

# CSCSE 638 Natural Language Processing

## Foundation and Techniques

Lecture 6: Sequential Labeling, Sequence-to-Sequence, Attention

Kuan-Hao Huang

Spring 2025



(Some slides adapted from Chris Manning, Karthik Narasimhan, Danqi Chen, and Vivian Chen)

# Project – Proposal

- Due: 3/3
- Page limit: 2 pages
- Format: [ACL style](#)
- The proposal should include
  - Introduction to the topic you choose
  - Related literature
  - Novelty and challenges
  - Evaluation metrics
  - The dataset, models, and approaches you plan to use

# Project Sign-Up

- <https://docs.google.com/spreadsheets/d/15Rj4AovtHtIZxILbX1ydrw7IEylamXuV7Dtg7cBD2EU/edit?usp=sharing>
- 3~4 members per team
  - Form teams on your own
  - No solo teams (We have too many students!)

3~4 members per team									
	Project Topic	Member 1 (Name)	Member 1 (E-mail)	Member 2 (Name)	Member 2 (E-mail)	Member 3 (Name)	Member 3 (E-mail)	Member 4 (Name)	Member 4 (E-mail)
Team 1									
Team 2									
Team 3									
Team 4									
Team 5									
Team 6									
Team 7									
Team 8									
Team 9									
Team 10									
Team 11									
Team 12									

# If You Need Teammates – Team Match

- <https://docs.google.com/spreadsheets/d/15Rj4AovtHtIZxILbX1ydrw7IEylamXuV7Dtg7cBD2EU/edit?usp=sharing>
- Put your names and emails, contact other classmates
- Some listed topics
  - Listed topics are broad directions for reference and **are not project topics**
  - You have to survey related literature and **propose a more specific one**

Topic (feel free to add new ones!)	Related Readings	Interested! (Name)	Interested! (E-mail)	Interested! (Name)	Interested! (E-mail)	Interested! (Name)	Interested! (E-mail)
In-Context Learning with LLMs	<a href="https://arxiv.org/abs/2202.12837">https://arxiv.org/abs/2202.12837</a> <a href="https://arxiv.org/abs/2402.05403">https://arxiv.org/abs/2402.05403</a> <a href="https://arxiv.org/abs/2311.00237">https://arxiv.org/abs/2311.00237</a>						
LLM Hallucination	<a href="https://arxiv.org/abs/2305.13534">https://arxiv.org/abs/2305.13534</a> <a href="https://arxiv.org/abs/2303.08896">https://arxiv.org/abs/2303.08896</a> <a href="https://arxiv.org/abs/2407.07071">https://arxiv.org/abs/2407.07071</a>						
Cross-Lingual Transfer Learning	<a href="https://arxiv.org/abs/1901.07291">https://arxiv.org/abs/1901.07291</a> <a href="https://arxiv.org/abs/1911.02116">https://arxiv.org/abs/1911.02116</a> <a href="https://arxiv.org/abs/1912.07840">https://arxiv.org/abs/1912.07840</a>						
LLM Reasoning	<a href="https://arxiv.org/abs/2201.11903">https://arxiv.org/abs/2201.11903</a> <a href="https://arxiv.org/abs/2305.10601">https://arxiv.org/abs/2305.10601</a>						
Math and Logical Reasoning	<a href="https://arxiv.org/abs/2410.05229">https://arxiv.org/abs/2410.05229</a> <a href="https://arxiv.org/abs/2405.18357">https://arxiv.org/abs/2405.18357</a> <a href="https://arxiv.org/abs/2410.15580">https://arxiv.org/abs/2410.15580</a>						
Prompt Optimization	<a href="https://arxiv.org/abs/2305.03495">https://arxiv.org/abs/2305.03495</a>						
Model Self-Correction	<a href="https://arxiv.org/abs/2203.11171">https://arxiv.org/abs/2203.11171</a> <a href="https://arxiv.org/abs/2303.17651">https://arxiv.org/abs/2303.17651</a> <a href="https://arxiv.org/abs/2211.00053">https://arxiv.org/abs/2211.00053</a> <a href="https://arxiv.org/abs/2310.01798">https://arxiv.org/abs/2310.01798</a>						
Model Editing	<a href="https://arxiv.org/abs/2202.05262">https://arxiv.org/abs/2202.05262</a> <a href="https://arxiv.org/abs/2310.20138">https://arxiv.org/abs/2310.20138</a> <a href="https://arxiv.org/abs/2308.08742">https://arxiv.org/abs/2308.08742</a>						

# If You Need Teammates – Team Match

- <https://docs.google.com/spreadsheets/d/15Rj4AovtHtIZxILbX1ydrw7IEylamXuV7Dtg7cBD2EU/edit?usp=sharing>
- Feel free to add new topics!

The following are proposed by students.			
Topic	Related Readings	Interested! (Name)	Interested! (E-mail)

# Project – Suggested Topics

- Choose a topic by selecting an existing problem discussed in class and developing **new ideas** around it
- Identify any **unresolved challenges** from a published paper and improve the proposed approach
- Implement **multiple baseline models** for a specific topic, make a **comprehensive comparison** of their performance, and report **findings and insights**
- Participate in shared tasks at SemEval, CoNLL, Kaggle, or relevant workshops, and present the **techniques** you apply

If you are not sure about whether the proposed project is appropriate or not, come to my office hour for a discussion

# Quiz 1

- Date: 2/17 (the same day as the deadline for assignment 1)
  - 10 minutes before the end of the lecture
  - 5 questions focusing on high-level concepts

Week	Date		Topic
W1	1/13	L1	Course Overview <a href="#">[slides]</a>
	1/15	L2	Text Classification <a href="#">[slides]</a>
W2	1/20		<b>Martin Luther King, Jr. Day (No Class)</b>
	1/22	L3	Word Representations <a href="#">[slides]</a>
W3	1/27	L4	Word Representations, Tokenization, Language Modeling <a href="#">[slides]</a>
	1/29	L5	Convolutional Neural Network, Recurrent Neural Network <a href="#">[slides]</a>
W4	2/3	L6	Sequential Labeling, Sequence-to-Sequence, Attention
	2/5	L7	Transformers

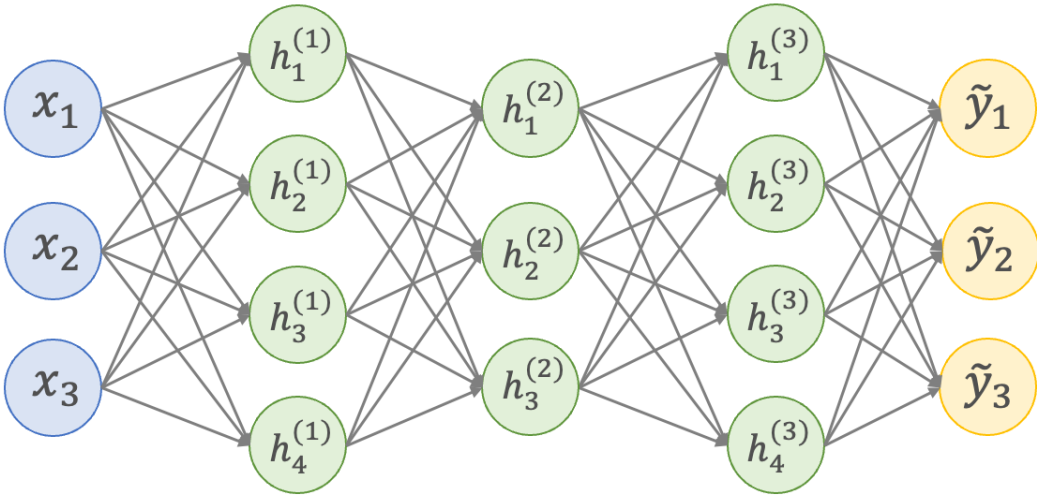
# Lecture Plan

- Sequential Labeling
- Sequence-to-Sequence
- Attention



# Recap: Convolutional Neural Network (CNN)

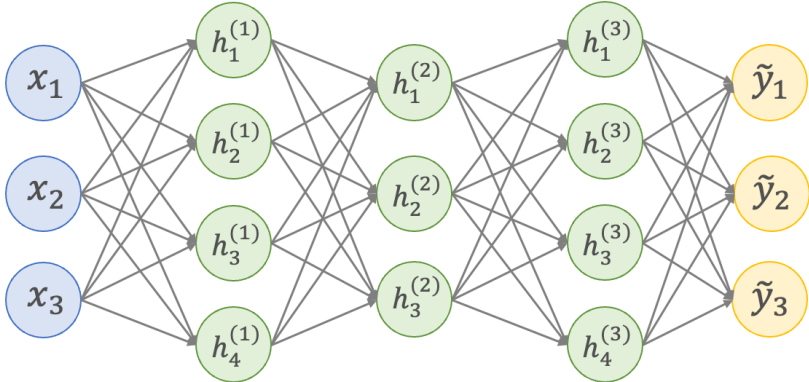
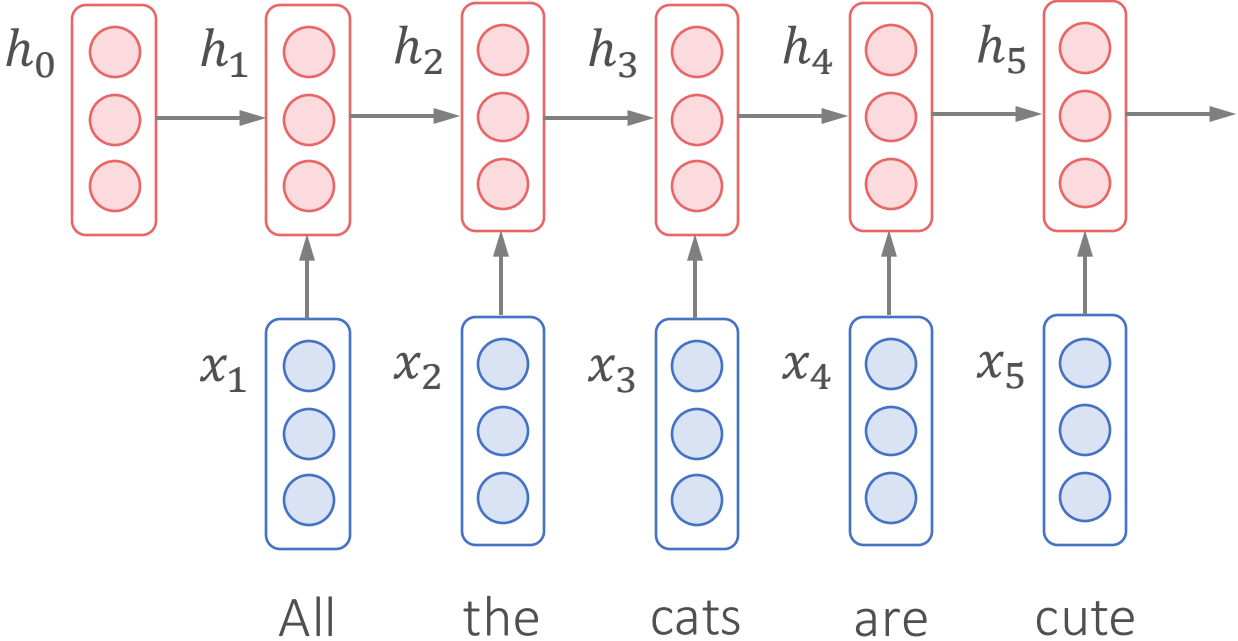
	Alice	treats	Bob	well
Dimension 1	0.7	2.7	-0.1	-5.7
Dimension 2	8.6	-3.9	6.7	-9.8
Dimension 3	-2.4	-5.6	1.5	-1.6
Dimension 4	2.3	1.1	2.0	-1.0



$$\mathcal{L}_{CE}(y, \tilde{y}) = - \sum_{c=0}^C y_c \log P(y = c | \mathbf{x})$$

# Recap: Recurrent Neural Network (RNN)

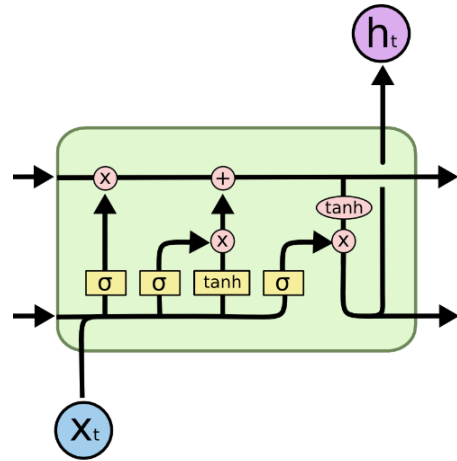
$$h_t = \sigma(W h_{t-1} + U x_t + b)$$



$$\mathcal{L}_{CE}(y, \tilde{y}) = - \sum_{c=0}^C y_c \log P(y = c | \mathbf{x})$$

# Recap: Long Short-Term Memory and Gated Recurrent Units

LSTM



$$i_t = \sigma(W^{(i)}h_{t-1} + U^{(i)}x_t + b^{(i)})$$

$$f_t = \sigma(W^{(f)}h_{t-1} + U^{(f)}x_t + b^{(f)})$$

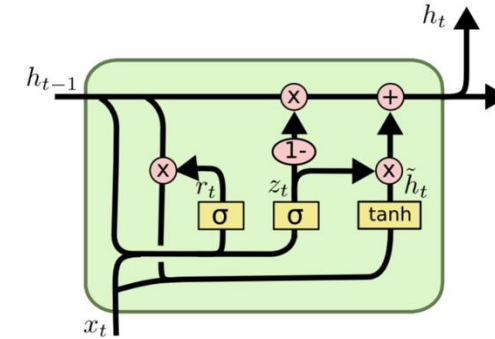
$$o_t = \sigma(W^{(o)}h_{t-1} + U^{(o)}x_t + b^{(o)})$$

$$\tilde{C}_t = \tanh(W^{(c)}h_{t-1} + U^{(c)}x_t + b^{(c)})$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

$$h_t = o_t * \tanh(C_t)$$

GRU



$$r_t = \sigma(W^{(r)}h_{t-1} + U^{(r)}x_t + b^{(r)})$$

$$z_t = \sigma(W^{(z)}h_{t-1} + U^{(z)}x_t + b^{(z)})$$

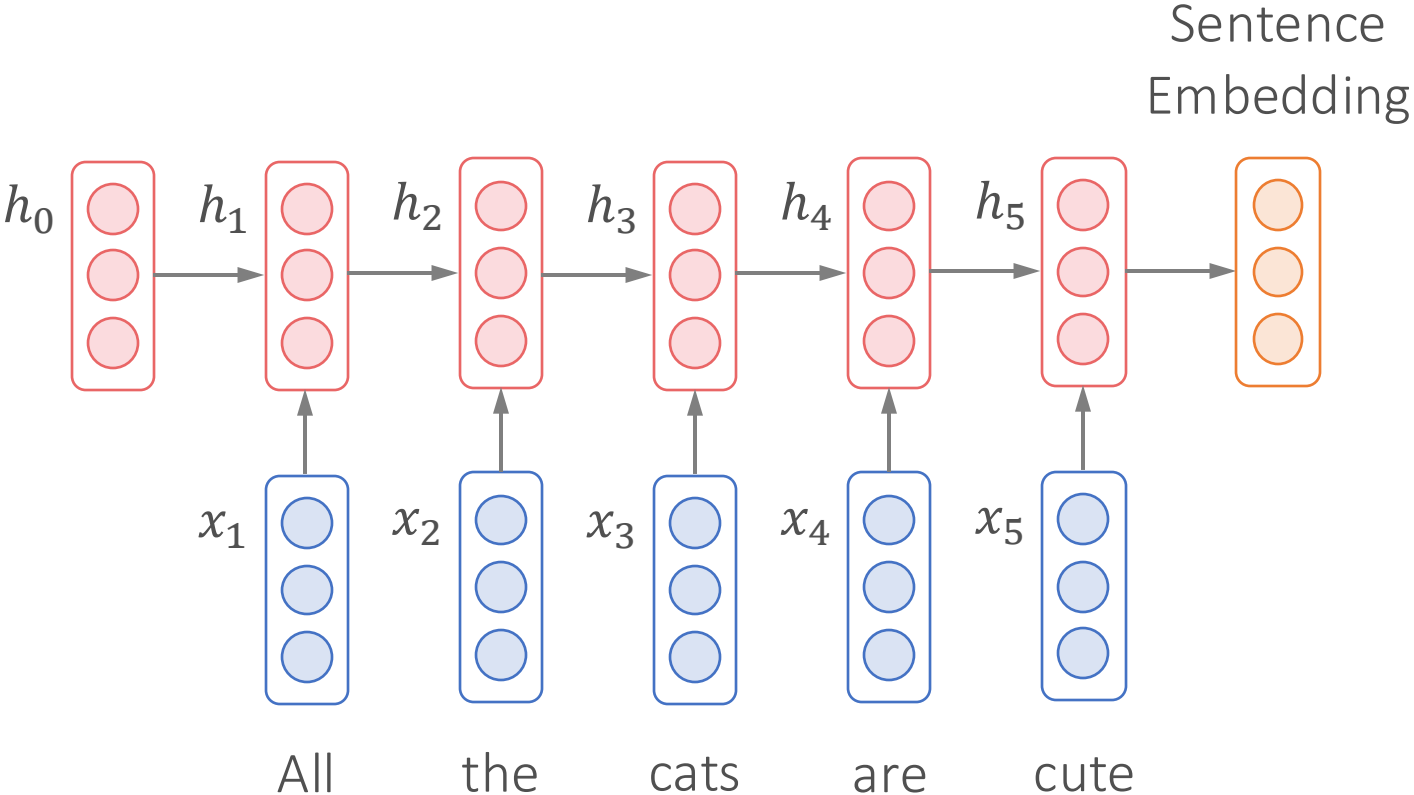
$$\tilde{h}_t = \tanh(W(r_t * h_{t-1}) + Ux_t + b)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

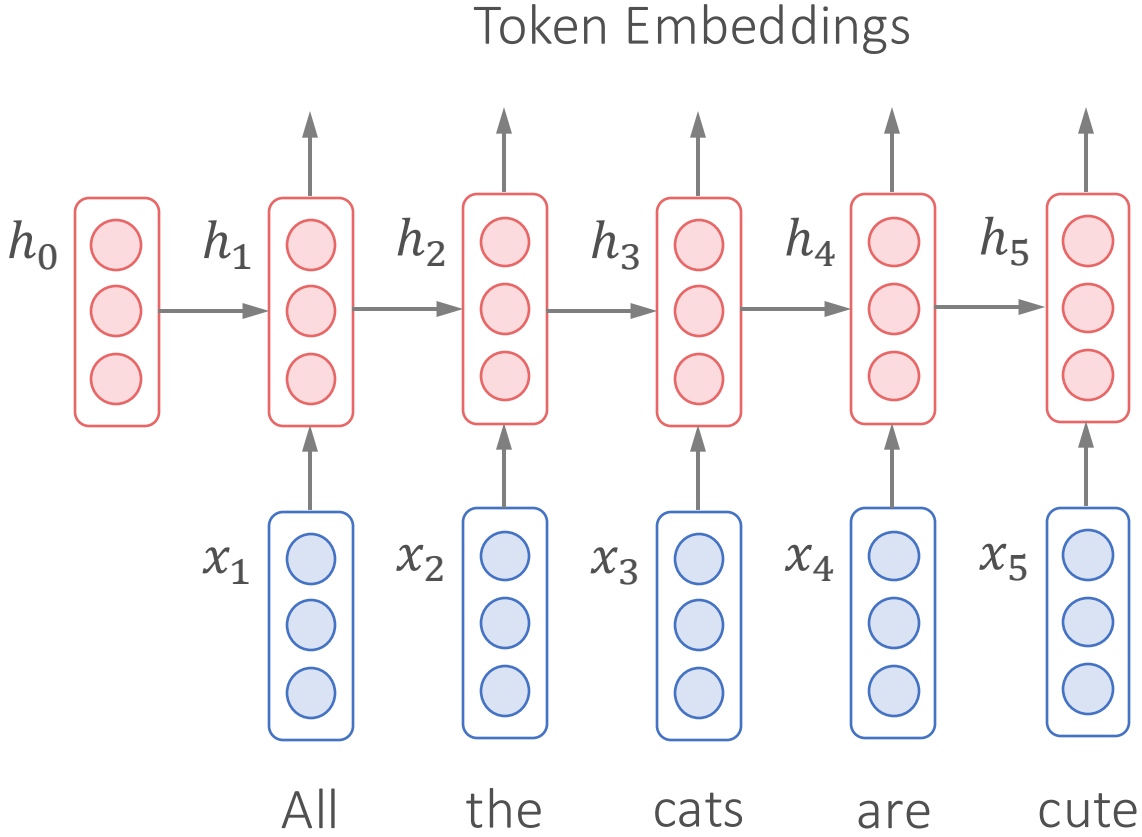
# RNN is Flexible

- Can be used for both **classification** and **generation**
  - Encoder
  - Decoder
  - Encoder-decoder

# RNN as Sentence-Level Encoder



# RNN as Token-Level Encoder



The embeddings are contextualized

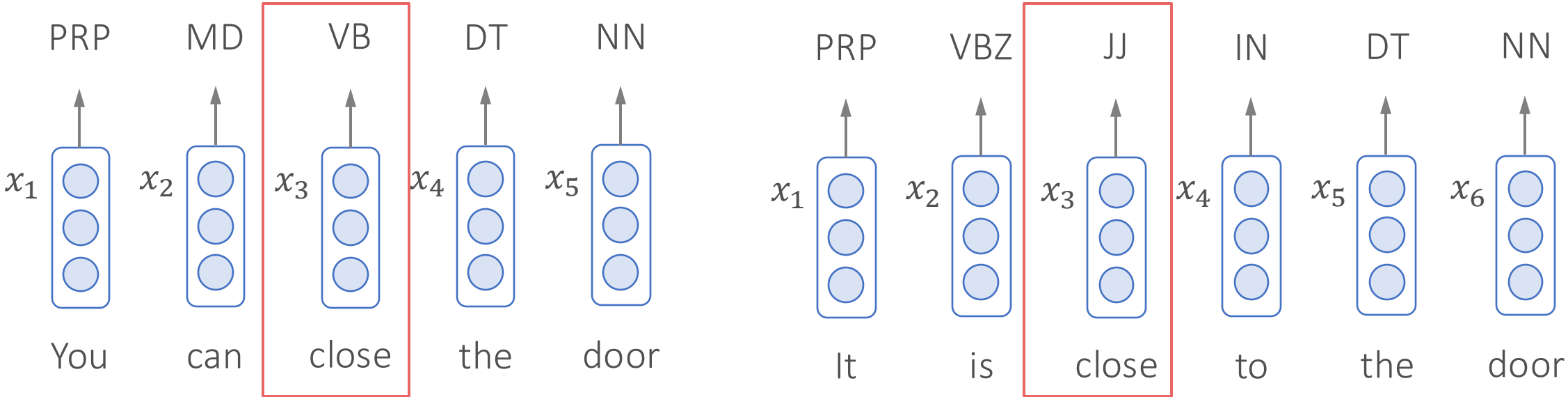
# Part-of-Speech (POS) Tagging

*You can close the door*  
PRP MD VB DT NN

*It is close to the door*  
PRP VBZ JJ IN DT NN

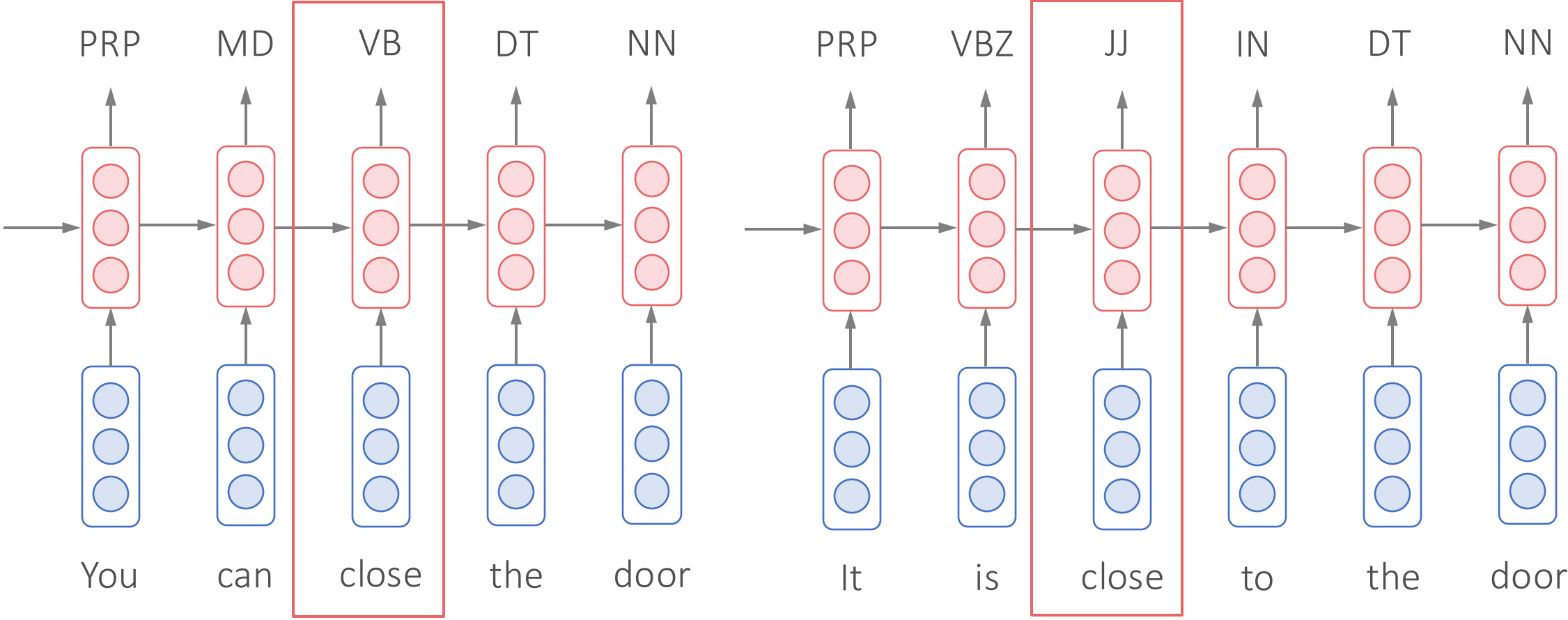
It's a structured prediction problem

# POS Tagging with Word Embeddings





# POS Tagging with Sequential Labeling



# Named Entity Recognition

*John* went to *New York City* to visit *Kuan-Hao Huang*

Entity

Entity

Entity

## BIO Sequence

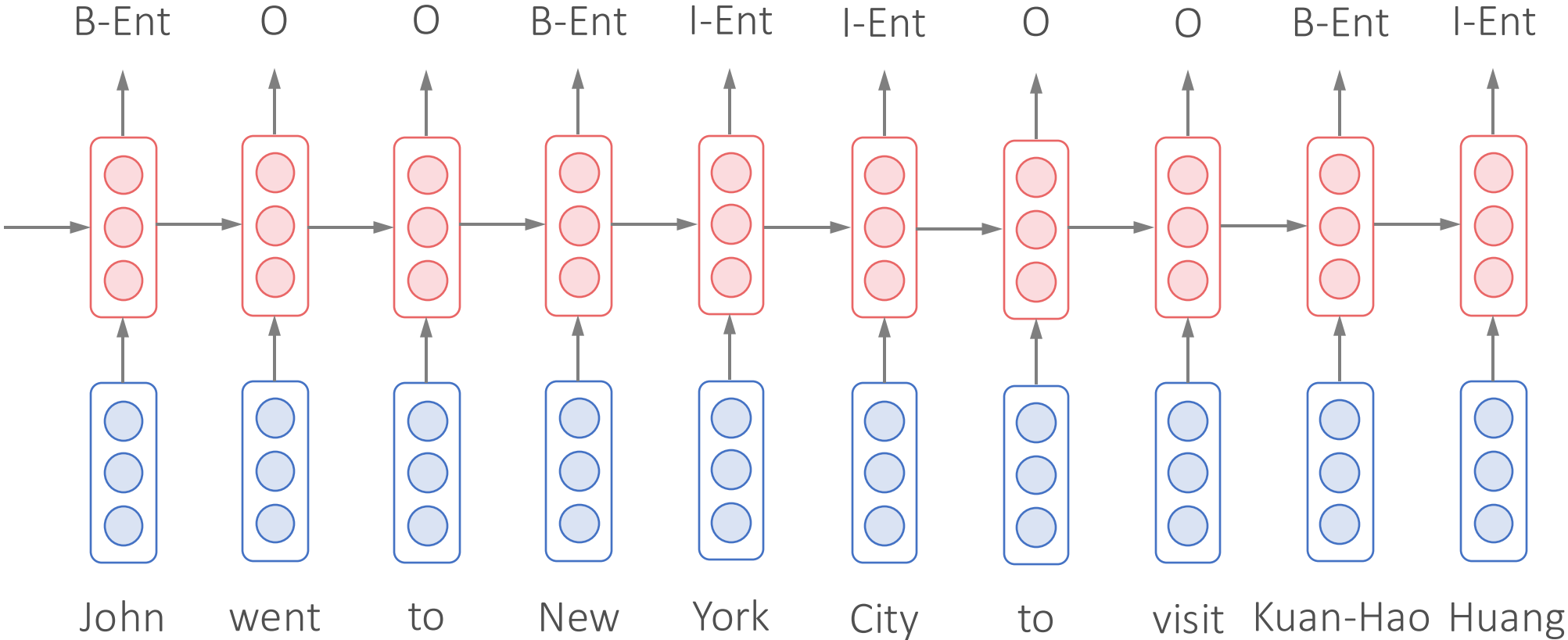
*John*   *went*   *to*   *New*   *York*   *City*   *to*   *visit*   *Kuan-Hao*   *Huang*

B-Entity   Other   Other   B-Entity   I-Entity   I-Entity   Other   Other   B-Entity   I-Entity

B-Entity: Begin of an entity span, I-Entity: Inside of an entity span

It's a structured prediction problem

# Named Entity Recognition as Sequential Labeling



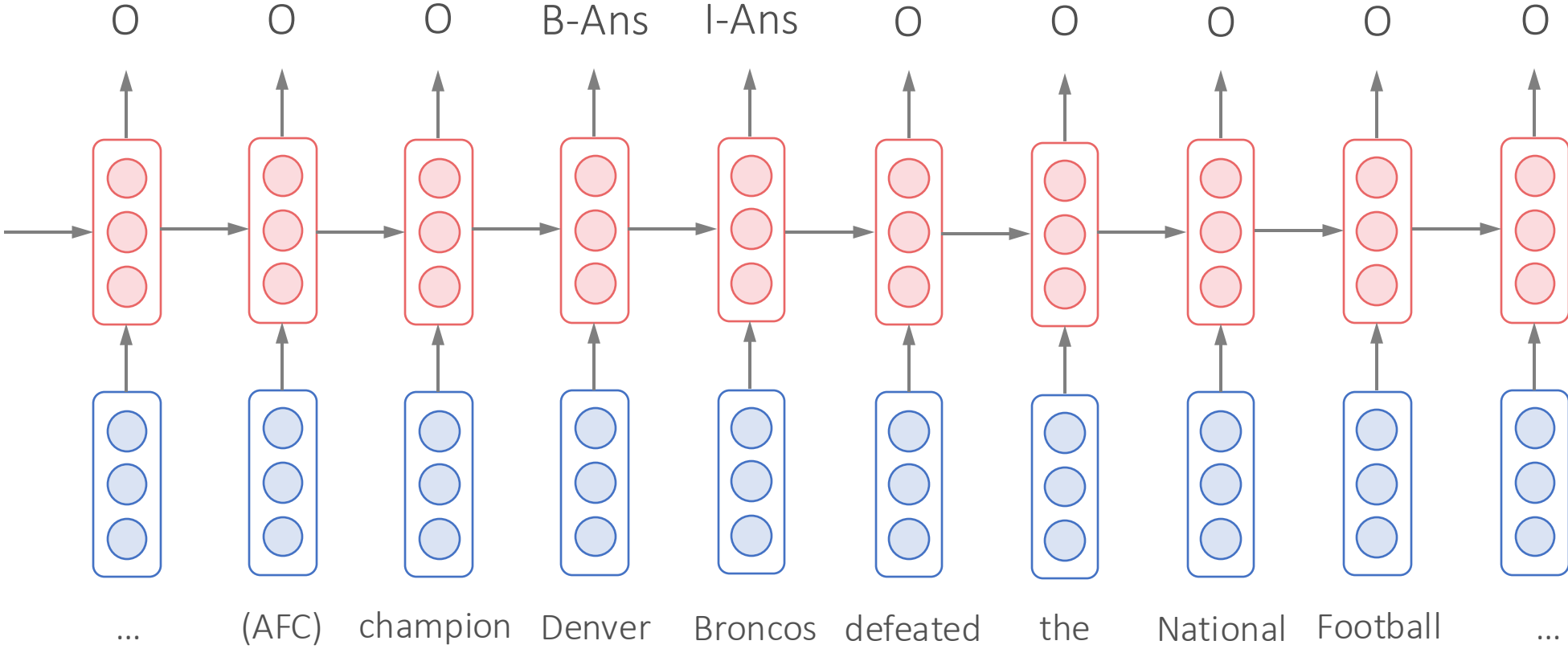
# Extractive Question Answering

Super Bowl 50 was an American football game to determine the champion of the National Football League (NFL) for the 2015 season. The American Football Conference (AFC) champion Denver Broncos defeated the National Football Conference (NFC) champion Carolina Panthers 24–10 to earn their third Super Bowl title. The game was played on February 7, 2016, at Levi's Stadium in the San Francisco Bay Area at Santa Clara, California. As this was the 50th Super Bowl, the league emphasized the "golden anniversary" with various gold-themed initiatives, as well as temporarily suspending the tradition of naming each Super Bowl game with Roman numerals (under which the game would have been known as "Super Bowl L"), so that the logo could prominently feature the Arabic numerals 50.

**Question:** Which NFL team represented the AFC at Super Bowl 50?

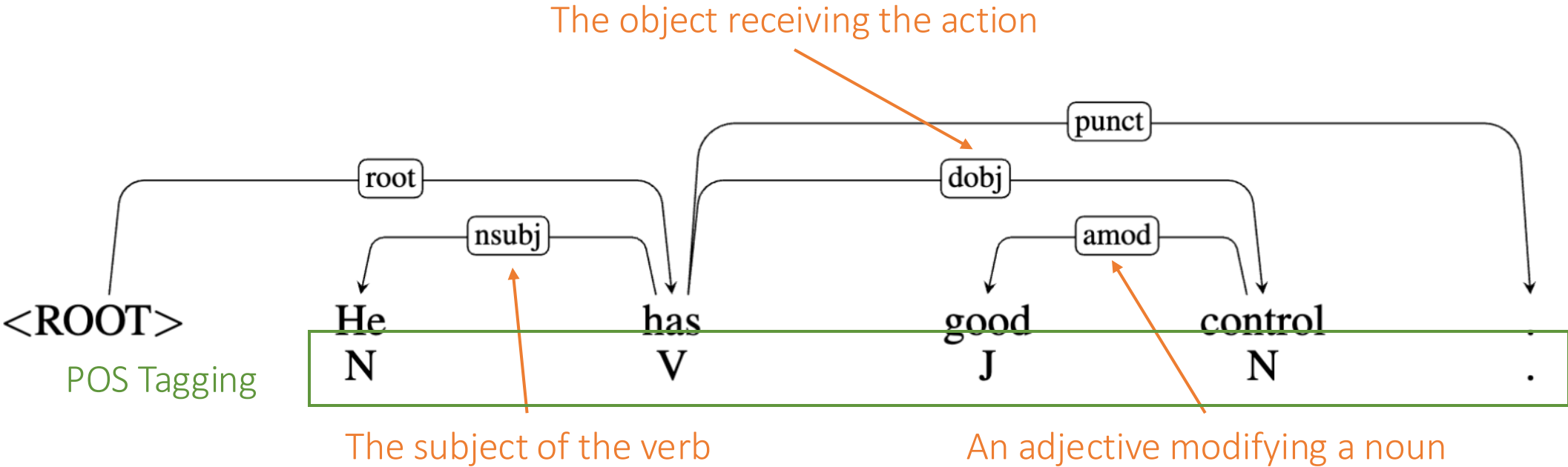
**Answer:** Denver Broncos

# Extractive Question Answering as Sequential Labeling



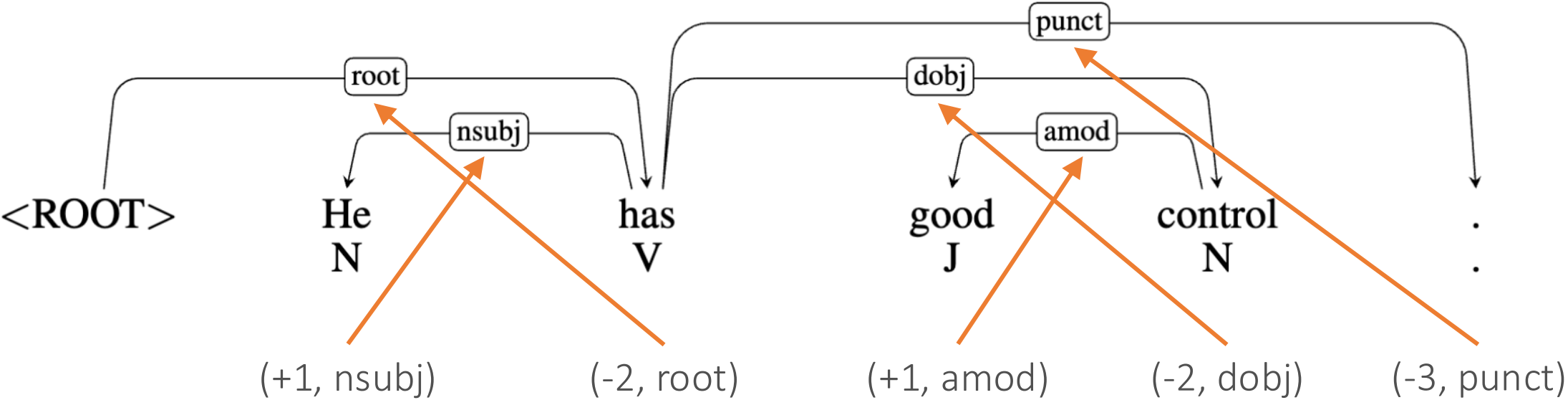
# Dependency Parsing

- Identify **dependency relations** between words
  - A **tree** structure

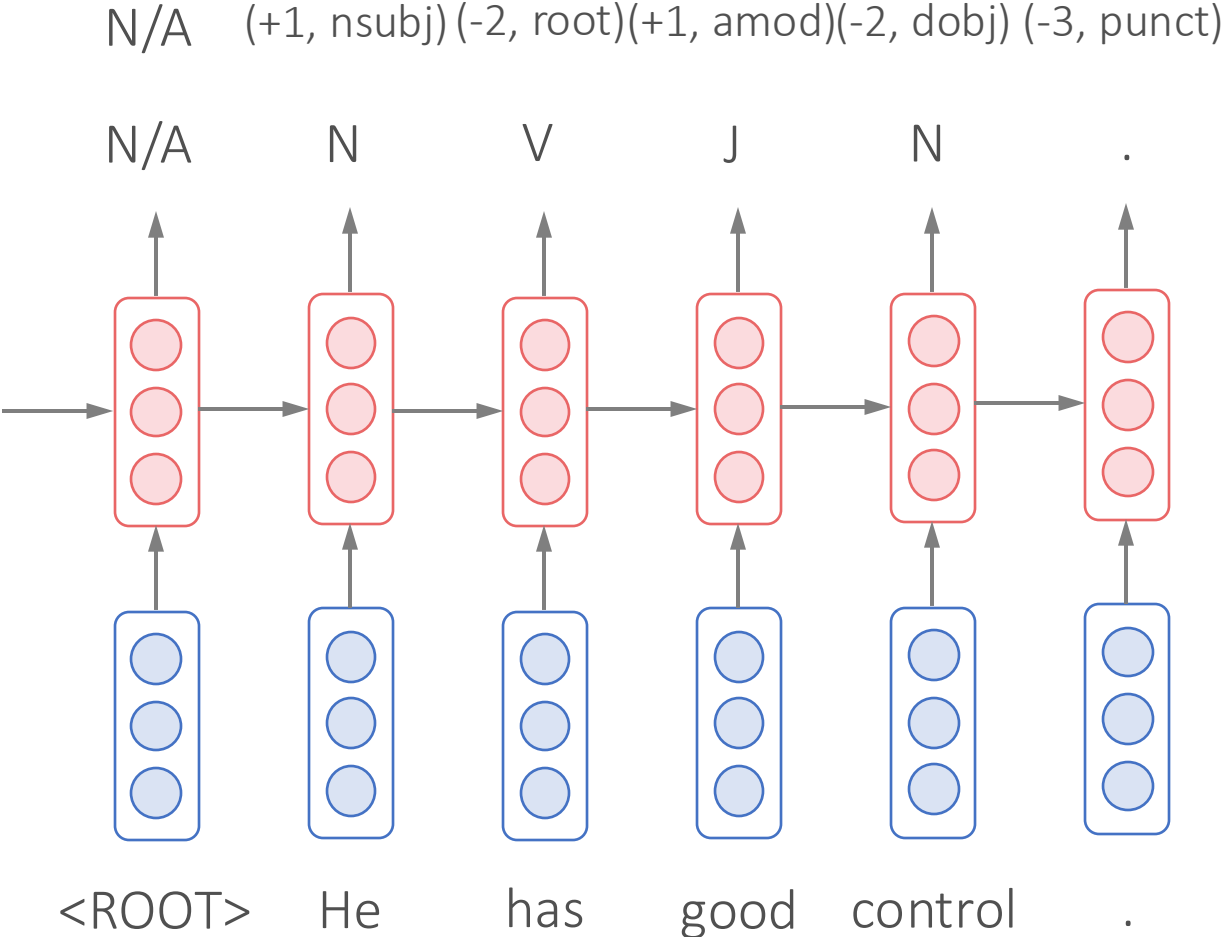


# Dependency Parsing

- Convert tree to sequential labels  $(p_i, l_i)$ 
  - $p_i$ : relative offset between the word and its head word
  - $l_i$ : dependency relation



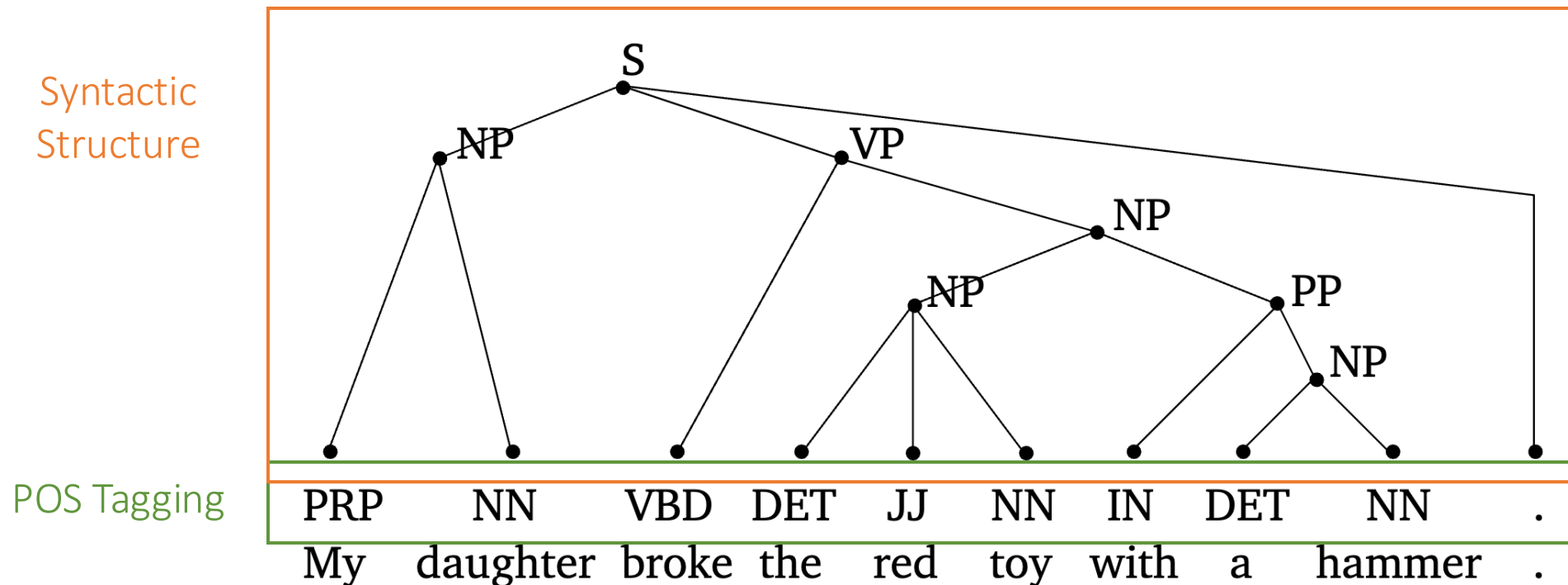
# Dependency Parsing as Sequential Labeling





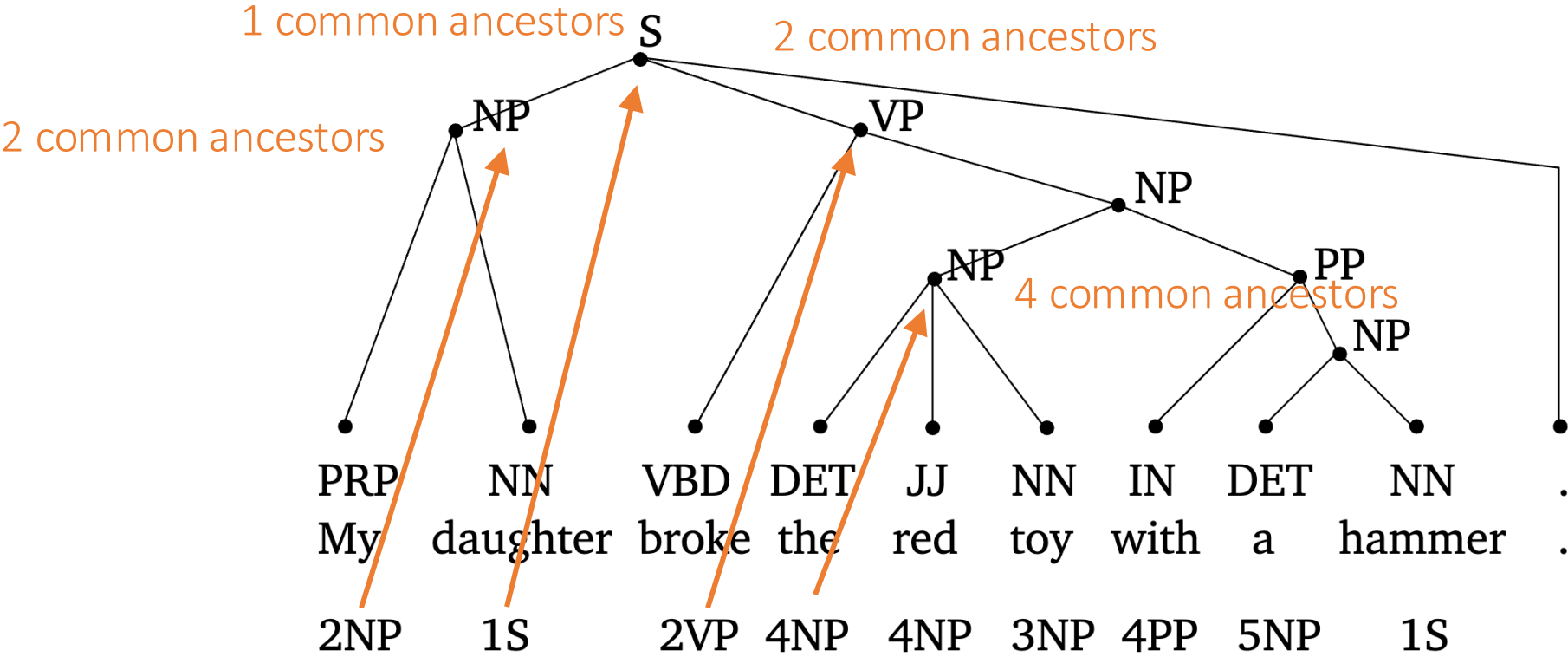
# Constituency Parsing

- Analyze the **syntactic** structure of a sentence
  - Break a sentence down into its **constituent** parts
  - Hierarchical **tree** structure

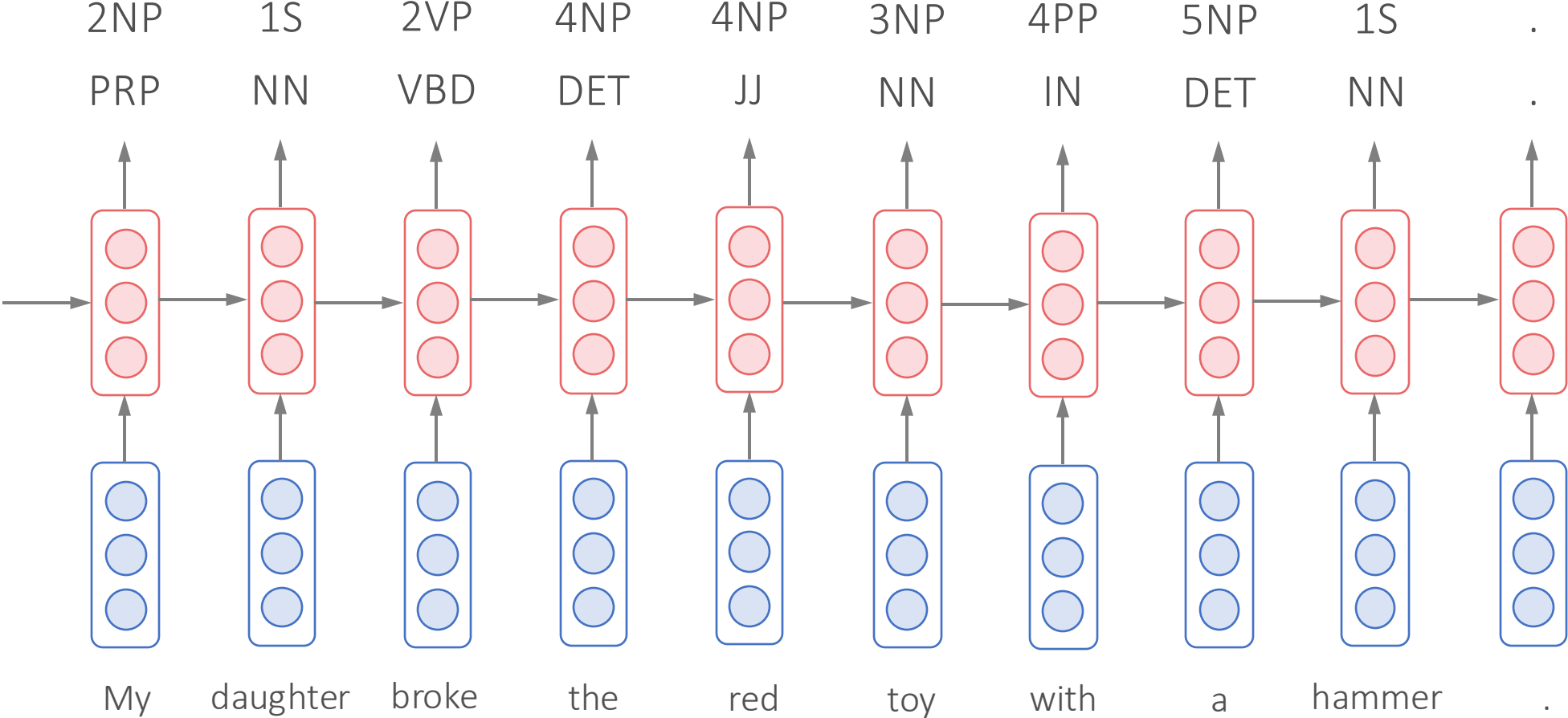


# Constituency Parsing

- Convert tree to sequential labels  $(n_i, c_i)$ 
  - $n_i$ : the number of common ancestors between  $w_i$  and  $w_{i+1}$
  - $c_i$ : the nonterminal symbol at the lowest common ancestor

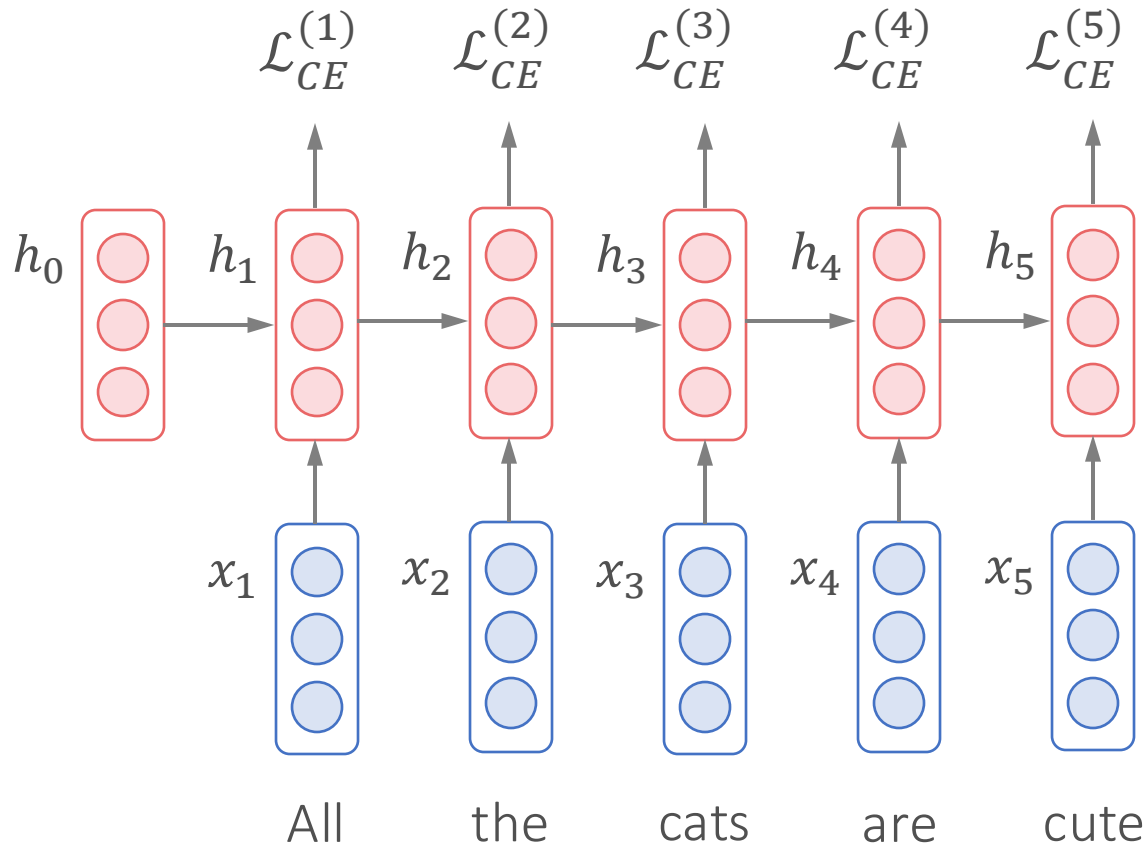


# Constituency Parsing as Sequential Labeling



# Sequential Labeling

- A sequence of **dependent** classification

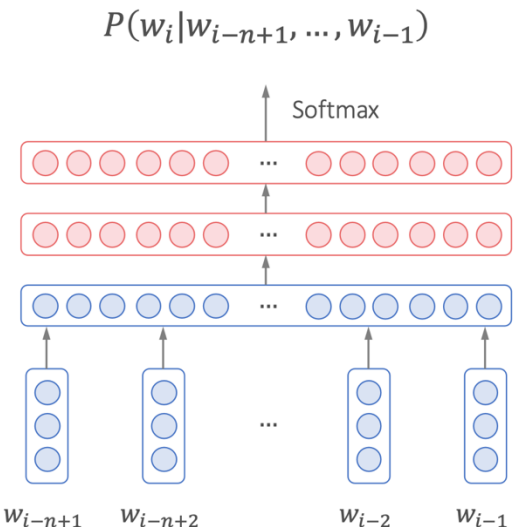


$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N \mathcal{L}_{CE}^{(i)}$$

# RNN as Decoder (Generator)

- RNN Language Modeling
  - Generation is a sequence of word classification

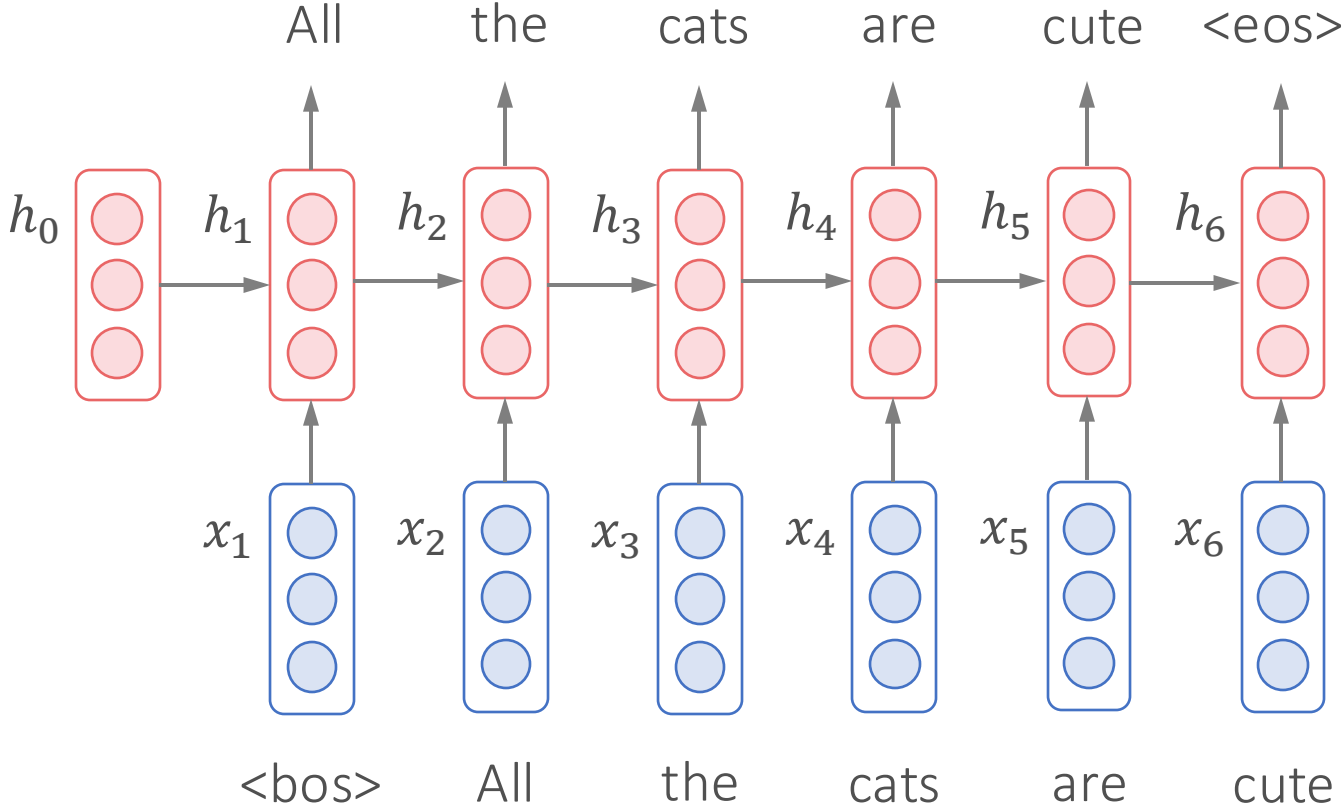
$$\begin{aligned}
 P(w_1, w_2, w_3, \dots, w_l) &= P(w_1)P(w_2, w_3, \dots, w_l|w_1) \\
 &= P(w_1)P(w_2|w_1)(w_3, \dots, w_l|w_1, w_2) \\
 &= P(w_1)P(w_2|w_1)P(w_3|w_1, w_2)(w_4, \dots, w_l|w_1, w_2, w_3) \\
 &= \prod_{i=1}^l P(w_i|w_1, w_2, \dots, w_{i-1})
 \end{aligned}$$



Neural language models with context window

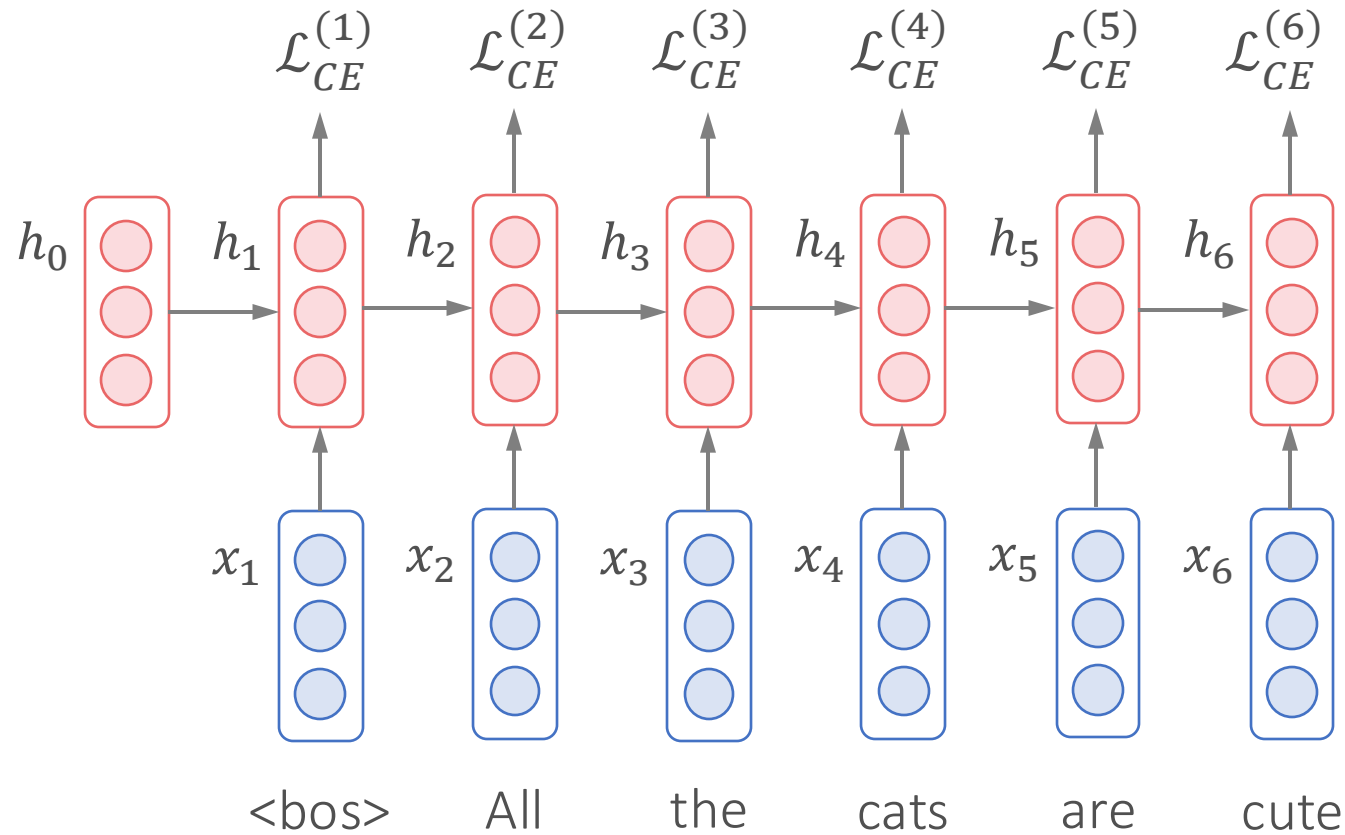
# RNN as Decoder (Generator)

- RNN Language Modeling
  - Generation is a sequence of word classification



# RNN as Decoder (Generator)

- RNN Language Modeling
  - Generation is a sequence of word classification



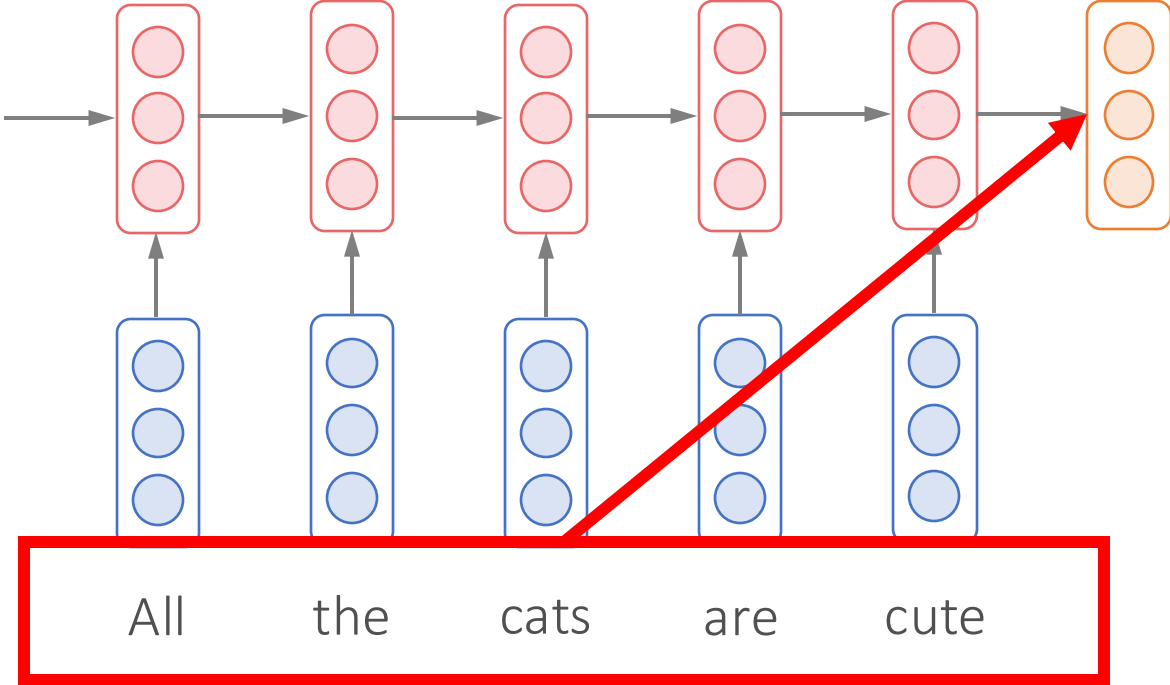
$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N \mathcal{L}_{CE}^{(i)}$$

# Encoder vs. Decoder

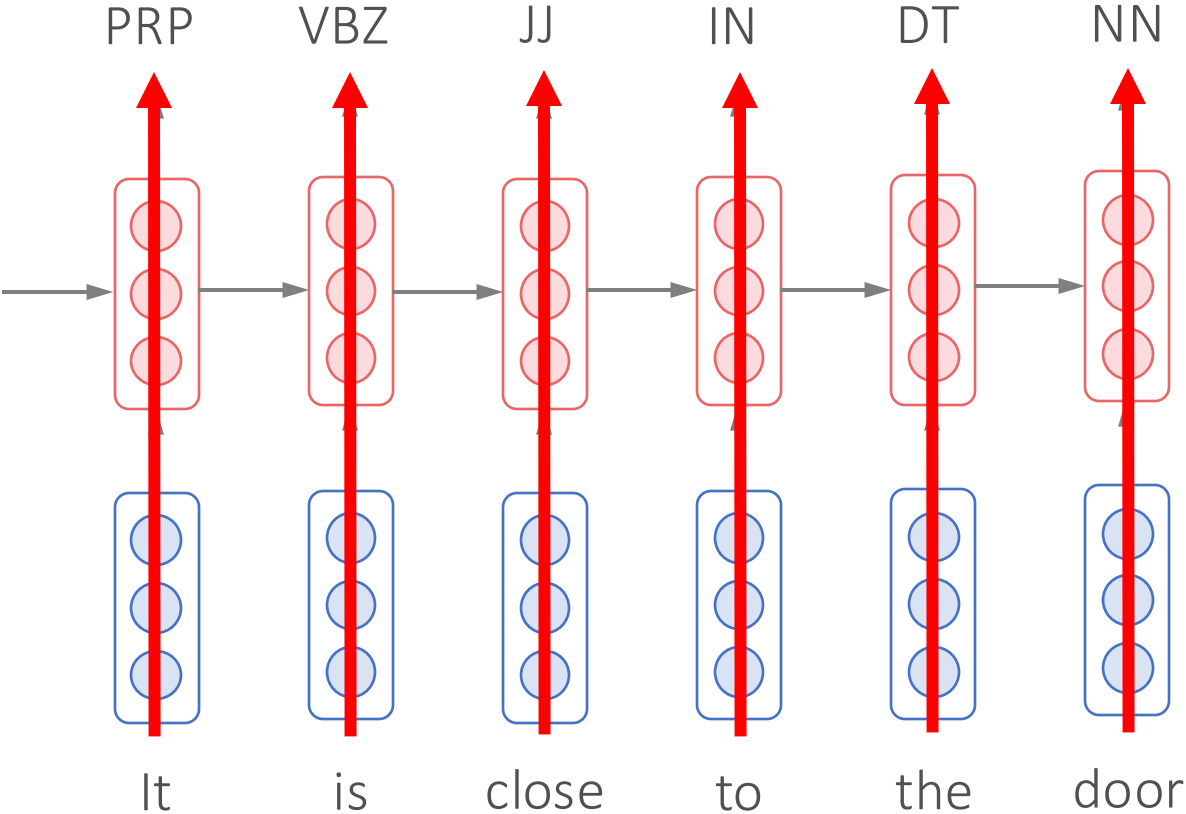
- Encoder
  - Focus more on **representations** and **understanding**
- Decoder
  - Focus on **generation**



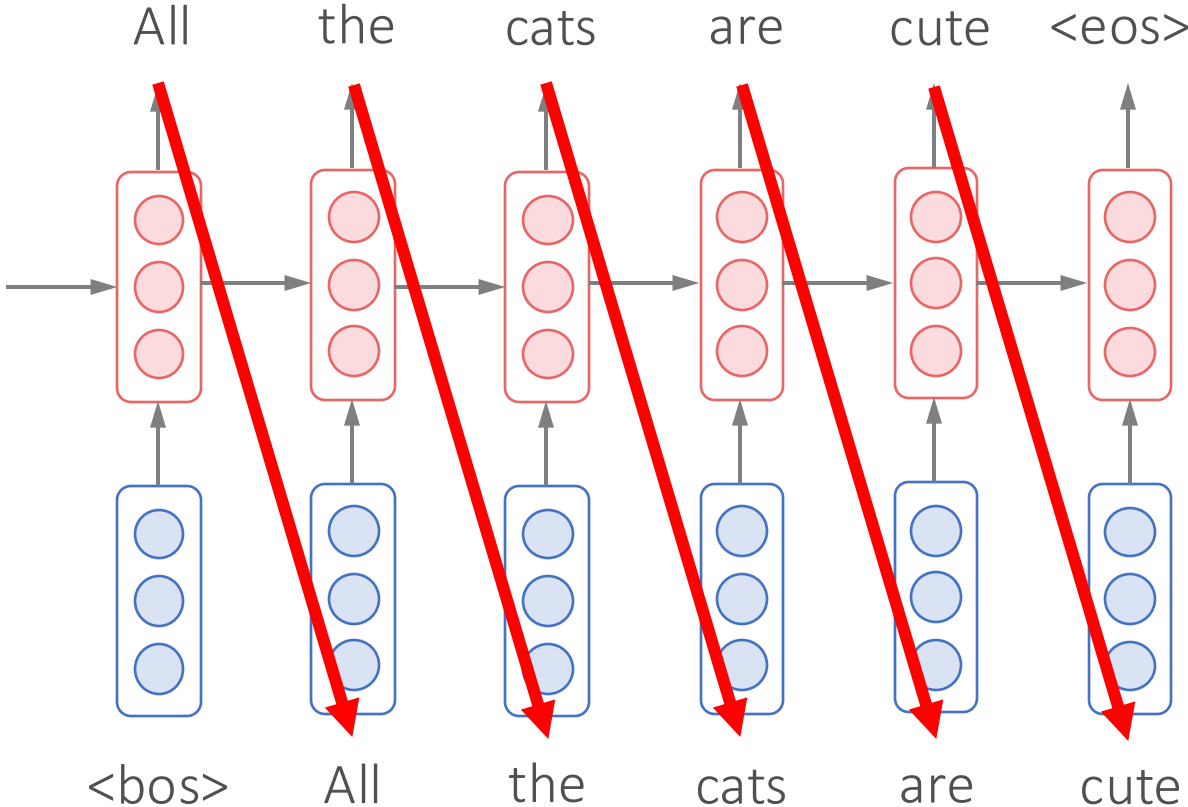
# Encoder



# Encoder

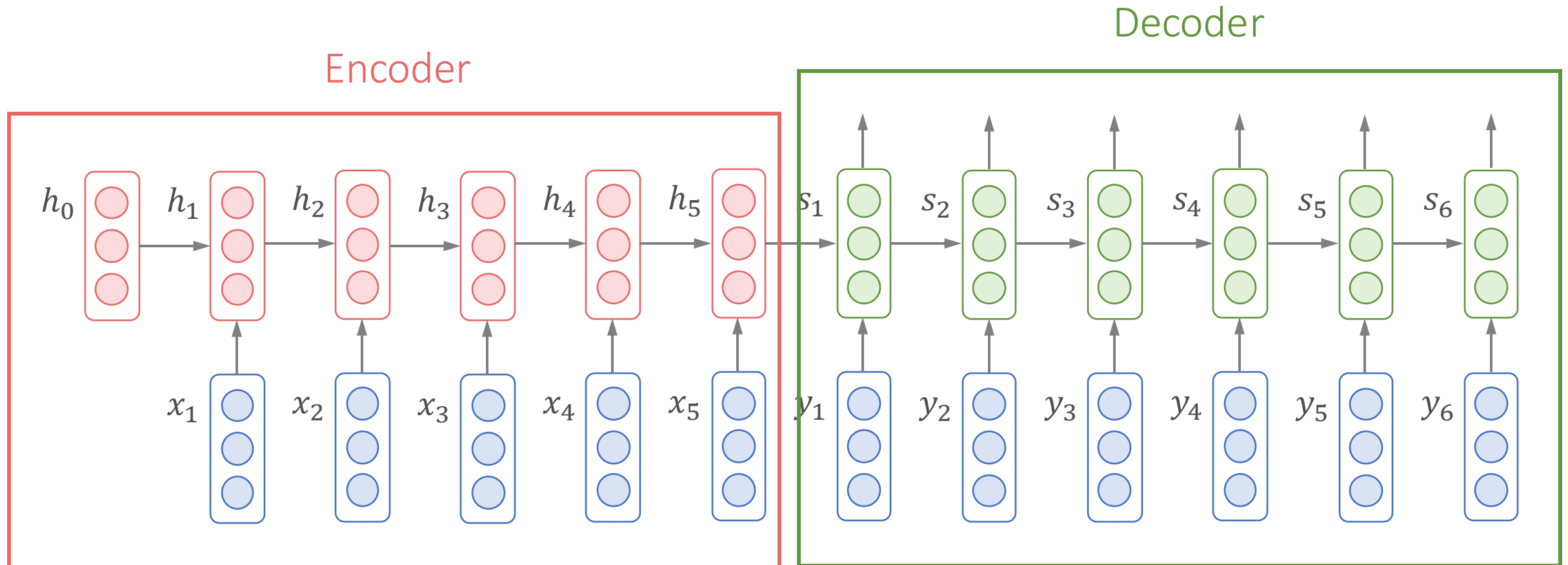


# Decoder



# Sequence-to-Sequence Models (Seq2Seq)





- When we need understanding and generation at the same time




# Sequence-to-Sequence Tasks


English - detected ↔ French

hello world × Bonjour le monde

I think I have an idea that should sort of improve campaign performance.

 Tone Suggestion

**Confident** 

I have an idea that should improve campaign performance.

**Rephrase** Dismiss

Provided proper attribution is provided, Google hereby grants permission to reproduce the tables and figures in this paper solely for use in journalistic or scholarly works.

**Attention Is All You Need**

Ashish Vaswani\*  
Google Brain  
avaswani@google.com

Noam Shazeer\*  
Google Brain  
noam@google.com

Niki Parmar\*  
Google Research  
nikip@google.com

Jakob Uszkoreit\*  
Google Research  
usz@google.com

Llion Jones\*  
Google Research  
llion@google.com

Aidan N. Gomez\*<sup>†</sup>  
University of Toronto  
aidan@cs.toronto.edu

Lukasz Kaiser\*  
Google Brain  
lukasz@kaiser@google.com


Illia Polosukhin\*<sup>†</sup>  
illia.polosukhin@gmail.com

**Abstract**

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

\*Equal contribution. Listing order is random. Jakob proposed replacing RNNs with self attention and started the effort to evaluate this idea. Ashish, with Illia, designed and implemented the first Transformer models and has been crucially involved in every aspect of this work. Noam proposed scaled dot-product attention, multi-head attention and the parameter-free position representation and became the other person involved in nearly every detail. Niki designed, implemented, tuned and evaluated countless model variants in our original codebase and tensor2tensor. Llion also experimented with novel model variants, was responsible for our initial codebase, and efficient inference and visualizations. Lukasz and Aidan spent countless long days designing various parts of and implementing tensor2tensor, replacing our earlier codebase, greatly improving results and massively accelerating our research.

<sup>†</sup>Work performed while at Google Brain.

 Summary

The document titled "Attention Is All You Need" introduces the Transformer model, a network architecture based solely on attention mechanisms, eliminating the need for recurrent or convolutional neural networks in sequence transduction tasks. The Transformer model achieves superior performance in machine translation tasks, demonstrating improved quality, parallelizability, and reduced training time compared to existing models. The key points and arguments presented in the document are as follows:

- The dominant sequence transduction models rely on complex recurrent or convolutional neural networks with an encoder-decoder structure and attention mechanisms.
- The Transformer model proposes a new architecture based solely on attention mechanisms, eliminating the need for recurrence and convolutions.
- Experiments show that the Transformer model outperforms existing models in machine translation tasks, achieving state-of-the-art results with reduced training time.
- The model utilizes self-attention to compute representations of input and output sequences, allowing for more parallelization and global dependencies.
- The Transformer model consists of stacked self-attention and fully connected layers for both the encoder and decoder, enabling efficient sequence transduction.
- Multi-Head Attention is employed to jointly attend to information from different representation subspaces at different positions, enhancing the model's performance.

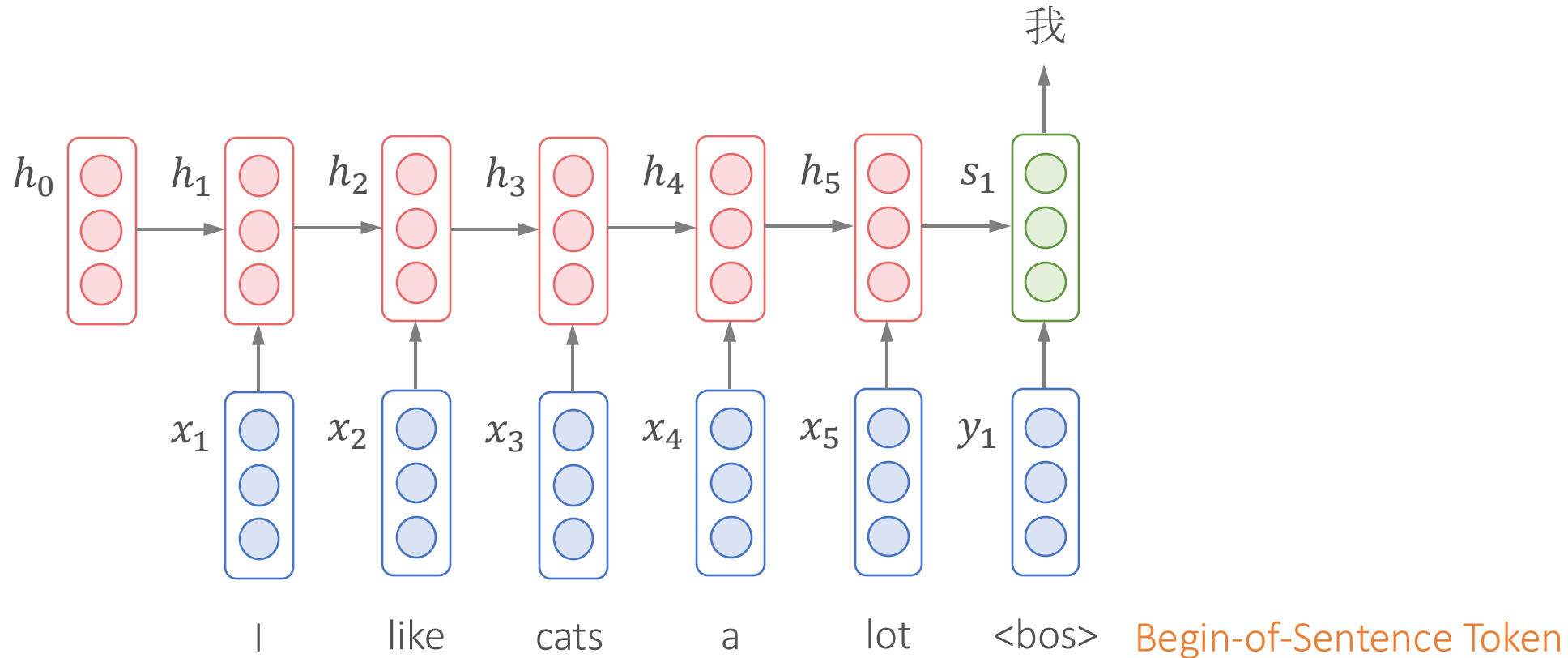
**Key Points:**

- Transformer model introduces a network architecture based solely on attention

# Translation

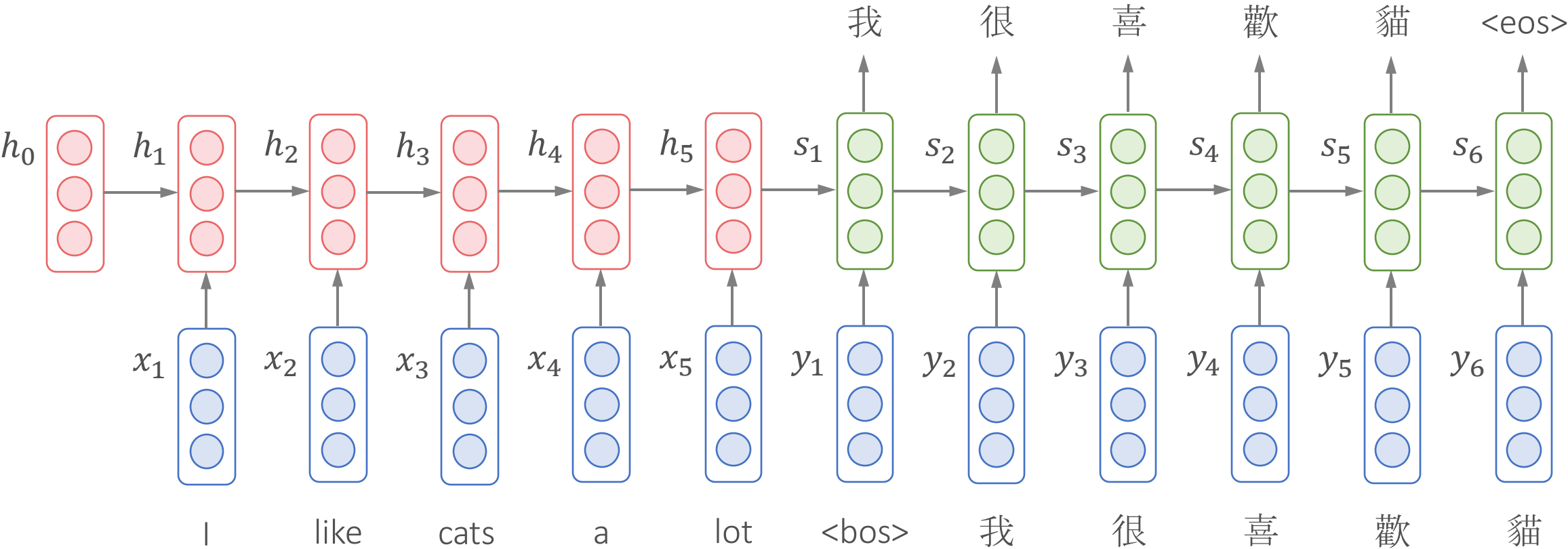
- Translate English to Chinese

Classification over the whole vocabulary



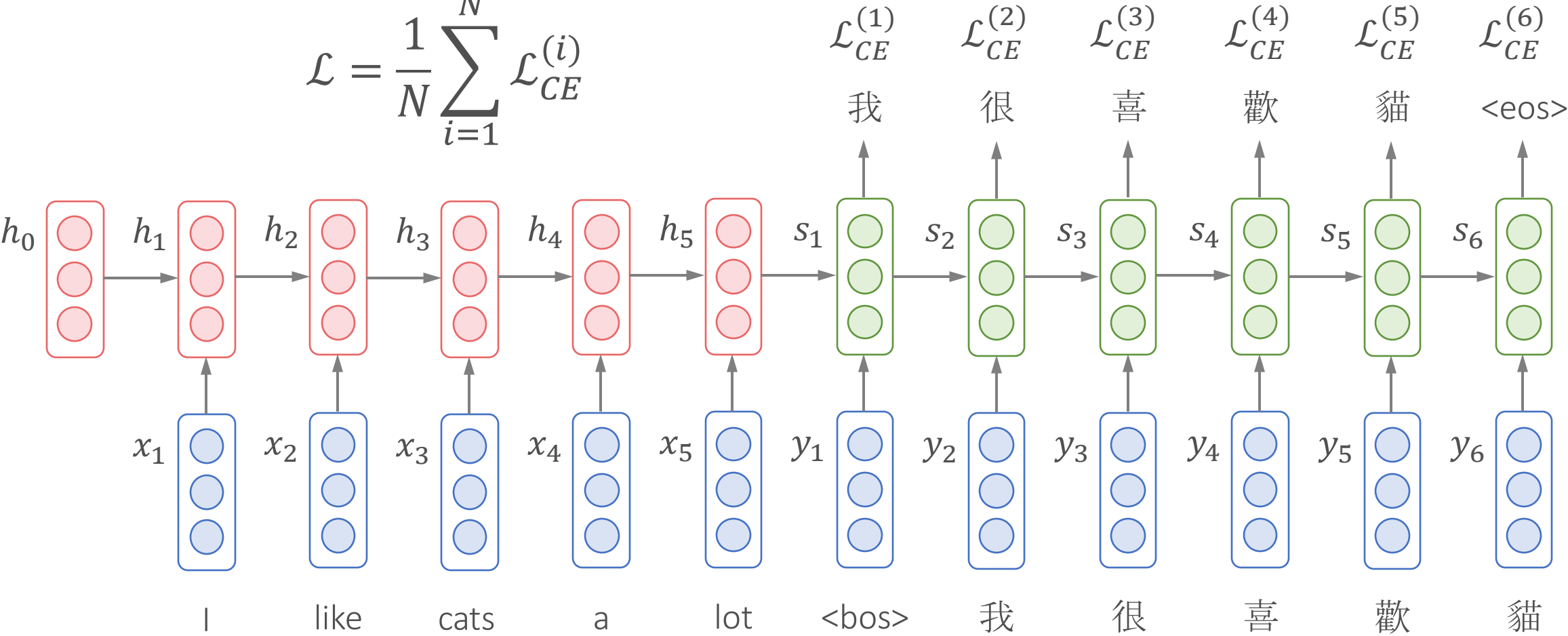
# Translation

- Translate English to Chinese



# Sequence-to-Sequence Model Loss

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N \mathcal{L}_{CE}^{(i)}$$



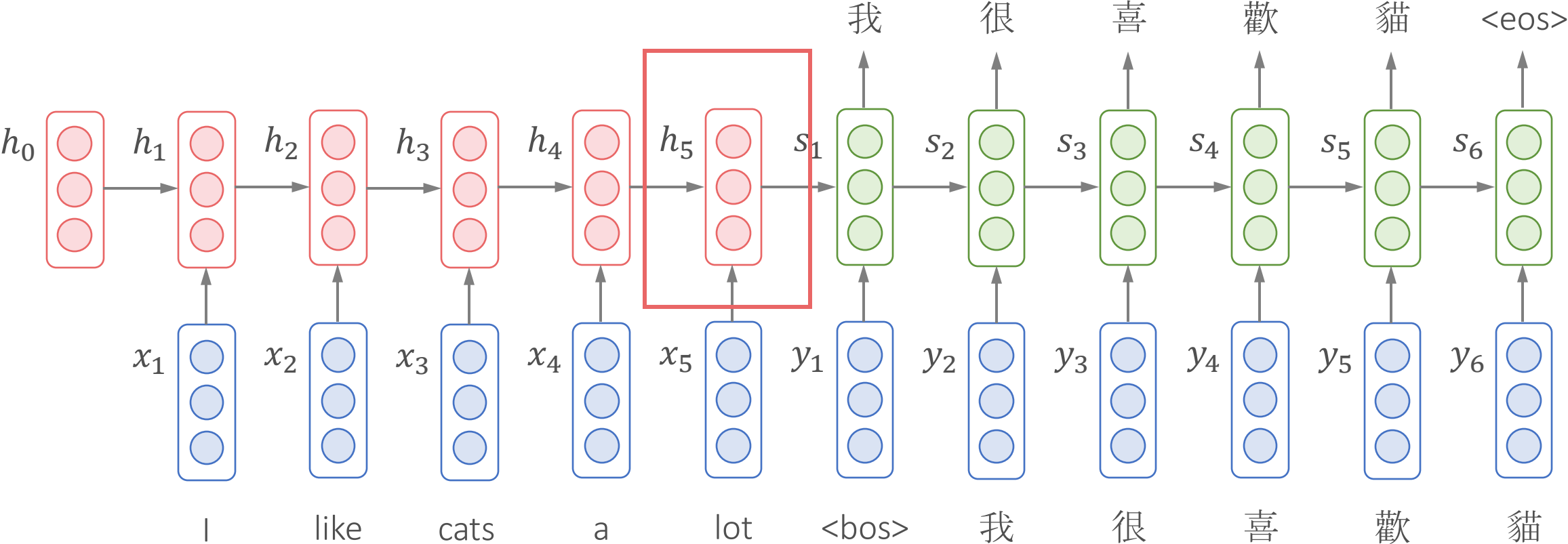


# Decoder-Only Models vs. Seq2Seq Models

- Decoder-only models with prompting
  - Continue writing
- Seq2Seq models
  - Encode first, then generate
- The difference becomes larger when we talk about Transformers!

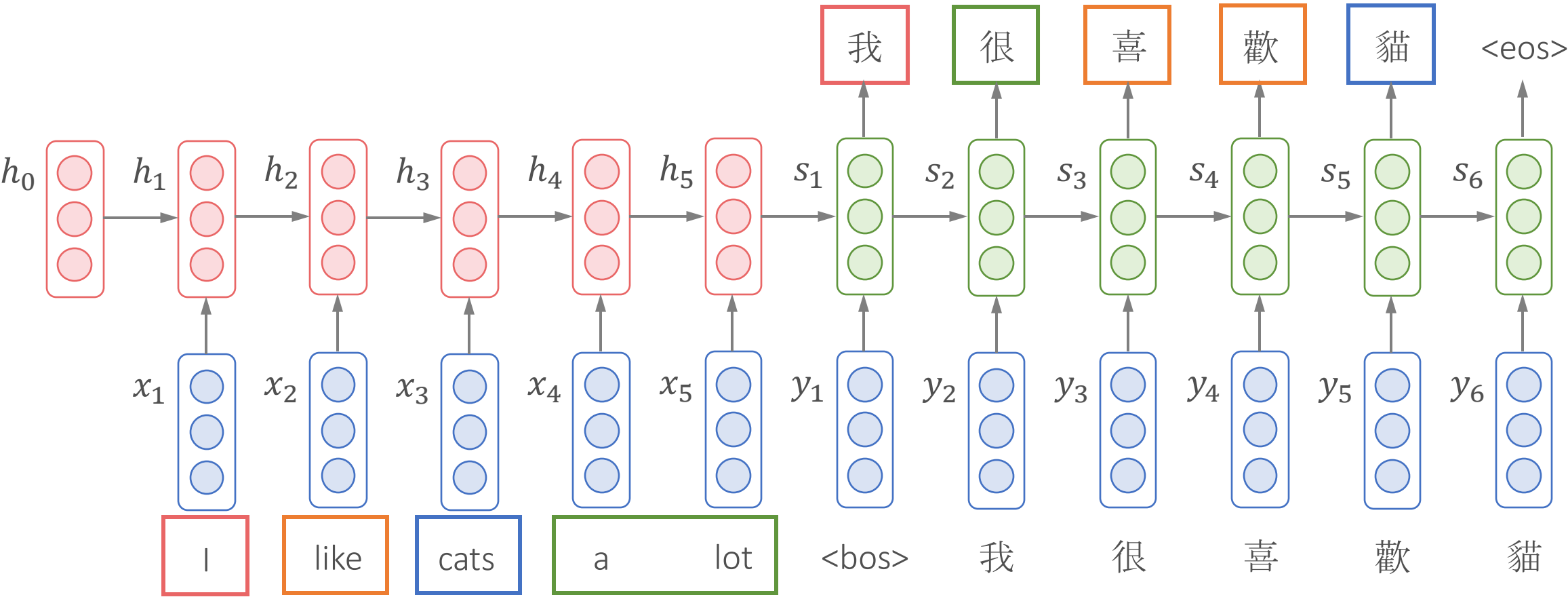
# Seq2Seq: Bottleneck

- A single vector needs to capture **all the information** about source sentence
- Longer sequences can still lead to **vanishing gradients**



# Focus on A Particular Part When Decoding

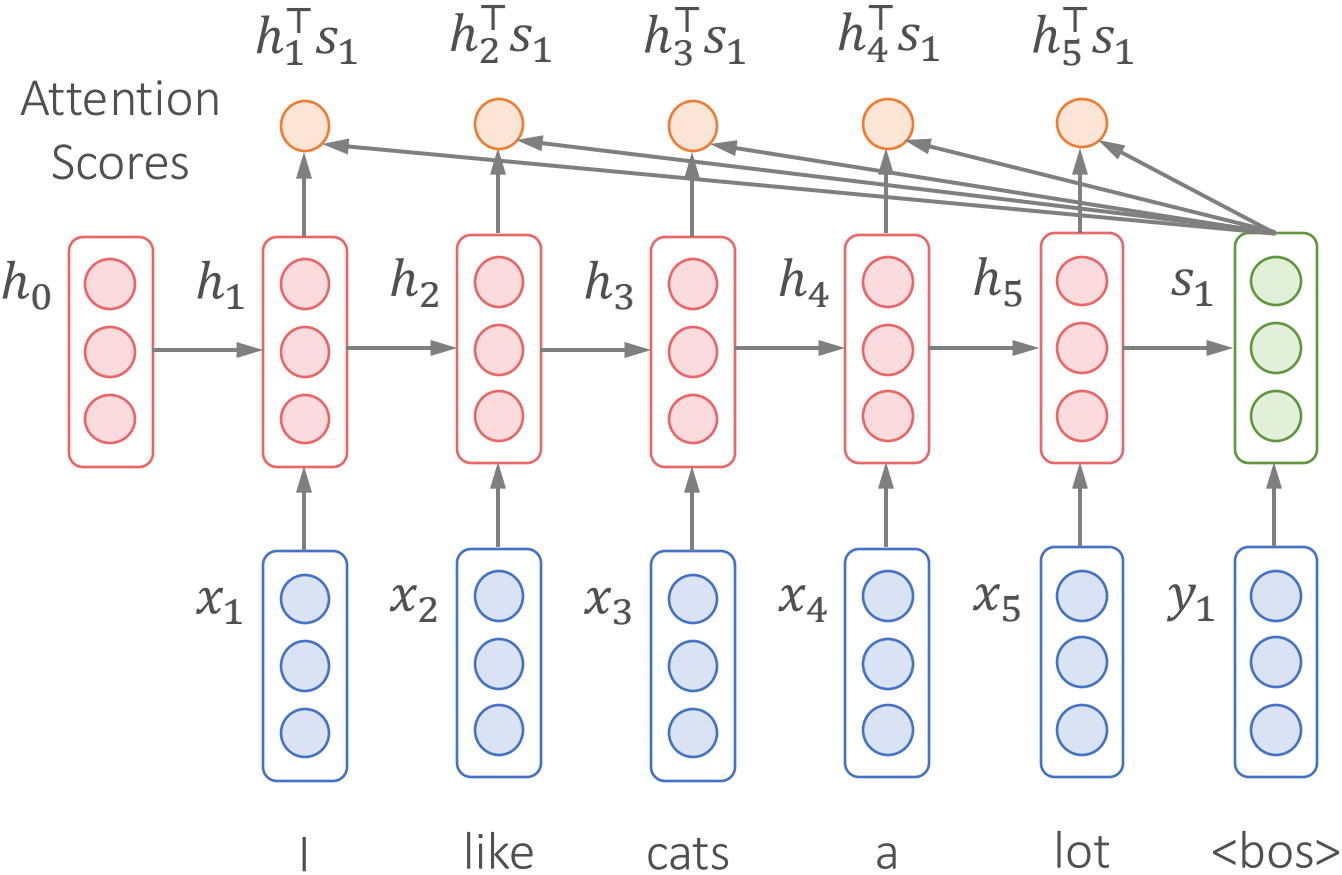
- Each token classification requires different part of information from source sentence



# Attention

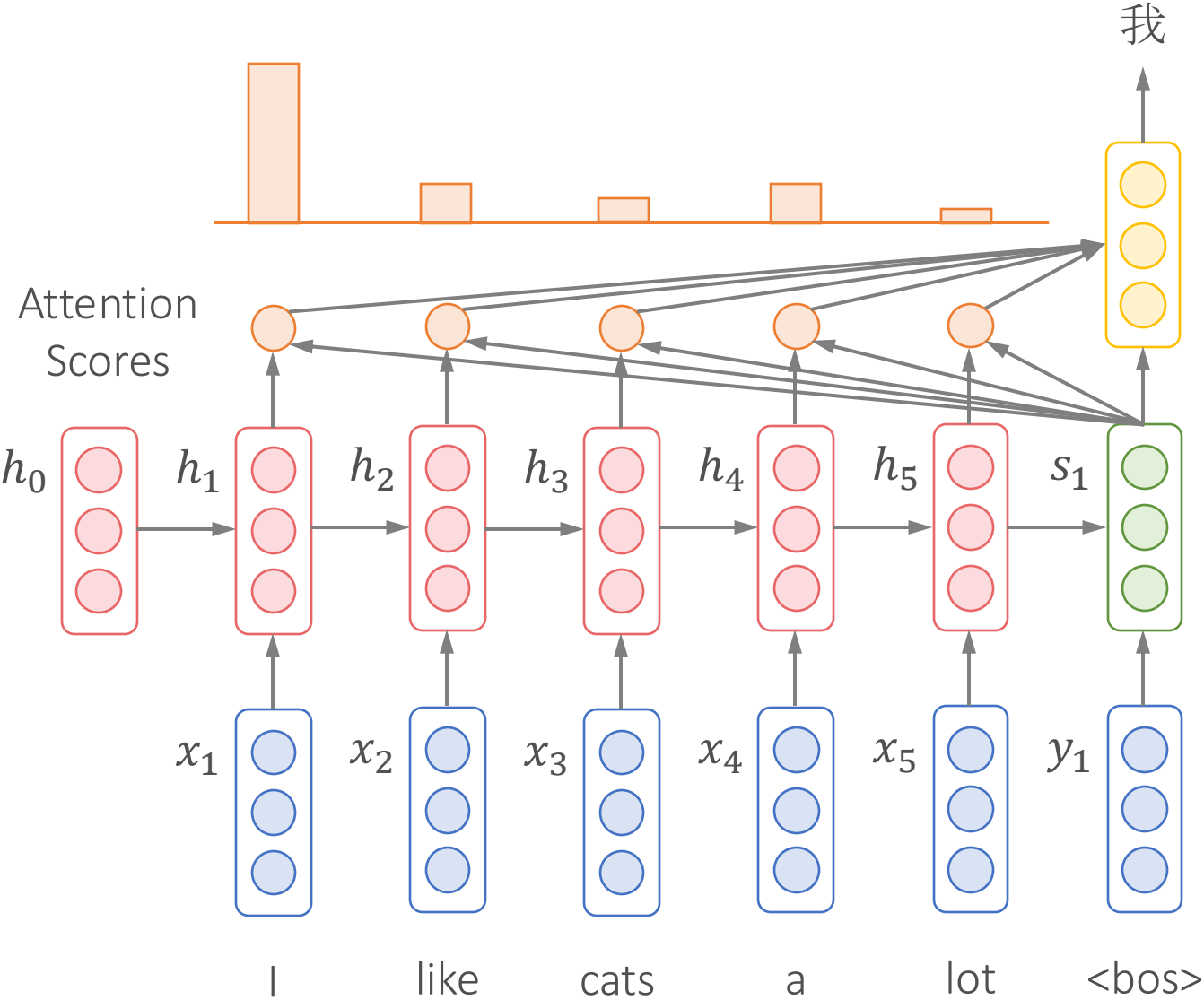
- Attention provides a solution to the bottleneck problem
- Key idea: At each time step during decoding, focus on **a particular part** of source sentence

# RNN with Attention



Attention Scores  $\alpha_i = h_i^T s_1$

# RNN with Attention



Attention Scores

$$\alpha_i = h_i^T s_1$$

Normalized Attention Scores

$$\hat{\alpha}_i = \text{softmax}(\alpha_i)$$

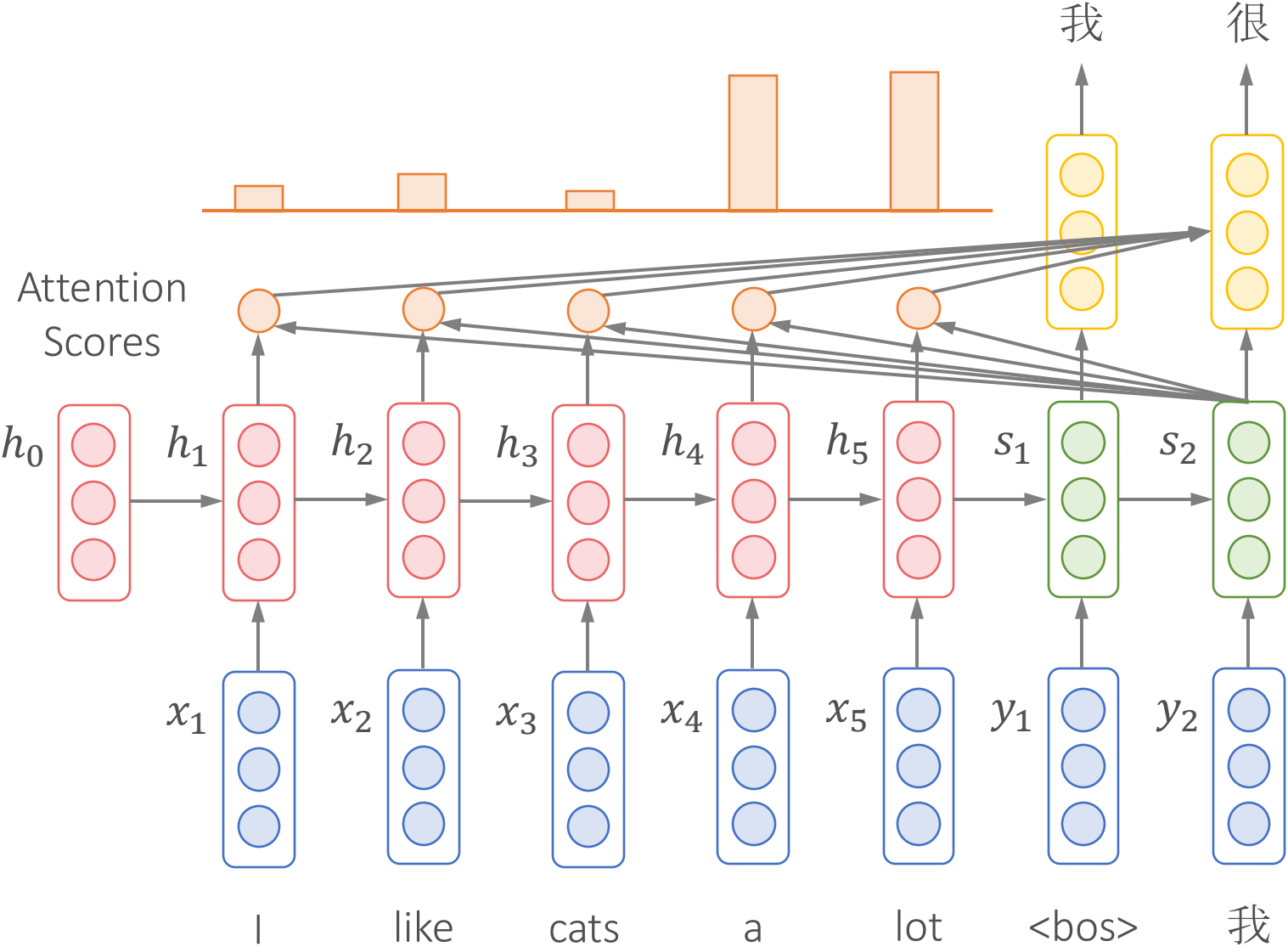
Weighted Sum

$$a = \sum_i \hat{\alpha}_i h_i$$

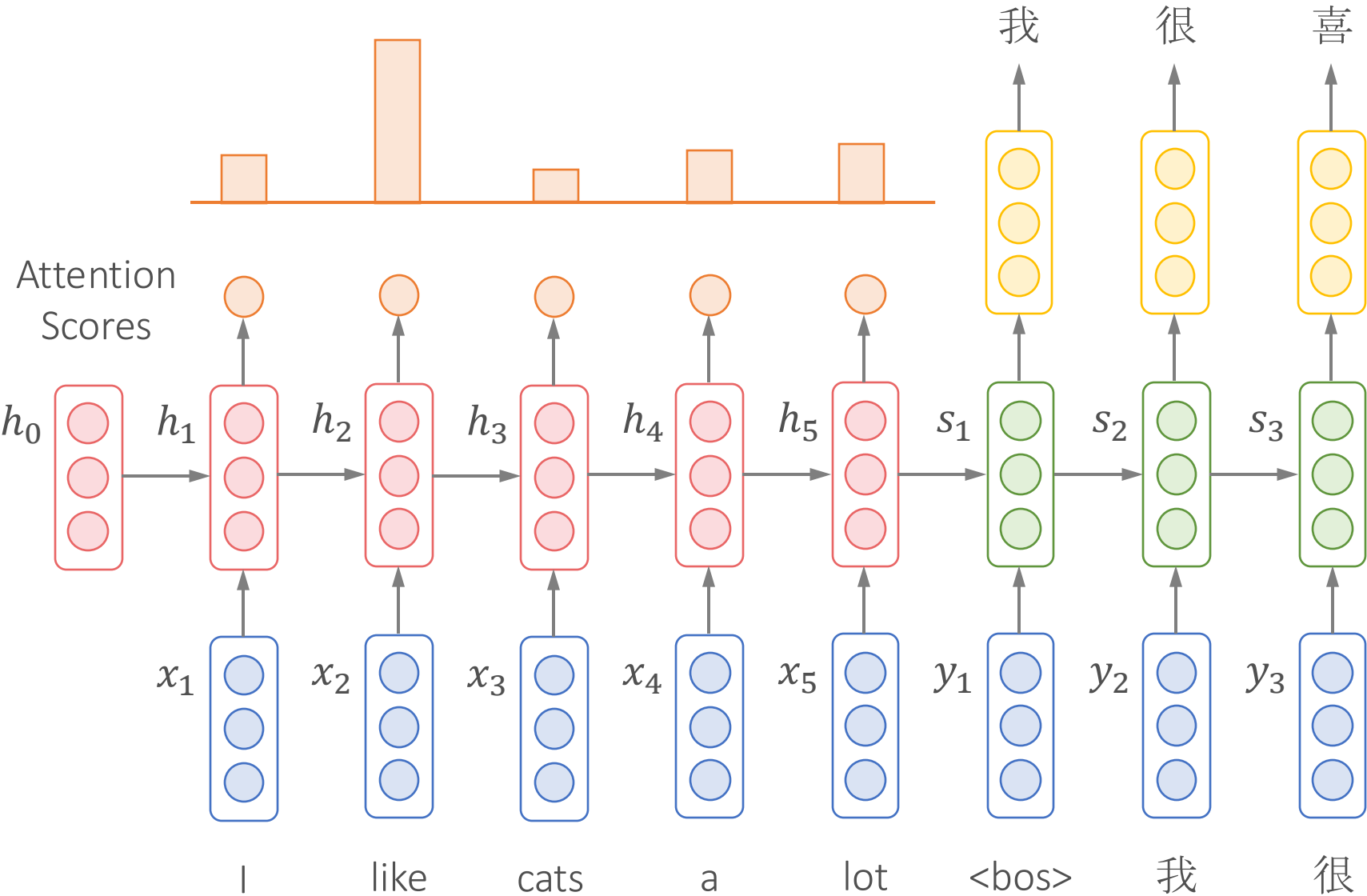
Attention Output

$$\tanh(\mathbf{W}[a; s_1])$$

# RNN with Attention

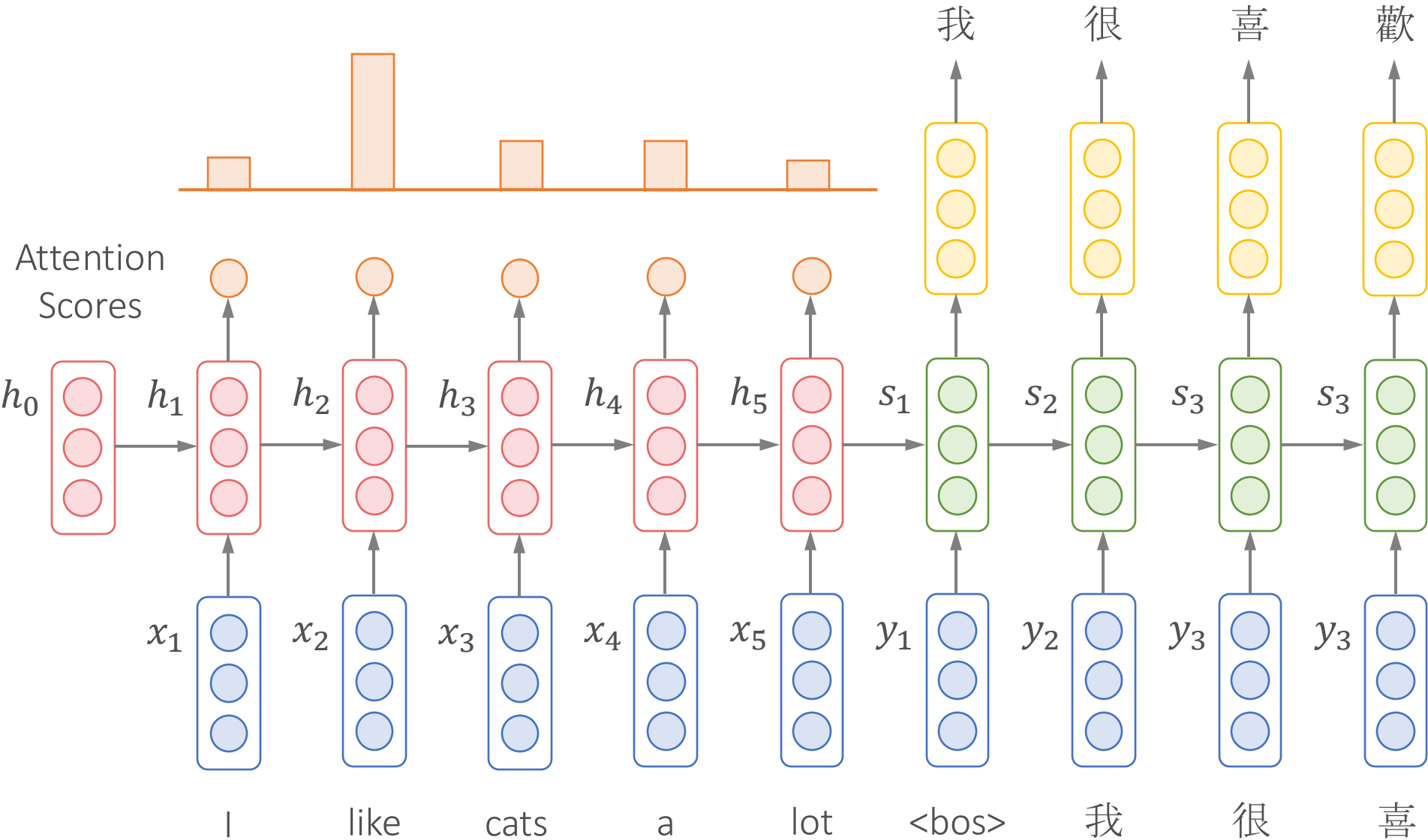


# RNN with Attention

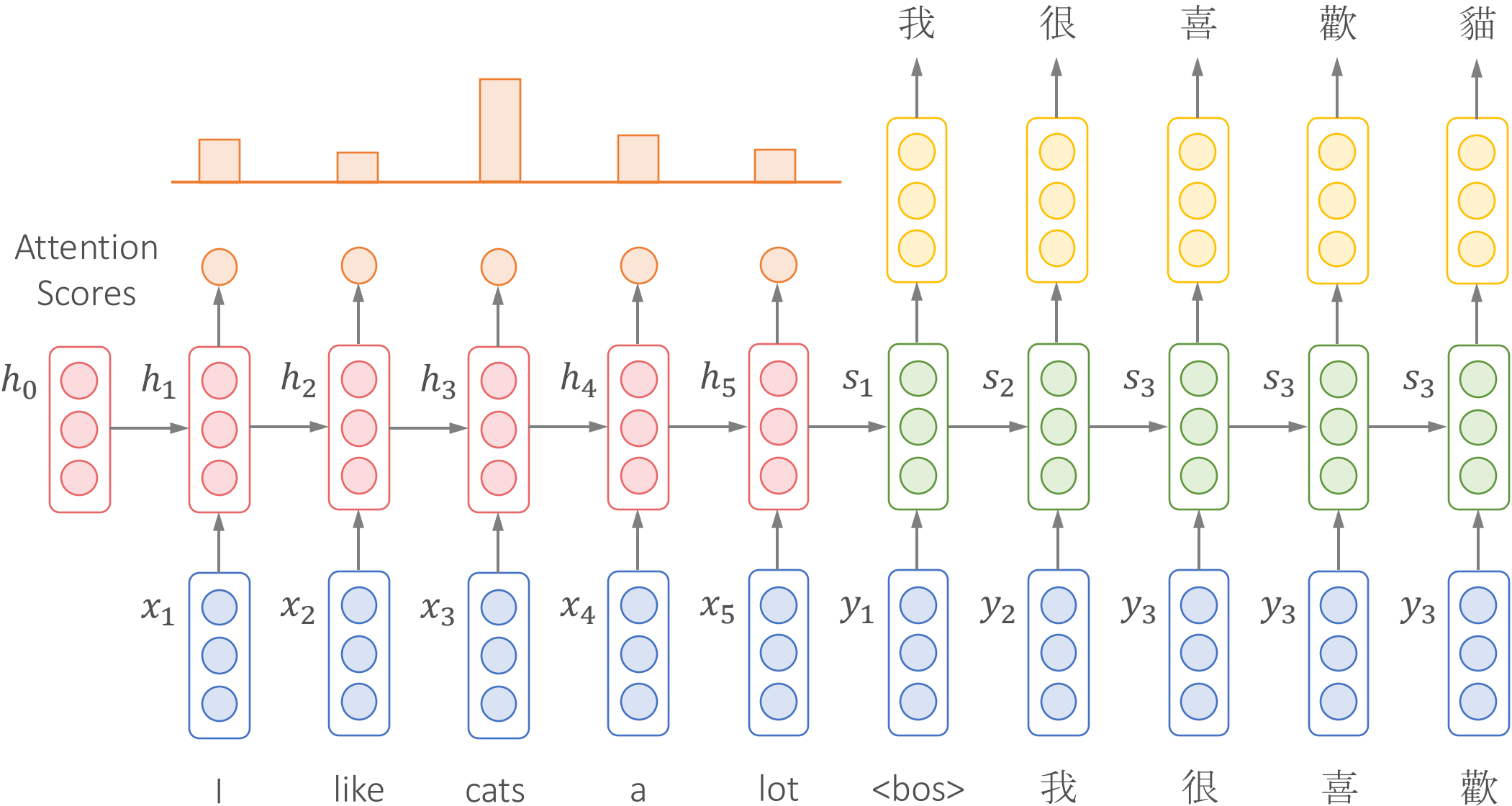




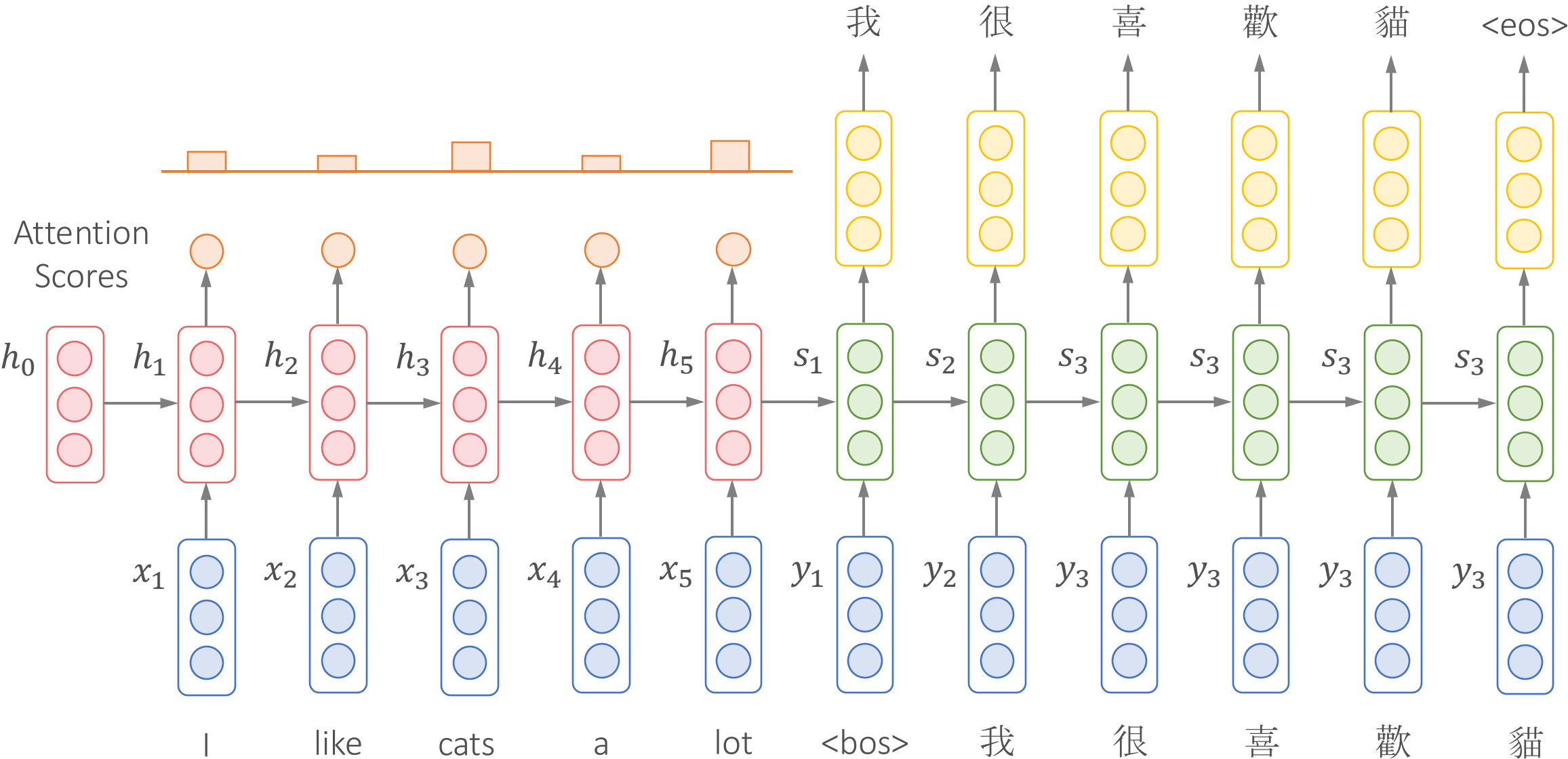
# RNN with Attention



# RNN with Attention



# RNN with Attention



# Different Types of Attention

Dot-Product Attention

$$h_i^\top s_j$$

Multiplicative Attention

$$h_i^\top W s_j$$

Additive Attention

$$v^\top \tanh(W_1 h_i + W_2 s_j)$$