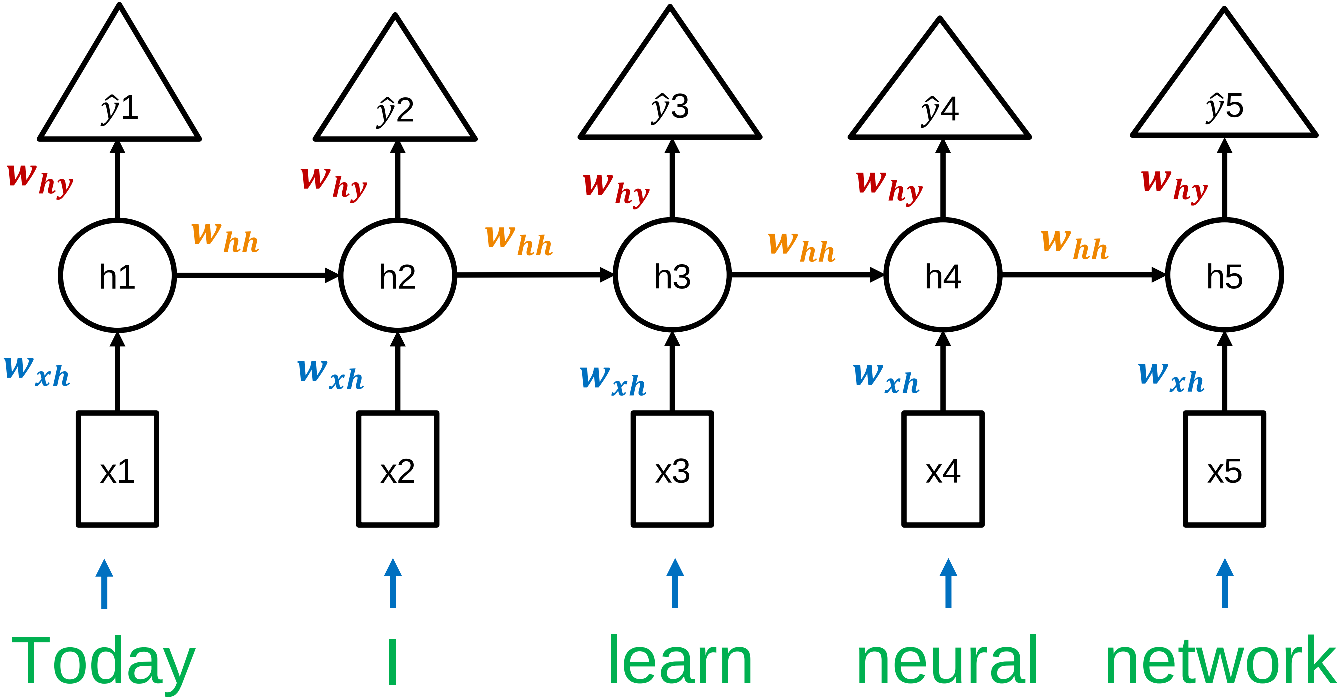
Nội dung

1. Cấu trúc Recurrent Neural Network (RNN)
2. Cách hoạt động
3. Backpropagation through time
4. Vanishing/Exploding gradients

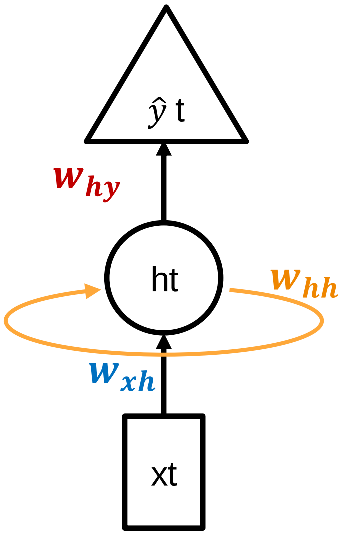
1 Cấu trúc RNN



1 Cấu trúc RNN



1 Cấu trúc RNN

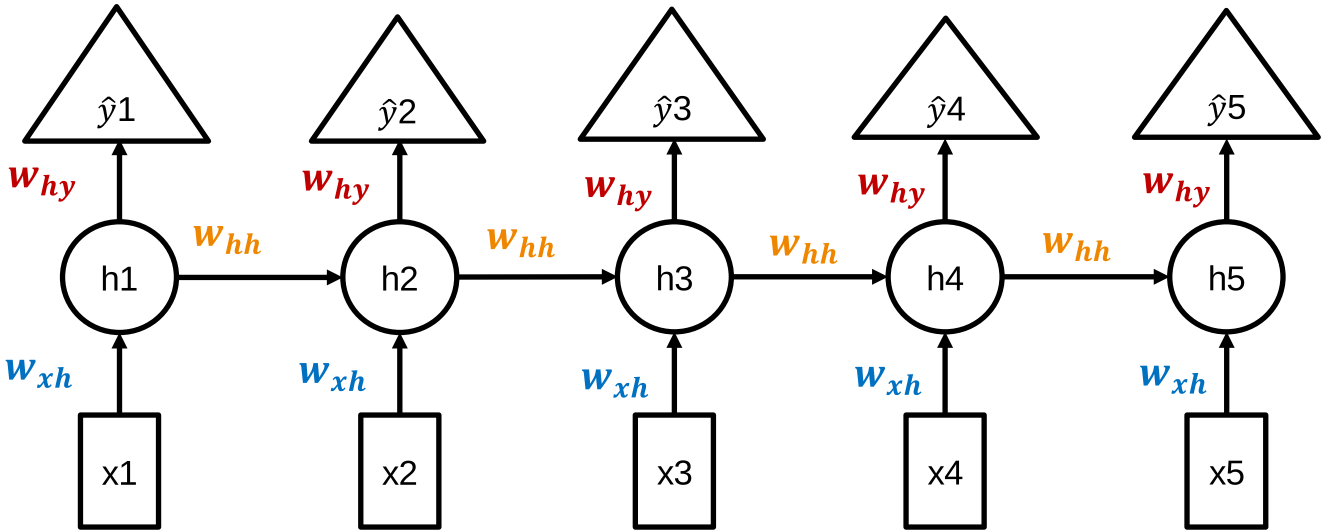


Today I learn neural network

Nội dung

1. Cấu trúc Recurrent Neural Network (RNN)
2. Cách hoạt động
3. Backpropagation through time
4. Vanishing/Exploding gradients

2 Cách hoạt động của RNN



Step 1: $h_t = \tanh(W_{hh} h_{t-1} + W_{xh} x_t + b_h)$

Step 2: $\hat{y}_t = \text{softmax}(W_{hy} h_t + b_y)$

$$\tanh(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}}$$

$$\text{softmax}(x_1) = \frac{e^{x_1}}{e^{x_1} + e^{x_2} + \cdots + e^{x_n}}$$

2 Cách hoạt động của RNN

Ví dụ: Tìm giá trị $\hat{y}_2$, biết:

$$x_1 = [0.1 \quad 0.2]$$

$$x_2 = y_1$$

$$w_{xh} = \begin{bmatrix} 0 & 1 & 0.1 \\ 0.3 & 0.5 & 1 \end{bmatrix}$$

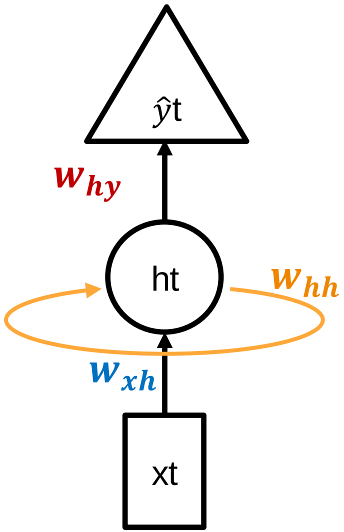
$$w_{hh} = \begin{bmatrix} 1 & 0.1 & 0.2 \\ 0.5 & 0.5 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$w_{hy} = \begin{bmatrix} 1 & 1 \\ 0.1 & 1 \\ 1 & 0.5 \end{bmatrix}$$

$$h_0 = [0 \quad 0 \quad 0]$$

$$b_h = [0.1 \quad 0.1 \quad 0]$$

$$b_y = [0 \quad 0.5]$$

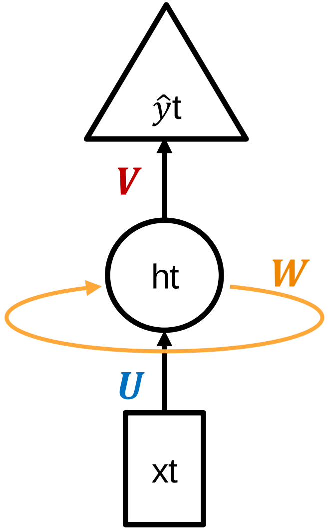


2 Cách hoạt động của RNN

Giải:

2 Quy định lại ký hiệu

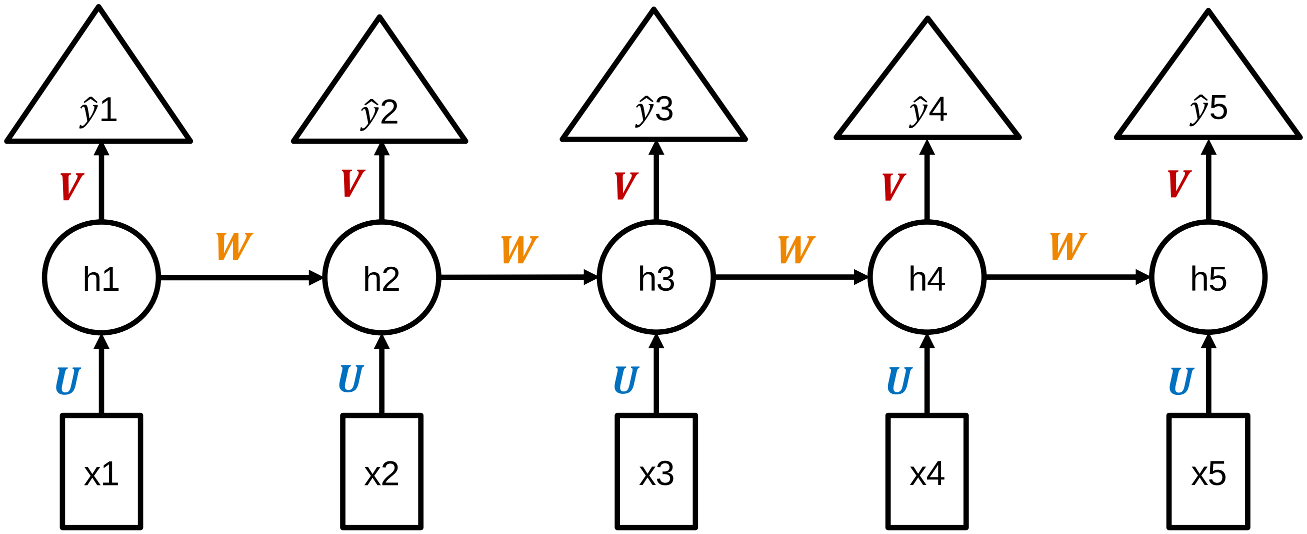
$$w_{xh} \equiv U$$
$$w_{hh} \equiv W$$
$$w_{hy} \equiv V$$



Nội dung

1. Cấu trúc Recurrent Neural Network (RNN)
2. Cách hoạt động
3. Backpropagation through time
4. Vanishing/Exploding gradients

3 Backpropagation through time



$$Loss^{<t>}(\hat{y}^{<t>}, y^{<t>}) = -y^{<t>} \log \hat{y}^{<t>} - (1 - y^{<t>}) \log(1 - \hat{y}^{<t>})$$

$$\mathbf{L}(\hat{y}, y) = \sum_{t=1}^{T} Loss^{<t>}(\hat{y}^{<t>}, y^{<t>})$$

3 Backpropagation through time

- Đạo hàm hàm Loss (tương tự Lecture 8)
- Tìm gradients tương ứng cho mỗi ma trận trọng số $\boldsymbol{U}$, $\boldsymbol{W}$, $\boldsymbol{V}$

$$\boldsymbol{U} = \boldsymbol{U} - \boldsymbol{\alpha \Delta} \mathbf{U}$$

$$\boldsymbol{W} = \boldsymbol{W} - \boldsymbol{\alpha \Delta W}$$

$$\boldsymbol{V} = \boldsymbol{V} - \boldsymbol{\alpha \Delta} \mathbf{V}$$

- Cần tìm $\frac{\partial \boldsymbol{L}}{\partial V}, \frac{\partial \boldsymbol{L}}{\partial W}, \frac{\partial \boldsymbol{L}}{\partial U}$

3 Backpropagation through time

Đạo hàm riêng hàm L trên biến *V*

$$\frac{\partial \boldsymbol{L}}{\partial V} = \sum_{t=1}^{T} \frac{\partial Loss^{<t>}}{\partial V}$$

Tại một thời điểm t nhất định:

$$= [-y \log \hat{y} - (1-y)\log(1-\hat{y})]'$$

$$= \left[-y \log\left(\text{softmax}(V h_t + b_y)\right) - (1-y)\log\left(1 - \text{softmax}(V h_t + b_y)\right)\right]'$$

→ Sử dụng đạo hàm hàm hợp

Step 1: $h_t = \tanh(W_{hh} h_{t-1} + W_{xh} x_t + b_h)$

Step 2: $\hat{y}_t = \text{softmax}(W_{hy} h_t + b_y)$

3 Backpropagation through time

Đạo hàm riêng hàm L trên biến W

$$\frac{\partial \boldsymbol{L}}{\partial W} = \sum_{t=1}^{T} \frac{\partial Loss^{<t>}}{\partial W}$$

- Tìm mối liên kết giữa hàm $Loss$ và ma trận W

➢ $\partial Loss^{<4>}$ phụ thuộc vào $h4$

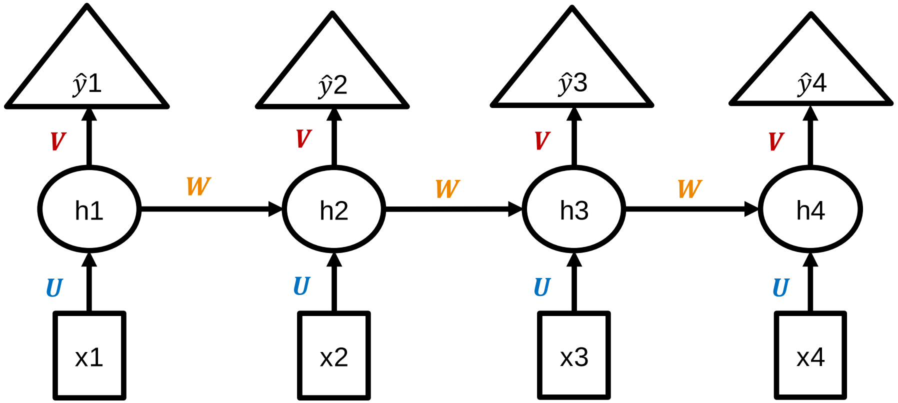
➢ $h4$ phụ thuộc vào $h3$ và W

➢ $h3$ phụ thuộc vào $h2$ và W

➢ $h2$ phụ thuộc vào $h1$ và W

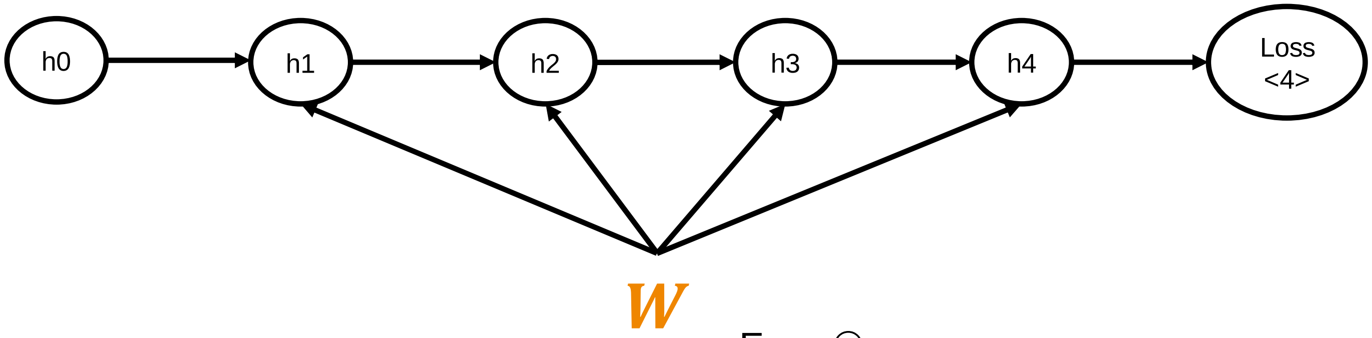
➢ $h1$ phụ thuộc vào $h0$ và W

h_0 là hằng số - starting state



3 Backpropagation through time

Đạo hàm riêng hàm L trên biến $\boldsymbol{W}$



- Áp dụng chain rule :

Easy ☺

Hard ☹

$$\frac{\partial Loss^{<4>}}{\partial W} = \frac{\partial Loss^{<4>}}{\partial h_4}\frac{\partial h_4}{\partial W}$$

$$= [-y \log \hat{y} - (1-y)\log(1-\hat{y})]'$$

$$= \left[-y \log\left(\text{softmax}(Vh_t + b_y)\right) - (1-y)\log\left(1 - \text{softmax}(Vh_t + b_y)\right)\right]'$$

3 Backpropagation through time

Đạo hàm riêng hàm L trên biến *W*

Tại sao $\frac{\partial h_4}{\partial W}$ hard ☹?

- Nhắc lại công thức tính h: $h_4 = tanh(Wh_3 + Ux_4 + b)$
- Bỏ qua activation function, đạo hàm của h_4 trên biến W là

$$\frac{\partial h_4}{\partial W} = h_3 \quad ???$$

- Trong khi $h_3 = (Wh_2 + Ux_3 + b)$ (W xuất hiện trong h_3, vậy h_3 không thể là hằng số)

3 Backpropagation through time

Đạo hàm riêng hàm L trên biến $\boldsymbol{W}$

Tại sao $\frac{\partial h_4}{\partial W}$ hard ☹?

- Nhắc lại công thức tính h: $h_4 = (Wh_3 + Ux_4 + b)$
- Không thể tính $\frac{\partial h_4}{\partial W}$ bằng cách xem h_3 là hằng số được vì h_3 cũng phụ thuộc vào W
- Để tính được $\frac{\partial h_4}{\partial W}$, ta chia đạo hàm thành 2 phần:
    - Explicit
    - Implicit

3 Backpropagation through time

Đạo hàm riêng hàm L trên biến W

$$\frac{\partial h_4}{\partial W} = \frac{\partial^+ h_4}{\partial W} + \frac{\partial h_4}{\partial h_3}\frac{\partial h_3}{\partial W}$$

$$= \frac{\partial^+ h_4}{\partial W} + \frac{\partial h_4}{\partial h_3}\left[\frac{\partial^+ h_3}{\partial W} + \frac{\partial h_3}{\partial h_2}\frac{\partial h_2}{\partial W}\right]$$

$$= \frac{\partial^+ h_4}{\partial W} + \frac{\partial h_4}{\partial h_3}\frac{\partial^+ h_3}{\partial W} + \frac{\partial h_4}{\partial h_3}\frac{\partial h_3}{\partial h_2}\left[\frac{\partial^+ h_2}{\partial W} + \frac{\partial h_2}{\partial h_1}\frac{\partial h_1}{\partial W}\right]$$

$$= \frac{\partial^+ h_4}{\partial W} + \frac{\partial h_4}{\partial h_3}\frac{\partial^+ h_3}{\partial W} + \frac{\partial h_4}{\partial h_3}\frac{\partial h_3}{\partial h_2}\frac{\partial^+ h_2}{\partial W} + \frac{\partial h_4}{\partial h_3}\frac{\partial h_3}{\partial h_2}\frac{\partial h_2}{\partial h_1}\left[\frac{\partial^+ h_1}{\partial W}\right]$$

$$\frac{\partial h_4}{\partial W} = \frac{\partial h_4}{\partial h_4}\frac{\partial^+ h_4}{\partial W} + \frac{\partial h_4}{\partial h_3}\frac{\partial^+ h_3}{\partial W} + \frac{\partial h_4}{\partial h_2}\frac{\partial^+ h_2}{\partial W} + \frac{\partial h_4}{\partial h_1}\left[\frac{\partial^+ h_1}{\partial W}\right] = \sum_{k=1}^{4}\frac{\partial h_4}{\partial h_k}\frac{\partial^+ h_k}{\partial W}$$

3 Backpropagation through time

Đạo hàm riêng hàm L trên biến *W*

$$\frac{\partial h_4}{\partial W} = \sum_{k=1}^{4} \frac{\partial h_4}{\partial h_k} \frac{\partial^+ h_k}{\partial W}$$

- Quay lại bài toán cần tìm

Easy ☺

Hard ☹

$$\frac{\partial Loss^{<4>}}{\partial W} = \frac{\partial Loss^{<4>}}{\partial h_4} \frac{\partial h_4}{\partial W}$$

Tổng quát hóa:

$$\frac{\partial Loss^{<t>}}{\partial W} = \frac{\partial Loss^{<t>}}{\partial h_t} \sum_{k=1}^{t} \frac{\partial h_t}{\partial h_k} \frac{\partial^+ h_k}{\partial W}$$

$$\frac{\partial \boldsymbol{L}}{\partial W} = \sum_{t=1}^{T} \frac{\partial Loss^{<t>}}{\partial W}$$

3 Backpropagation through time

Đạo hàm riêng hàm L trên biến $\boldsymbol{U}$

Nội dung

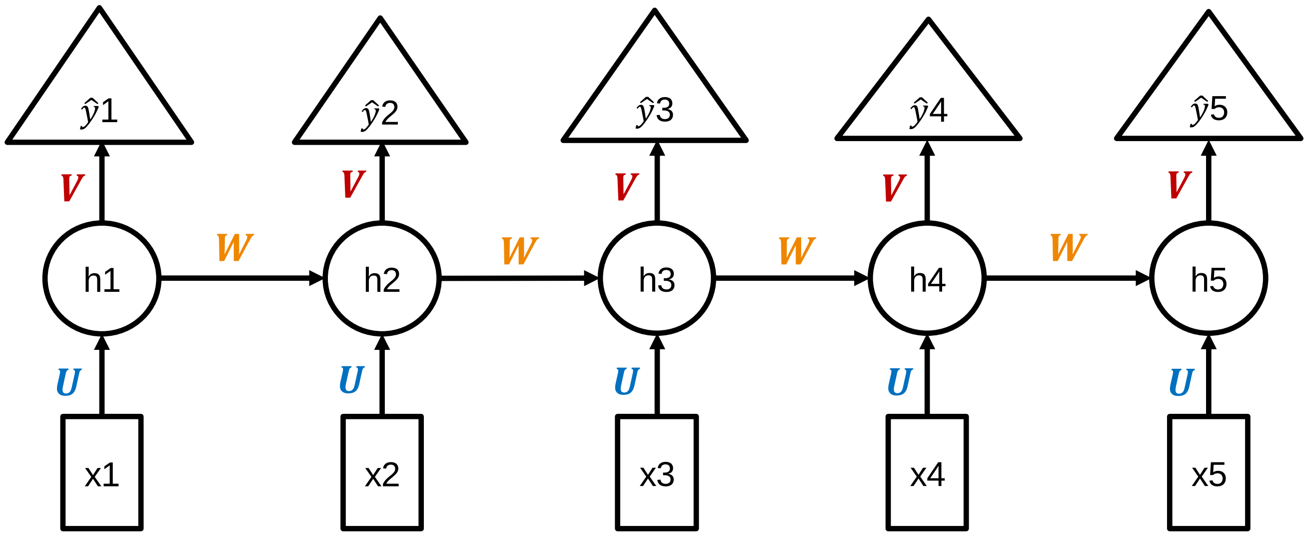
1. Cấu trúc Recurrent Neural Network (RNN)
2. Cách hoạt động
3. Backpropagation through time
4. Vanishing/Exploding gradients

4 Long-term dependency

"In France, I had a great time and I learnt some of the ____ language."

our parameters are not trained to capture long-term dependencies, so the word we predict will mostly depend on the previous few words, not much earlier ones

4 Vanishing/Exploding gradients



4 Vanishing/Exploding gradients

$$a_j = [a_{j1}, a_{j2}, a_{j3}, \ldots a_{jd},]$$
$$s_j = [\sigma(a_{j1}), \sigma(a_{j2}), \ldots \sigma(a_{jd})]$$

$$\frac{\partial s_j}{\partial a_j} = \begin{bmatrix} \frac{\partial s_{j1}}{\partial a_{j1}} & \frac{\partial s_{j2}}{\partial a_{j1}} & \frac{\partial s_{j3}}{\partial a_{j1}} & \cdots \\ \frac{\partial s_{j1}}{\partial a_{j2}} & \frac{\partial s_{j2}}{\partial a_{j2}} & \ddots & \\ \vdots & \vdots & \vdots & \frac{\partial s_{jd}}{\partial a_{jd}} \end{bmatrix}$$

$$= \begin{bmatrix} \sigma'(a_{j1}) & 0 & 0 & 0 \\ 0 & \sigma'(a_{j2}) & 0 & 0 \\ 0 & 0 & \ddots & \\ 0 & 0 & \ldots & \sigma'(a_{jd}) \end{bmatrix}$$

$$= diag(\sigma'(a_j))$$

- We are interested in $\frac{\partial s_j}{\partial s_{j-1}}$

$$a_j = W s_{j-1} + b$$
$$s_j = \sigma(a_j)$$

$$\frac{\partial s_j}{\partial s_{j-1}} = \frac{\partial s_j}{\partial a_j} \frac{\partial a_j}{\partial s_{j-1}}$$
$$= diag(\sigma'(a_j))W$$

- We are interested in the magnitude of $\frac{\partial s_j}{\partial s_{j-1}} \leftarrow$ if it is small (large) $\frac{\partial s_t}{\partial s_k}$ and hence $\frac{\partial \mathcal{L}_t}{\partial W}$ will vanish (explode)

4 Vanishing/Exploding gradients

$$\left\|\frac{\partial s_j}{\partial s_{j-1}}\right\| = \left\|diag(\sigma'(a_j))W\right\| \leq \left\|diag(\sigma'(a_j))\right\| \|W\|$$

$\because \sigma(a_j)$ is a bounded function (sigmoid, tanh) $\sigma'(a_j)$ is bounded

$$\sigma'(a_j) \leq \frac{1}{4} = \gamma \text{ [if } \sigma \text{ is logistic]}$$
$$\leq 1 = \gamma \text{ [if } \sigma \text{ is tanh]}$$

$$\left\|\frac{\partial s_j}{\partial s_{j-1}}\right\| \leq \gamma \|W\| \leq \gamma\lambda$$

$$\left\|\frac{\partial s_t}{\partial s_k}\right\| = \left\|\prod_{j=k+1}^{t} \frac{\partial s_j}{\partial s_{j-1}}\right\| \leq \prod_{j=k+1}^{t} \gamma\lambda \leq (\gamma\lambda)^{t-k}$$

- If $\gamma\lambda < 1$ the gradient will vanish
- If $\gamma\lambda > 1$ the gradient could explode

4 Vanishing/Exploding gradients

- Dùng activation function khác
- Khởi tạo lại ma trận trọng số
- Dùng các biến thể của RNN
  - Long-short term memory
  - Gated Recurrent Units