

Bài 9

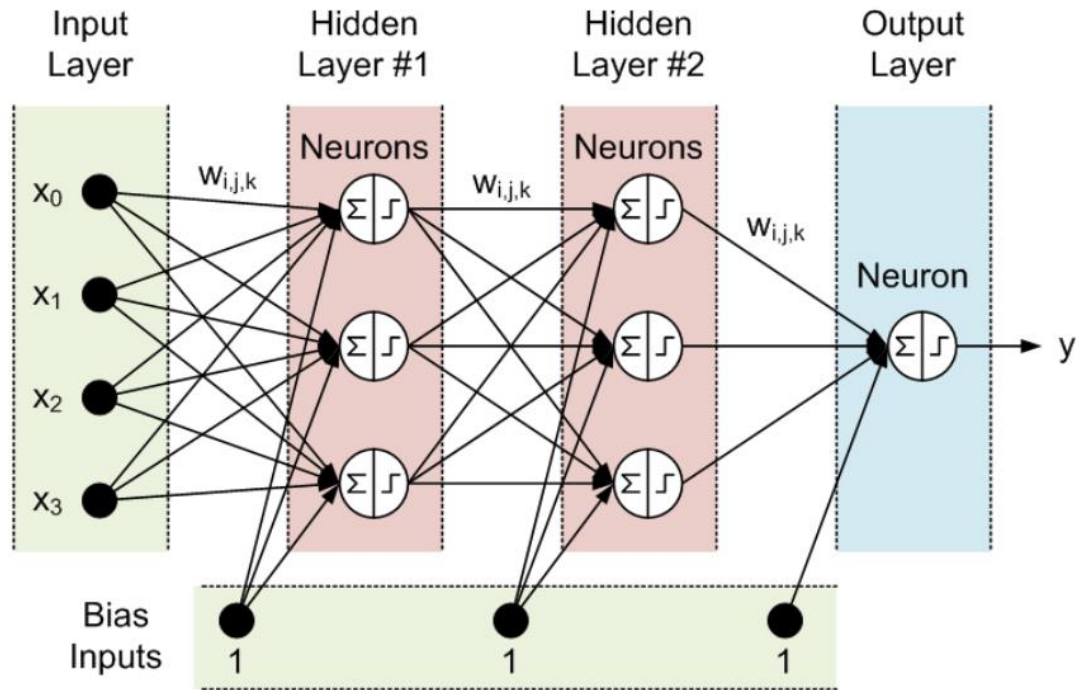
Mạng LSTM và các ứng dụng

Lê Thanh Hương
Trường Công nghệ Thông tin và Truyền thông, ĐHBKHN

Nội dung buổi học

- Giới thiệu RNN
- Giới thiệu LSTM
- Các ứng dụng với RNN và LSTM
 - LSTM cho mô hình ngôn ngữ
 - LSTM cho gán nhãn chuỗi
 - LSTM cho phân loại
 - LSTM cho dịch máy

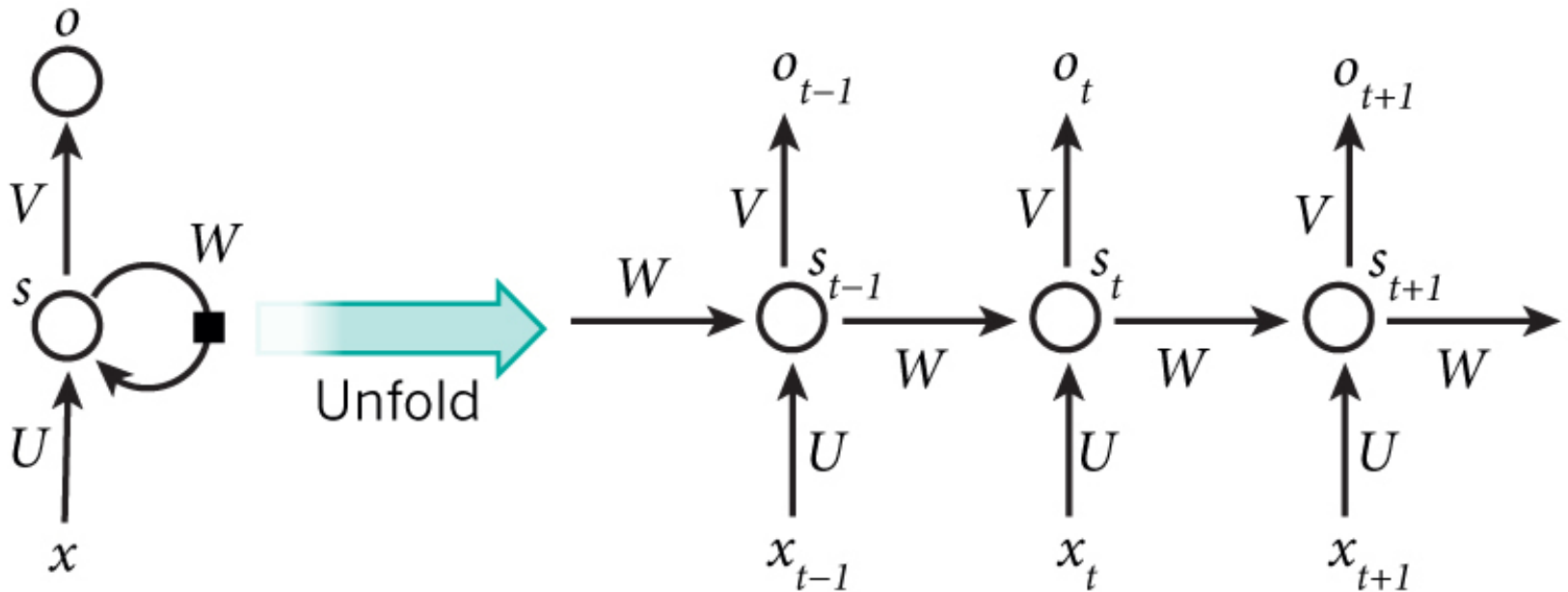
Mạng nơ ron truyền thẳng



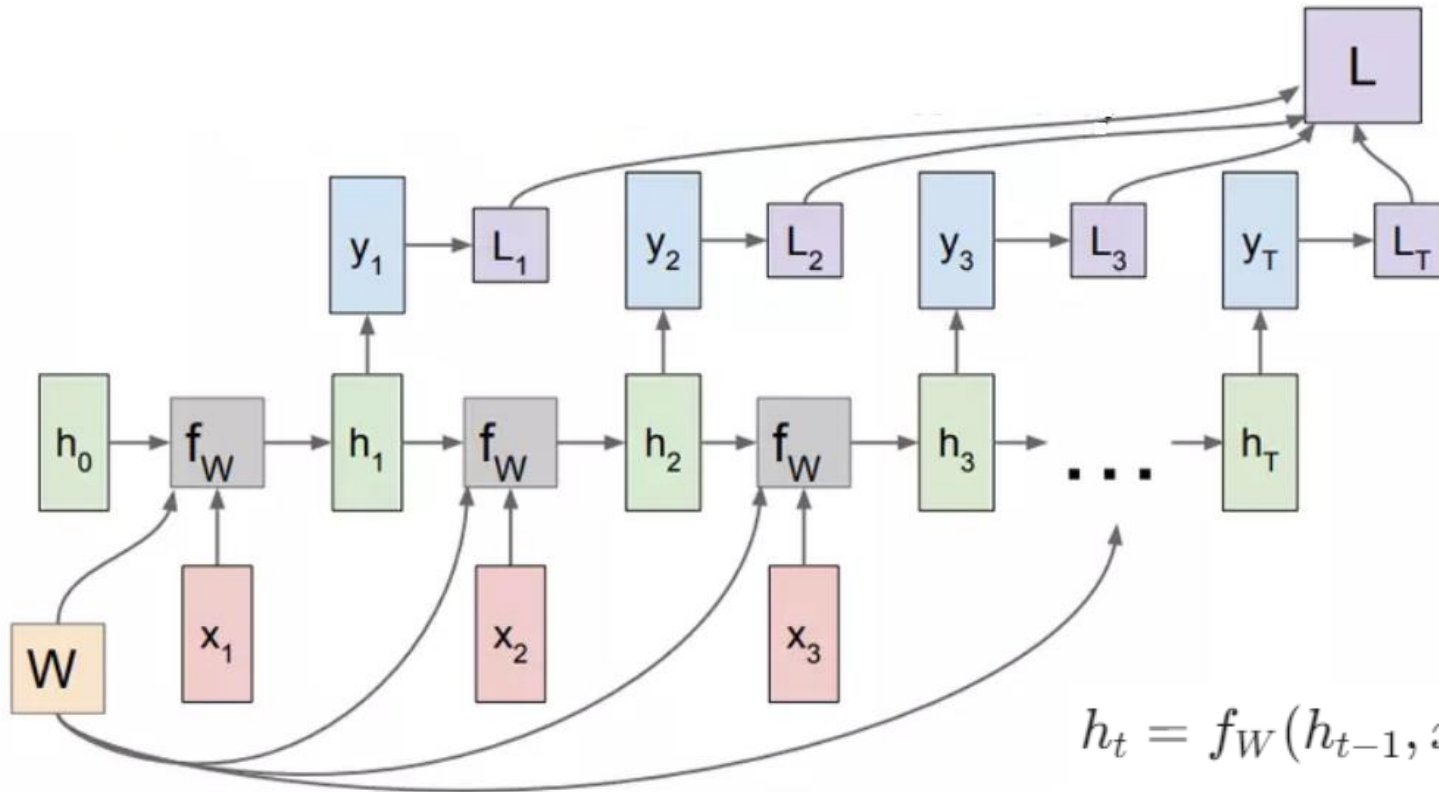
- Input layer x đi qua hidden layer h và cho ra output layer y với **full connected** giữa các layer
 - Không phù hợp với những bài toán dạng chuỗi như mô tả, hoàn thành câu, ...

Mạng nơ ron hồi quy

Recurrent Neural Networks (RNNs)



Recurrent Neural Networks



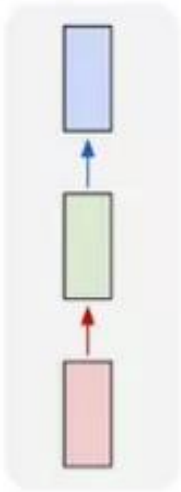
$$h_t = f_W(h_{t-1}, x_t)$$

$$h_t = \tanh(W_{hh}h_{t-1} + W_{hx}x_t)$$

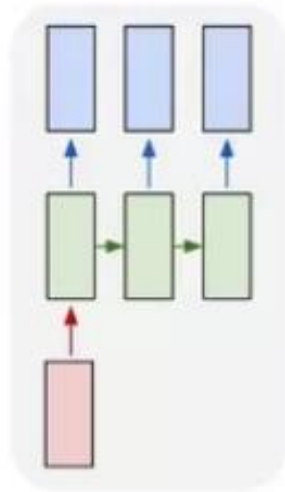
$$y_t = W_{hy}h_t$$

Các mô hình RNN

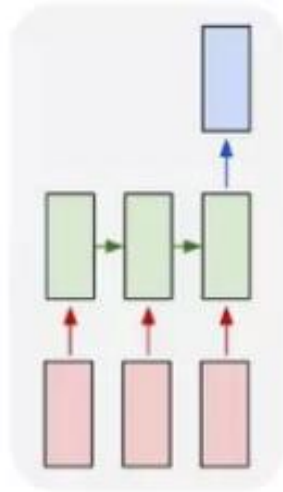
one to one



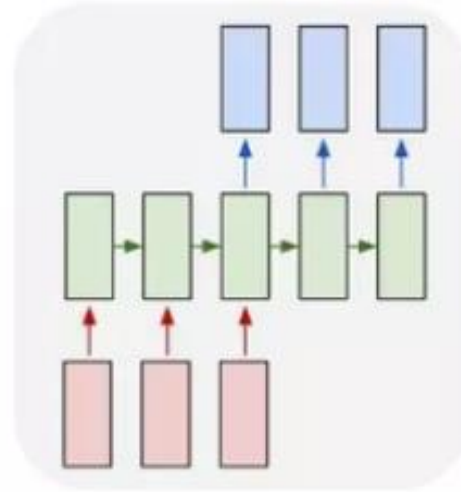
one to many



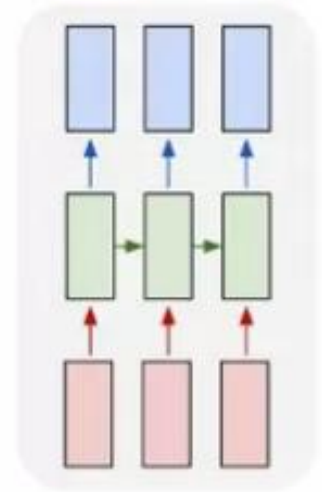
many to one



many to many



many to many



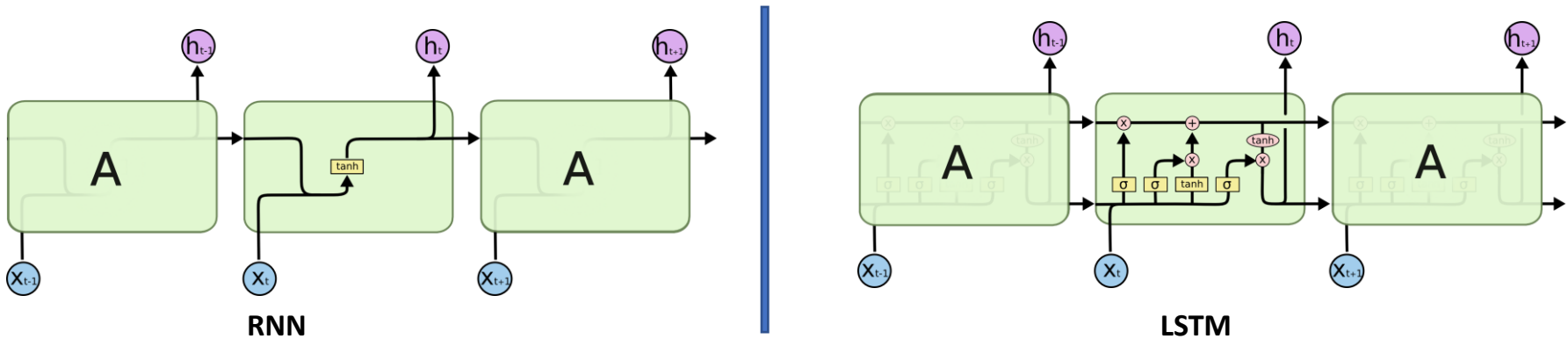
Vấn đề với RNN

- Giả sử chuỗi có 30 phần tử
- Đạo hàm của L với W ở state thứ i:

$$\frac{\partial L}{\partial W} = \frac{\partial L}{\partial y} * \frac{\partial y}{\partial s_{30}} * \frac{\partial s_{30}}{\partial s_i} * \frac{\partial s'_i}{\partial W}, \text{ trong đó } \frac{\partial s_{30}}{\partial s_i} = \prod_{j=i}^{29} \frac{\partial s_{j+1}}{\partial s_j}$$

- Vấn đề : Phụ thuộc xa, dễ “quên thông tin cũ”
=> Mong muốn: RNN có khả năng “nhớ”

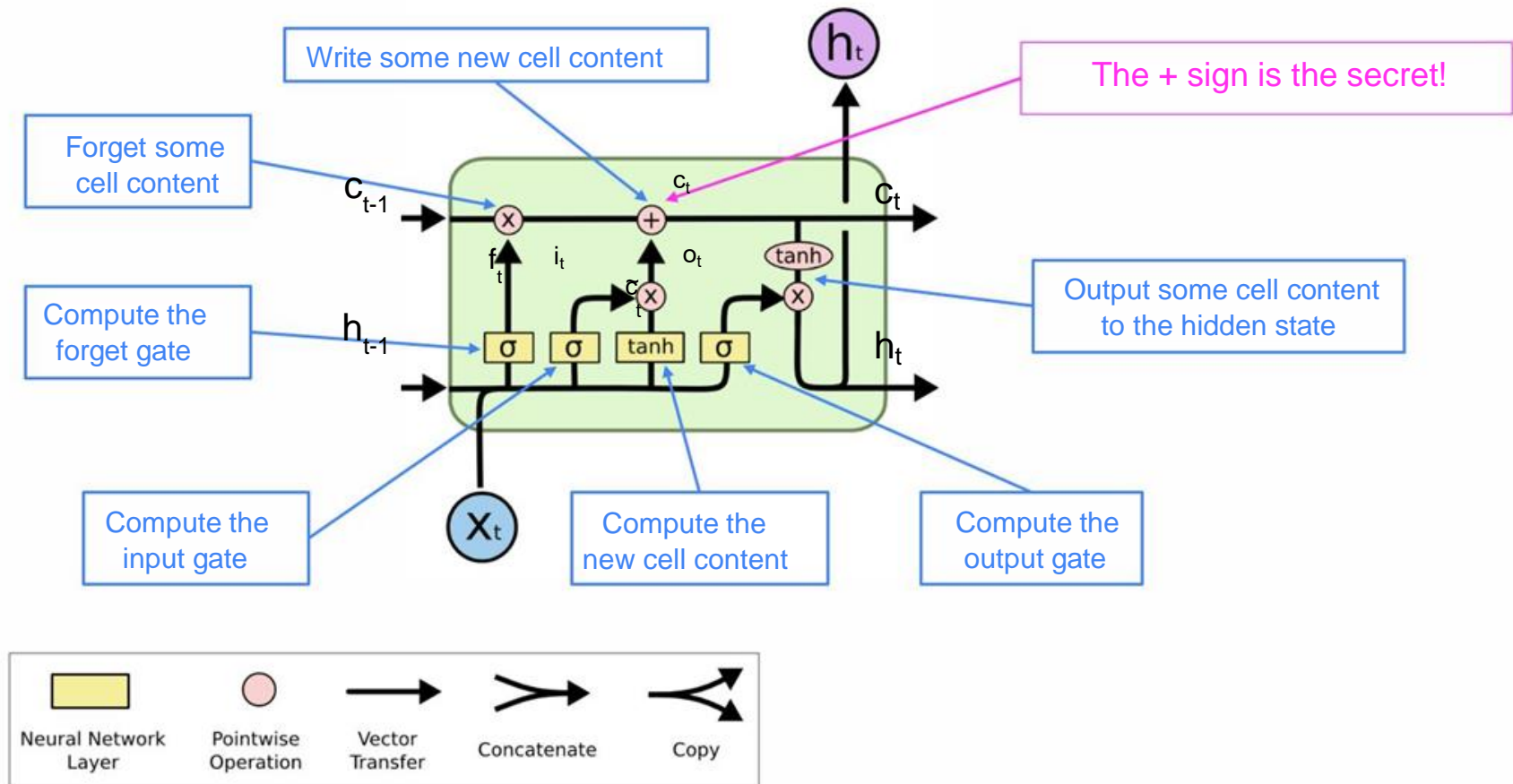
Long Short-Term Memory



- Ý tưởng: Sử dụng một đường **trực tiếp** từ các **đầu vào** phía **trước** đến **đầu ra** hiện tại

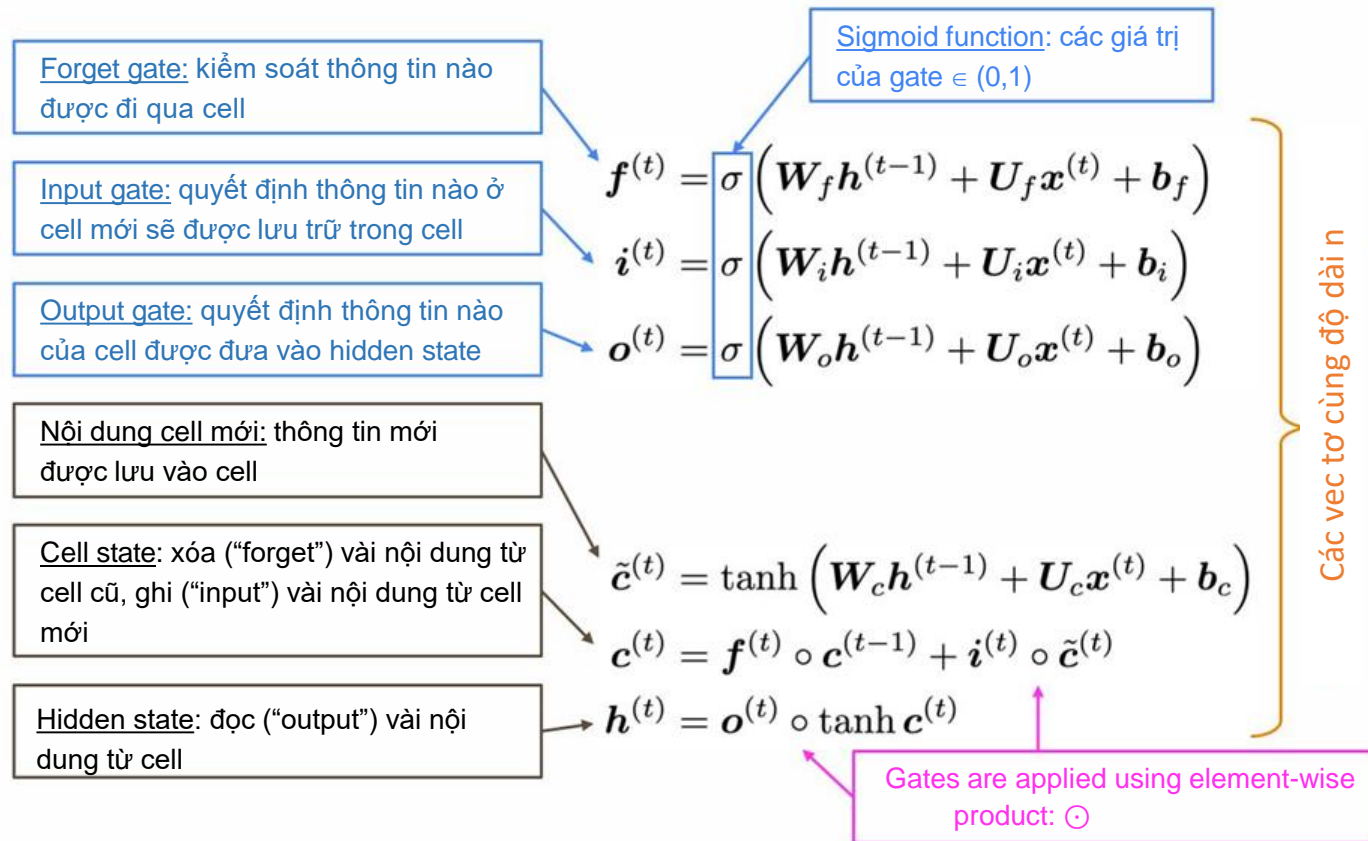
Long Short-Term Memory

Các công thức tính trong LSTM:



Long Short-Term Memory

Cho chuỗi đầu vào $x^{(t)}$, tính chuỗi trạng thái ẩn $h^{(t)}$ và các trạng thái cell $c^{(t)}$. Tại bước t :



LSTM giải quyết vấn đề quên thế nào?

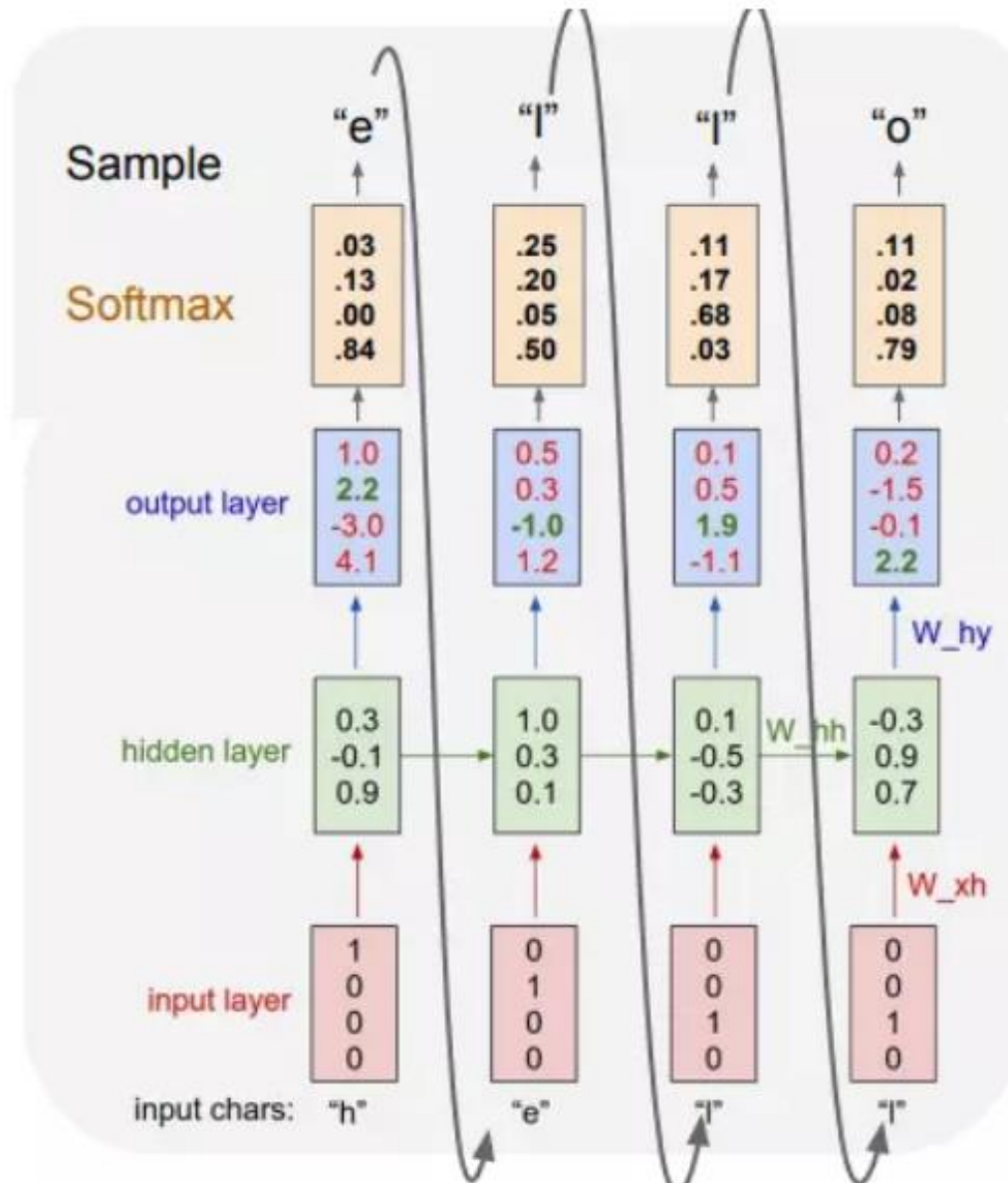
- Kiến trúc LSTM cho phép lưu thông tin qua thời gian dài dễ dàng hơn so với RNN
 - Ví dụ, nếu cổng forget = 1 cho mọi cell và cổng input = 0 thì thông tin trong cell đó được giữ mãi mãi
- LSTM không đảm bảo không bị quên, nhưng có thể nhớ các phụ thuộc dài dễ hơn so với RNN

Các ứng dụng với RNN và LSTM

Các ứng dụng

- Mô hình ngôn ngữ
 - Tiên đoán ký tự/từ tiếp theo
 - Tính xác suất sâu
- Gán nhãn chuỗi
 - Gán nhãn từ loại, nhận diện thực thể có tên, ...
- Phân loại
 - Phân tích quan điểm, ...
- Dịch máy
 - $P(\text{high winds tonight}) > P(\text{large winds tonight})$
- Sửa lỗi văn bản
 - The office is about fifteen **minuets** from my house
 - $P(\text{"about fifteen minutes from"}) > P(\text{about fifteen minuets from})$
- Nhận dạng giọng nói
 - $P(\text{I saw a van}) > P(\text{eyes awe of an})$
- Tóm tắt, hỏi – đáp, ..

Dự đoán ký tự tiếp theo với RNN



Tính xác suất câu với RNN

$$\hat{y}^{(4)} = P(x^{(5)} | \text{the students opened their})$$

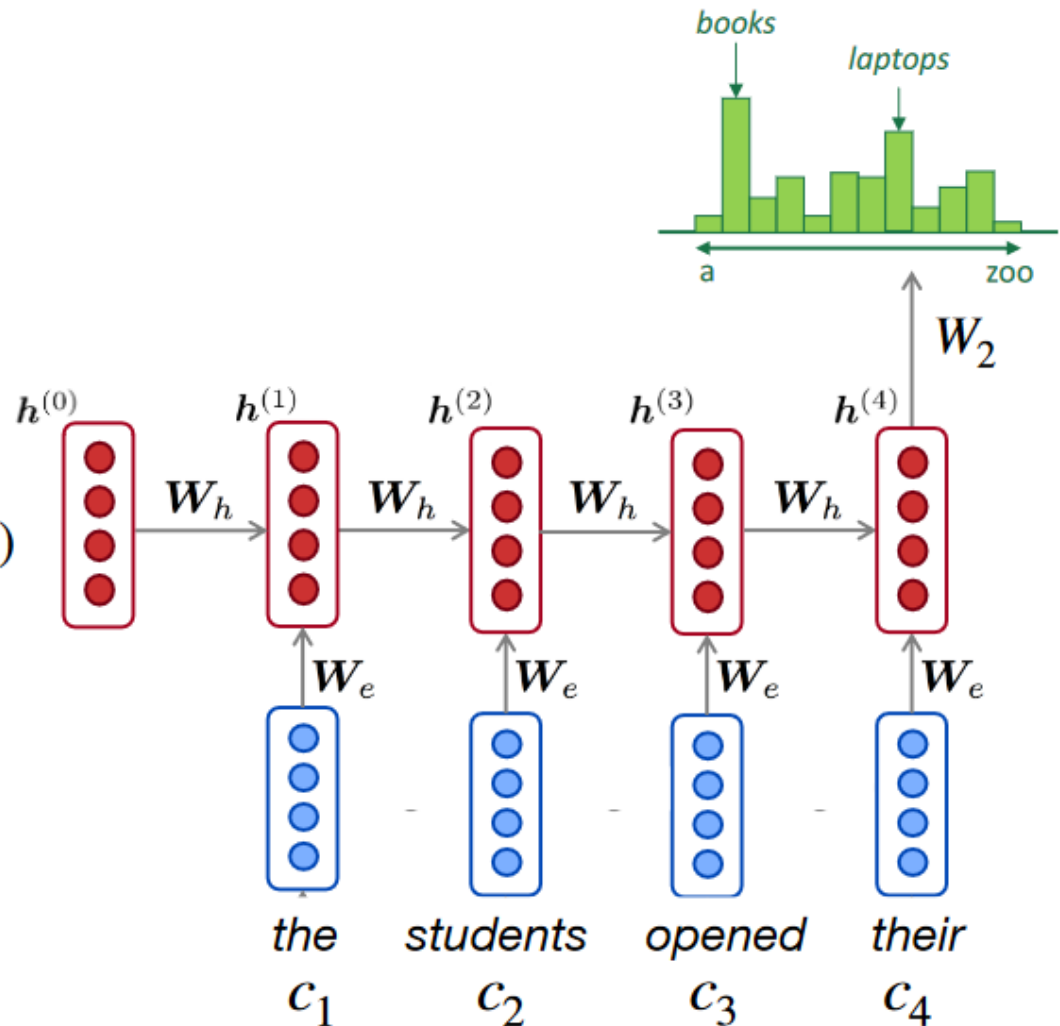
output distribution

$$\hat{y} = \text{softmax}(W_2 h^{(t)} + b_2)$$

hidden states
 $h^{(t)} = f(W_h h^{(t-1)} + W_e c_t + b_1)$
 $h^{(0)}$ is initial hidden state!

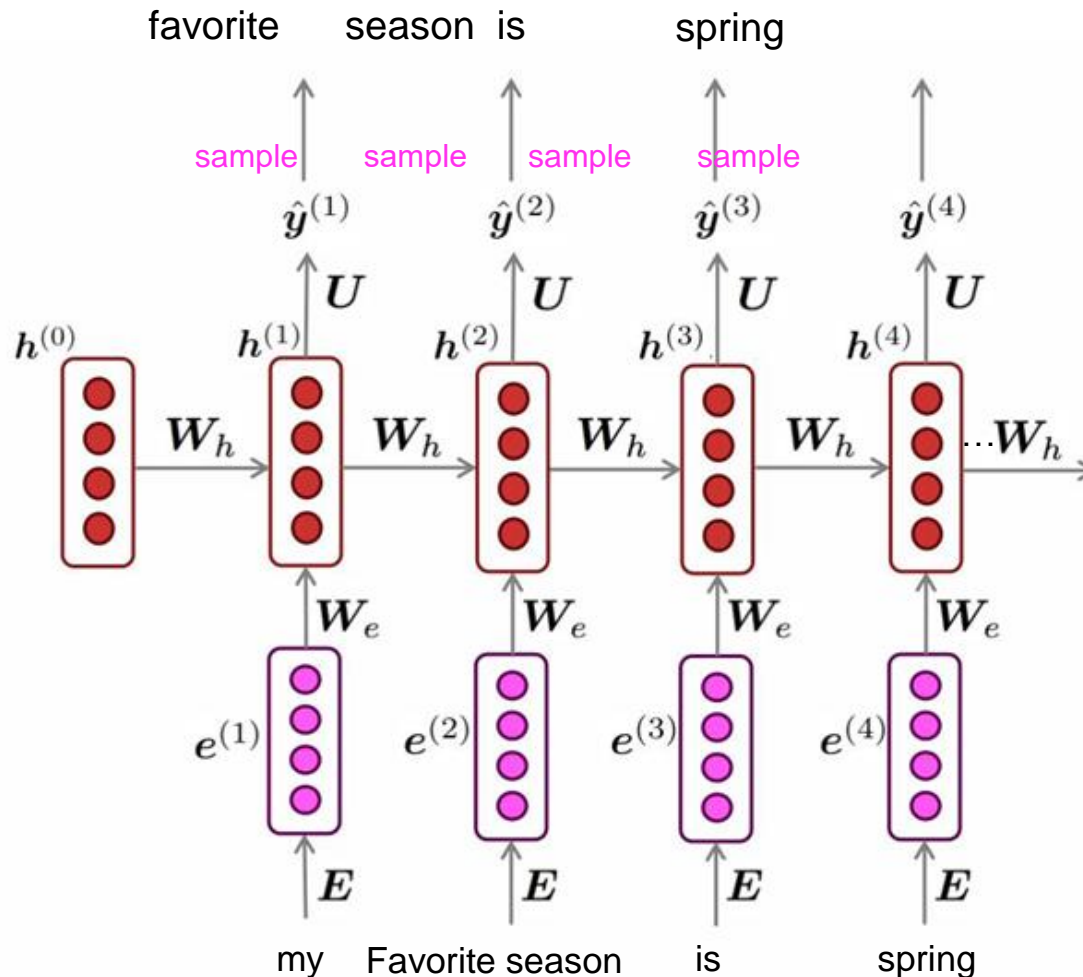
word embeddings

c_1, c_2, c_3, c_4



Sinh văn bản sử dụng RNN

- Tương tự mô hình ngôn ngữ n-gram, ta có thể sử dụng RNN để sinh văn bản bằng cách lặp lại quá trình lấy mẫu. Đầu ra 1 bước trở thành đầu vào bước sau.



Đánh giá mô hình

- Độ đo chuẩn: perplexity.

$$\text{perplexity} = \underbrace{\prod_{t=1}^T \left(\frac{1}{P_{\text{LM}}(\mathbf{x}^{(t+1)} | \mathbf{x}^{(t)}, \dots, \mathbf{x}^{(1)})} \right)^{1/T}}_{\text{Nghịch đảo xác suất trên tập ngữ liệu}}$$

Chuẩn hóa theo số từ

- Tương đương với hàm mất mát cross-entropy loss

$$= \prod_{t=1}^T \left(\frac{1}{\hat{\mathbf{y}}_{\mathbf{x}_{t+1}}^{(t)}} \right)^{1/T} = \exp \left(\frac{1}{T} \sum_{t=1}^T -\log \hat{\mathbf{y}}_{\mathbf{x}_{t+1}}^{(t)} \right) = \exp(J(\theta))$$

Perplexity càng thấp càng tốt

So sánh kết quả sử dụng n-gram và RNN

n-gram model →

Tăng độ phức tạp của RNNs ↓

Model	Perplexity
Interpolated Kneser-Ney 5-gram (Chelba et al., 2013)	67.6
RNN-1024 + MaxEnt 9-gram (Chelba et al., 2013)	51.3
RNN-2048 + BlackOut sampling (Ji et al., 2015)	68.3
Sparse Non-negative Matrix factorization (Shazeer et al., 2015)	52.9
LSTM-2048 (Jozefowicz et al., 2016)	43.7
2-layer LSTM-8192 (Jozefowicz et al., 2016)	30
Ours small (LSTM-2048)	43.9
Ours large (2-layer LSTM-2048)	39.8

Cải thiện perplexity (thấp là tốt) ↓

Source: <https://research.fb.com/building-an-efficient-neural-language-model-over-a-billion-words/>

LSTM cho gán nhãn chuỗi

- Gán nhãn từ loại (Part Of Speech Tagging – POS)

- Mỗi từ trong câu được gán với 1 từ loại

John saw the saw and decided to take it to the table.
NNP VBD DT NN CC VBD TO VB PRP IN DT NN

- Trích chọn thông tin

- Xác định cụm từ trong văn bản liên quan đến một nội dung cụ thể.
- Trích thông tin trong quảng cáo xe hơi

make model year mileage price

– For sale, 2002 Toyota Prius, 20,000 mi, \$15K or best offer.
Available starting July 30, 2006.

- Nhận diện thực thể có tên (Named entity recognition)

people organizations places

– Michael Dell is the CEO of Dell Computer Corporation and lives in Austin Texas.

LSTM cho gán nhãn chuỗi

- Semantic Role Labeling

- Với mỗi vế, xác định vai trò ngữ nghĩa của mỗi cụm danh từ so với động từ

agent patient source destination instrument

– John drove Mary from Austin to Dallas in his Toyota Prius.

– The hammer broke the window.

- Gán nhãn cho chuỗi gen trong phân tích bộ gen

exon intron

– AGCTAACGTTTCGATACGGATTACAGCCT

Nhận dạng thực thể có tên (NER)

- Mục đích: Phát hiện các thực thể có tên trong văn bản và phân loại vào các lớp được định nghĩa trước
- Các thực thể thông dụng: người (PER), vị trí (LOC), tổ chức (ORG)
- Các thực thể khác (tùy thuộc vào ứng dụng): datetime, email, quantity, product_name, price, URL, ...

Nhận dạng thực thể có tên

- Ví dụ

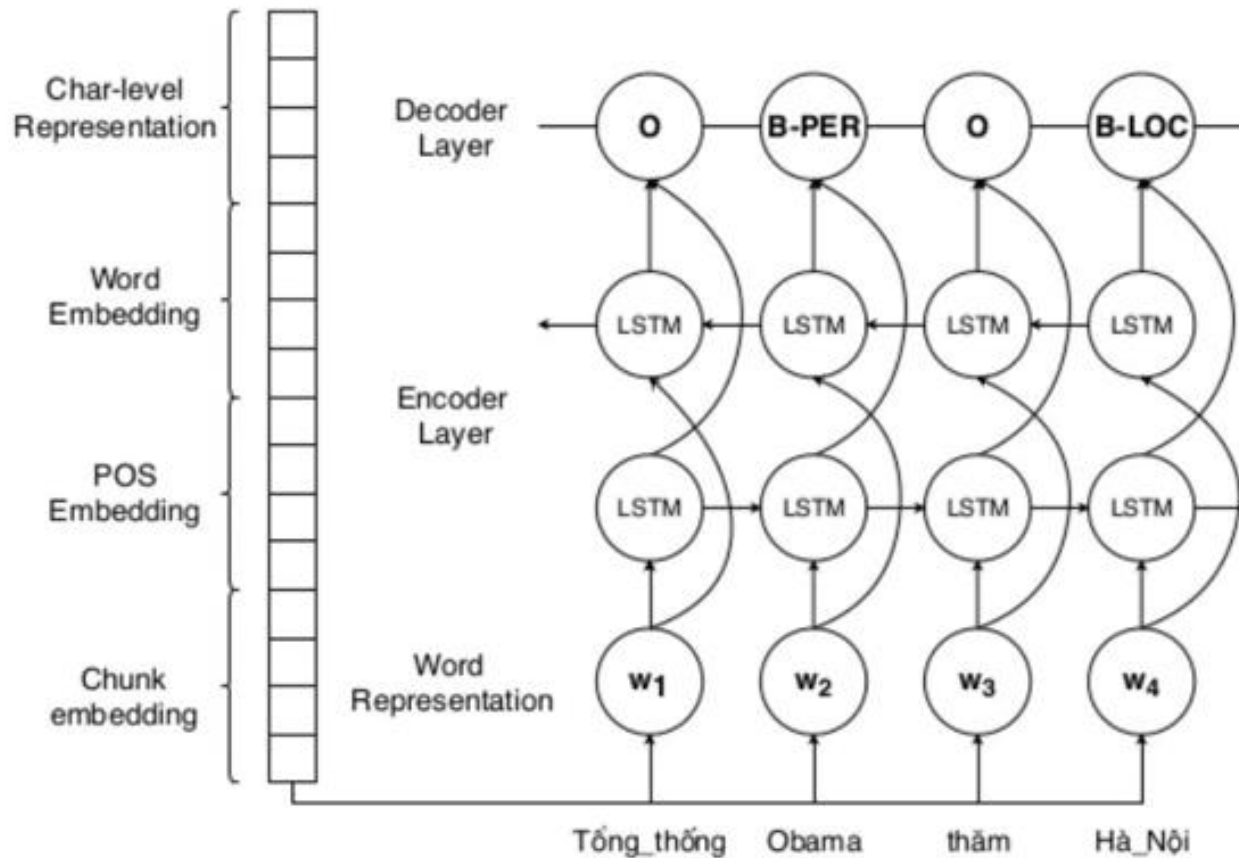
Lady Gaga is playing a concert for the Bushes in Texas next September

Person Person Location Time

- Ứng dụng của NER:

- Dịch máy
- Hỏi – đáp
- Truy vấn thông tin
- Tổng hợp giọng nói

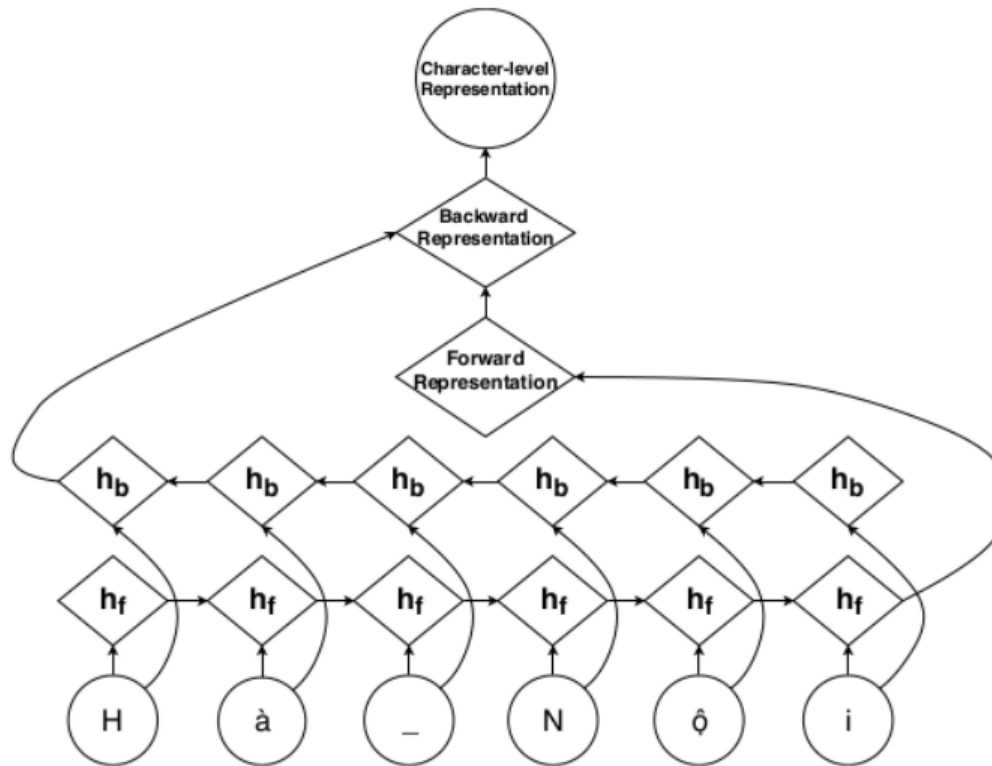
NER dựa trên biLSTM



Tầng đầu vào

- Biểu diễn nhúng kết hợp:
 - Biểu diễn từ: Sử dụng từ nhúng huấn luyện trước bởi word2vec trên 2 triệu văn bản
 - Biểu diễn ký tự: Sử dụng mạng LSTM hai chiều để học biểu diễn ký tự với khởi tạo ngẫu nhiên
 - Biểu diễn từ loại: Biểu diễn one-hot
 - Biểu diễn cụm: Biểu diễn one-hot

Học biểu diễn ký tự



LSTM hai chiều

- Sử dụng hai mạng LSTM theo chiều tiến và chiều lùi
 - Mục đích: Các từ ở đầu câu có thể sử dụng cả thông tin ở cuối câu để dự đoán và ngược lại
- Đầu ra được ghép nối để đưa vào tầng đầu ra

Tầng đầu ra

- Dự đoán các nhãn BIO ứng với các loại thực thể
 - VD: Với 3 loại thực thể ORG, PER, LOC, tập nhãn có 7 nhãn (B-ORG, I-ORG, B-PER, I-PER, B-LOC, I-LOC, O)
- Tầng đầu ra có thể được đưa vào một mô hình CRFs để thể hiện quan hệ với nhãn ở thời điểm trước thông qua xác suất chuyển đổi

Đánh giá kết quả

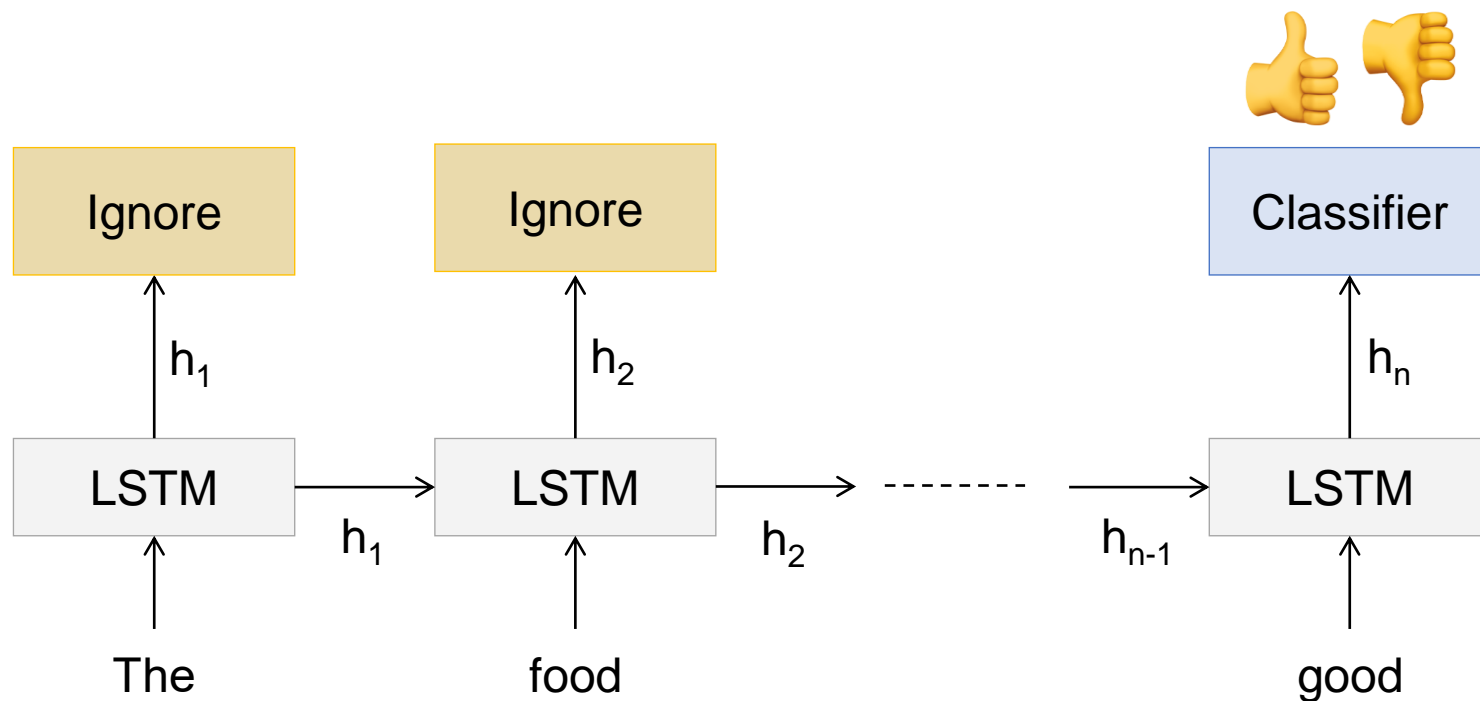
Method	P	R	F1	F1 (w.o char)
Feature-rich CRFs [25]	93.87	93.99	93.93	-
NNVLP [7]	92.76	93.07	92.91	-
BiLSTM-CRFs	90.97	87.52	89.21	76.43
BiLSTM-CRFs + POS	90.90	90.39	90.64	86.06
BiLSTM-CRFs + Chunk	95.24	92.16	93.67	87.13
BiLSTM-CRFs + POS + Chunk	95.44	94.33	94.88	91.36

BiLSTM-CRFs sử dụng thêm các thông tin PoS và phân cụm

BiLSTM-CRFs không kết hợp biểu diễn mức kí tự

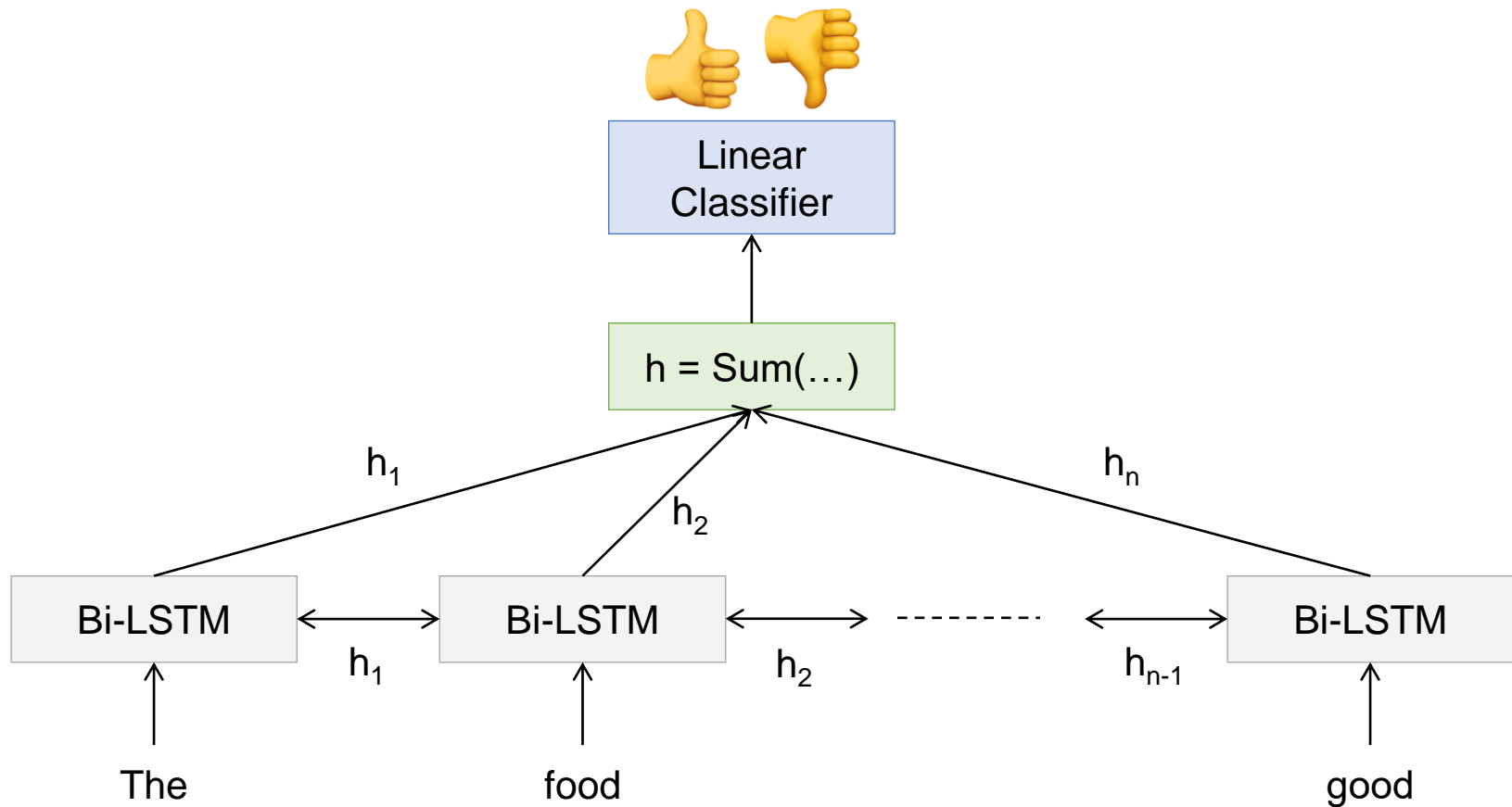
Phân loại quan điểm với LSTM

- Đầu ra của từ cuối cùng được sử dụng làm đặc trưng để xây dựng bộ phân lớp



Phân loại quan điểm với LSTM

- Đặc trưng được tính dựa trên đầu ra các từ



LSTM cho dịch máy

- Dịch: Chuyển từ câu nguồn f (tiếng Pháp) sang câu đích e (tiếng Anh)
- Ước lượng $P(e|f)$ sử dụng NN

$$\begin{aligned} p(e|f) &= p(e_1, e_2, \dots, e_m|f) \\ &= p(e_1|f) \cdot p(e_2|e_1, f) \cdot p(e_3|e_2, e_1, f) \cdot \dots \\ &= \prod_{i=1}^m p(e_i|e_1, \dots, e_{i-1}, f) \end{aligned}$$

Mô hình sequence to sequence

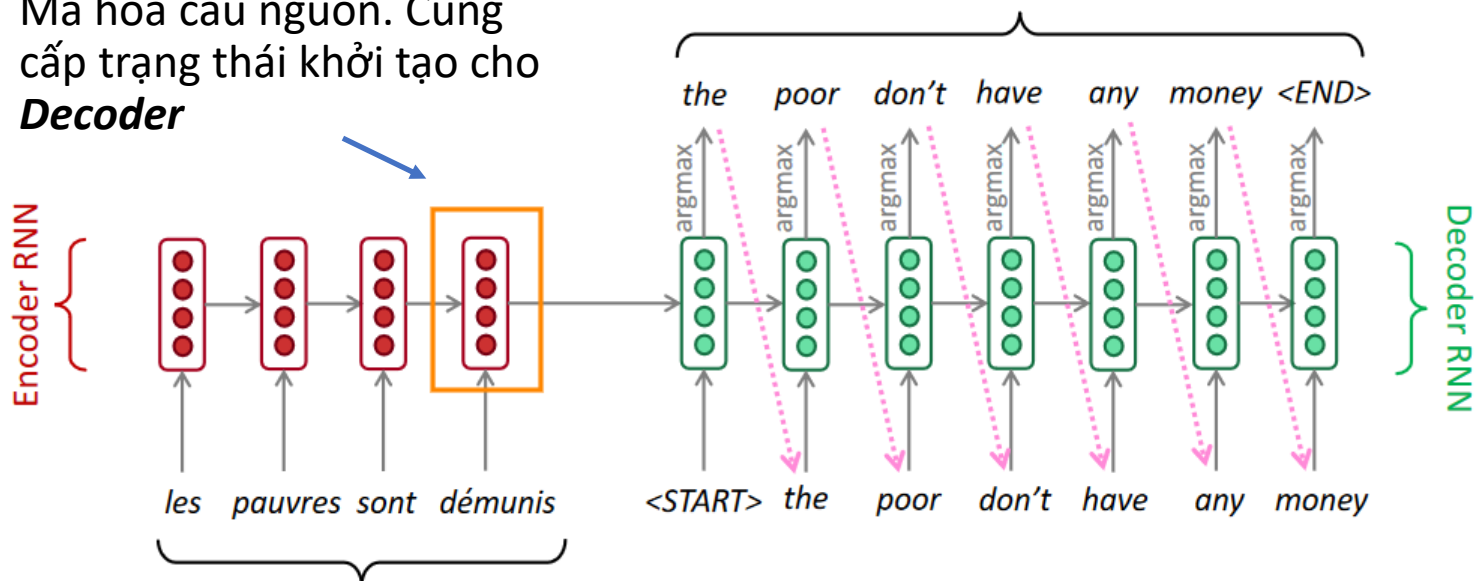
- Gồm 2 RNNs khác nhau để có mô hình

$$\prod_{i=1}^m p(e_i | e_1, \dots, e_{i-1}, f)$$

- RNN đầu tiên gọi là encoder, mã hóa câu f (tiếng Pháp)
- RNN thứ 2 gọi là decoder, sinh câu e (Tiếng Anh)

Mô hình Seq2Seq

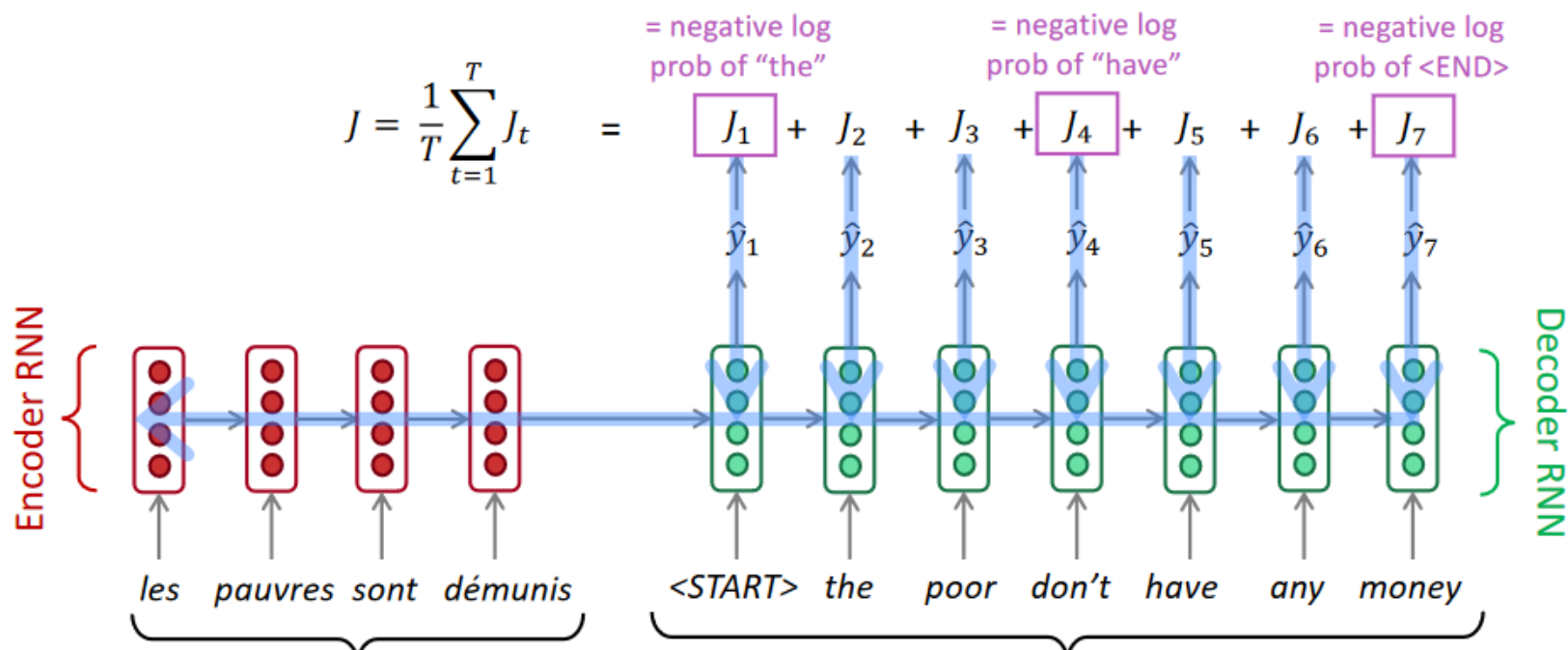
- Mã hóa câu nguồn. Cung cấp trạng thái khởi tạo cho **Decoder**



1) Encoder: Nén câu **nguồn** thành **vector**

2) Decoder: Là mô hình ngôn ngữ, sinh câu đích với điều kiện **vector nguồn**

Huấn luyện mô hình Seq2Seq



- Tham số của mô hình

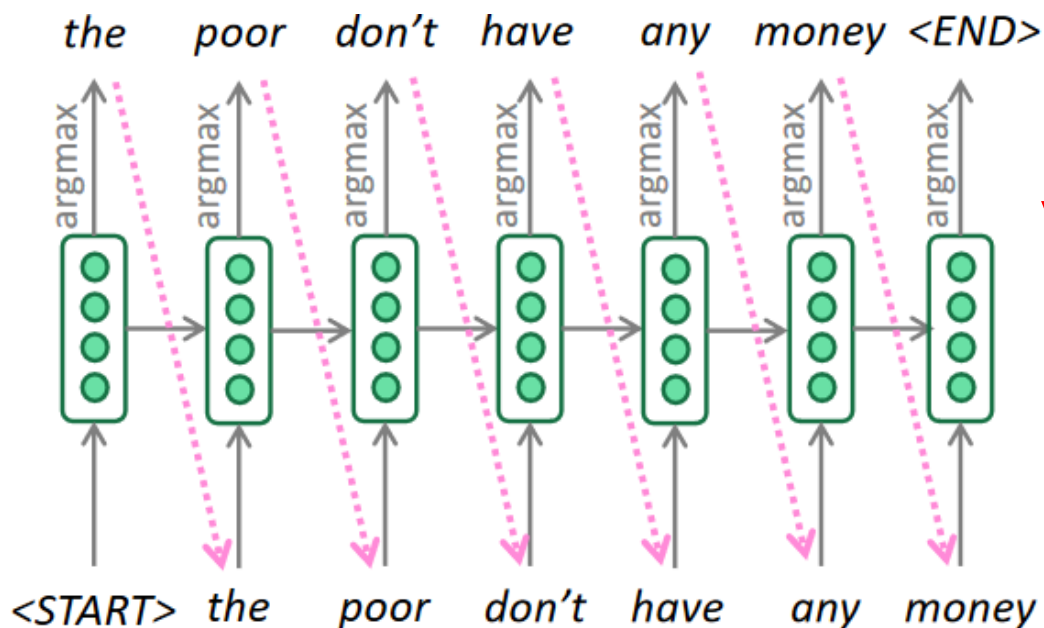
$$W^{enc}, W^{dec}, W_{out}$$

Bước giải mã

- Với mô hình Seq2Seq đã huấn luyện: Cho 1 câu tiếng Pháp, cần xác định câu tiếng Anh có xác suất cao nhất.

- Tìm
$$\arg \max \prod_{i=1}^m p(e_i | e_1, \dots, e_{i-1}, f)$$

- Giải pháp đơn giản nhất: giải mã tham lam



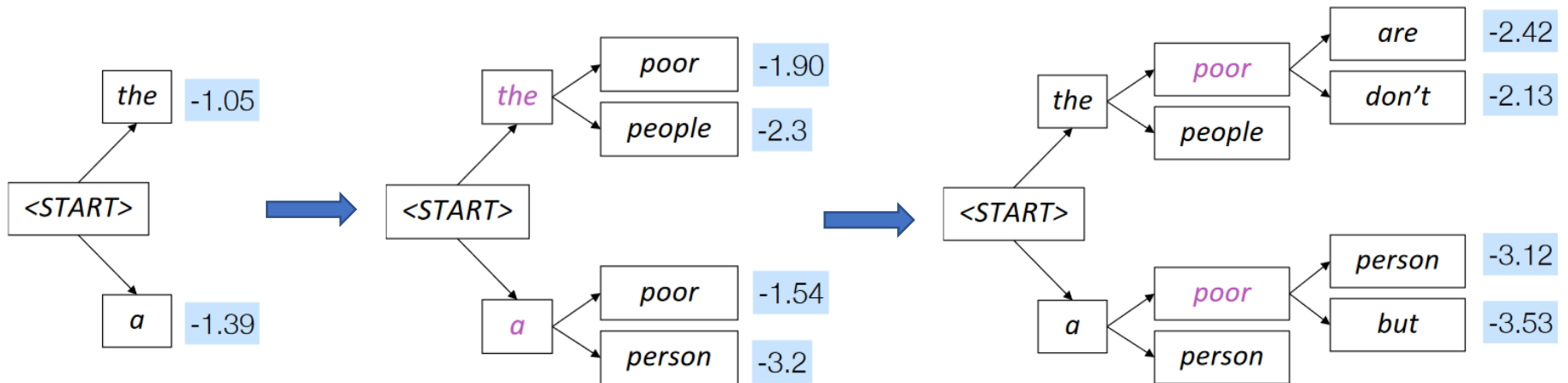
Vấn đề gặp phải?

Dùng beam search

- Giải mã tham lam: Chúng ta không thể quay lại để kiểm tra quyết định của bước trước.
 - les pauvres sont démunis (the poor don't have any money)
 - → the _____
 - → the poor _____
 - → the poor **are** _____
- Ý tưởng: Sử dụng beam search.
 - Lưu k bản dịch tại mỗi thời điểm thay vì chỉ 1
 - Thường beam size k khoảng 5 - 10

Ví dụ beam search

- Beam size = 2



Q&A

Thank you!