

Introduction to NLP, Language Models

COMP7607 — Week 1

Lingpeng Kong

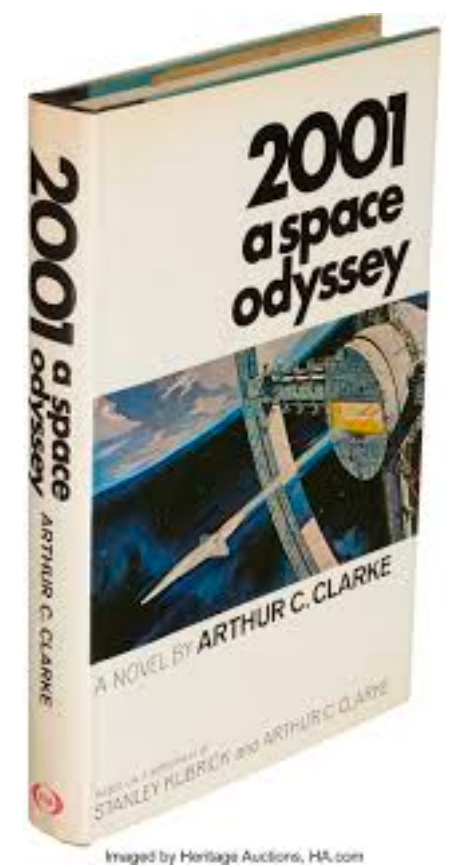
Department of Computer Science, The University of Hong Kong

Many materials from CSE517@UW, COMS W4705@Columbia, 11-711@CMU with special thanks!

“I’m sorry Dave, I’m afraid I can’t do that.”



<https://youtube.com/watch?v=ARJ8cAGm6JE>



2001: A Space Odyssey

Zero-shot Ability

Lingpeng Kong

"Good day Author!
Call For Books,
Unique Pub International (UPI), a self-sufficient and autonomous company, is a global service provider.
Click Here for more information 🖱️ UPI..... BOOKS
Email us if you have any questions editorialoffice@uniquepubinternational.com or upipublications@gmail.com
Publication takes 15–30 Days.
Author Benefits: Book will be published with ISBN
Prompt Email Notification
Low Processing Fee
Publication certificate will be provided to each author
🌸Regards🌸
Unique Pub International
Contact no: [REDACTED]
web: [REDACTED]
Email: [REDACTED]
[REDACTED].com" is this a spam email?

28/08/2023, 10:44:17

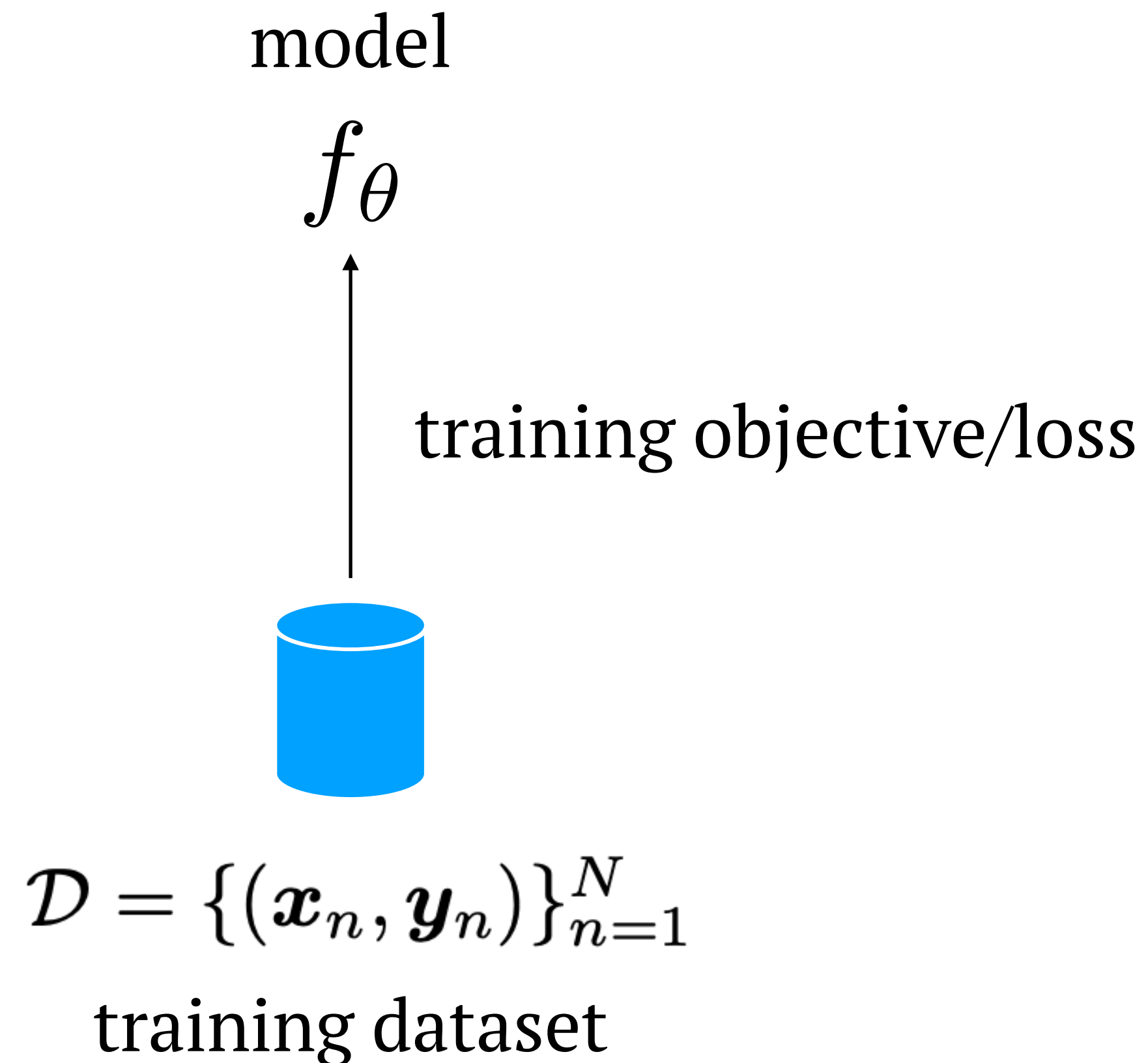


ChatGPT (Tokens Used 544)

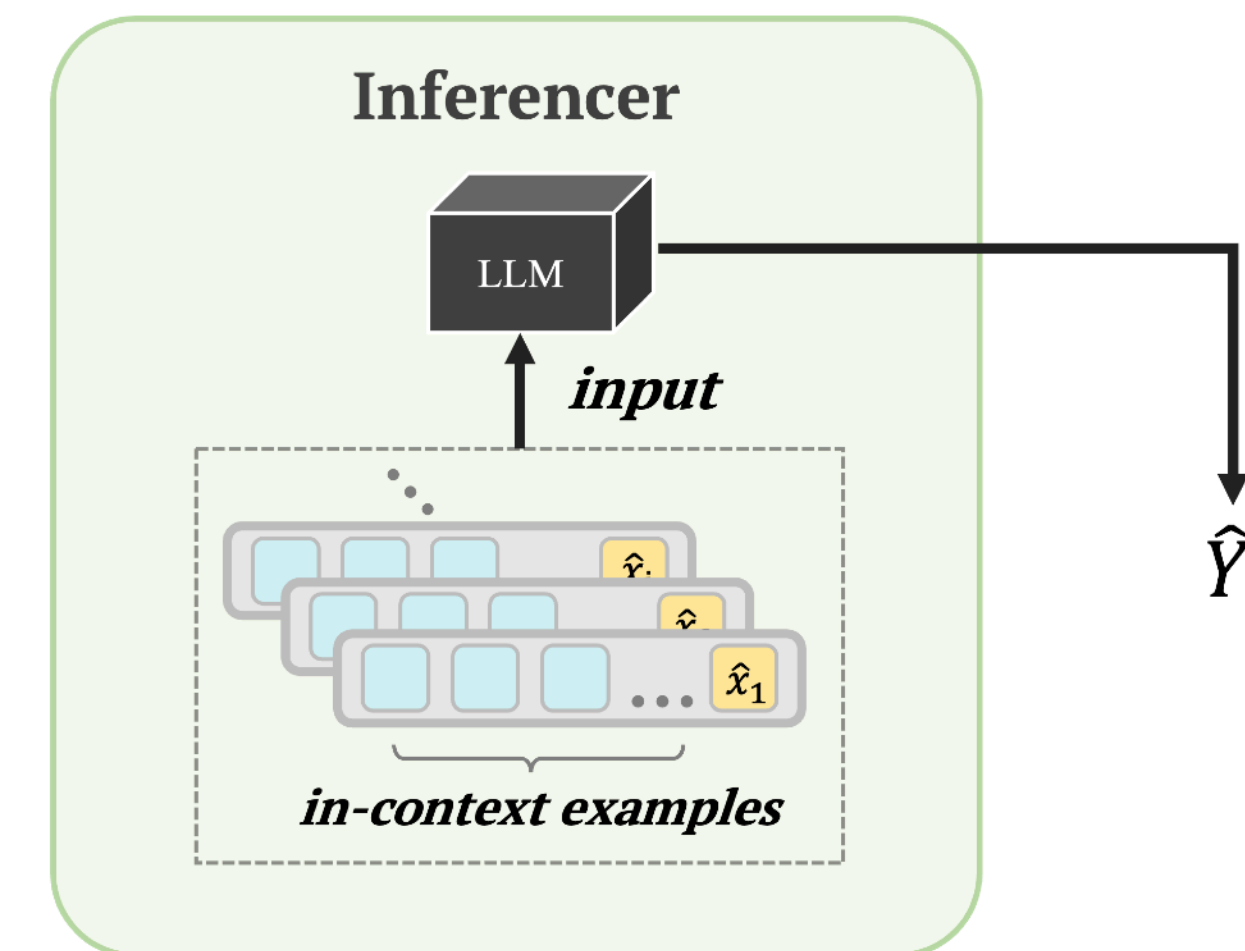
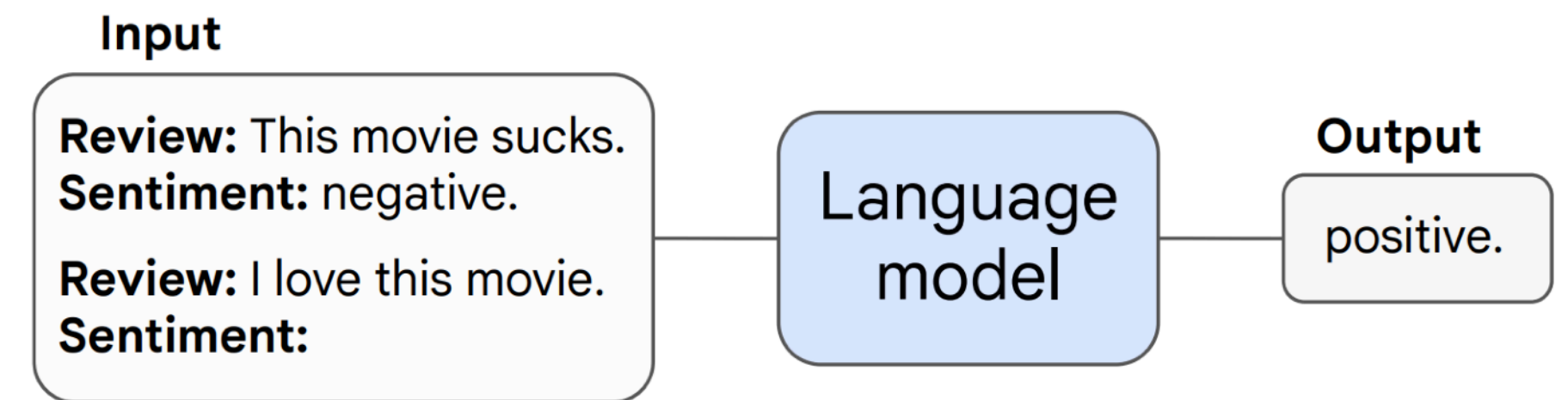
This email is likely a solicitation for self-publishing services, which may or may not be of interest to authors. However, as it is unsolicited and potentially sent to a large number of recipients, it could be considered spam.

28/08/2023, 10:44:17

Machine Learning (Today)



Supervised Learning



In-Context Learning

Powerful Large Language Models

Kylar went to the store to buy glasses for his new apartment. One glass costs \$5, but every second glass costs only 60% of the price. Kylar wants to buy 16 glasses. How much does he need to pay for them?



Steve decides to start eating more tomatoes and decides to grow his own cherry tomatoes. He eats twice as much as his girlfriend. He eats 6 per day. If a vine can produce 3 tomatoes per week how many vines does he need?



Powerful Large Language Models

Kylar went to the store to buy glasses for his new apartment. One glass costs \$5, but every second glass costs only 60% of the price. Kylar wants to buy 16 glasses. How much does he need to pay for them?

 ChatGPT 

To calculate the total amount Kylar needs to pay for the glasses, we can break down the purchase into two parts: the first glass and the rest of the glasses.

The first glass costs \$5. For the remaining 15 glasses, every second glass costs only 60% of the price, which means they cost $60\% * \$5 = \3 each.

Let's calculate the total cost:

First glass: \$5

Remaining glasses (15 glasses): $15 \text{ glasses} * \$3/\text{glass} = \45

Total cost: $\$5 + \$45 = \$50$

Therefore, Kylar needs to pay \$50 for the 16 glasses.

Steve decides to start eating more tomatoes and decides to grow his own cherry tomatoes. He eats twice as much as his girlfriend. He eats 6 per day. If a vine can produce 3 tomatoes per week how many vines does he need?

 ChatGPT 

To determine the number of vines Steve needs to meet his tomato consumption, we can calculate the weekly tomato requirement and then divide it by the number of tomatoes produced by each vine per week.

Steve eats 6 cherry tomatoes per day, which means he consumes $6 * 7 = 42$ tomatoes per week.

If a vine can produce 3 tomatoes per week, Steve would need $42 \text{ tomatoes} / 3 \text{ tomatoes per vine} = 14$ vines.

Therefore, Steve would need 14 cherry tomato vines to meet his weekly tomato consumption.

Powerful Large Language Models



	GPT-3.5 (ChatGPT)	GPT-4	Llama2
GSM8K (8-shot)	57.1	92.0	56.8

71.7 (ChatGPT as of Aug 30, 2023, zero-shot)

An arrow pointing from the text "71.7 (ChatGPT as of Aug 30, 2023, zero-shot)" to the value "57.1" in the table.

Why not every AI is that powerful?

What can I help you with?

Play a good song.

Sorry, I couldn't find 'a good song' in your music.



You need to do a better job understanding me.

Noted.

Yeah, make a note of that.

Here's your note:



What is NLP? Wait, what is language?

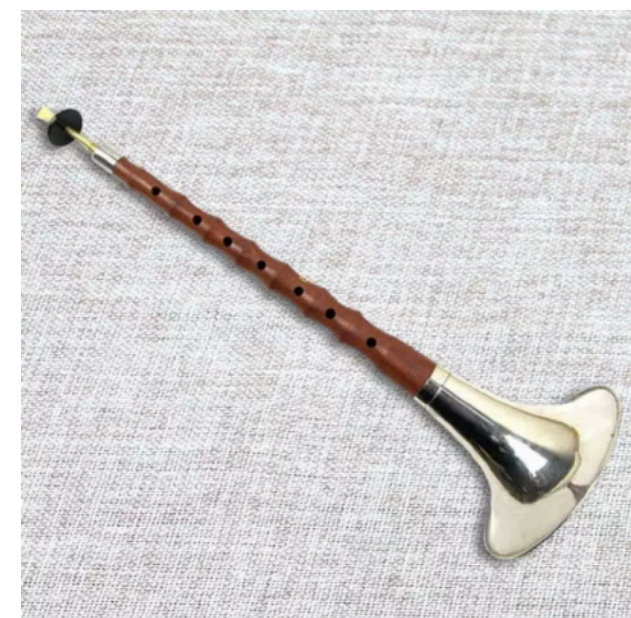
The abstraction of the real world – different languages take you to different worlds!



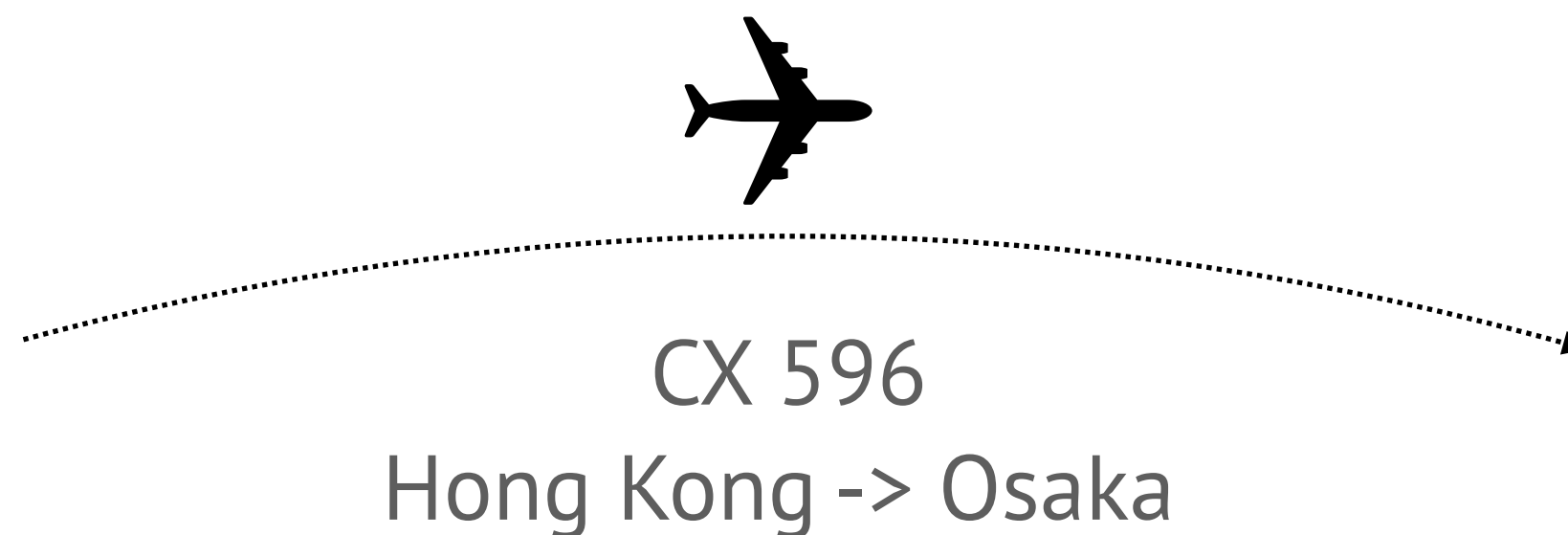
餃子



dumplings



唄



チャルメラ

Something that makes sharp long voice, like screaming???

Do AI “understand”? Let’s play a game!

The cat is thrown out of the _____

door, window, dog

This year, I am going to do an internship in _____

Queen Mary Hospital, HSBC, Google, Amazon

Majoring in computer science, this year, I am going to do an internship in _____

Queen Mary Hospital, HSBC, Google, Amazon

Shannon Game



Claude Elwood Shannon
(April 30, 1916 – February 24, 2001)

A photograph of a chalkboard with the mathematical formula for entropy written in white chalk. The formula is $H = -\sum p(x) \log p(x)$. The chalkboard is dark, and the formula is the central focus. The board is framed by wooden borders on the left and right sides.

Information Theory; Entropy

Language models, and how to build it.



Dice, and how do we roll them
(probabilistic model)



Transformers, neural networks and many others
(powerful functions, and inside configurations)

Generative Language Model

I am going to do an internship in Google



Making the dice



bag of words

(@Carnegie Mellon University)

- 1 Belief
- 2 Evidence
- 3 Reason
- 4 Claim
- 5 Think
- 6 Justify
- 7 Also
- ...
- 99 Therefore
- 100 Google

Vocabulary



Generative Language Model

I



I

Generative Language Model

I am



am

Generative Language Model

I am going



going

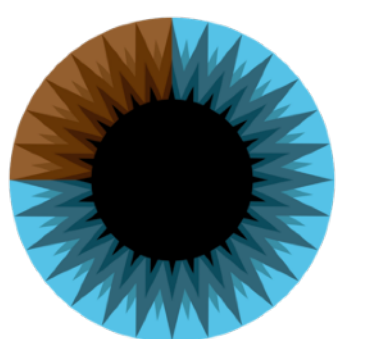
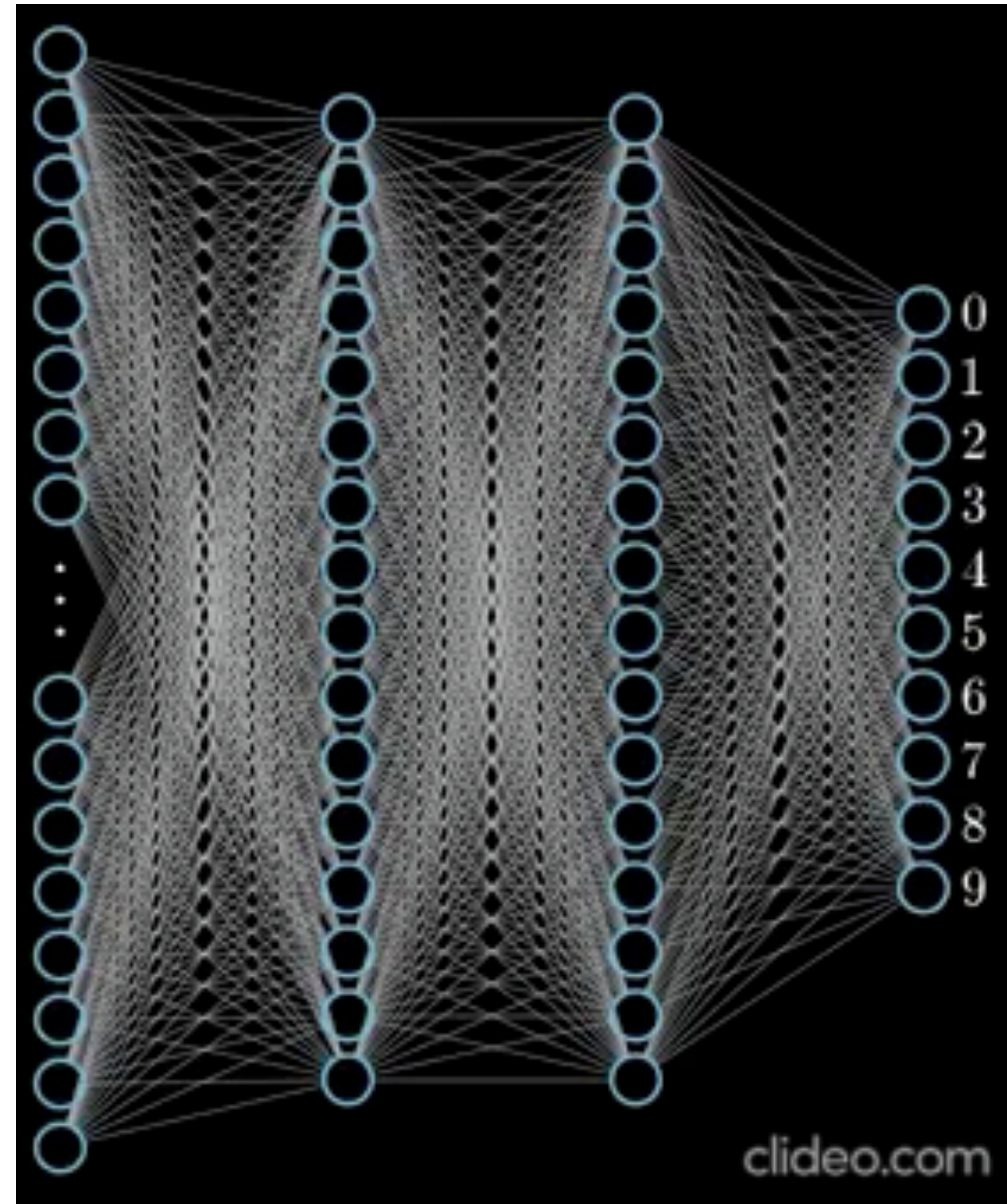
Generative Language Model

I am going to do an internship in Google

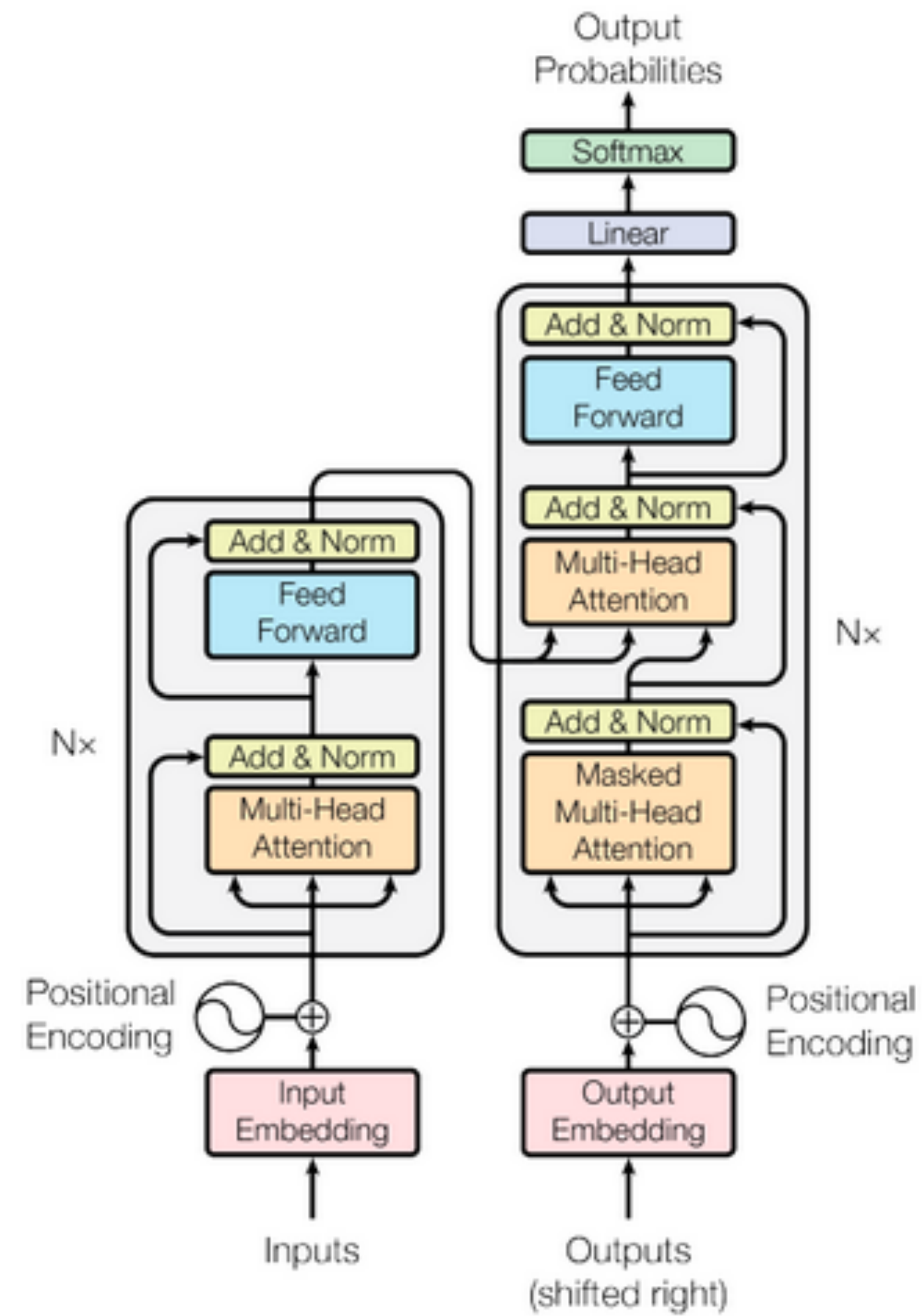


Google

Neutralize the dice!

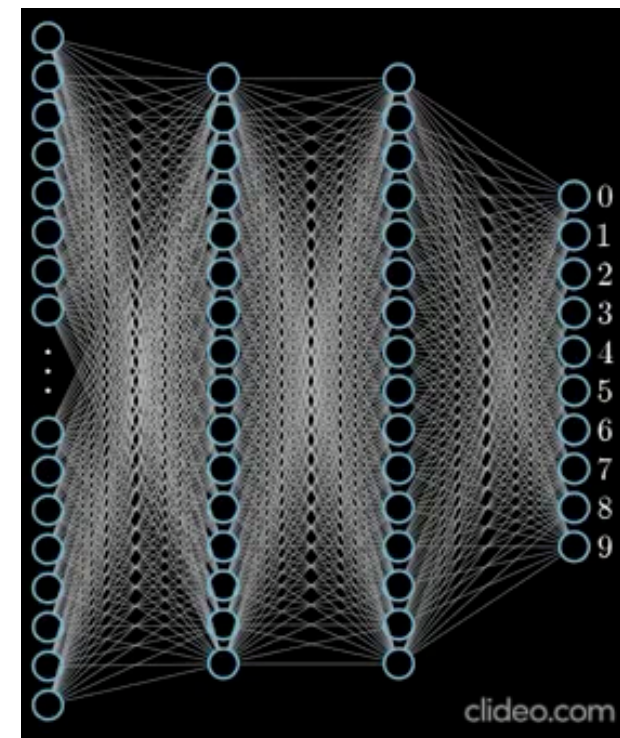


Neural Networks (e.g. Transformers)



Generative Language Model

I am going to do an internship in Google



Google

Language models, and how to build it.



Dice, and how do we roll them
(probabilistic model)



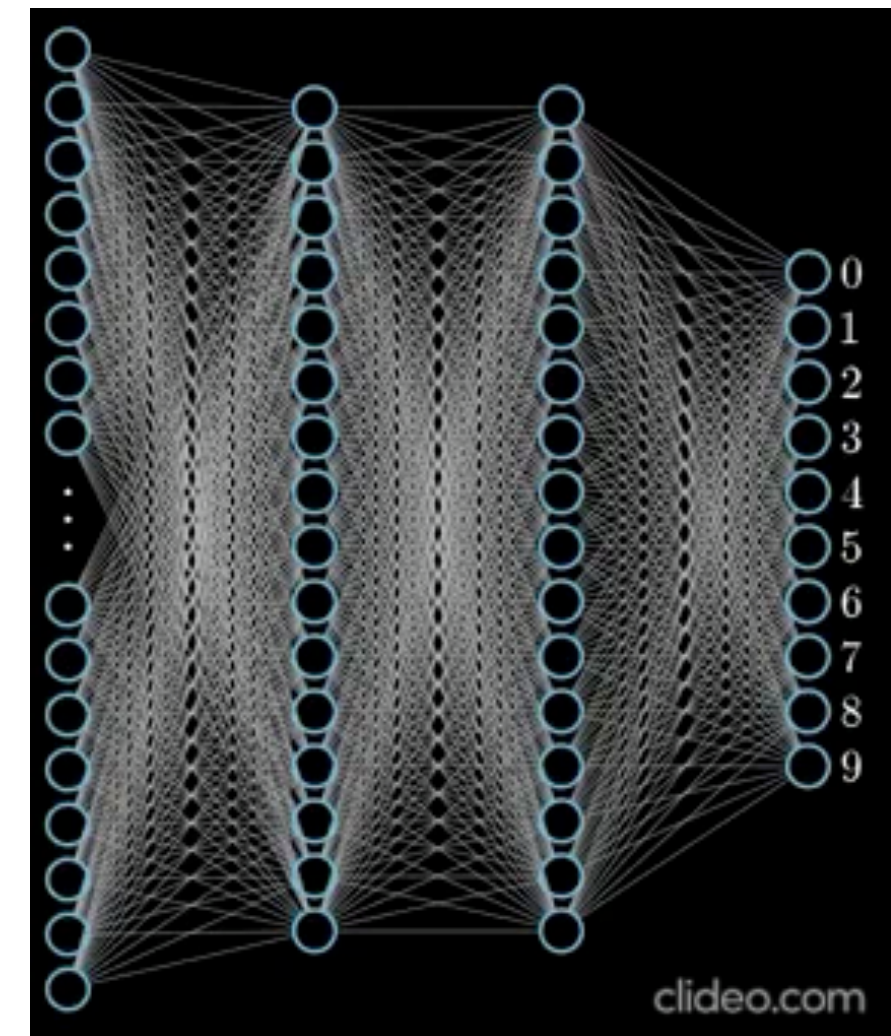
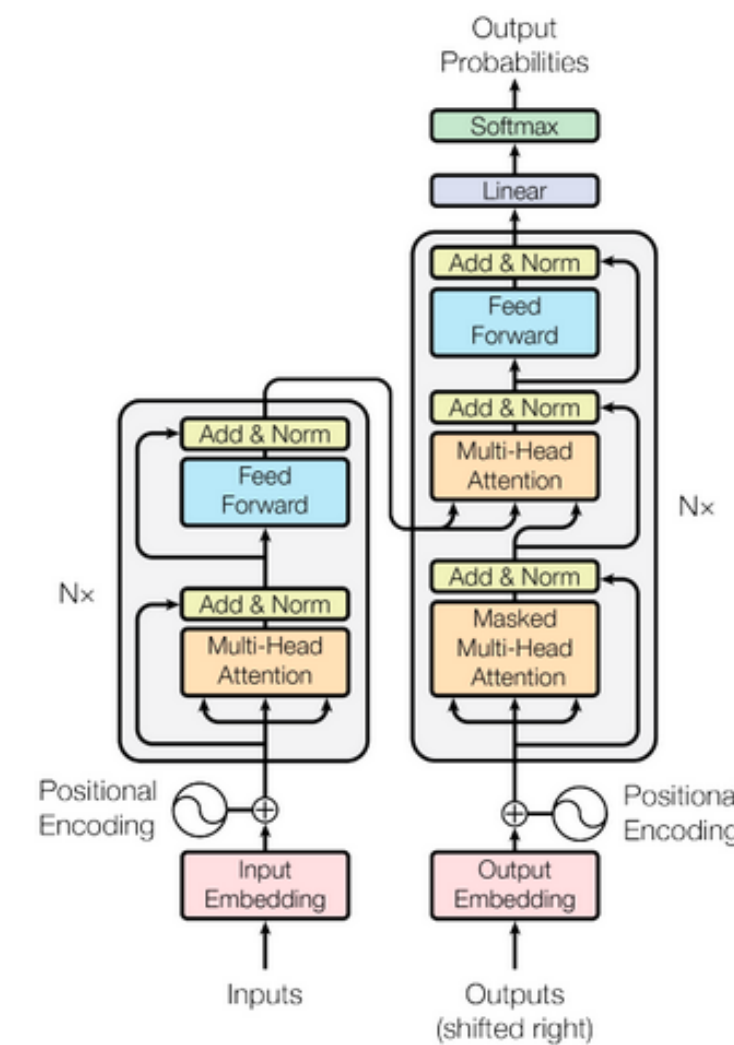
Transformers, neural networks and many others
(powerful functions)

$$p(\mathbf{x}) = \prod_i p(x_i | \mathbf{x}_{<i})$$

Learn



Parameterize



First problem – the language modeling problem

Given a finite vocabulary

$\mathcal{V} = \{\text{belief, evidence, reason, claim, } \dots \text{ Google, therefore}\}$

We have an infinite set of strings, \mathcal{V}^\dagger

$\langle s \rangle$ I am going to an internship in Google $\langle /s \rangle$

$\langle s \rangle$ an internship in Google $\langle /s \rangle$

$\langle s \rangle$ I am going going $\langle /s \rangle$

$\langle s \rangle$ Google is am $\langle /s \rangle$

$\langle s \rangle$ internship is going $\langle /s \rangle$

Formally:

$$p(x_1, x_2, \dots, x_n)$$

$$p(x_i \mid x_{i-1}, x_{i-2}, \dots, x_1)$$

Can we learn a “model” for this “generative process”? We need to “learn” a probability distribution:

$$\sum_{x \in \mathcal{V}^\dagger} p(x) = 1, p(x) \geq 0 \text{ for all } x \in \mathcal{V}^\dagger$$

Learn from what we've seen

The Language Modeling Problem

Given a *training sample* of example sentences, we need to “learn” a probabilistic model that assigns probabilities to every possible string:

$$p(\langle s \rangle \text{ I am going to an internship in Google } \langle /s \rangle) = 10^{-12}$$

$$p(\langle s \rangle \text{ an internship in Google } \langle /s \rangle) = 10^{-8}$$

$$p(\langle s \rangle \text{ I am going going } \langle /s \rangle) = 10^{-15}$$

...

It is a probability distribution p over strings, i.e., p is a function that satisfies

$$\sum_{x \in \mathcal{V}^{\dagger}} p(x) = 1, \quad p(x) \geq 0 \text{ for all } x \in \mathcal{V}^{\dagger}$$

The Language Modeling Problem

<s> Sam I am </s>

<s> I am Sam </s>

<s> I do not like green eggs and ham </s>

Training Corpus

Is this a good model?

$$P(\text{<s> Sam I am </s>}) = 1/3$$

$$P(\text{<s> I am Sam </s>}) = 1/3$$

$$P(\text{<s> I do not like green eggs and ham </s>}) = 1/3$$

$$P(\text{<s> I am </s>}) = 0$$

$$P(\text{<s> green Sam </s>}) = 0$$

Naïve Model

The Language Modeling Problem

<s> Sam I am </s>

<s> I am Sam </s>

<s> I do not like green eggs and ham </s>

Training Corpus

The probability of the word “<s>” followed by the word “I”:

$$P(I \mid \langle s \rangle) = 2/3$$

The Language Modeling Problem

<s> Sam I am </s>

<s> I am Sam </s>

<s> I do not like green eggs and ham </s>

Training Corpus

$$P(I \mid \langle s \rangle) = 2/3$$

$$P(\langle /s \rangle \mid \text{Sam}) = 1/2$$

Naïve Model

The Language Modeling Problem

<s> Sam I am </s>

<s> I am Sam </s>

<s> I do not like green eggs and ham </s>

Training Corpus

$$P(I \mid \langle s \rangle) = 2/3$$

$$P(\text{am} \mid I) = ?$$

$$P(\langle s \rangle \mid \text{Sam}) = 1/2$$

Naïve Model

The Language Modeling Problem

<s> Sam I am </s>

<s> I am Sam </s>

<s> I do not like green eggs and ham </s>

Training Corpus

$$P(I \mid \langle s \rangle) = 2/3$$

$$P(\text{am} \mid I) = 2/3$$

$$P(\langle s \rangle \mid \text{Sam}) = 1/2$$

Naïve Model

The Language Modeling Problem

<s> Sam I am </s>

<s> I am Sam </s>

<s> I do not like green eggs and ham </s>

Training Corpus

$$P(\text{<s> Sam I am </s>}) = P(\text{Sam} \mid \text{<s>}) * P(\text{I} \mid \text{Sam}) * P(\text{am} \mid \text{I}) * P(\text{</s>} \mid \text{am})$$

Bi-gram Model

Course Logistics

Course Logistics

Website:

<https://nlp.cs.hku.hk/comp7607-fall2023>

Prerequisites:

COMP3314 or COMP3340, MATH1853

We will assume a lot things from Machine Learning, Statistics, and Programming



This NLP course will be very difficult if you haven't taken these courses.

Assessment:

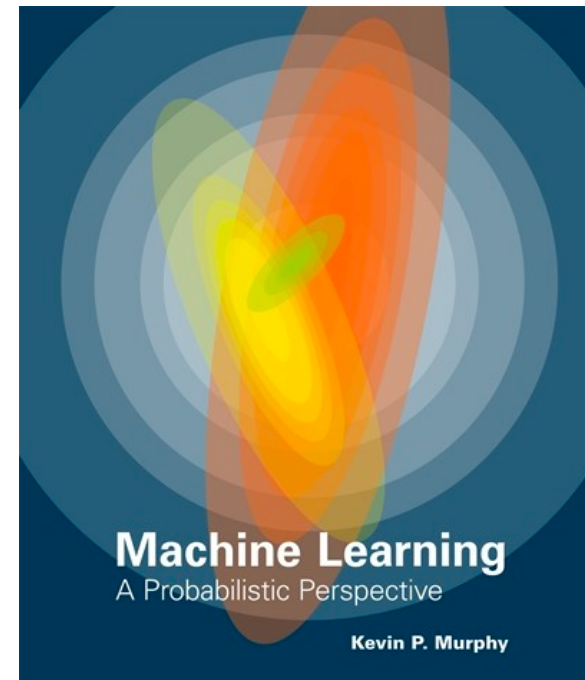
50% continuous assessment, 50% final project

TA:

Qington Li (<https://qtli.github.io/>), Xubin Ren (<https://rxubin.com/>)

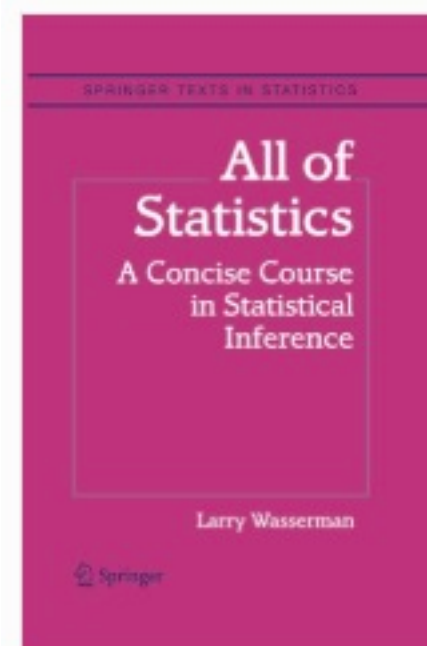
Course Logistics

We will assume a lot things from Machine Learning, Statistics, and Programming

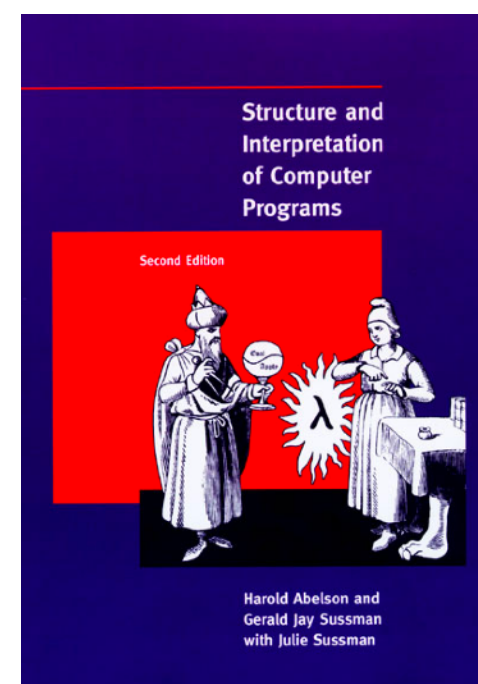


Supervised learning, unsupervised learning, regression, classification, loss function, neural networks, regularization ...

(COMP3314)



Random variables, joint probability, conditional probability, Bayes' theorem ...



Data structures, dynamic programming, time/space complexity ...

Course Logistics

Textbook recommendation (J&M):

<https://web.stanford.edu/~jurafsky/slp3/>

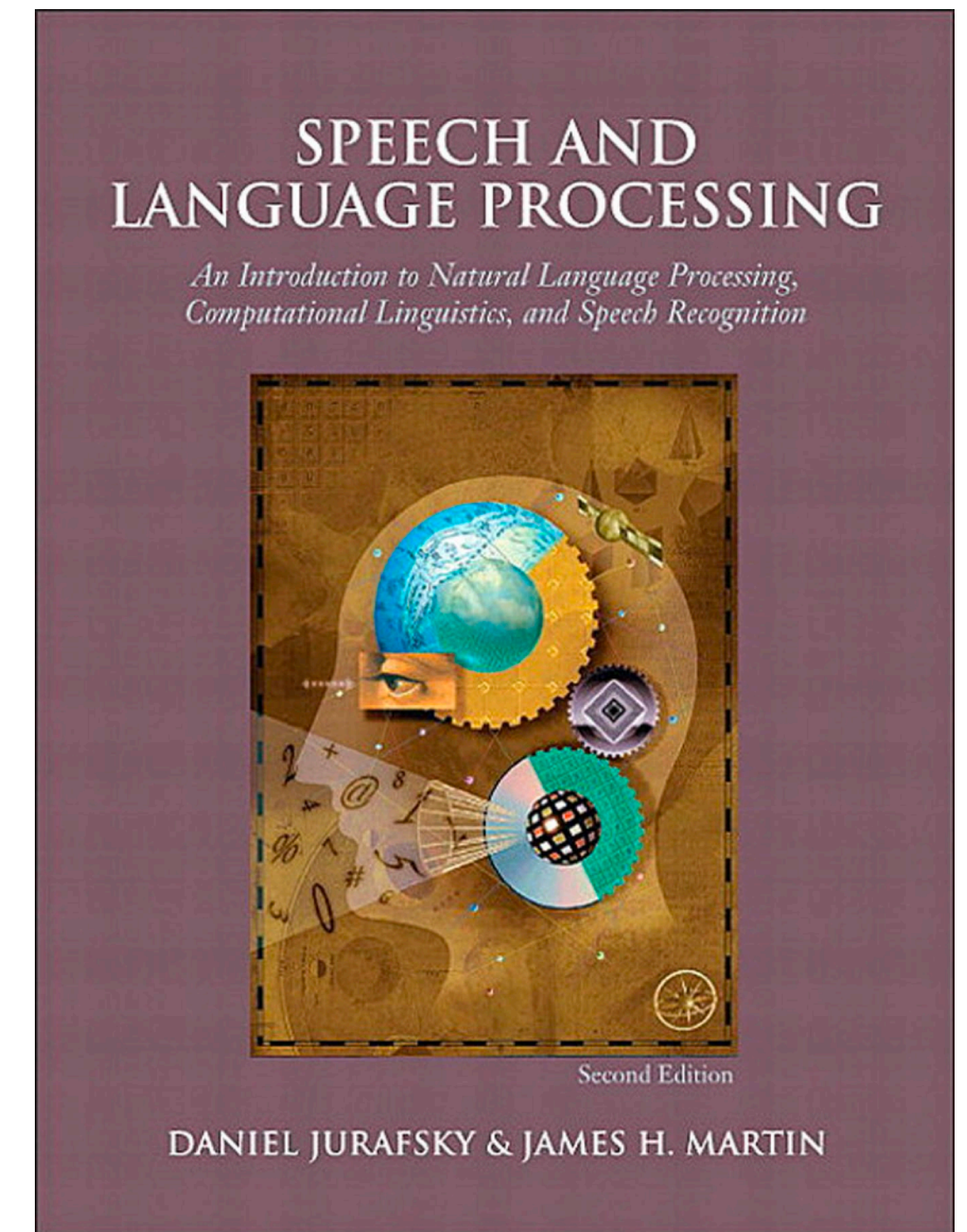
Assessments (in total ~4):

Programming problems

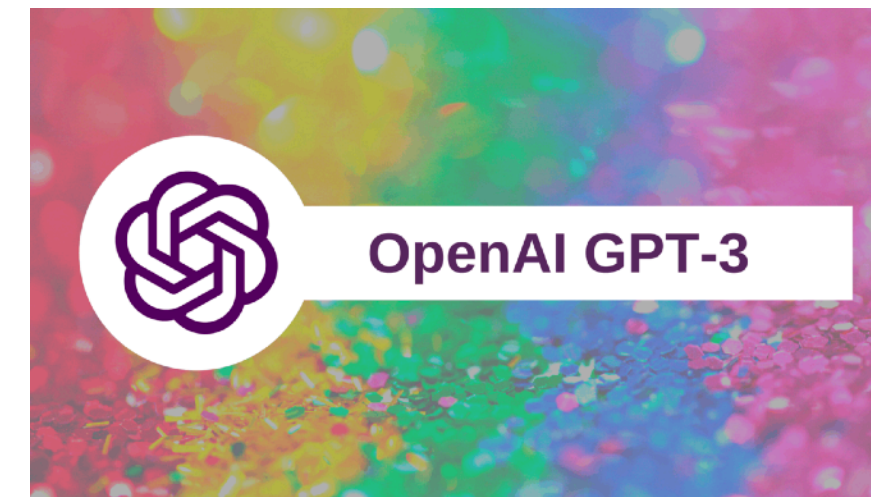
Problem sets

Honor code:

You are free to form study groups and discuss homeworks and projects. However, you must write up homeworks and code from scratch independently, and you must acknowledge in your submission all the students you discussed with.



What's Next?



BERT, GPT-3, Word2vec, Glove, T5 ...



N-Gram Models, Hidden Markov Models ...



LSTMs, Recurrent Neural Networks, MLP, Transformers ...

It's not about human language. It's the language of life!



Protein folding explained

<https://www.youtube.com/watch?v=KpedmJdrTpY>

 Google DeepMind

AlphaFold (Jumper et al, 2021)

Probabilistic Language Models

Assign a probability to a sentence

$P(\text{"I am going to school"}) > P(\text{"I are going to school"})$

Grammar Checking

I had some coffee this morning.

$P(\text{"我今早喝了一些咖啡"}) > P(\text{"我今早吃了一些咖啡"})$

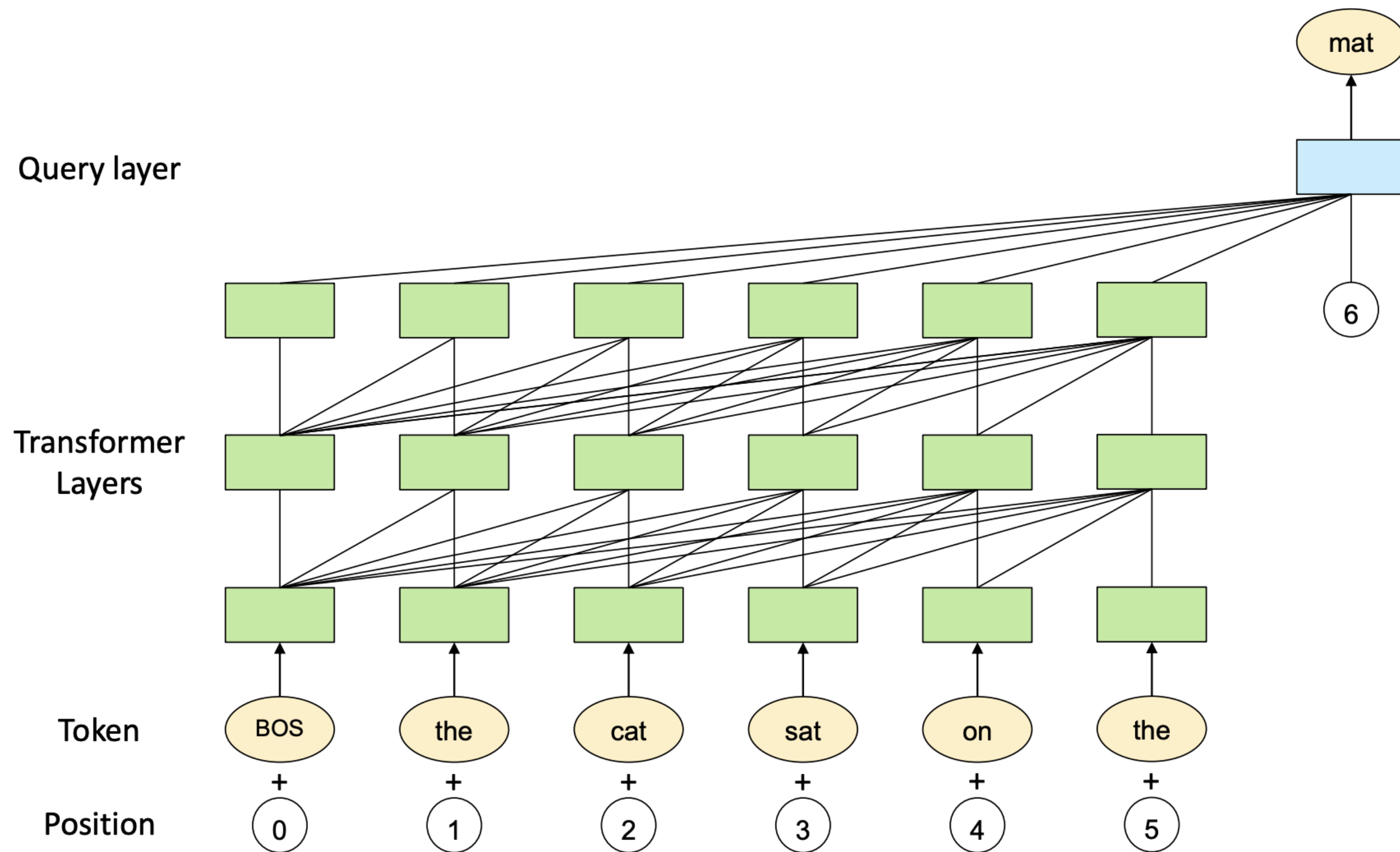
Machine translation

$P(\text{"Can we put an elephant into the refrigerator? No, we can't.}) > P(\text{"Can we put an elephant into the refrigerator? Yes, we can.})$

Question Answering

Probabilistic Language Models

This is the reality since 2021!



Input (Prompt):

How smart can a cat be?

Output:



They're incredibly independent animals, they can understand numerous things (even though they may choose to ignore you) and they even have fantastic short- and long-term memories!

PanGu- α (Zeng et al, April 2021)

Probabilistic Language Models

$$\mathcal{V} = \{\text{the, dog, laughs, saw, barks, cat, \dots}\}$$

A sentence in the language is a sequence of words

$$x_1 x_2 \dots x_n$$

For example

the dog barks STOP

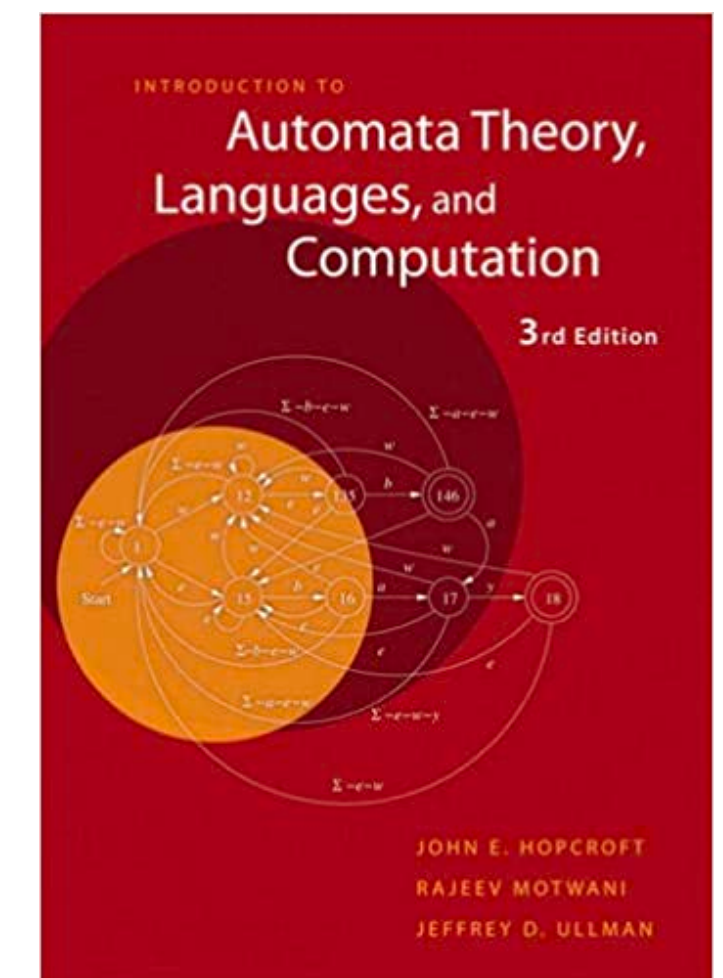
the cat saw the dog STOP

...

Definition (Language Mode)

1. For any $\langle x_1 \dots x_n \rangle \in \mathcal{V}^\dagger$, $p(x_1, x_2, \dots, x_n) \geq 0$

2. In addition,
$$\sum_{\langle x_1 \dots x_n \rangle \in \mathcal{V}^\dagger} p(x_1, x_2, \dots, x_n) = 1$$



(Hopcroft, Motwani, Ullman)

A (very bad) method for learning a LM

Number of times the sentence $x_1 \dots x_n$ is seen in the training corpus

$$c(x_1 \dots x_n)$$

Total number of sentences in the training corpus N

$$p(x_1 \dots x_n) = \frac{c(x_1 \dots x_n)}{N}$$

Why this is very bad?

Markov Models

Consider a sequence of random variables X_1, X_2, \dots, X_n , each take any value in \mathcal{V}

The joint probability of a sentence is

$$\begin{aligned} &P(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) \\ &= P(X_1 = x_1) \prod_{i=2}^n P(X_i = x_i | X_1 = x_1, \dots, X_{i-1} = x_{i-1}) \end{aligned}$$



$$= P(X_1 = x_1) \prod_{i=2}^n P(X_i = x_i | X_{i-1} = x_{i-1})$$

First-order Markov Assumption

Trigram Language Models

A trigram language model consists of a finite set \mathcal{V} , and a parameter $q(w \mid u, v)$

For each trigram u, v, w , such that $w \in \mathcal{V} \cup \{\text{STOP}\}$, $u, v \in \mathcal{V} \cup \{*\}$.

$q(w \mid u, v)$ can be interpreted as the probability of seeing the word w immediately after the bigram (u, v) .

For any sentence $x_1 \dots x_n$, where $x_i \in \mathcal{V}$ for $i = 1 \dots (n - 1)$, and $x_n = \text{STOP}$

$$p(x_1 \dots x_n) = \prod_{i=1}^n q(x_i \mid x_{i-2}, x_{i-1})$$

where we define $x_0 = x_{-1} = *$

Trigram Language Models

For example, for the sentence

the dog barks STOP

$$p(\text{the dog barks STOP}) = q(\text{the} \mid *, *) \times q(\text{dog} \mid *, \text{the}) \times q(\text{barks} \mid \text{the, dog}) \times q(\text{STOP} \mid \text{dog, barks})$$

Problem solved? How can we find $q(w \mid u, v)$

Parameters (of the model)

$$q(w \mid u, v)$$

How many parameters?

How to “estimate” them from training data?

Trigram Language Models

Parameters (of the model)

$$q(w \mid u, v)$$

How many parameters?

$$|\mathcal{V}|^3$$

How to “estimate” them from training data?

$$q(w \mid u, v) = \frac{c(u, v, w)}{c(u, v)}$$

$$q(\text{barks} \mid \text{the, dog}) = \frac{c(\text{the, dog, barks})}{c(\text{the, dog})}$$

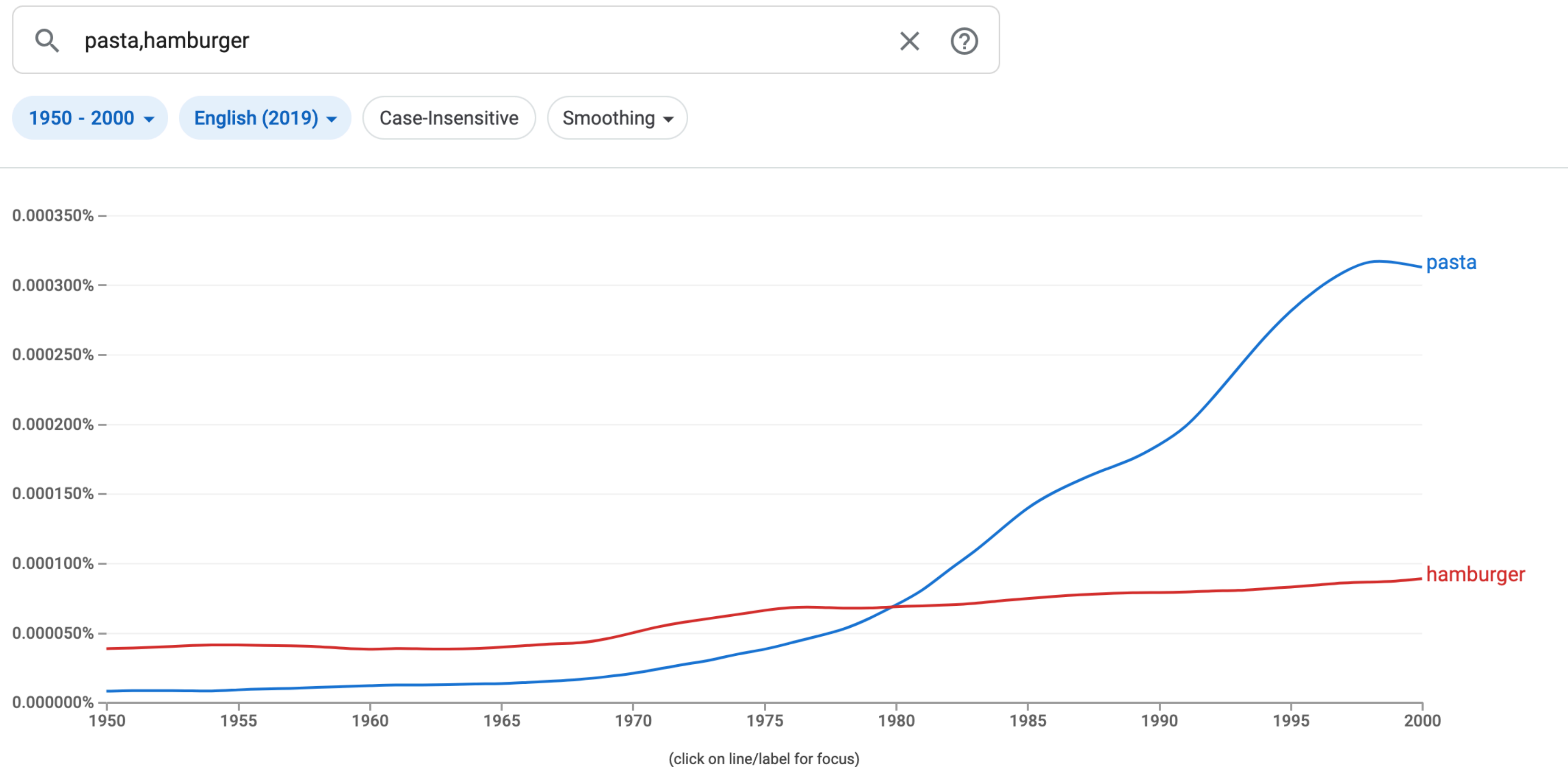
Trigram Language Models

How to “estimate” them from training data?

$$q(w | u, v) = \frac{c(u, v, w)}{c(u, v)}$$

$$q(\text{barks} | \text{the, dog}) = \frac{c(\text{the, dog, barks})}{c(\text{the, dog})}$$

N-gram counts!



Pasta v.s. Hamburger ([Google Books Ngram Viewer](#))

Sparse Data Problems

Maximum likelihood estimate:

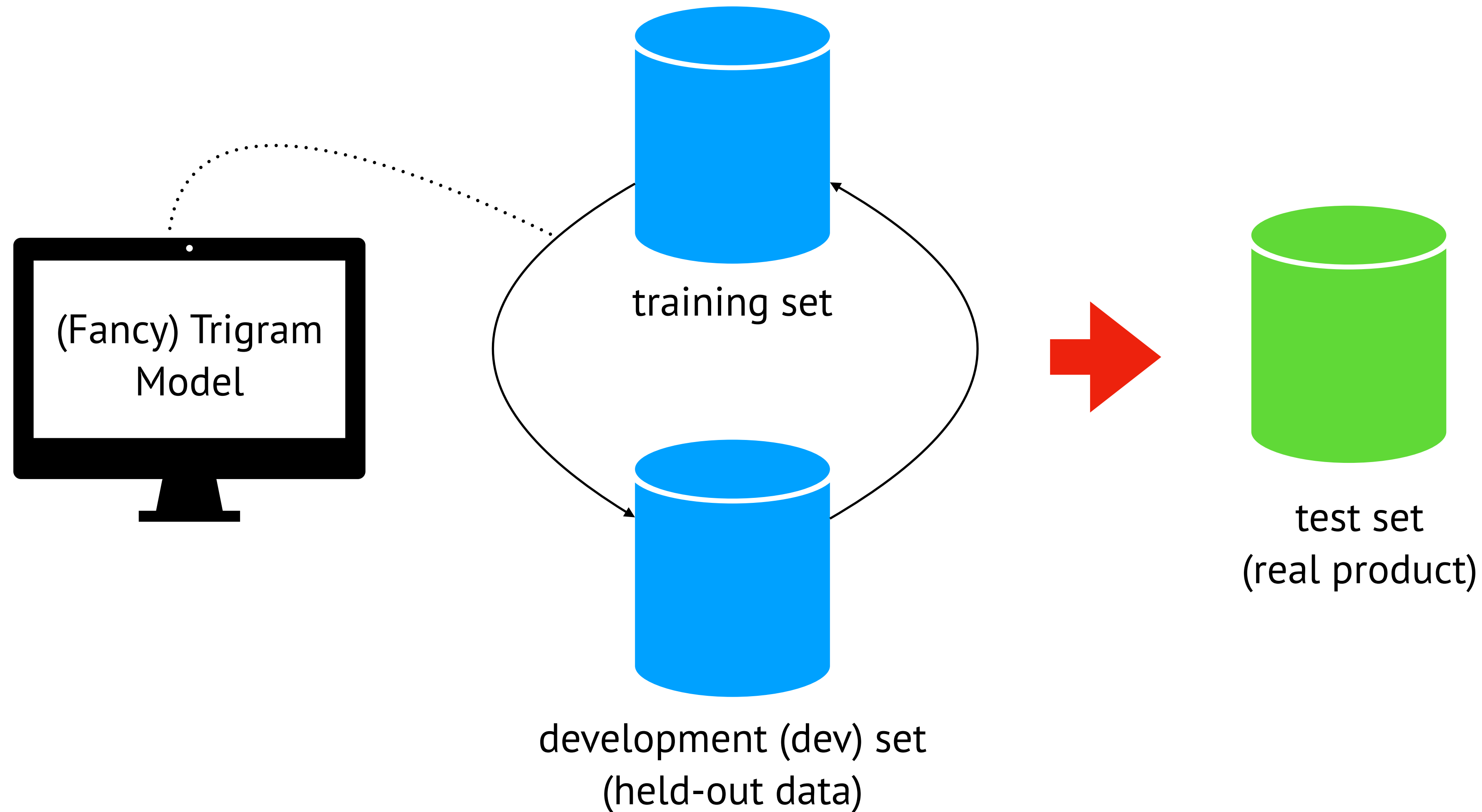
$$q(w \mid u, v) = \frac{c(u, v, w)}{c(u, v)}$$

$$q(\text{barks} \mid \text{the, dog}) = \frac{c(\text{the, dog, barks})}{c(\text{the, dog})}$$

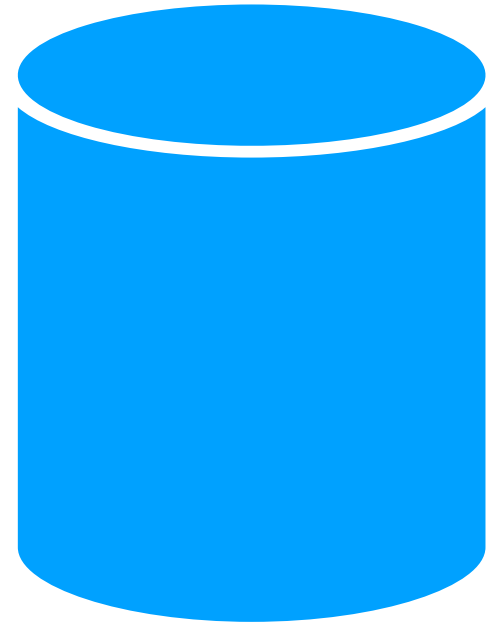
$|\mathcal{V}|^3$

Say vocabulary size is 20000. We have $8 * 10^{12}$ parameters!!

Evaluating Language Models: Perplexity



Evaluating Language Models: Perplexity



development (dev) set
(held-out data)

...

$x^{(i)}$ the cat laughs STOP

$x^{(i+1)}$ the dog laughs at the cat STOP

...

We can compute the probability it assigns to the entire set of test sentences

$$\prod_{i=1}^m p(x^{(i)})$$

The **higher** this quantity is, the better the language model is at modeling unseen sentences.

Evaluating Language Models: Perplexity

The **higher** this quantity is, the better the language model is at modeling unseen sentences.

$$\prod_{i=1}^m p(x^{(i)})$$

Perplexity on the test corpus is derived as a direction transformation of this.

$$\text{ppl} = 2^{-l}$$

$$l = \frac{1}{M} \sum_{i=1}^m \log_2 p(x^{(i)})$$

M is the total length of the sentences in the test corpus.

What if the model estimate $q(w | u, v) = 0$ and the trigram appears in the dataset?

Wait, why we love this number in the first place?

Let the model predicts $q(w | u, v) = 1/N$

$$l = \frac{1}{M} \sum_{i=1}^m \log_2 p(x^{(i)})$$

$$\text{ppl} = 2^{-l} = N$$

A uniform probability model — The perplexity is equal to the vocabulary size!

Perplexity can be thought of as the effective vocabulary size under the model!
For example, the perplexity of the model is 120 (even though the vocabulary size is say 10,000), then this is roughly equivalent to having an effective vocabulary of 120.

How much your language model updates the uniform guess!

Bayes Factors:

<https://www.youtube.com/watch?v=lG4VkPoG3ko>

Smoothing for Language Models

If the model estimate $q(w | u, v) = 0$ and the trigram appears in the test data, ppl goes up to infinity.

When we have **sparse** statistics:

P(w | denied the)

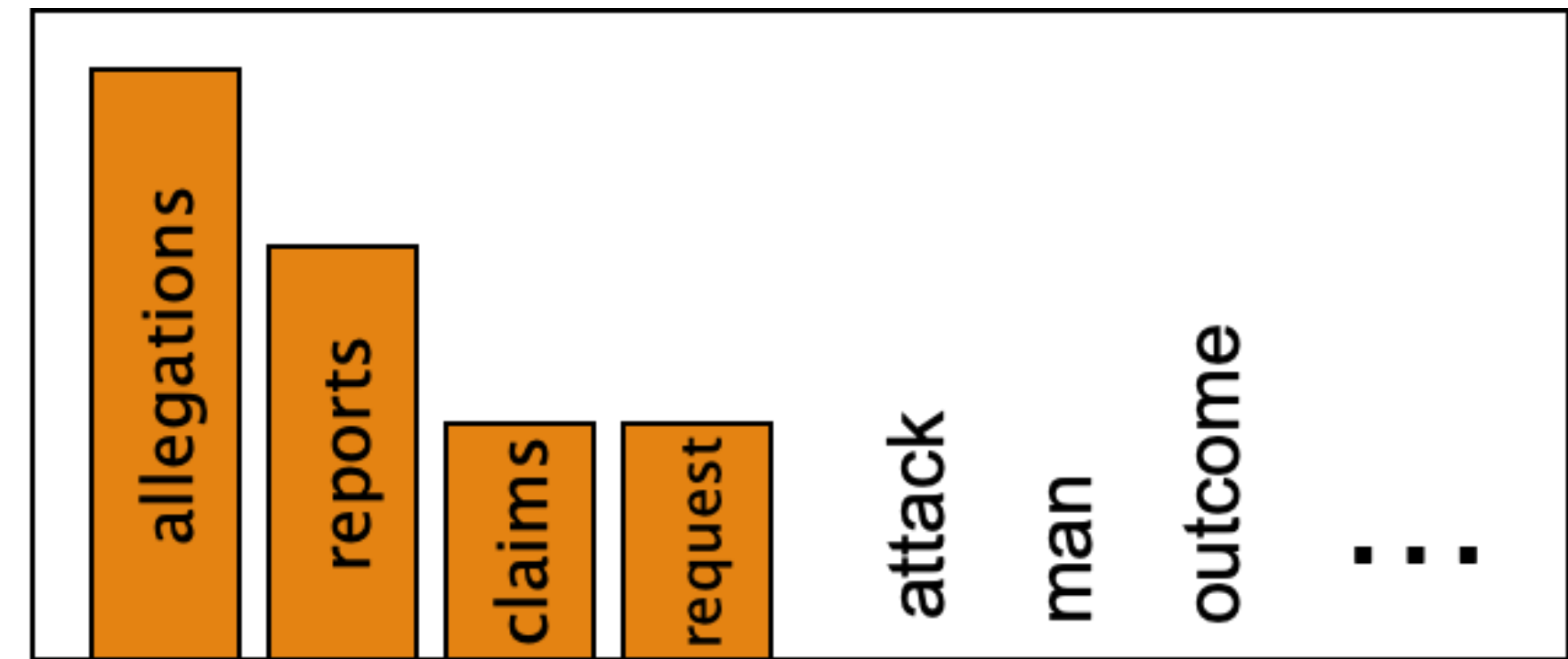
3 allegations

2 reports

1 claims

1 request

7 total



Steal probability mass to generalize better:

P(w | denied the)

2.5 allegations

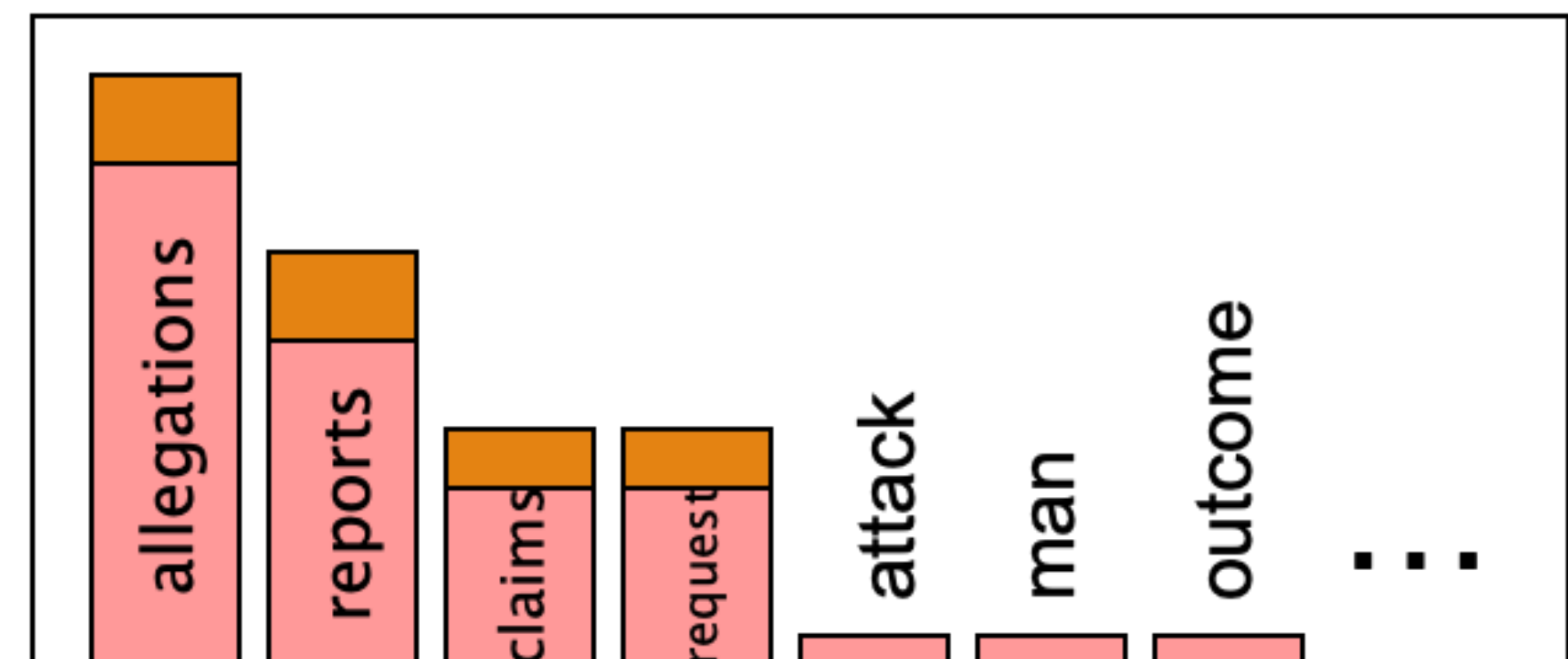
1.5 reports

0.5 claims

0.5 request

2 other

7 total



Example from Dan Klein

Add-one (Laplace) smoothing

Considering a bigram model here, pretend we saw each word one more time than we did.

MLE estimate:

$$q_{\text{MLE}}(w_i | w_{i-1}) = \frac{c(w_{i-1}, w_i)}{c(w_{i-1})}$$

Add-one smoothing:

$$q_{\text{Laplace}}(w_i | w_{i-1}) = \frac{c(w_{i-1}, w_i) + 1}{c(w_{i-1}) + |\mathcal{V}|}$$

Linear Interpolation (Stupid Backoff)

Trigram Model, Bigram Model, Unigram Model

Trigram maximum-likelihood estimate: $q(w_i | w_{i-2}, w_{i-1}) = \frac{c(w_{i-2}, w_{i-1}, w_i)}{c(w_{i-2}, w_{i-1})}$

Bigram maximum-likelihood estimate: $q(w_i | w_{i-1}) = \frac{c(w_{i-1}, w_i)}{c(w_{i-1})}$

Unigram maximum-likelihood estimate: $q(w_i) = \frac{c(w_i)}{c(\cdot)}$

Which one suffers from the data sparsity problem the most?
Which one is more accurate?

Linear Interpolation (Stupid Backoff)

$$q(w_i | w_{i-2}, w_{i-1}) = \lambda_1 \times q_{\text{ML}}(w_i | w_{i-2}, w_{i-1}) \\ + \lambda_2 \times q_{\text{ML}}(w_i | w_{i-1}) \\ + \lambda_3 \times q_{\text{ML}}(w_i)$$

where $\lambda_1 + \lambda_2 + \lambda_3 = 1$, and $\lambda_i \geq 0$ for all i .

How to choose the value of $\lambda_1, \lambda_2, \lambda_3$

Use the held-out corpus

Hyperparameters



maximize the probability of held-out data.

Markov Models in Retrospect

Consider a sequence of random variables X_1, X_2, \dots, X_n , each take any value in \mathcal{V}

The joint probability of a sentence is

$$\begin{aligned} &P(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) \\ &= P(X_1 = x_1) \prod_{i=2}^n P(X_i = x_i | X_1 = x_1, \dots, X_{i-1} = x_{i-1}) \\ &= P(X_1 = x_1) \prod_{i=2}^n P(X_i = x_i | X_{i-1} = x_{i-1}) \end{aligned}$$

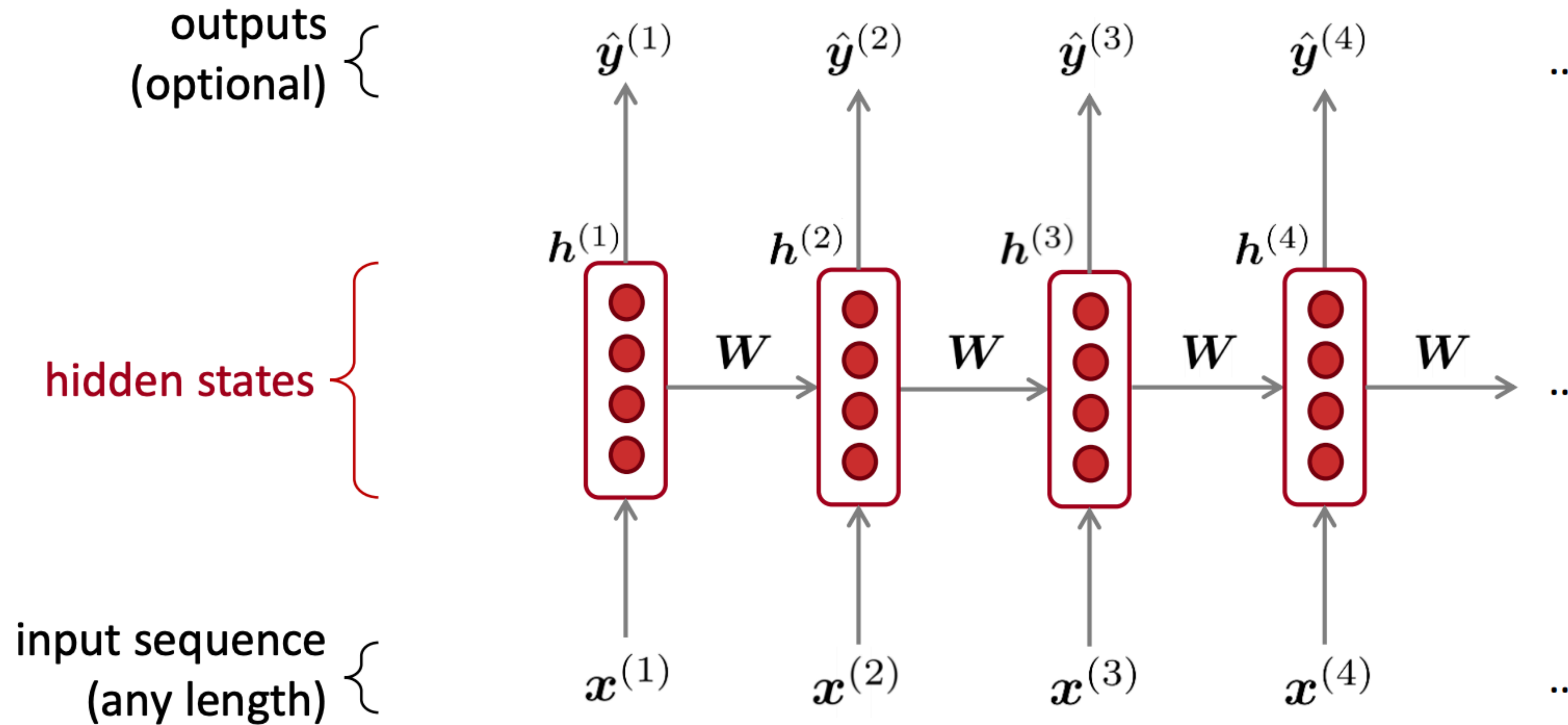


First-order Markov Assumption

$$P(X_i = x_i | X_1 = x_1, \dots, X_{i-1} = x_{i-1})$$

Is it possible to directly model this probability?

Recurrent Neural Networks (RNNs)



$$p(x^{(t)} \mid x^{(t-1)}, \dots, x^{(1)})$$

A RNN Language Model

output distribution

$$\hat{y}^{(t)} = \text{softmax}(U\mathbf{h}^{(t)} + \mathbf{b}_2) \in \mathbb{R}^{|V|}$$

hidden states

$$\mathbf{h}^{(t)} = \sigma(\mathbf{W}_h\mathbf{h}^{(t-1)} + \mathbf{W}_e\mathbf{e}^{(t)} + \mathbf{b}_1)$$

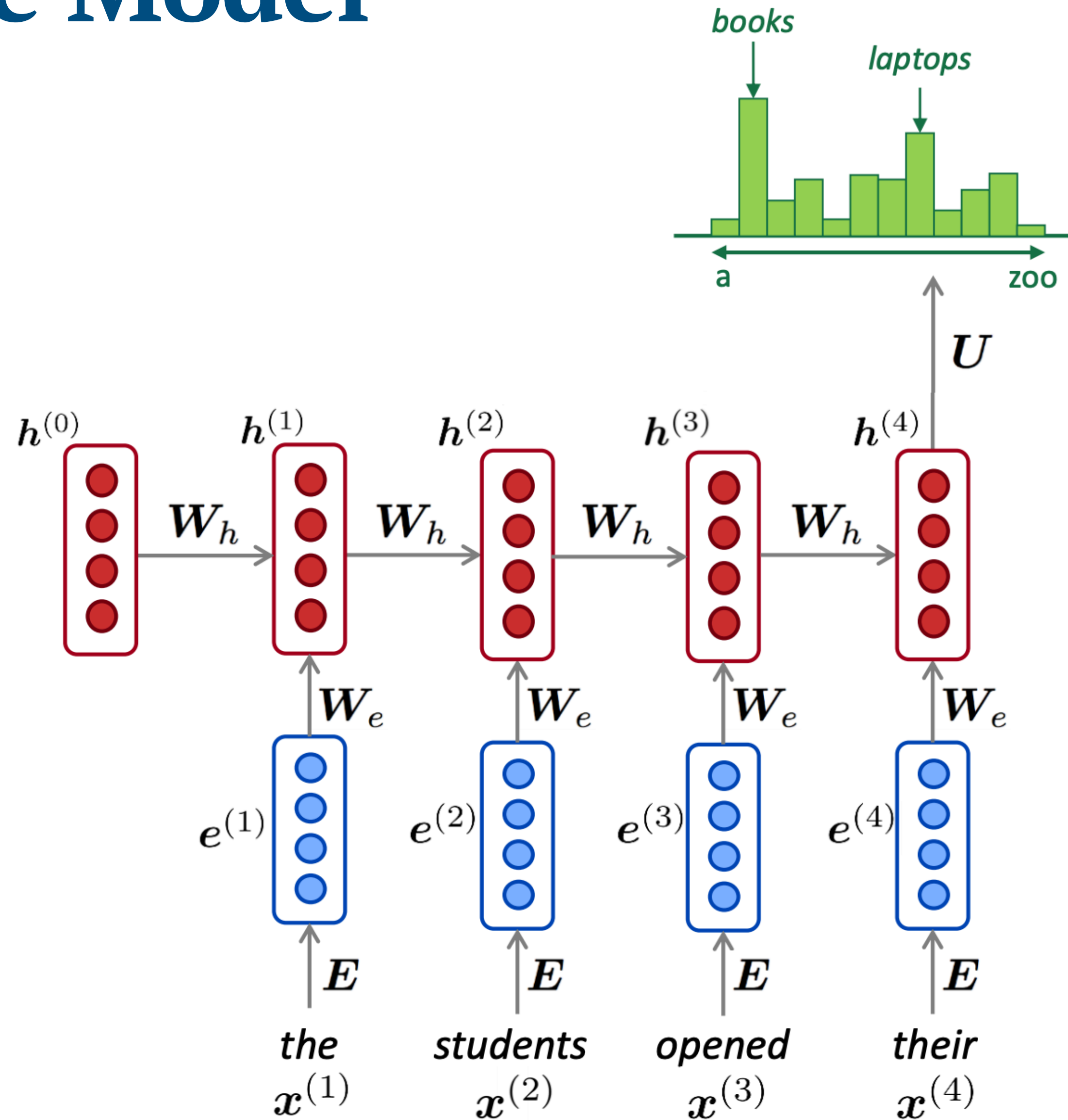
$\mathbf{h}^{(0)}$ is the initial hidden state

word embeddings

$$\mathbf{e}^{(t)} = \mathbf{E}\mathbf{x}^{(t)}$$

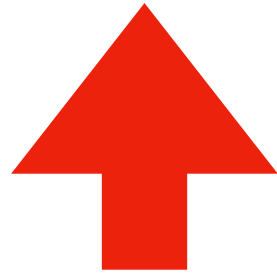
words / one-hot vectors

$$\mathbf{x}^{(t)} \in \mathbb{R}^{|V|}$$



A RNN Language Model

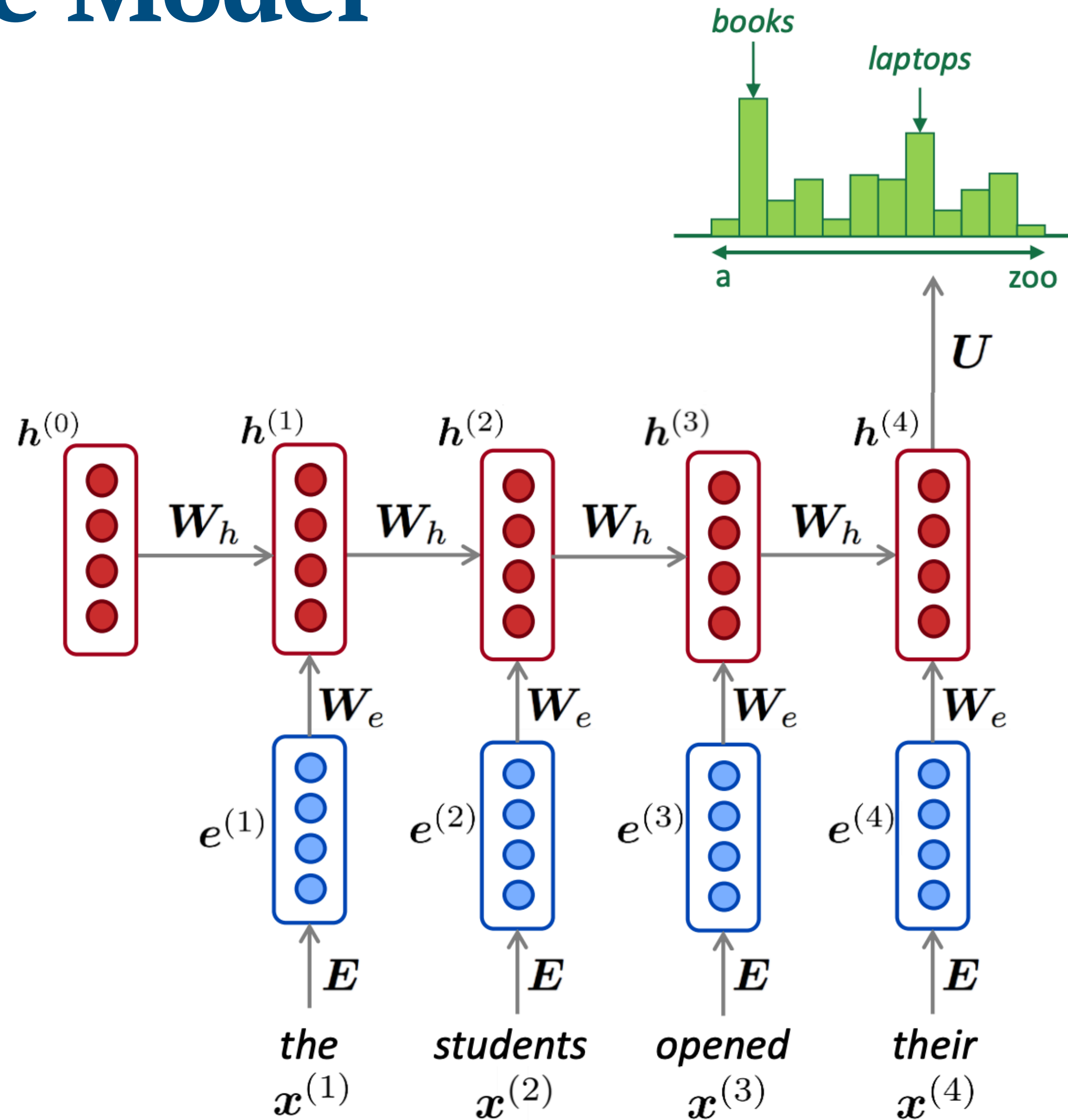
$$p(x^{(t)} \mid x^{(t-1)}, \dots, x^{(1)})$$



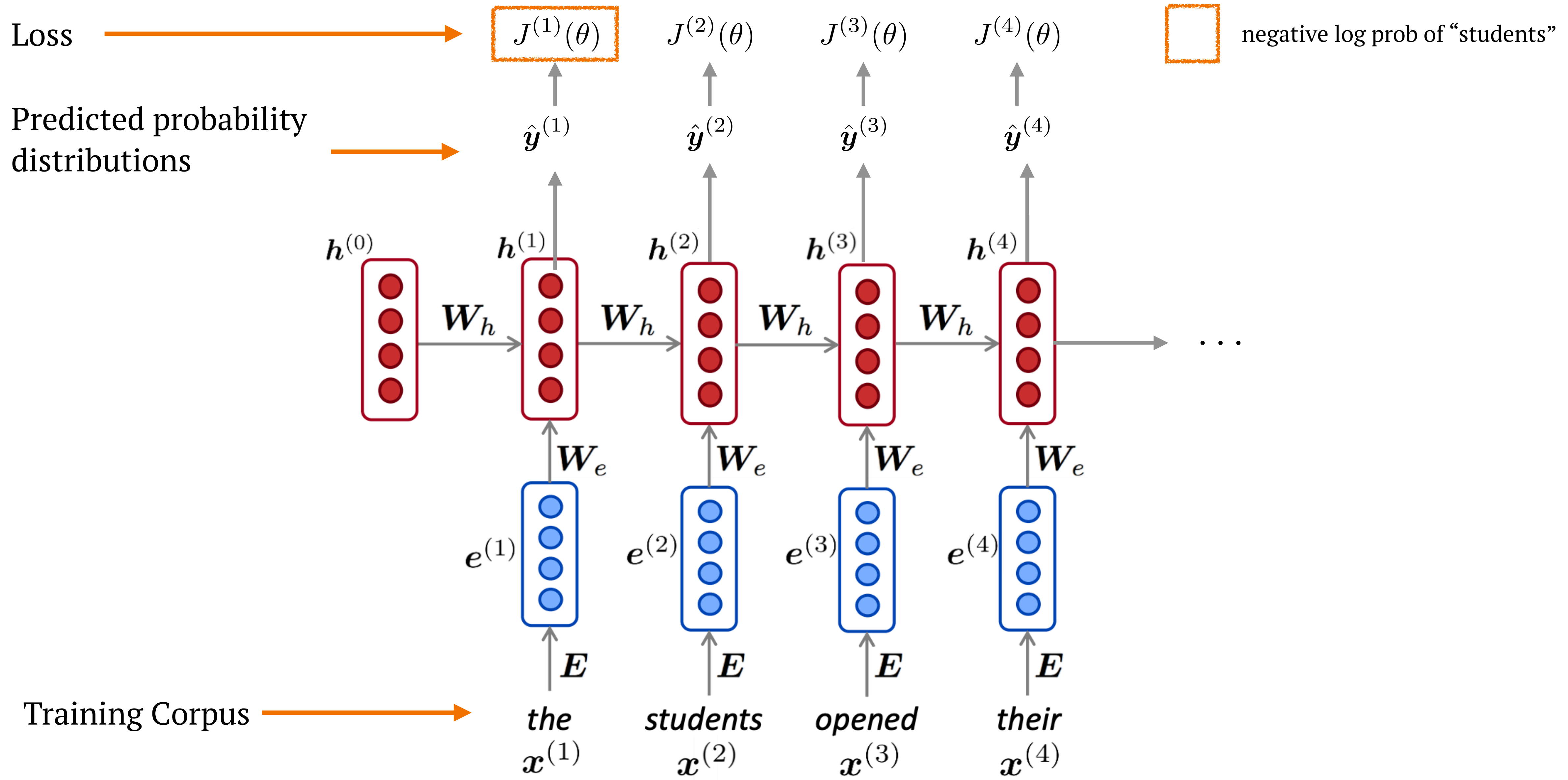
$$\hat{y}^{(t)} = \text{softmax}(U h^{(t)} + b_2) \in \mathbb{R}^{|V|}$$
$$h^{(t)} = \sigma(W_h h^{(t-1)} + W_e e^{(t)} + b_1)$$

$h^{(0)}$ is the initial hidden state

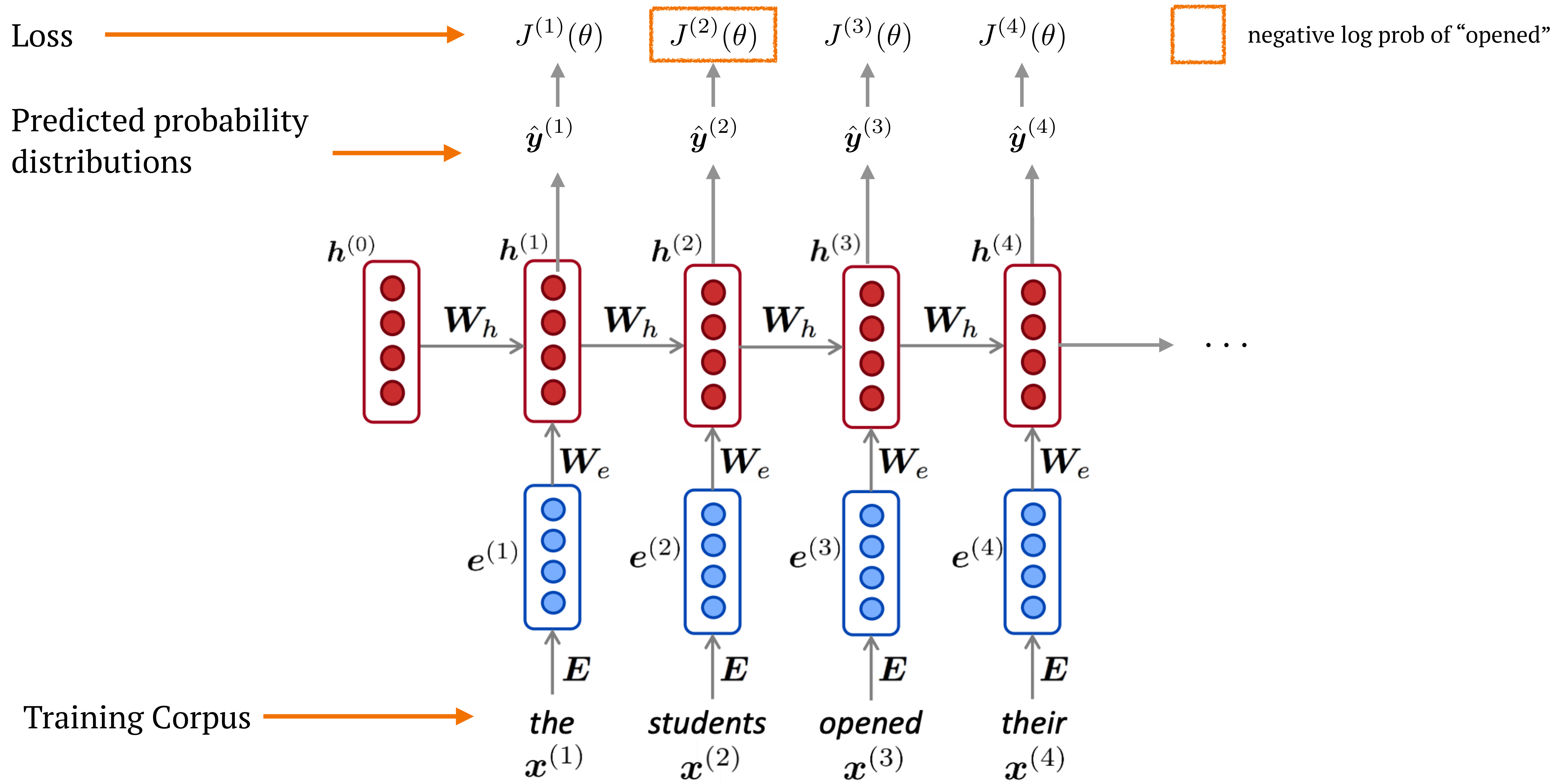
The recurrent function here, takes into consideration **all** the history!



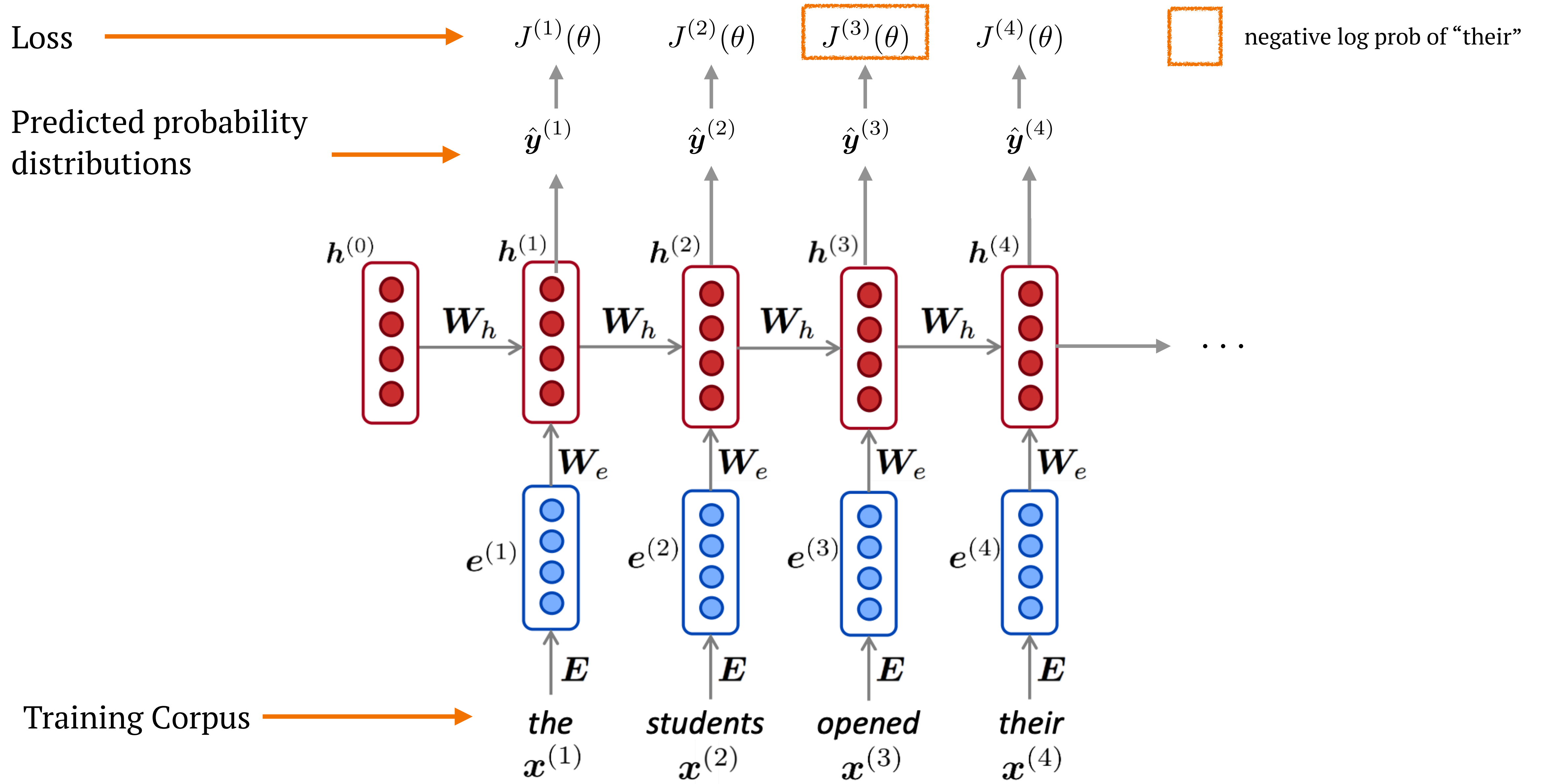
Training a RNN Language Model



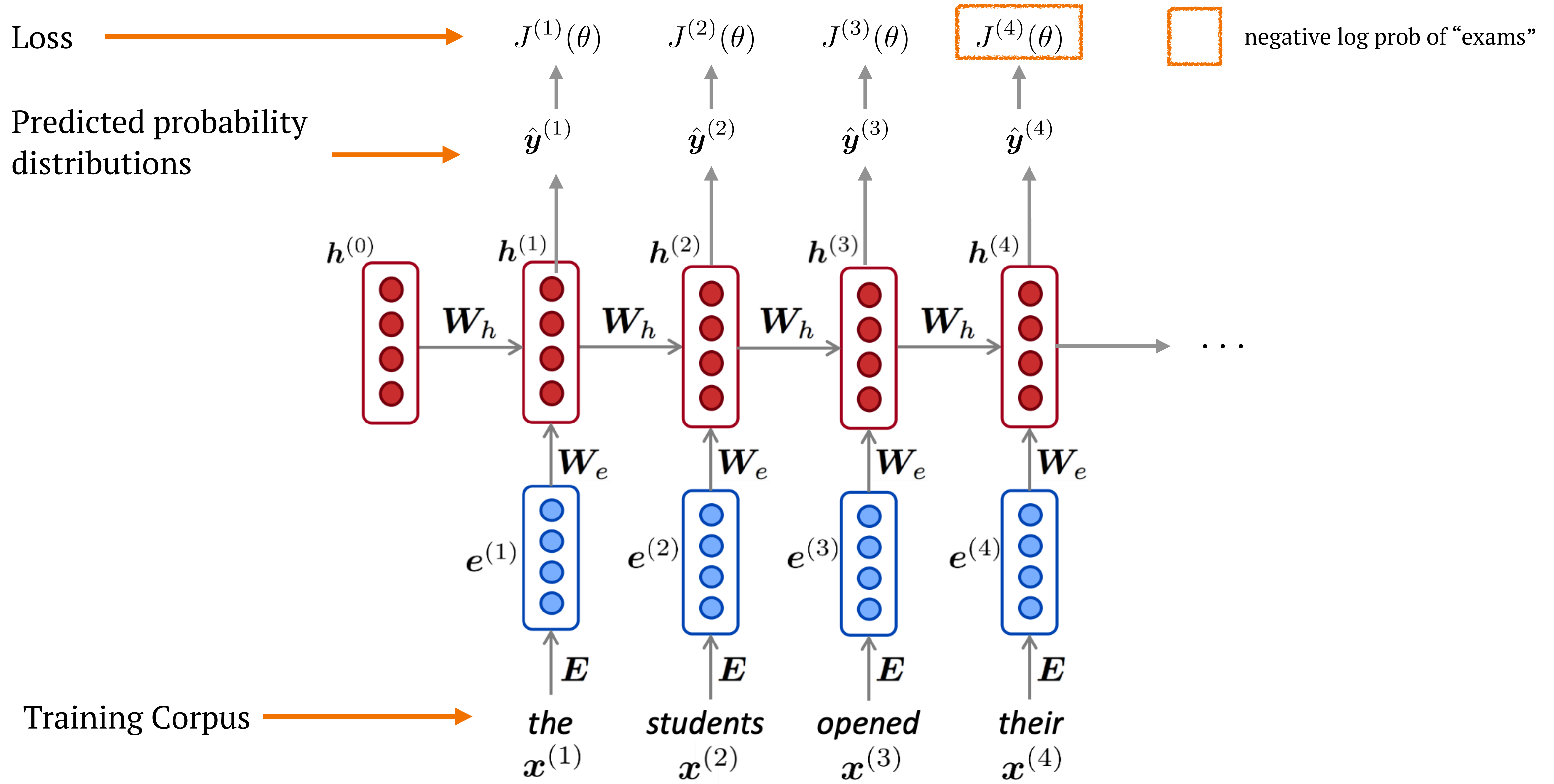
Training a RNN Language Model



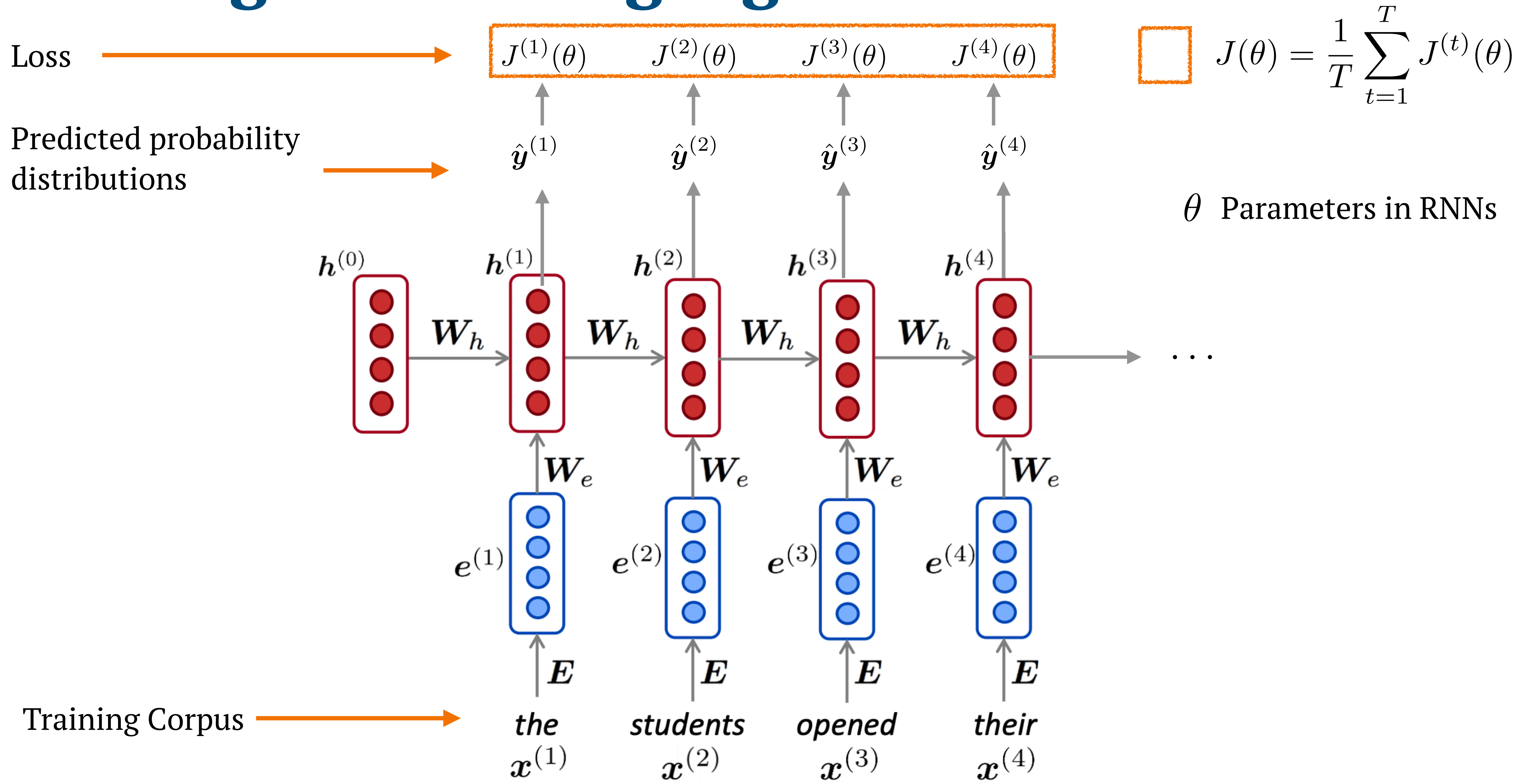
Training a RNN Language Model



Training a RNN Language Model



Training a RNN Language Model



GPT-3 Model

Loss



Predicted probability distributions



$$J^{(1)}(\theta) \quad J^{(2)}(\theta) \quad J^{(3)}(\theta) \quad J^{(4)}(\theta)$$

$$\hat{y}^{(1)} \quad \hat{y}^{(2)} \quad \hat{y}^{(3)} \quad \hat{y}^{(4)}$$



$$\square J(\theta) = \frac{1}{T} \sum_{t=1}^T J^{(t)}(\theta)$$

θ Parameters in Transformer

What else can we do?

Assign a probability to a sentence

$P(\text{"I am going to school"}) > P(\text{"I are going to school"})$

Grammar Checking

I had some coffee this morning.

$P(\text{"我今早喝了一些咖啡"}) > P(\text{"我今早吃了一些咖啡"})$

Machine translation

$P(\text{"Can we put an elephant into the refrigerator? No, we can't.}) > P(\text{"Can we put an elephant into the refrigerator? Yes, we can.})$

Question Answering