

# An Introduction to (Large) Language Models

Swabha Swayamdipta  
Assistant Professor of Computer Science  
Introduction to Engineering by Prof. Darin Gray  
June 27, 2024

USC Viterbi

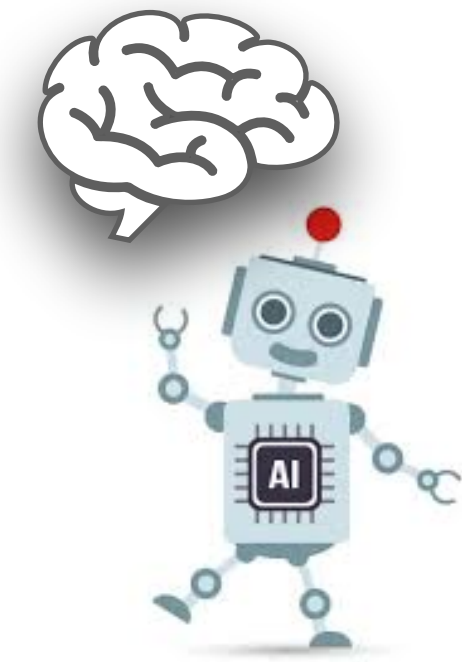
School of Engineering

Some slides adapted from [Greg Durrett, UT Austin](#) and [Google Cloud Tech](#)

## Artificial Intelligence

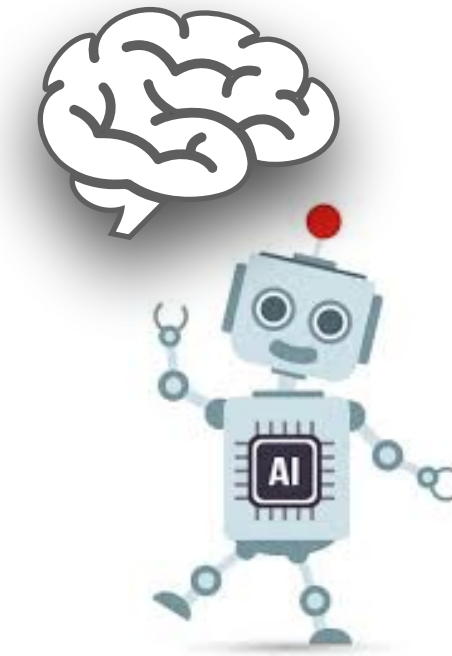
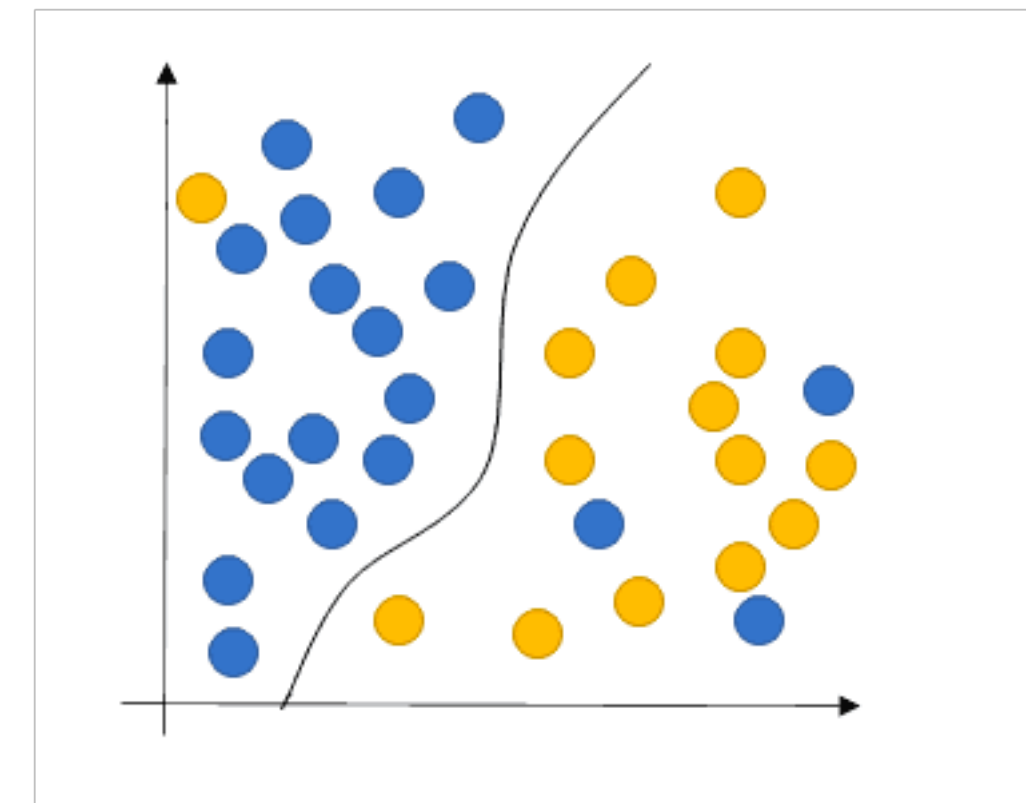


## Artificial Intelligence



## Artificial Intelligence

### Machine Learning and Natural Language Processing

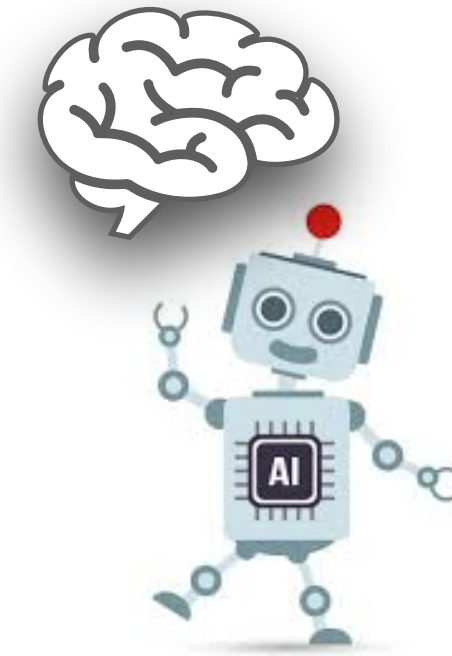
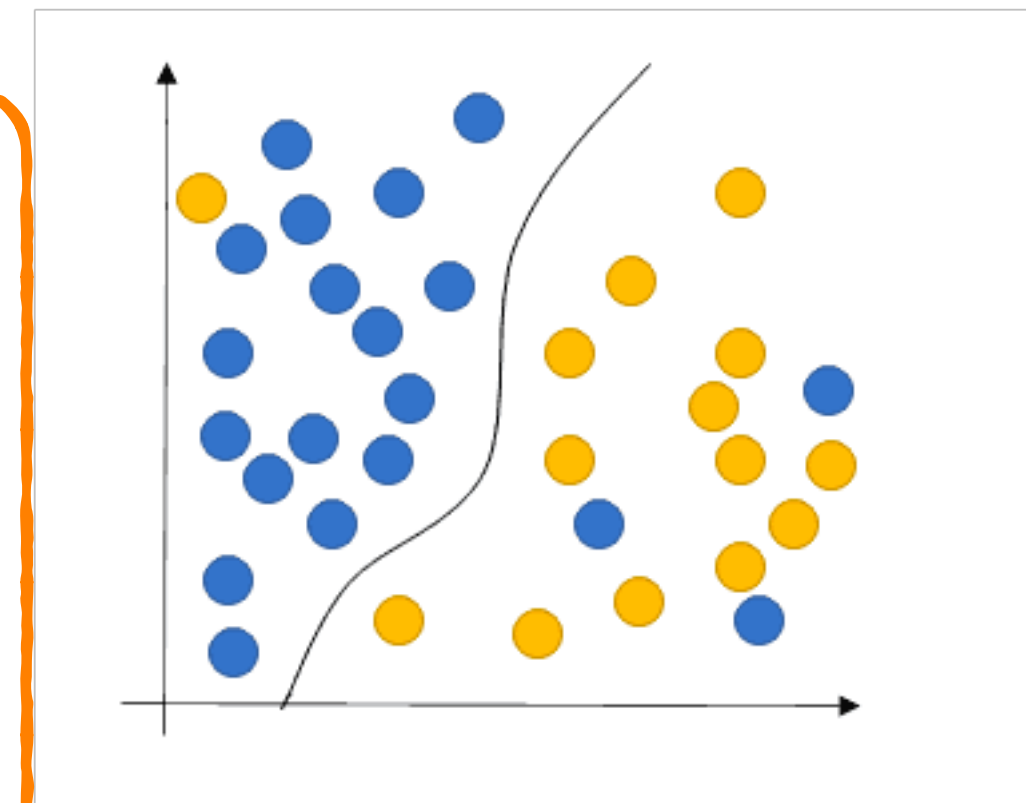
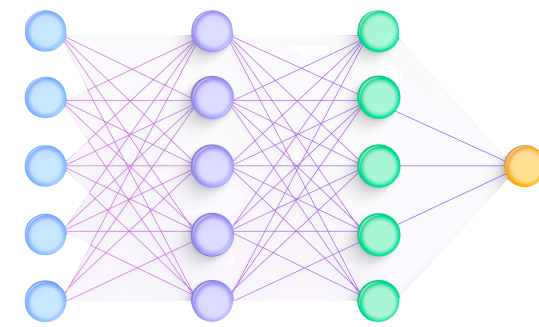




## Artificial Intelligence

### Machine Learning and Natural Language Processing

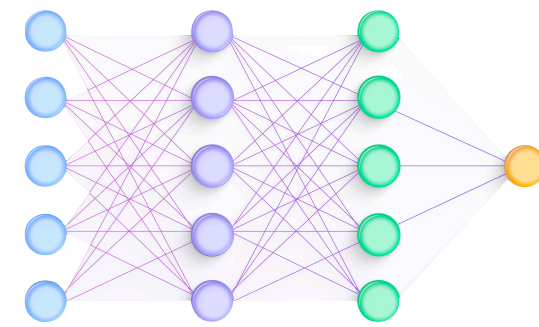
#### Deep Learning



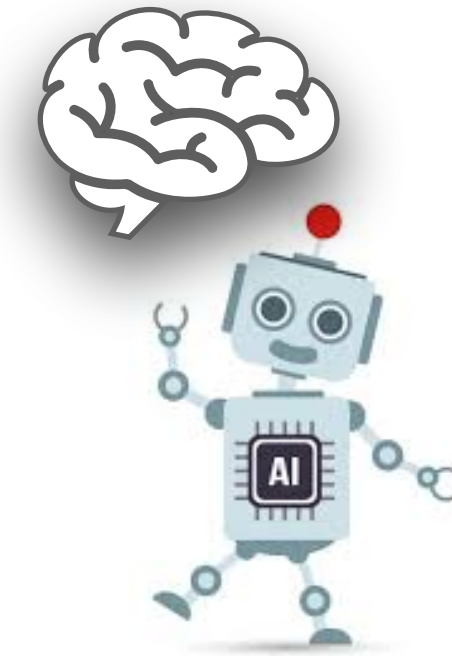
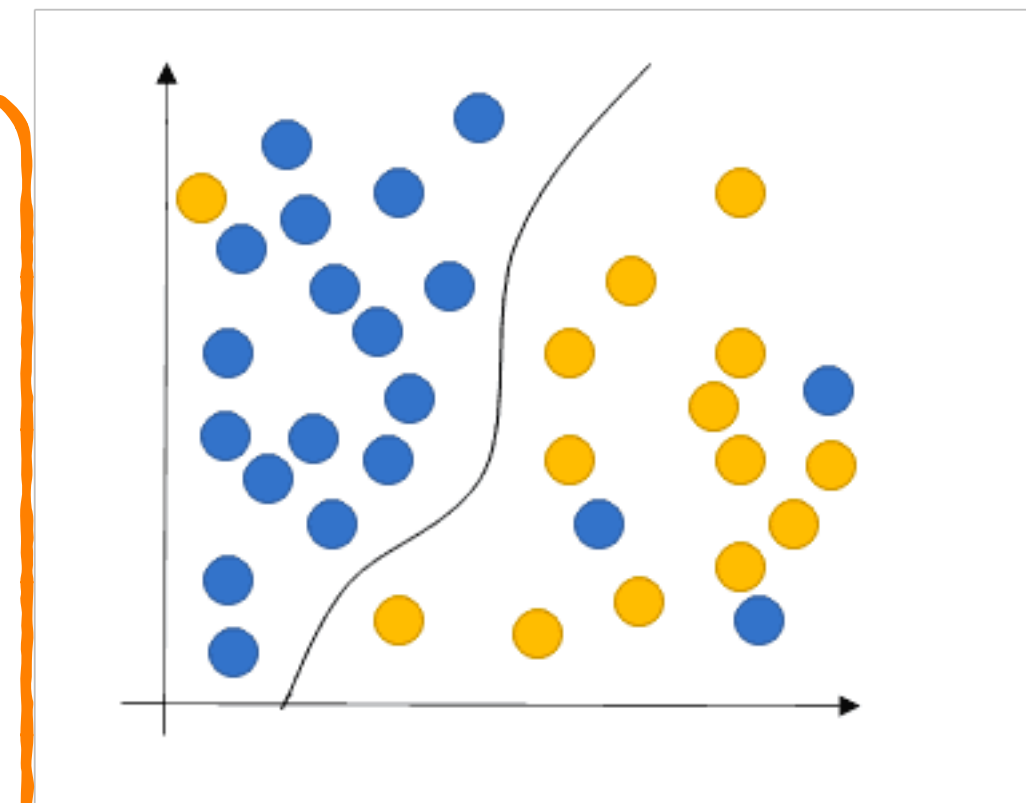
## Artificial Intelligence

### Machine Learning and Natural Language Processing

#### Deep Learning



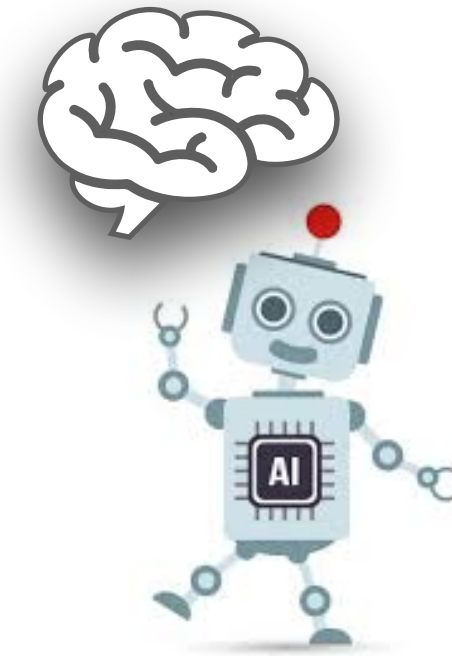
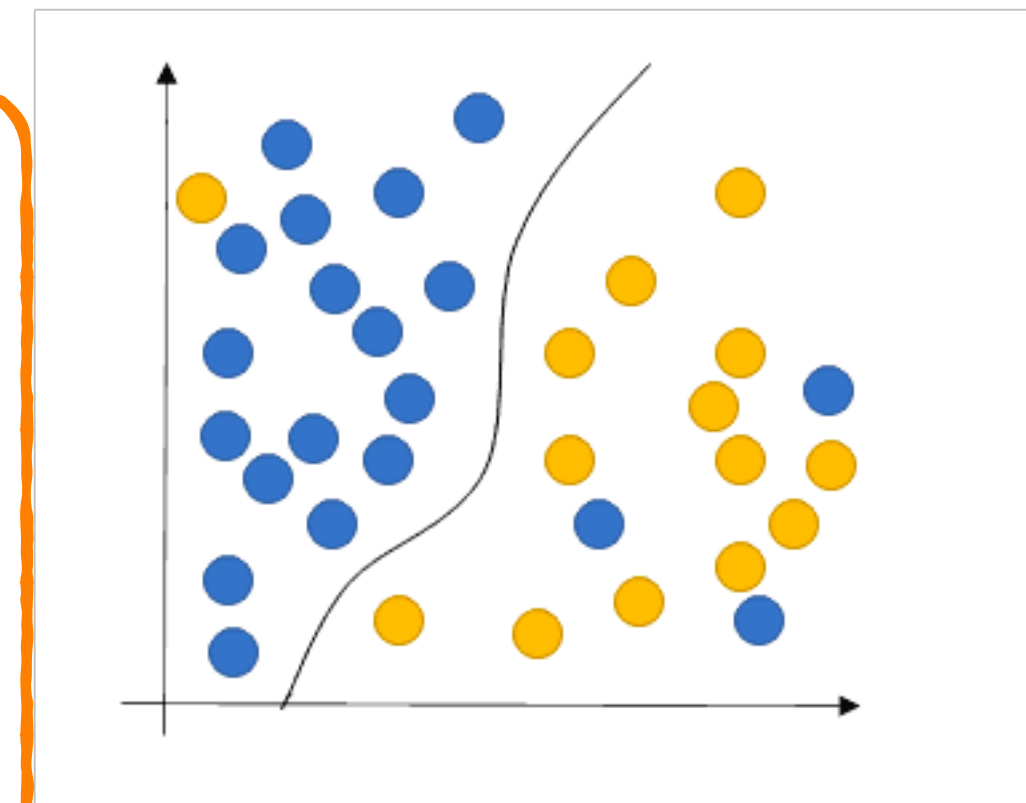
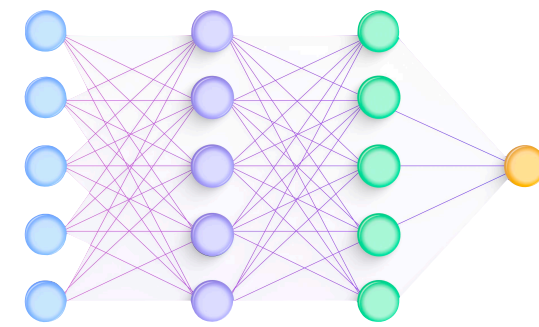
#### Generative AI



## Artificial Intelligence

### Machine Learning and Natural Language Processing

#### Deep Learning



#### Generative AI

Language  
Models



# Language Models Are Everywhere



**Virtual  
Assistants**



**Translation**



**Content  
Creation**



**Code  
Generation**



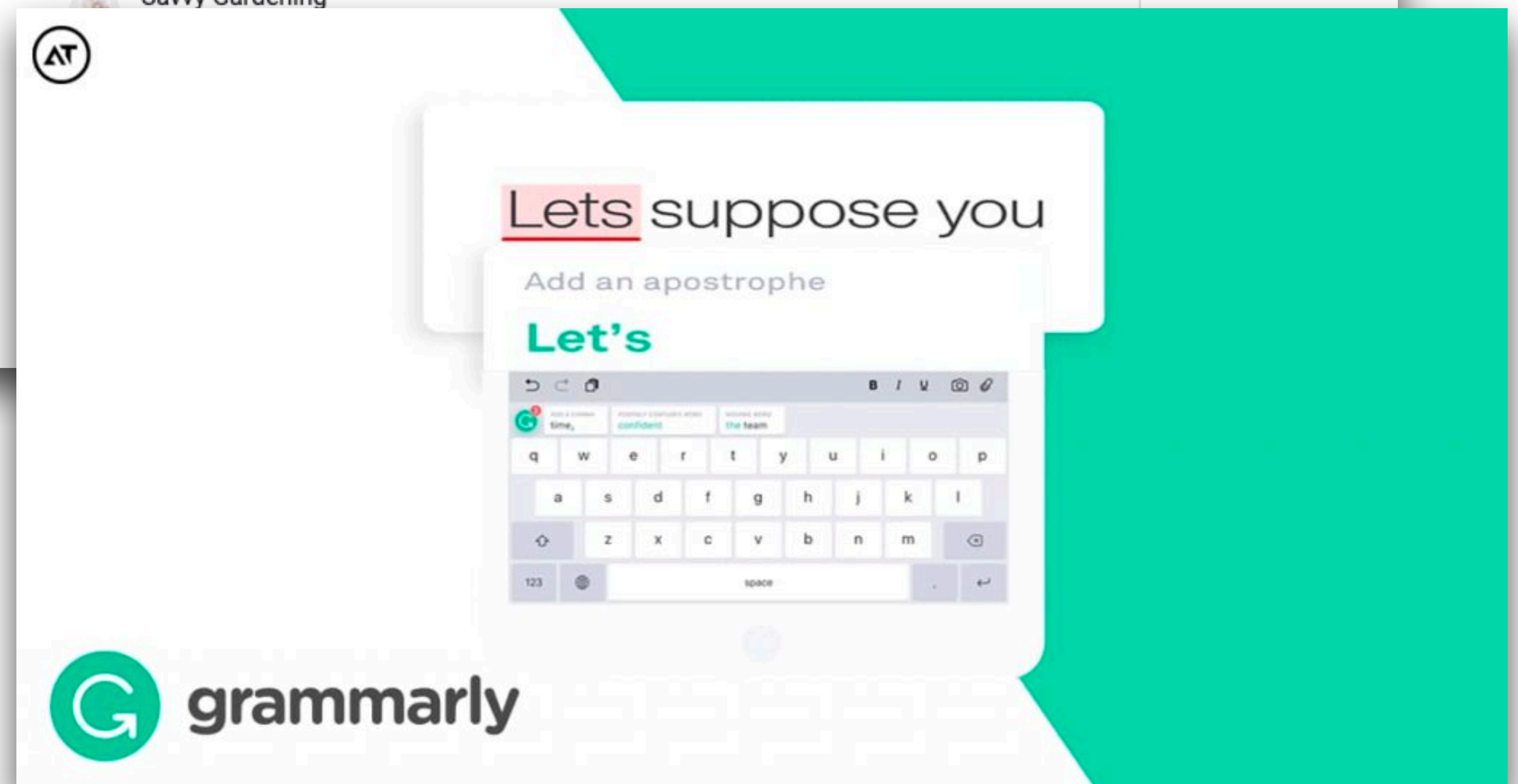
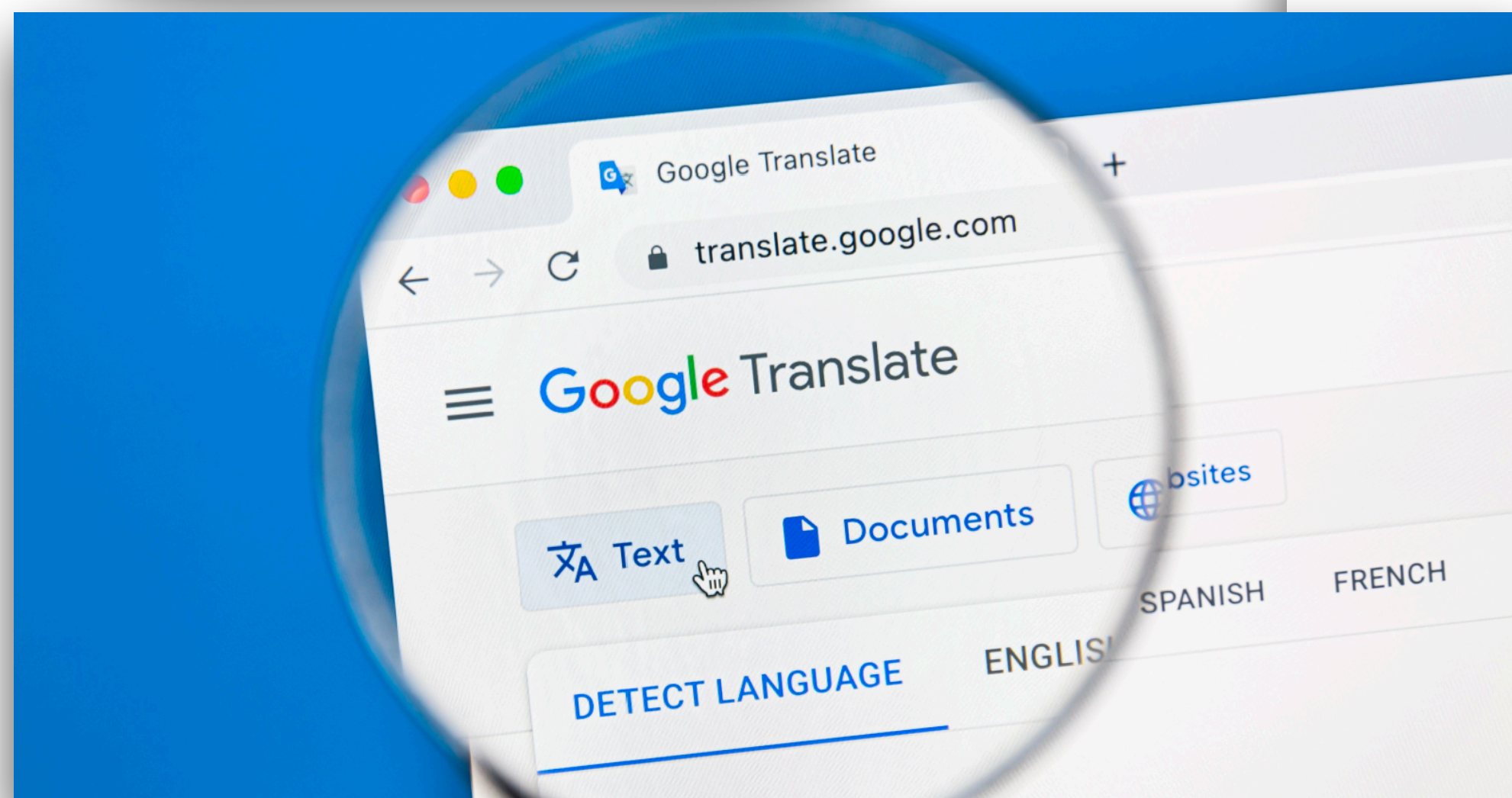
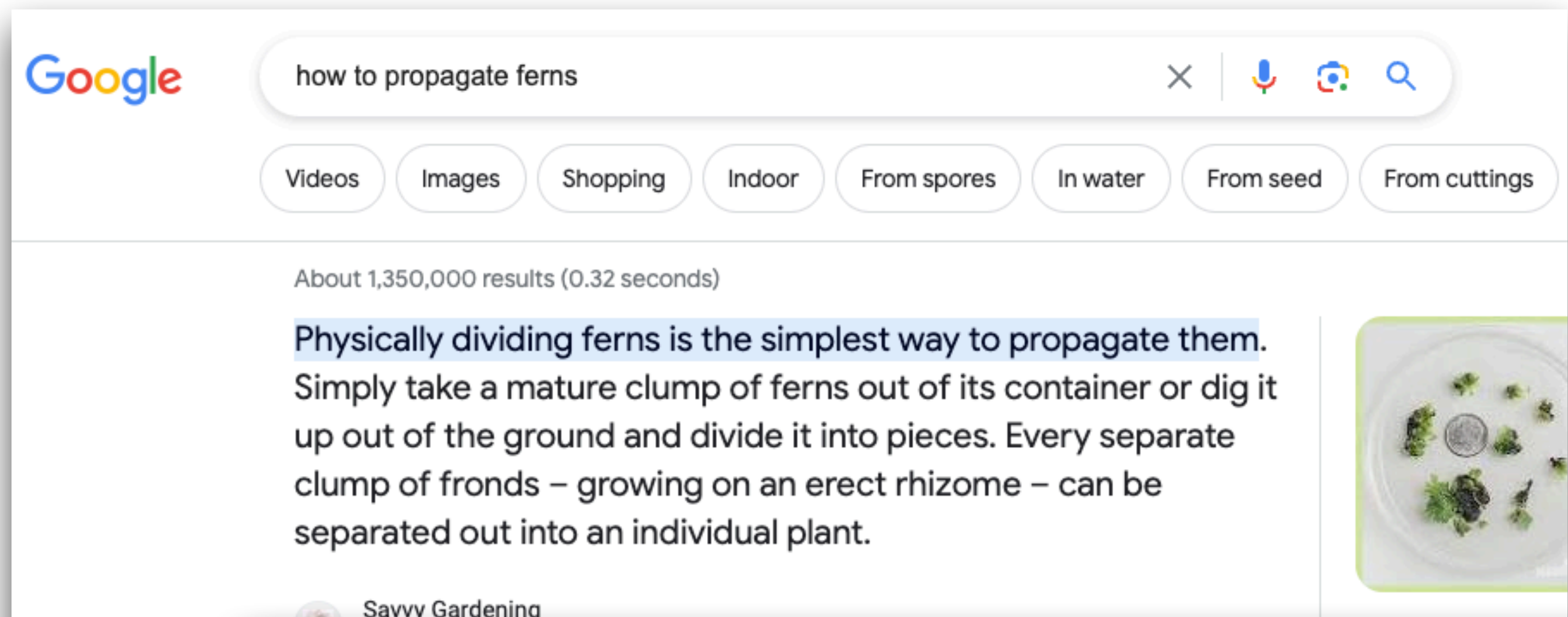
**Transcription**



**Sentiment  
Analysis**









# Apple's Siri



# Apple's Siri





# Apple's Siri



Hey Siri, set an  
alarm for 7am every  
day



# Apple's Siri



Hey Siri, set an  
alarm for 7am every  
day

Okay, your  
alarm is set



# Apple's Siri

- Understands the user



Hey Siri, set an  
alarm for 7am every  
day

Okay, your  
alarm is set



# Apple's Siri

- Understands the user



Hey Siri, set an alarm for 7am every day

Okay, your alarm is set

When is my next alarm?





# Apple's Siri

- Understands the user



Hey Siri, set an alarm for 7am every day

Okay, your alarm is set

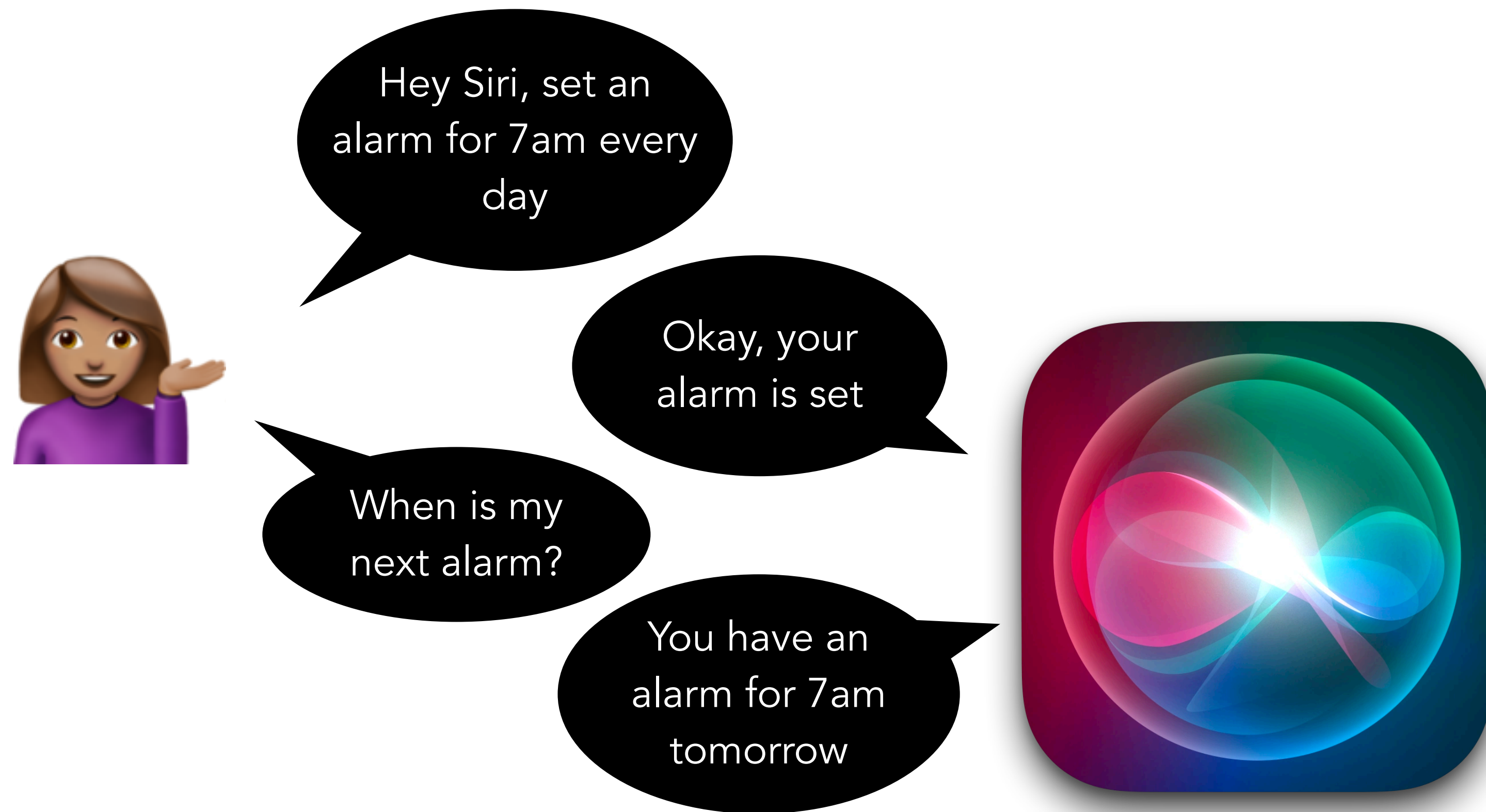
When is my next alarm?

You have an alarm for 7am tomorrow



# Apple's Siri

- Understands the user
- Remembers what the user said earlier



# Apple's Siri

- Understands the user
- Remembers what the user said earlier





# Apple's Siri

- Understands the user
- Remembers what the user said earlier
- Can understand which alarm she is referring to

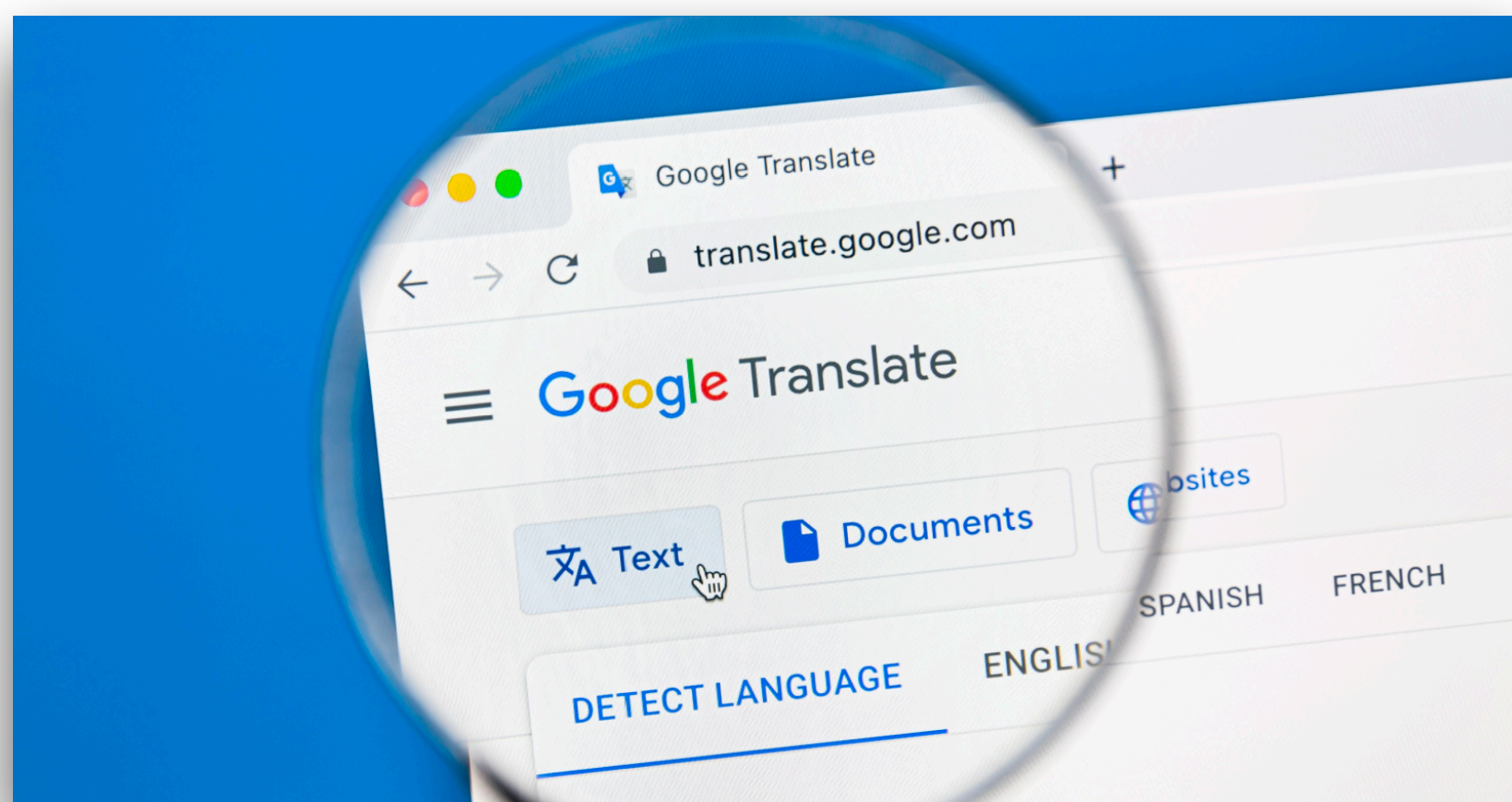




# Google Translate



# Google Translate



中共中央政治局7月30日召开会议，会议分析研究当前经济形势，部署下半年经济工作。

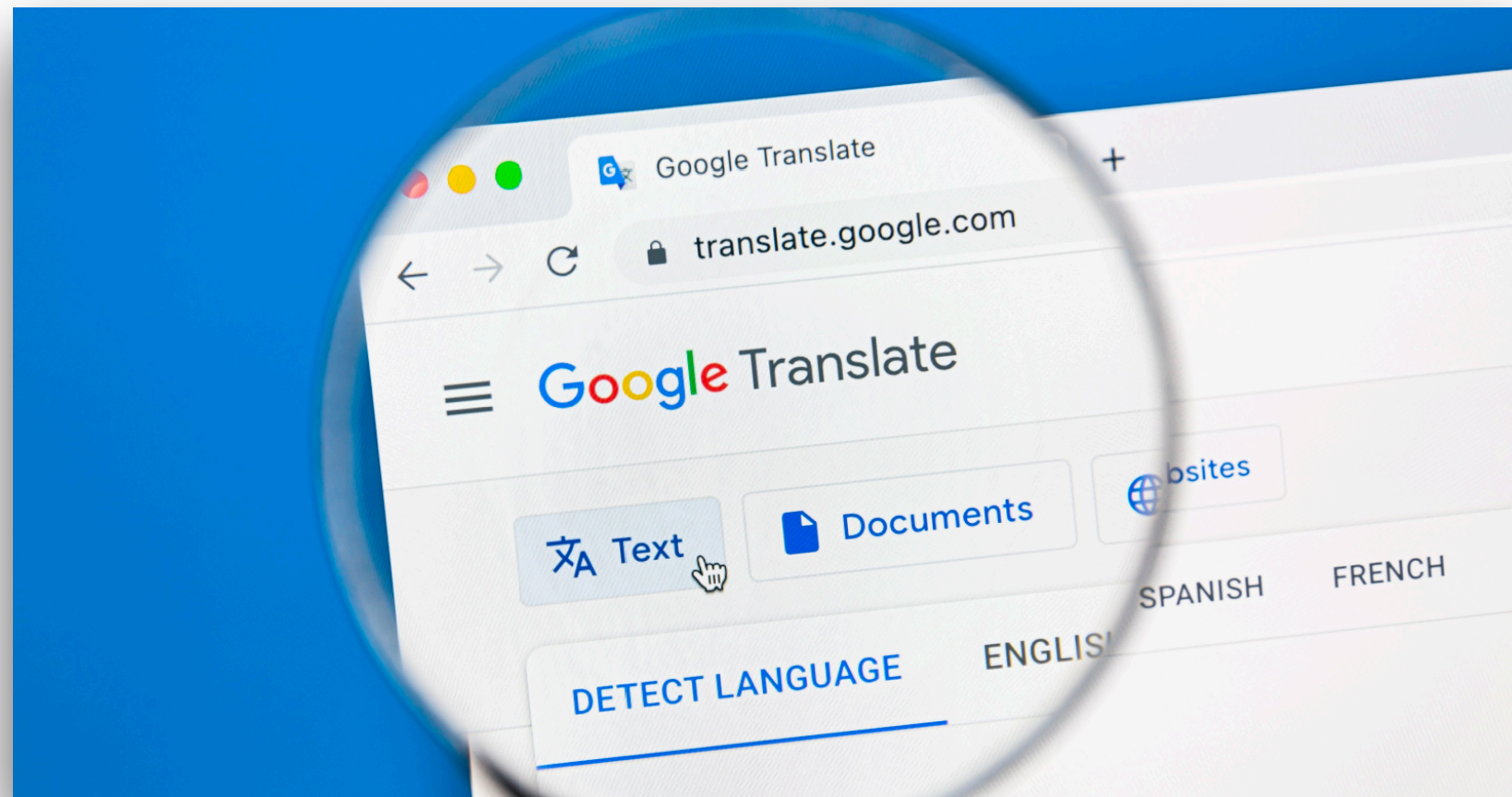
People's Daily, August 10, 2020

Translate

The Political Bureau of the CPC Central Committee held a meeting on July 30 to analyze and study the current economic situation and plan economic work in the second half of the year.



# Google Translate



- Detects language automatically

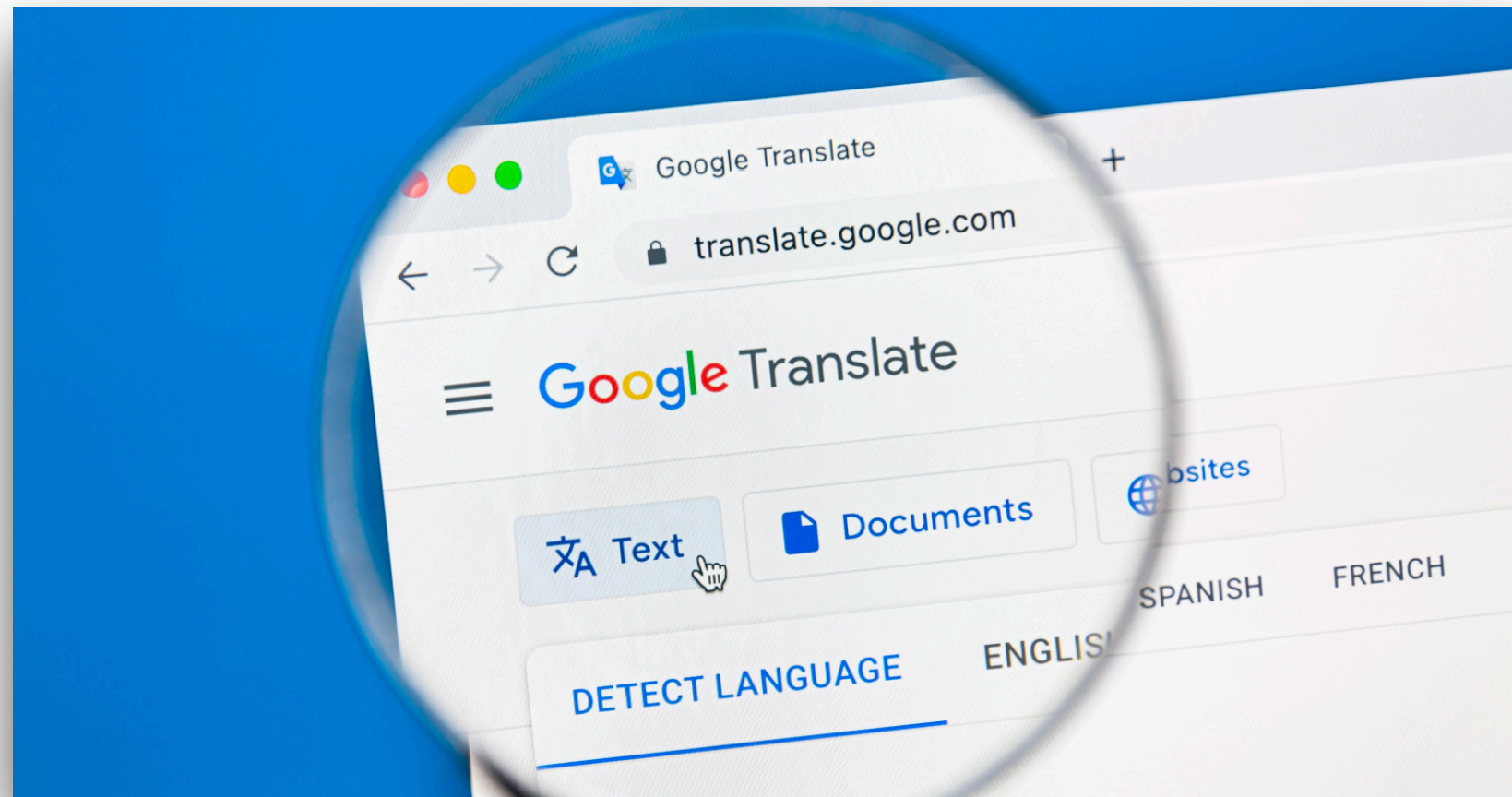
中共中央政治局7月30日召开会议，会议分析研究当前经济形势，部署下半年经济工作。

People's Daily, August 10, 2020

Translate

The Political Bureau of the CPC Central Committee held a meeting on July 30 to analyze and study the current economic situation and plan economic work in the second half of the year.

# Google Translate



- Detects language automatically

The Political Bureau  
of the CPC Central  
Committee July 30 hold a meeting

中共中央政治局7月30日召开会议，会议分析研究当前经济  
济形势，部署下半年经济工作。

People's Daily, August 10, 2020

Translate

The Political Bureau of the CPC Central Committee held a meeting on July 30 to analyze and study the current economic situation and plan economic work in the second half of the year.



# Google Translate



- Detects language automatically
- Can reorder spans in text on the fly

The Political Bureau  
of the CPC Central  
Committee July 30 hold a meeting

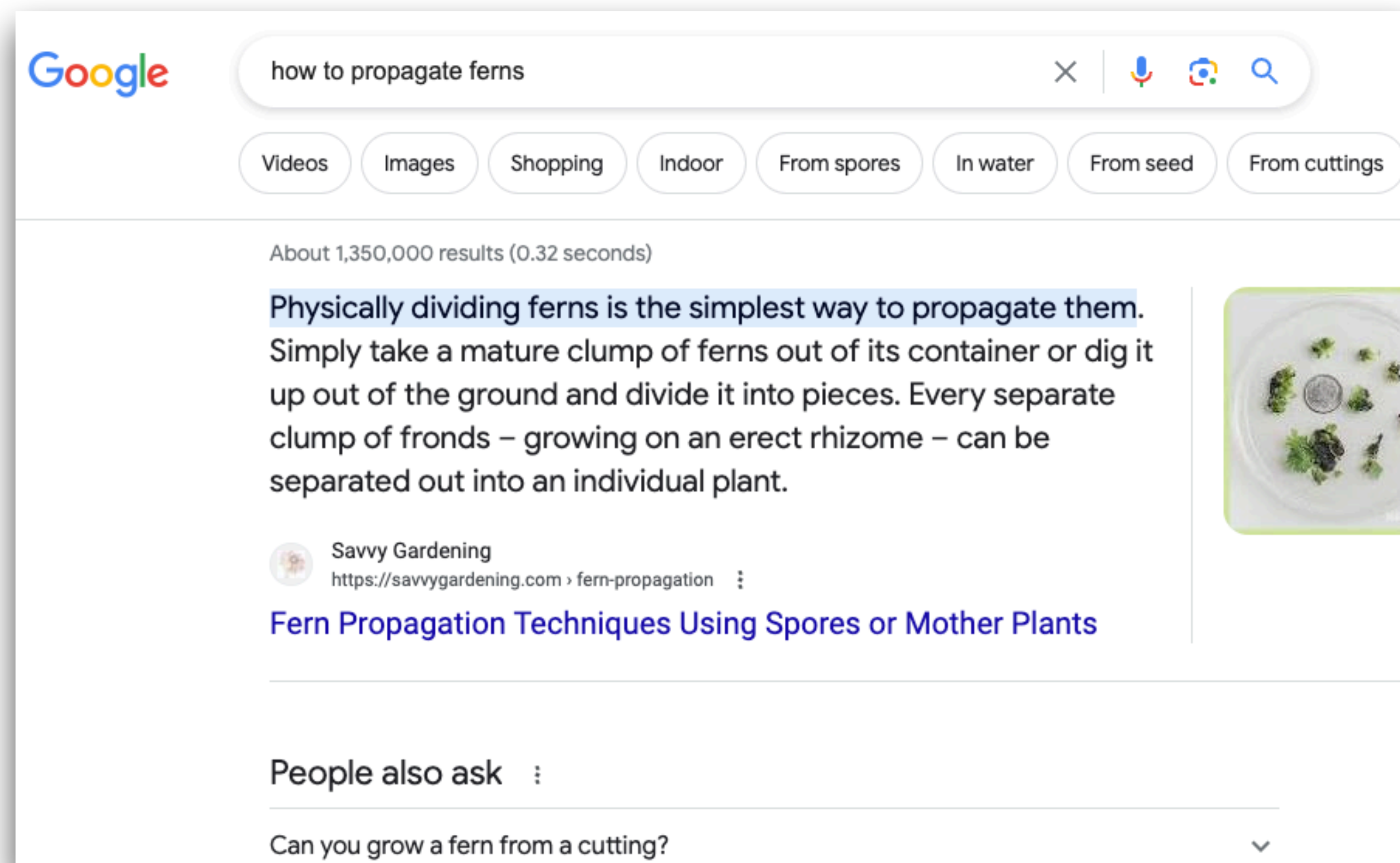
中共中央政治局7月30日召开会议，会议分析研究当前经济  
济形势，部署下半年经济工作。

People's Daily, August 10, 2020

Translate

The Political Bureau of the CPC Central Committee held a meeting on July 30 to analyze and study the current economic situation and plan economic work in the second half of the year.

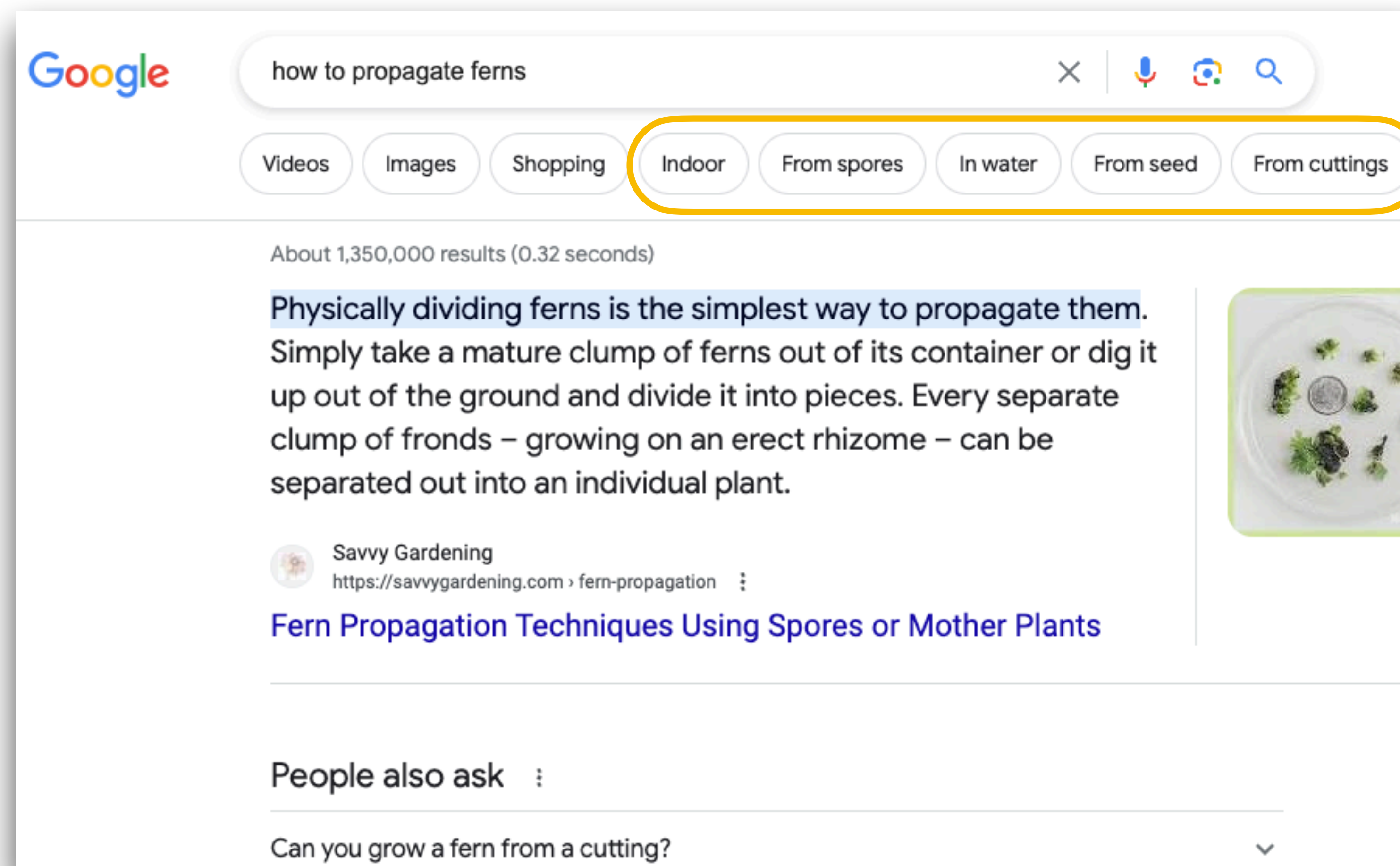
# Google Search





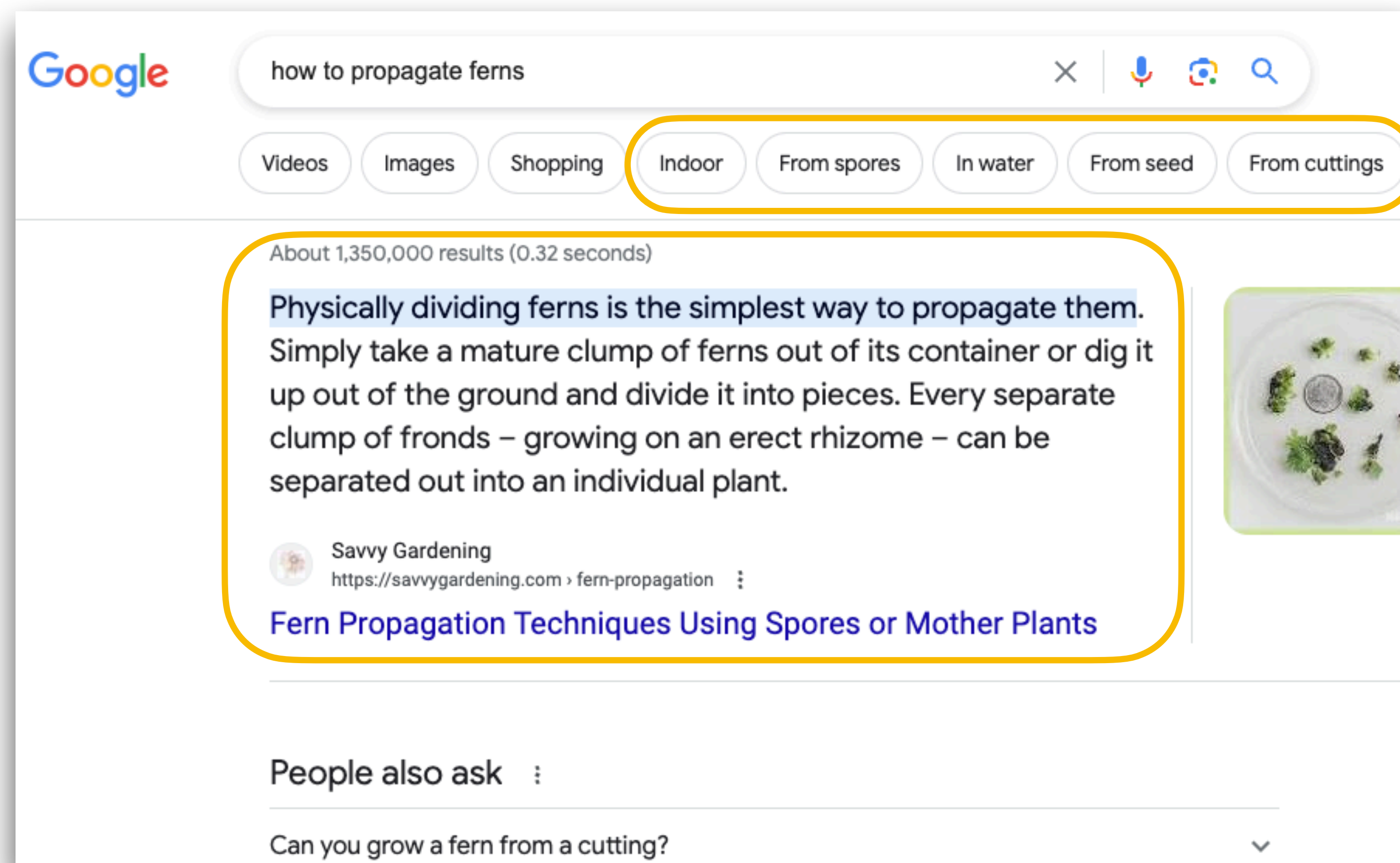
# Google Search

- Understands that a fern can be indoor, can be propagated either from seed or from cuttings



# Google Search

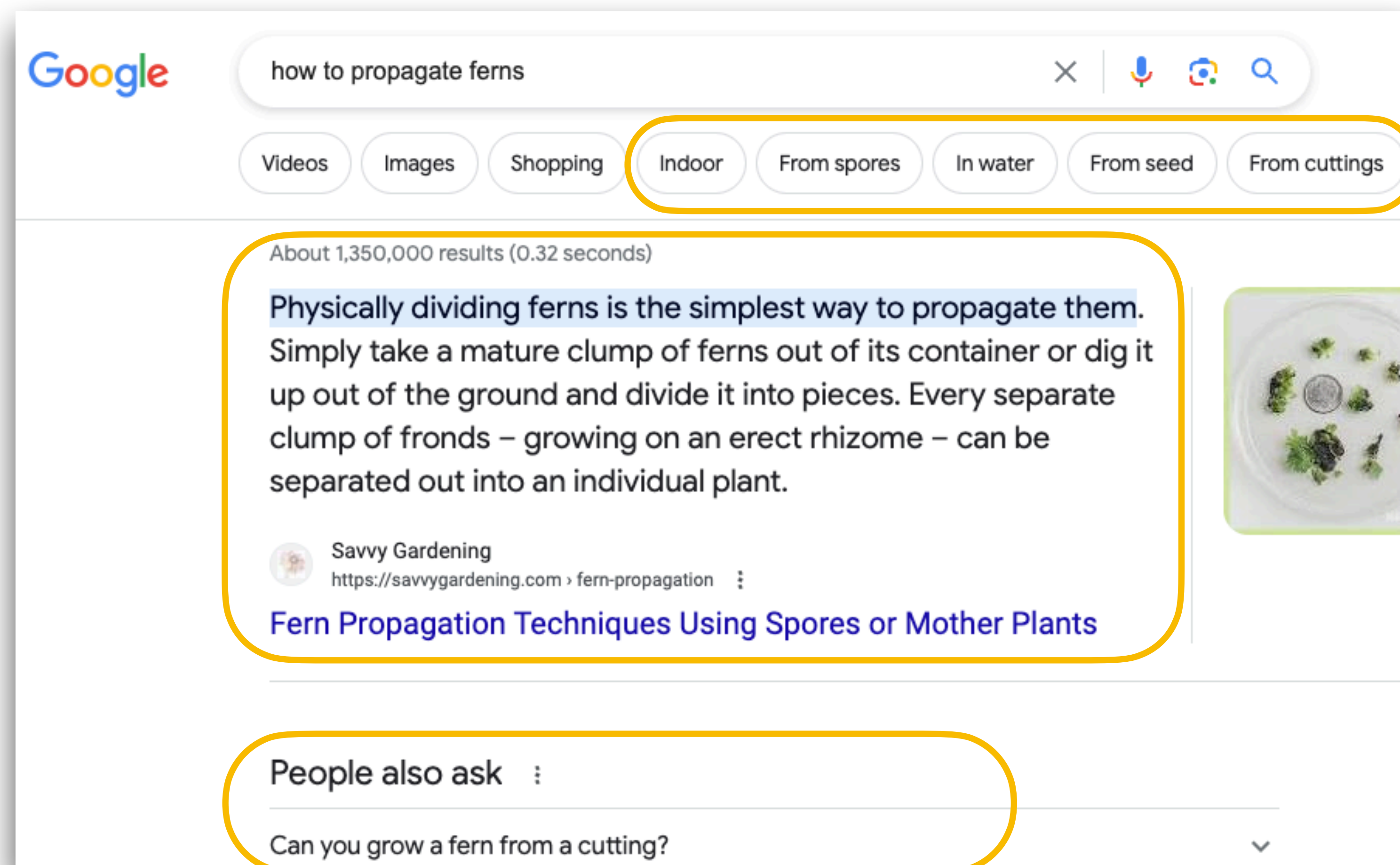
- Understands that a fern can be indoor, can be propagated either from seed or from cuttings
- Can find the exact passage in a webpage that answers the questions





# Google Search

- Understands that a fern can be indoor, can be propagated either from seed or from cuttings
- Can find the exact passage in a webpage that answers the questions
- Can find related (in meaning) questions



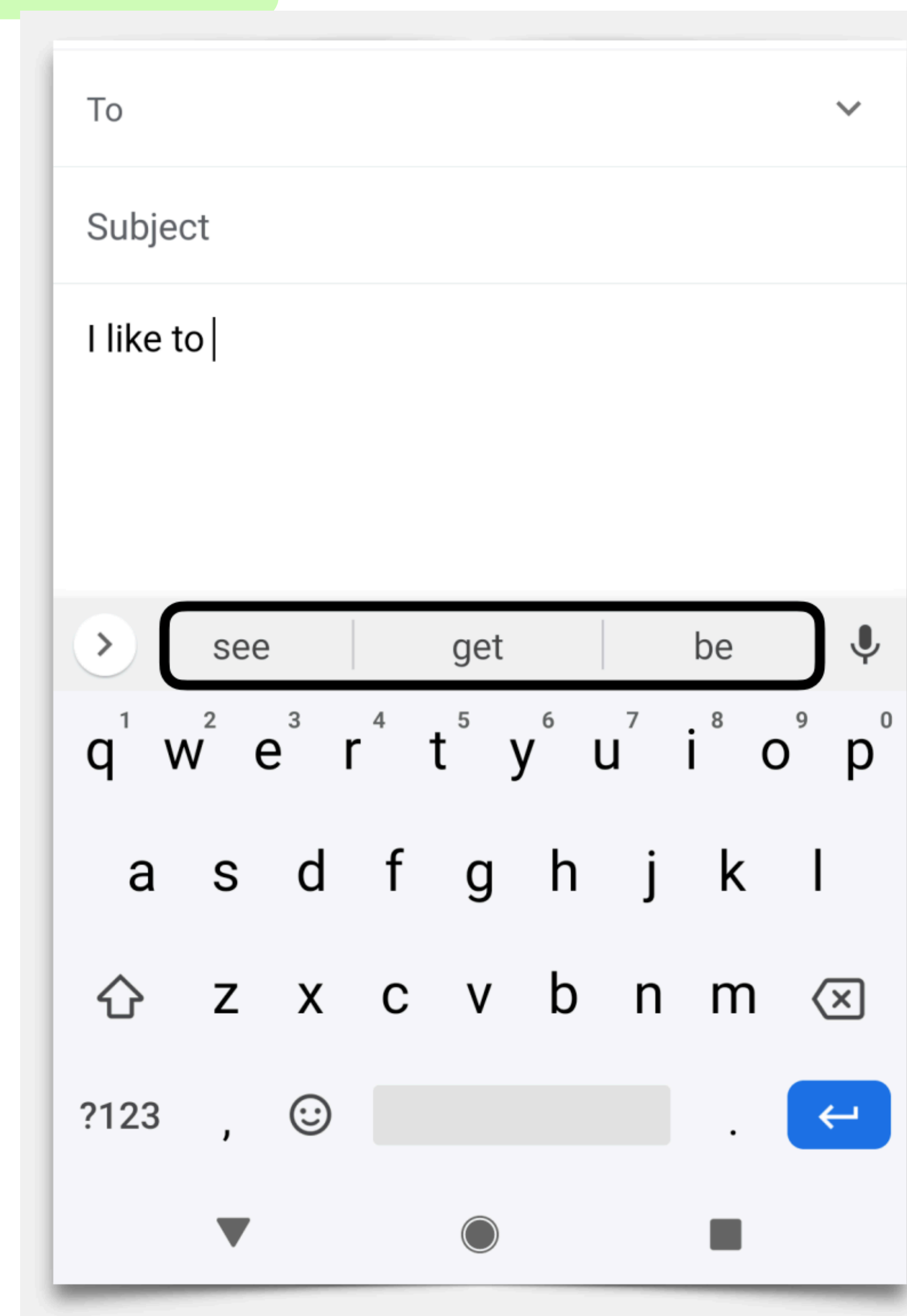
# Language Models

# Language Models

- Task: Given a sequence of words so far (**the context**), predict what comes next

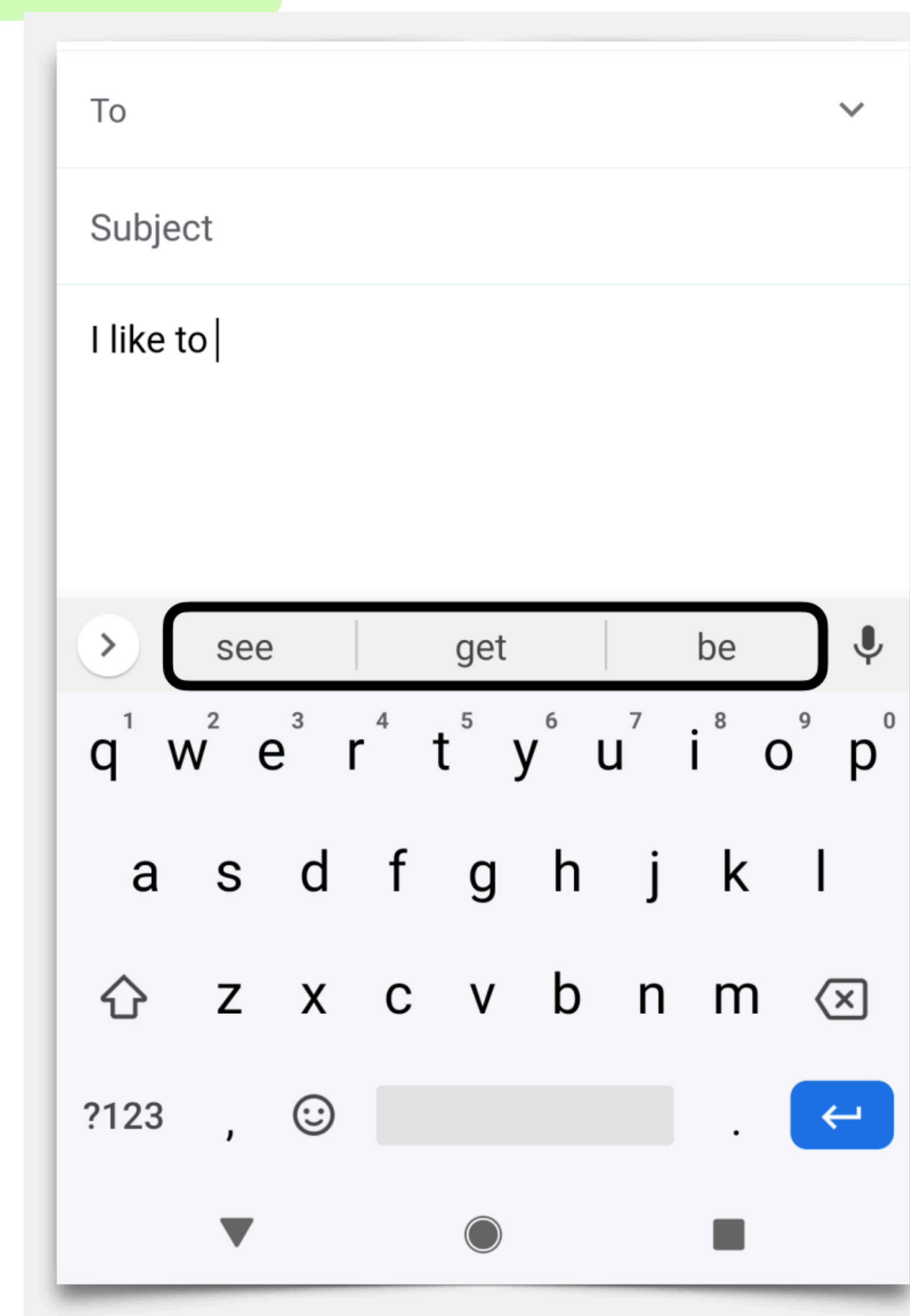
# Language Models

- Task: Given a sequence of words so far (**the context**), predict what comes next
  - Like (a supercharged) autocomplete!



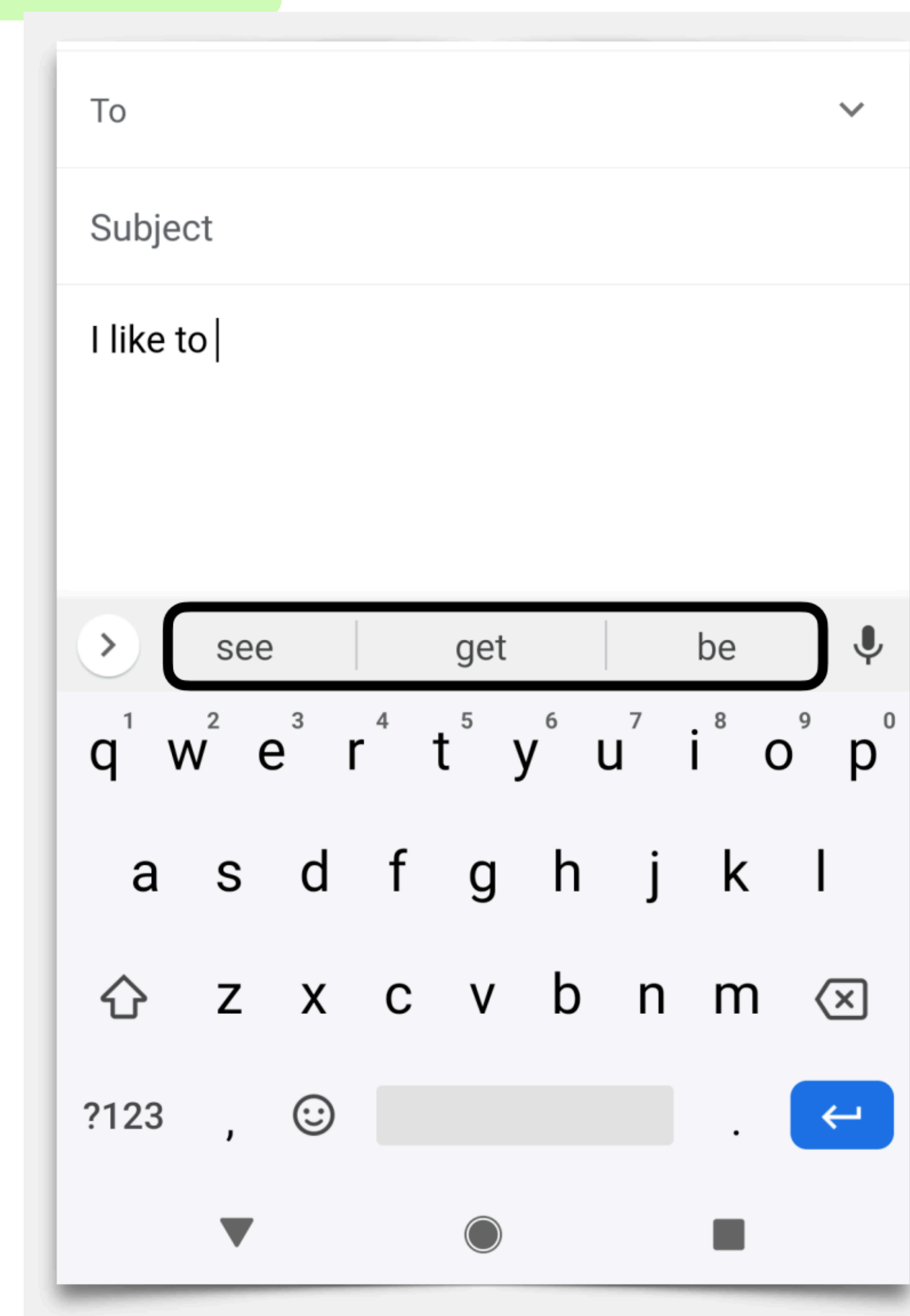
# Language Models

- Task: Given a sequence of words so far (**the context**), predict what comes next
  - Like (a supercharged) autocomplete!
- We never know for sure what comes next, but we can still make good guesses!



# Language Models

- Task: Given a sequence of words so far (**the context**), predict what comes next
  - Like (a supercharged) autocomplete!
- We never know for sure what comes next, but we can still make good guesses!
- To understand how these work we need to familiarize ourselves with the basics of machine learning



# Concrete Outcomes



# Concrete Outcomes

- Learn some basic ideas of machine learning (a statistical model)









# Concrete Outcomes

- Learn some basic ideas of machine learning (a statistical model)
- Learn how to build a simple language model (statistical model for predicting what word should come next in this sentence?)

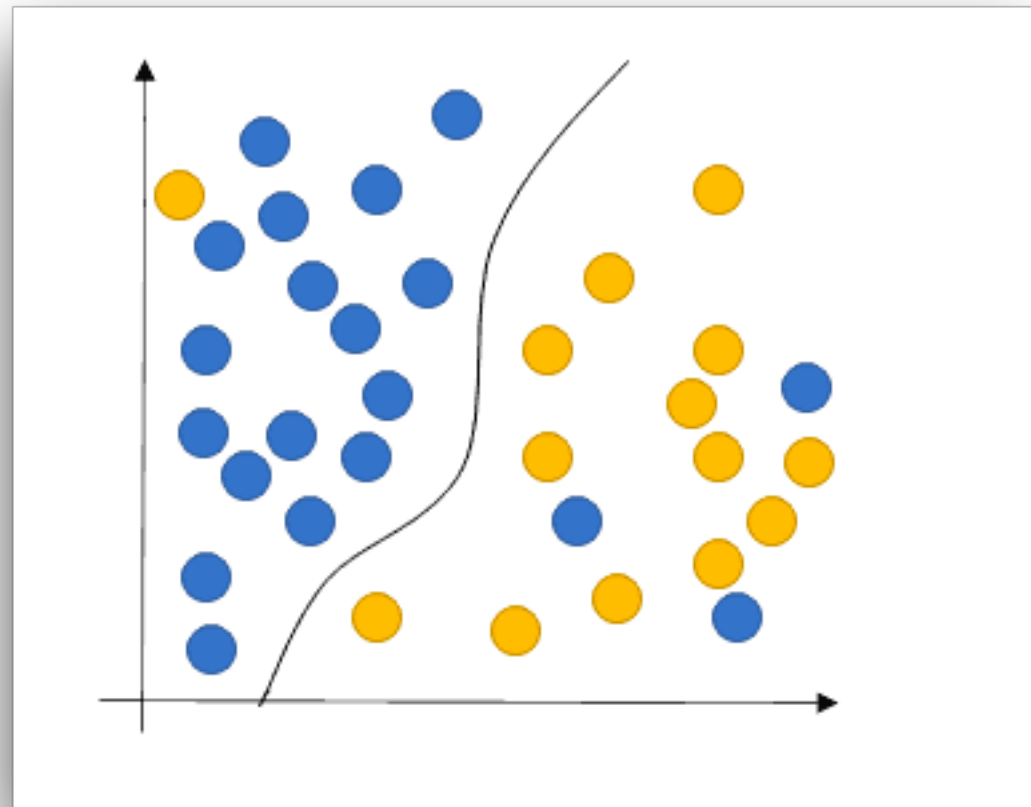


# Concrete Outcomes

- Learn some basic ideas of machine learning (a statistical model) 
- Learn how to build a simple language model (statistical model for predicting what word should come next in this sentence?) 
- Learn the connections between this language model and models such as OpenAI's ChatGPT / GPT-4 models and how to prompt them 
- Learn some limitations of language models 

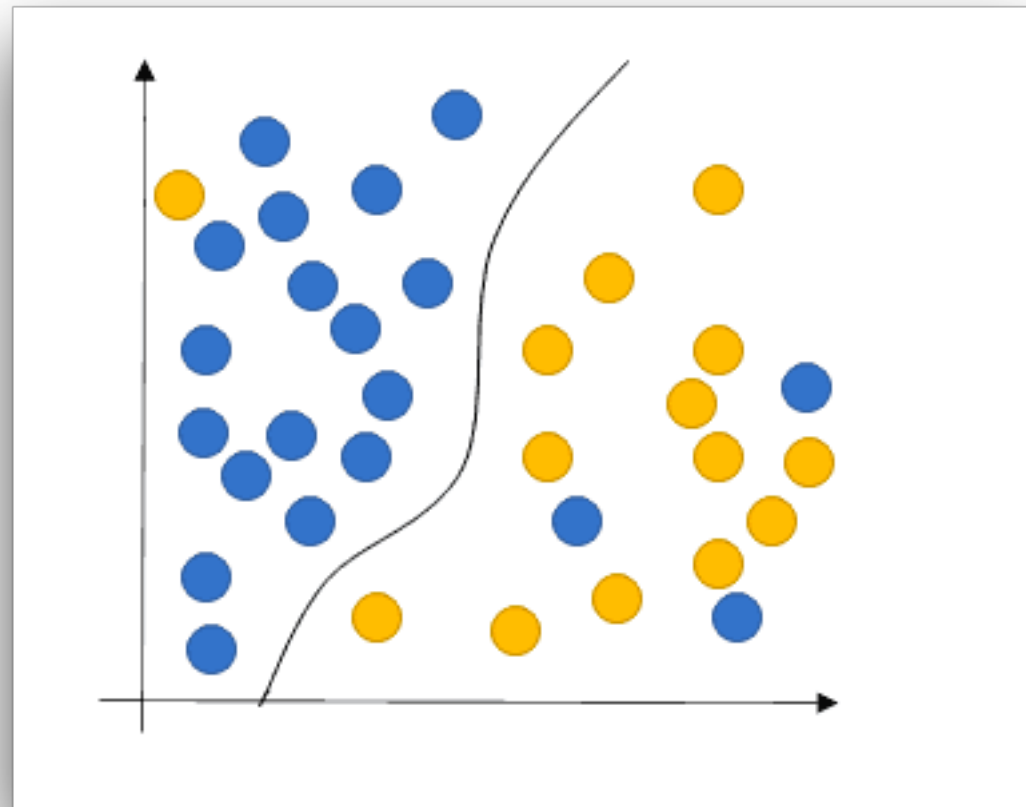
# Outline

# Outline



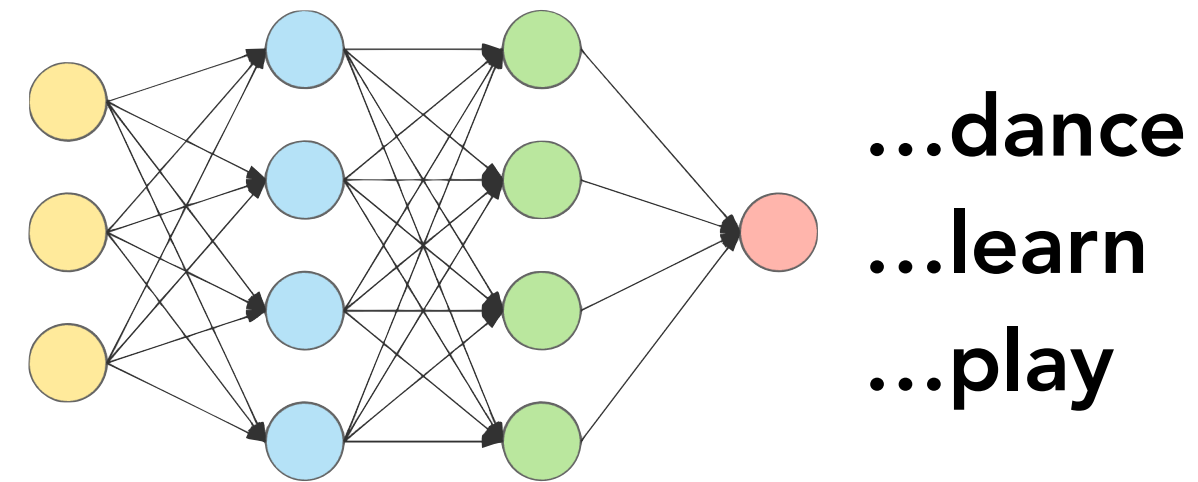
Machine Learning

# Outline



Machine Learning

So you want to ...



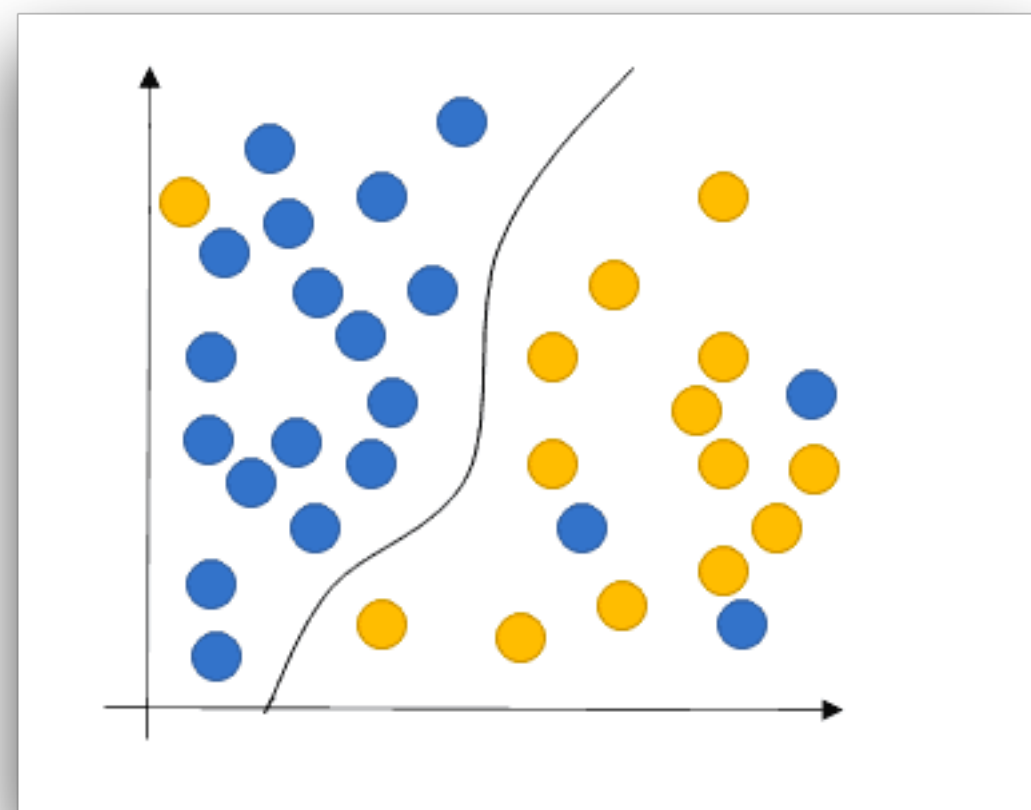
Language Modeling





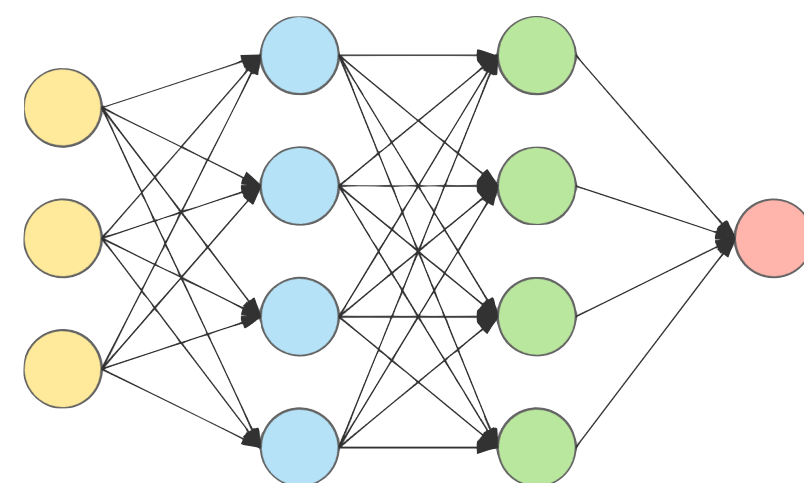


# Outline



Machine Learning

So you want to ...

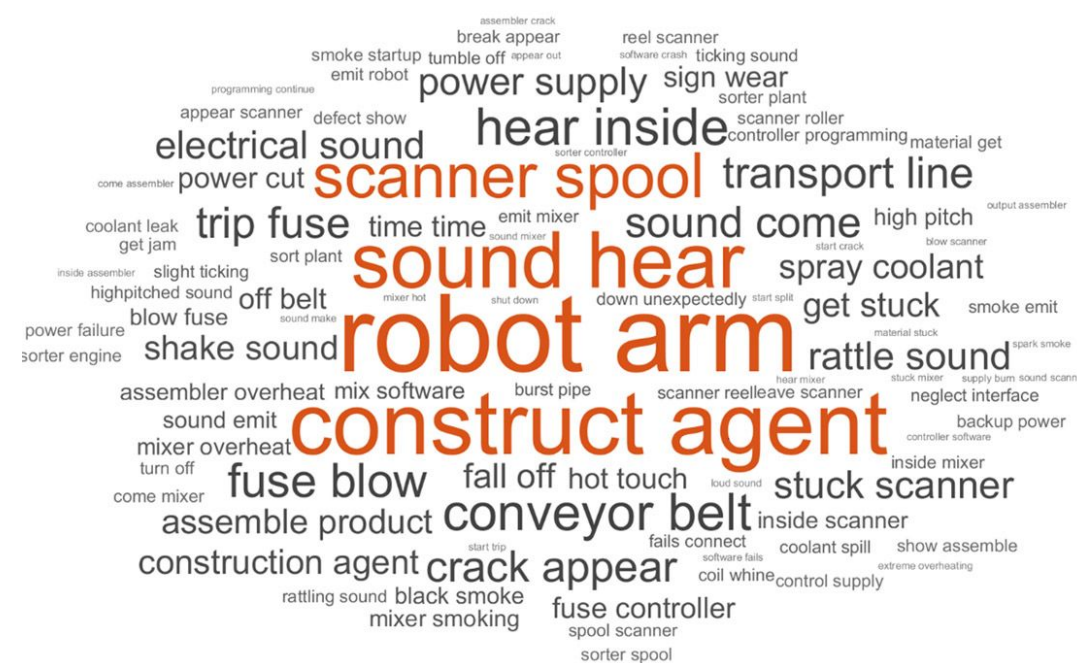


...dance  
...learn  
...play

Language Modeling



What's Next?



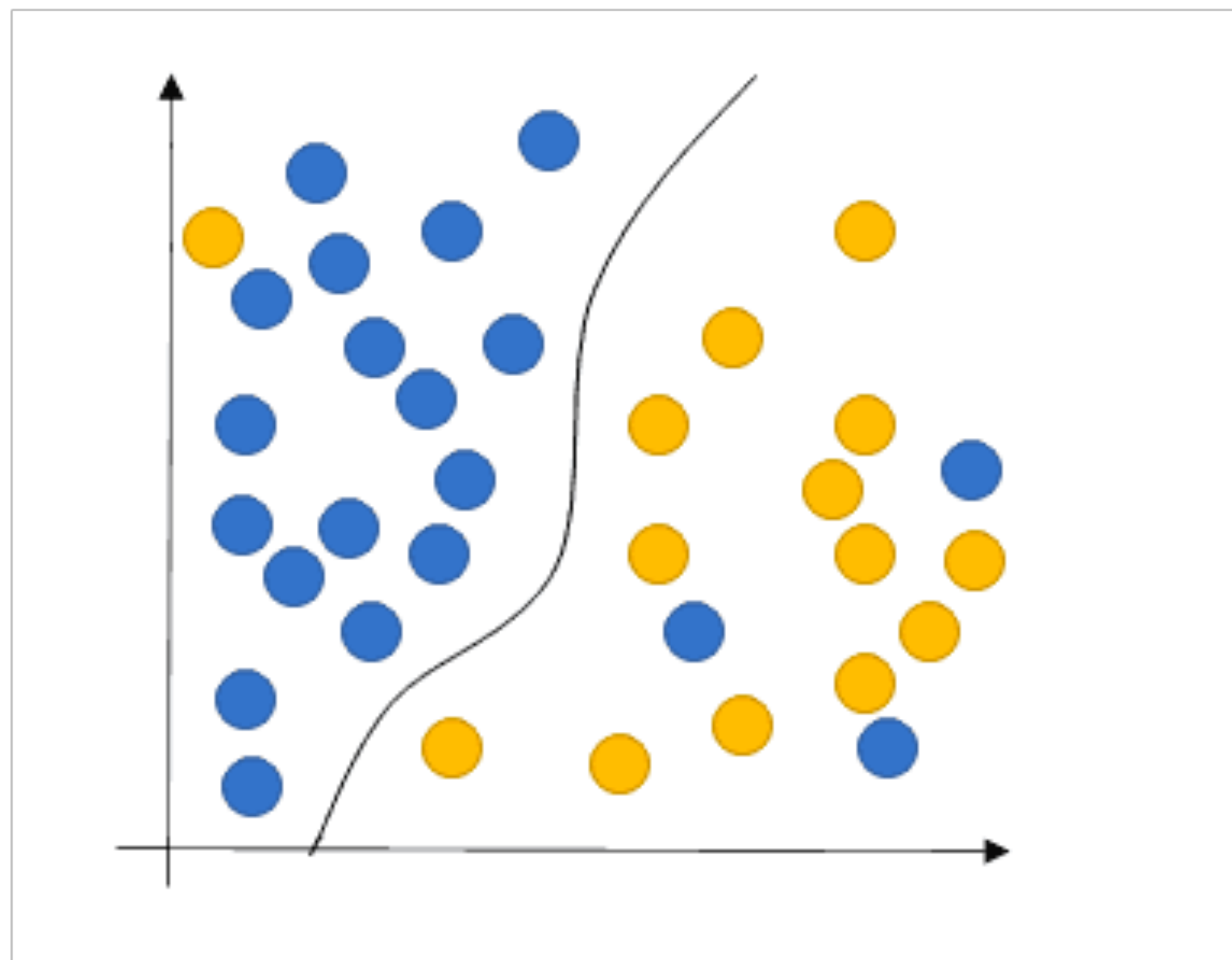
n-gram Language Models



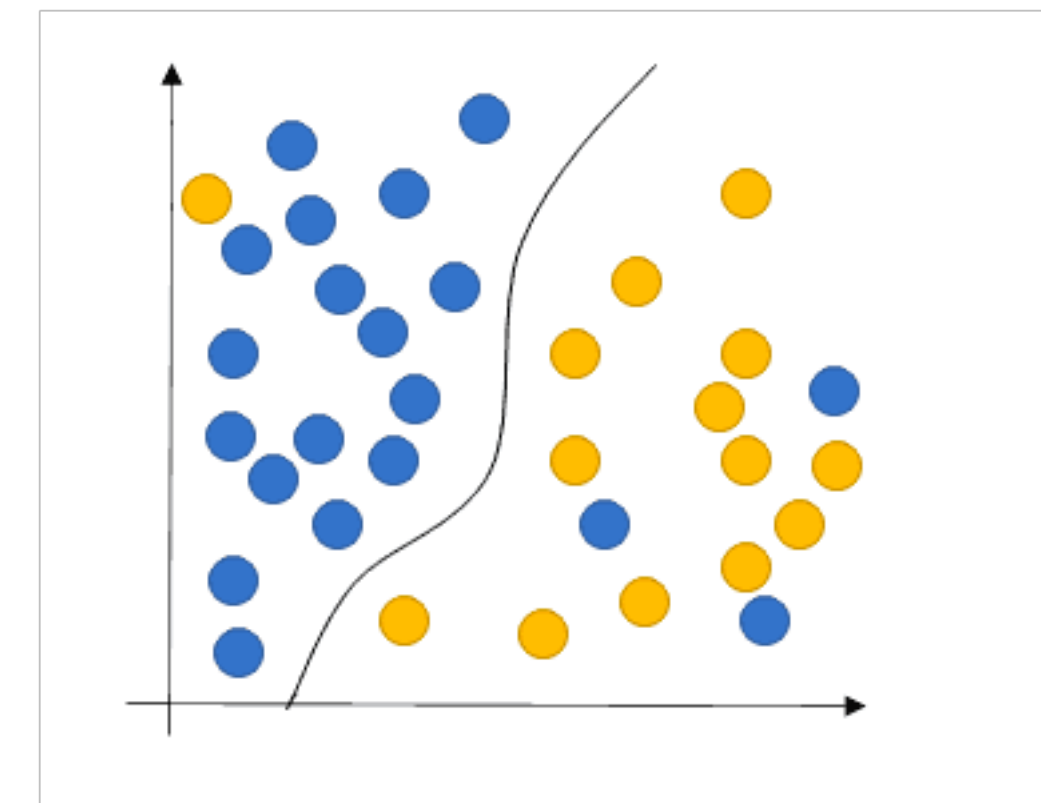
Large Language Models



# Machine Learning

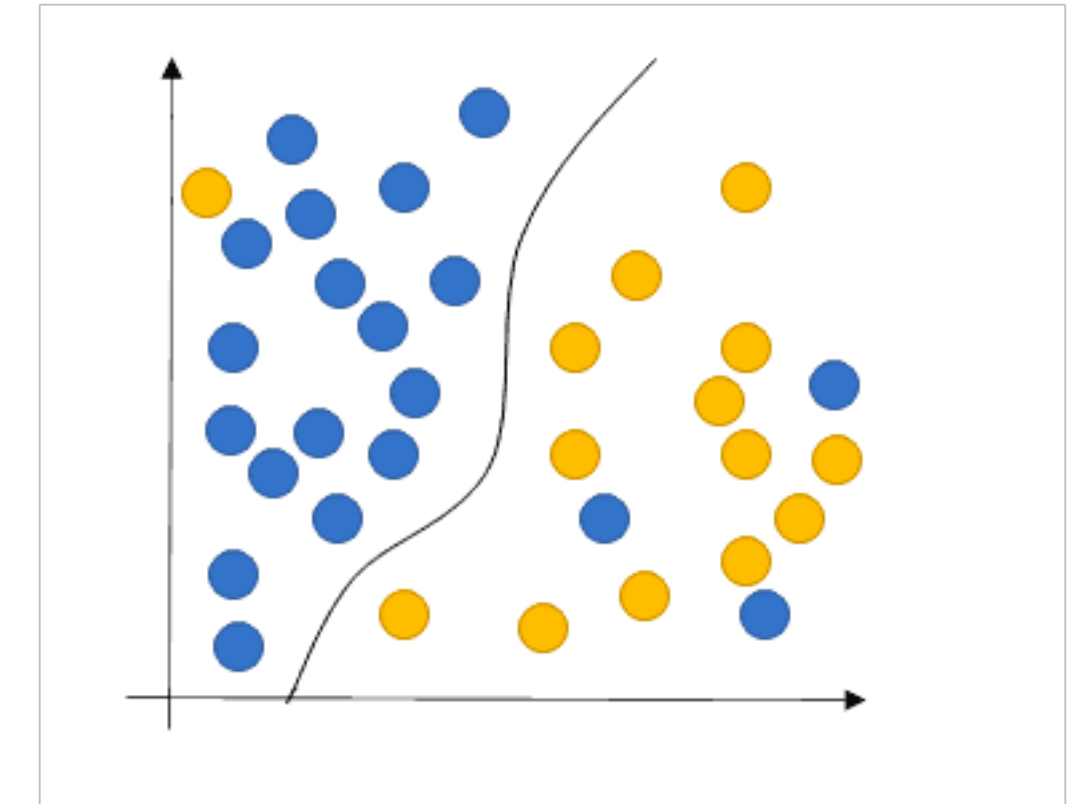


# Machine Learning



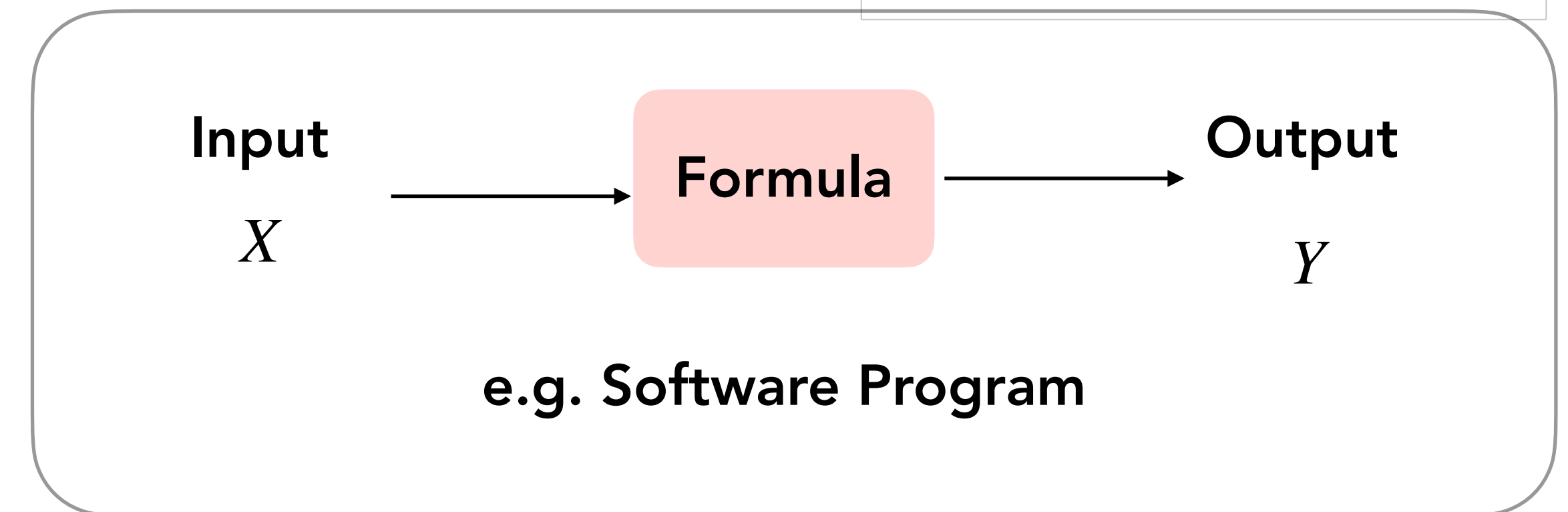
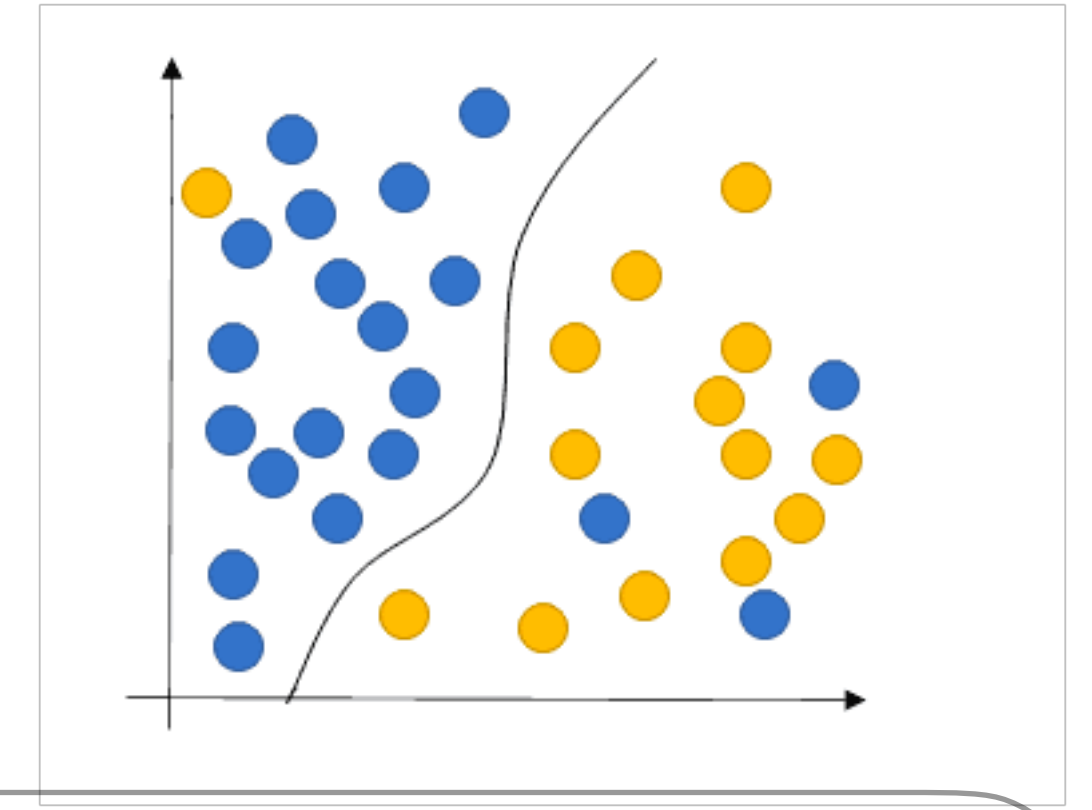
# Machine Learning

- All about predictions: Input  $X$  and Output  $Y$

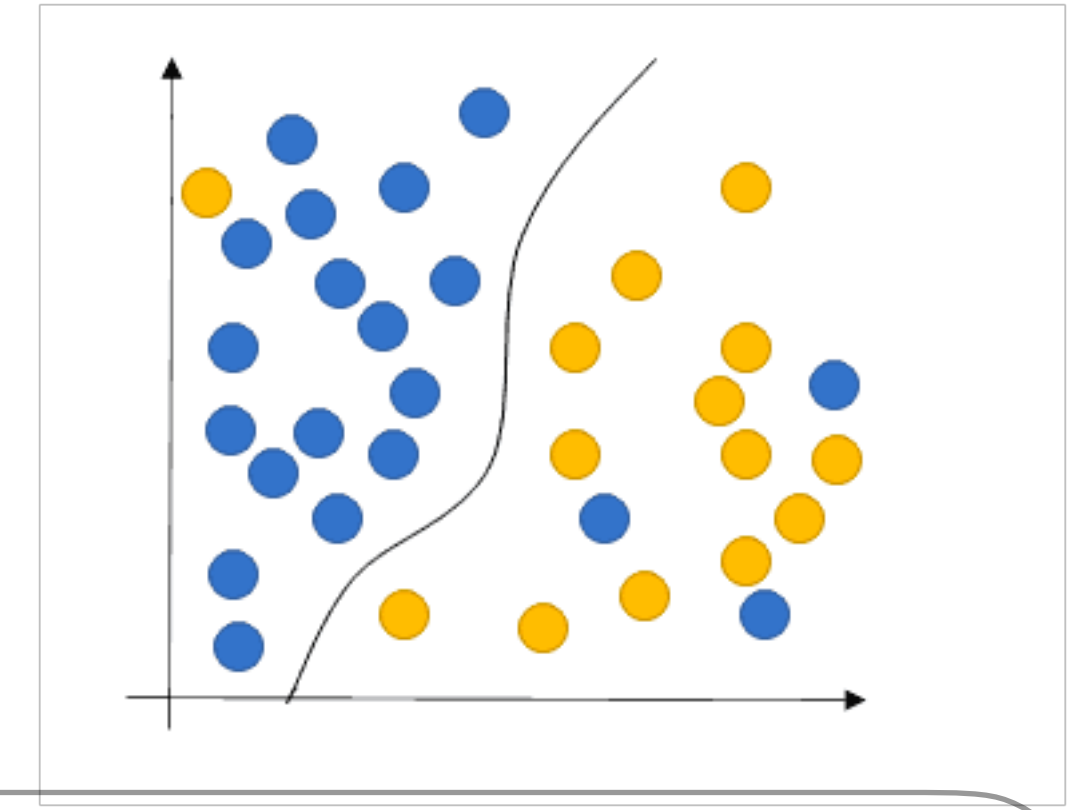


# Machine Learning

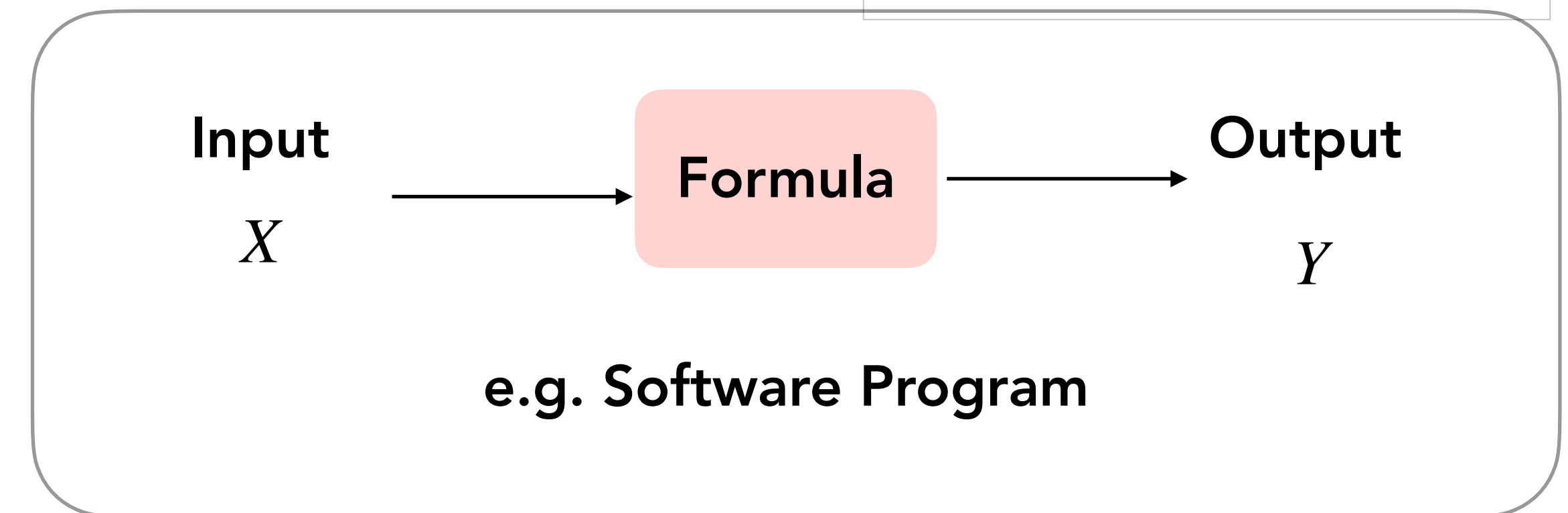
- All about predictions: Input  $X$  and Output  $Y$



# Machine Learning

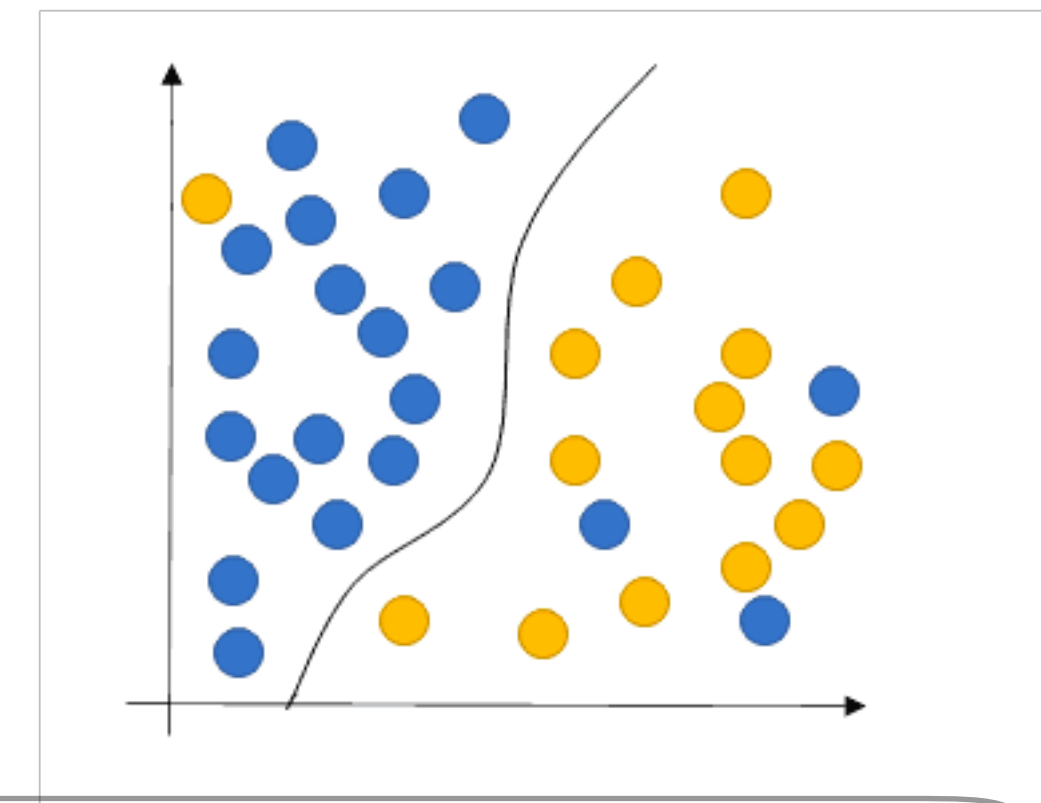


- All about predictions: Input  $X$  and Output  $Y$
- In most real life problems, there is no simple formula to obtain  $Y$  from  $X$

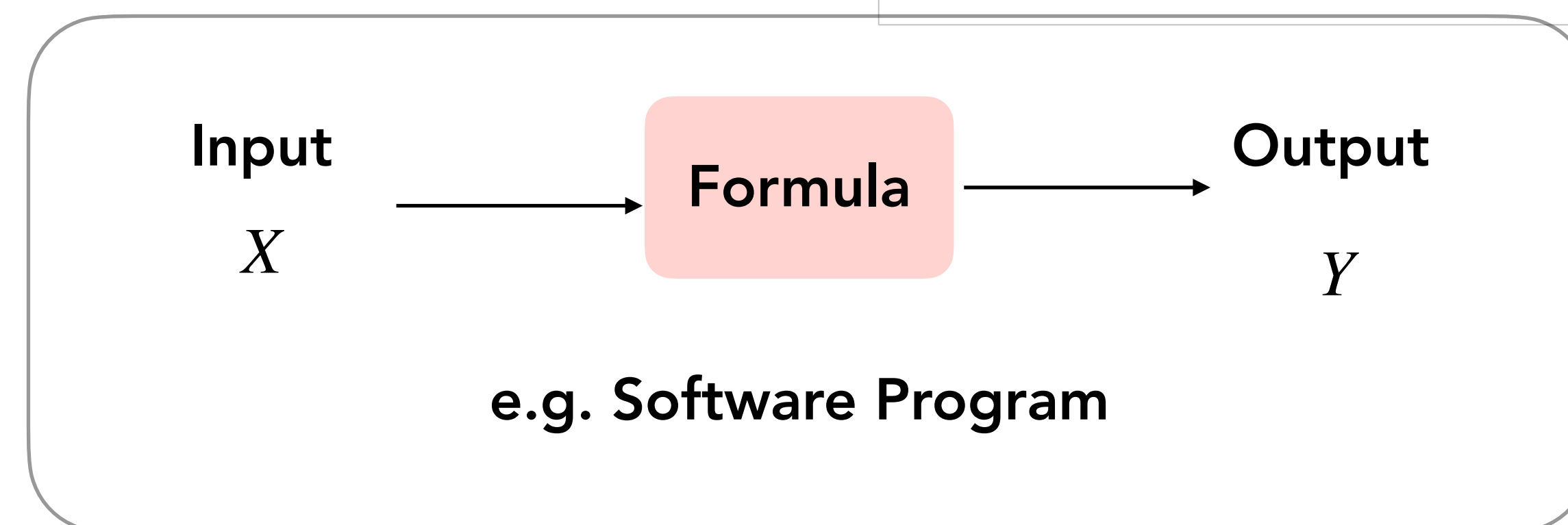




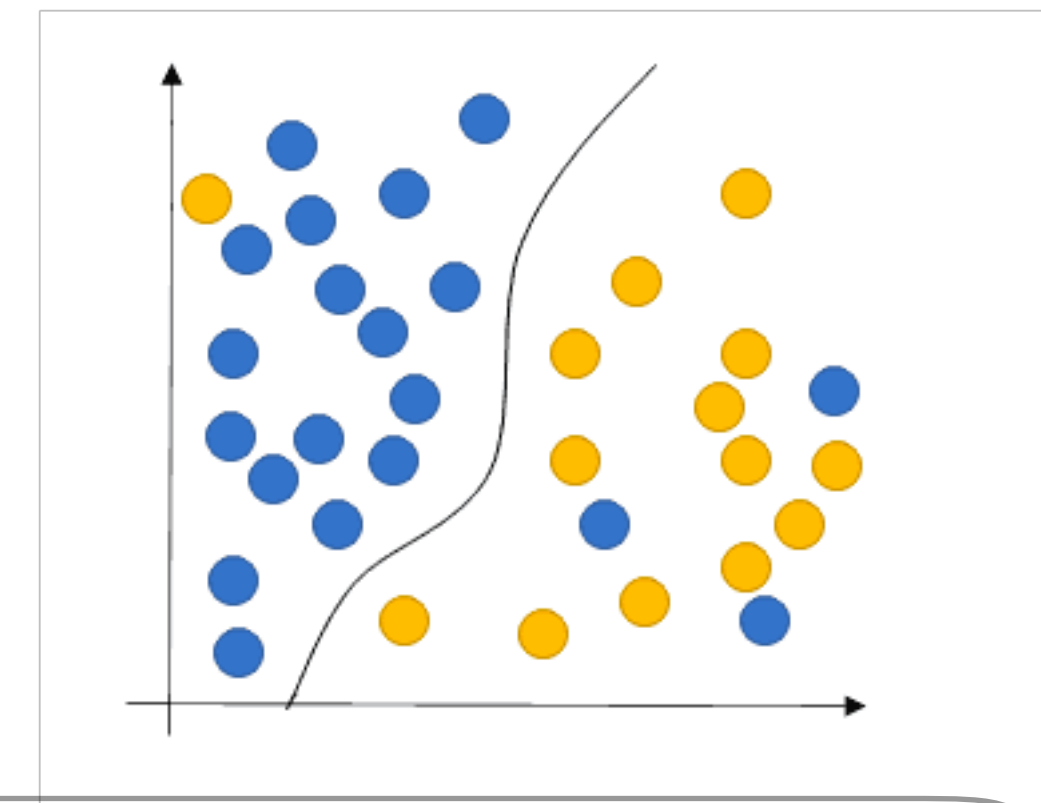
# Machine Learning



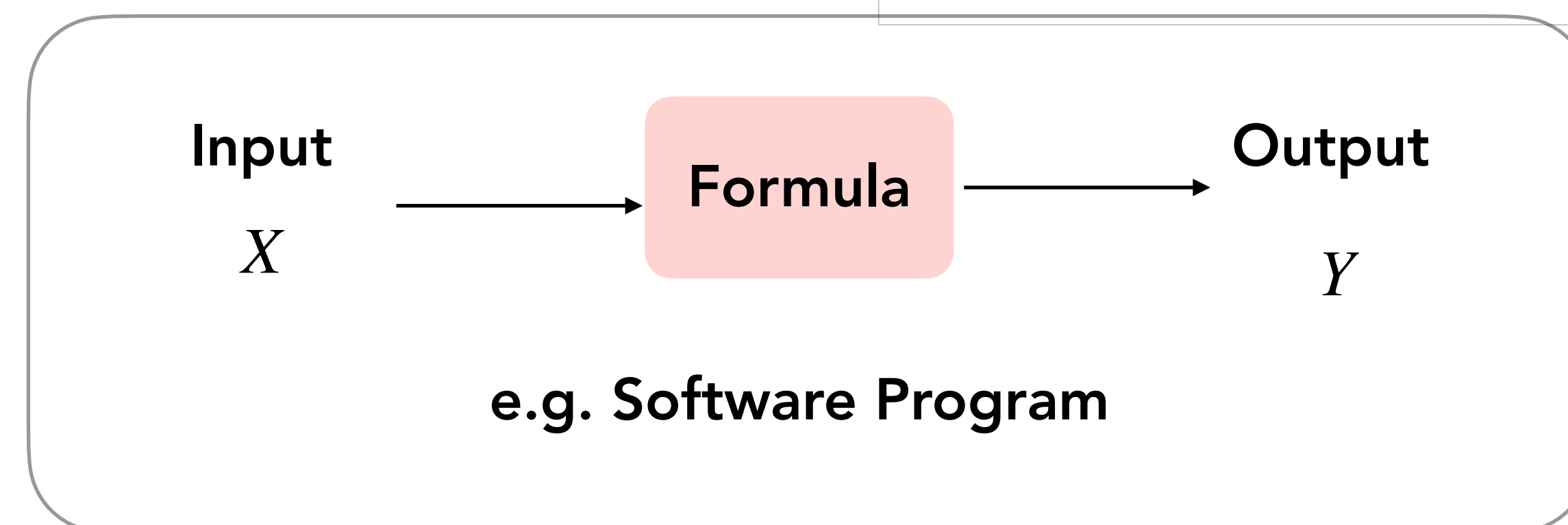
- All about predictions: Input  $X$  and Output  $Y$
- In most real life problems, there is no simple formula to obtain  $Y$  from  $X$
- Machine learning uses statistical analysis to figure out what would be the probability of the output  $Y$ , written as  $p(Y)$ , given the input  $X$



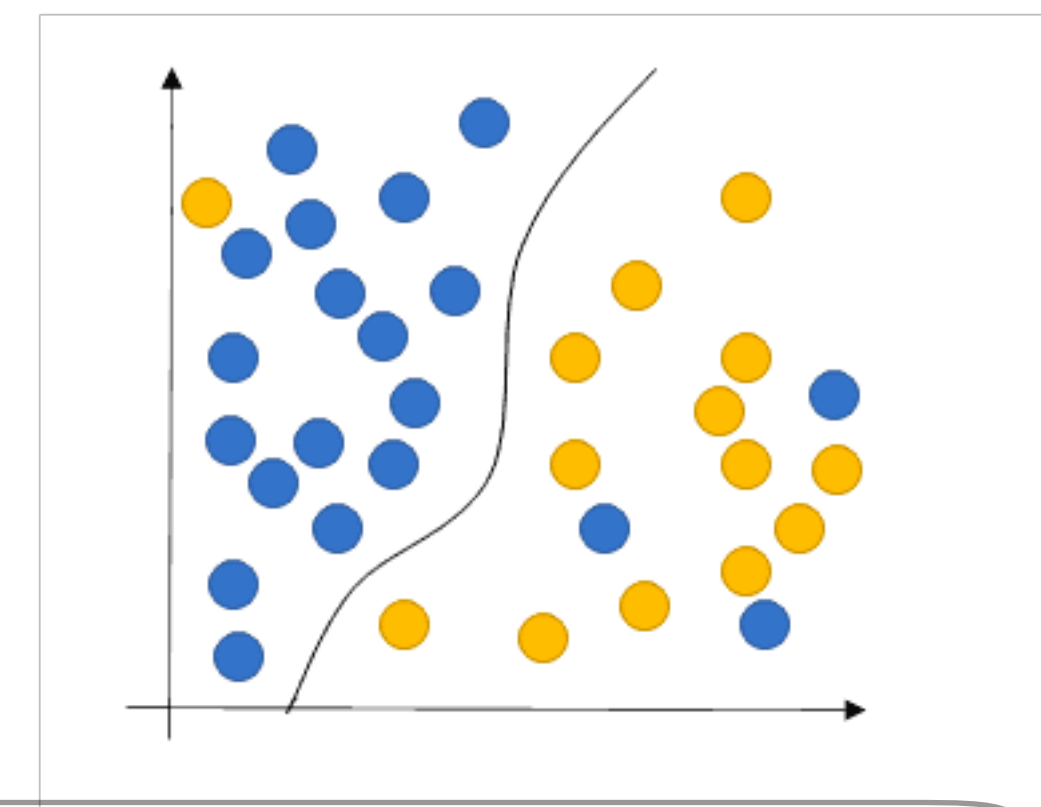
# Machine Learning



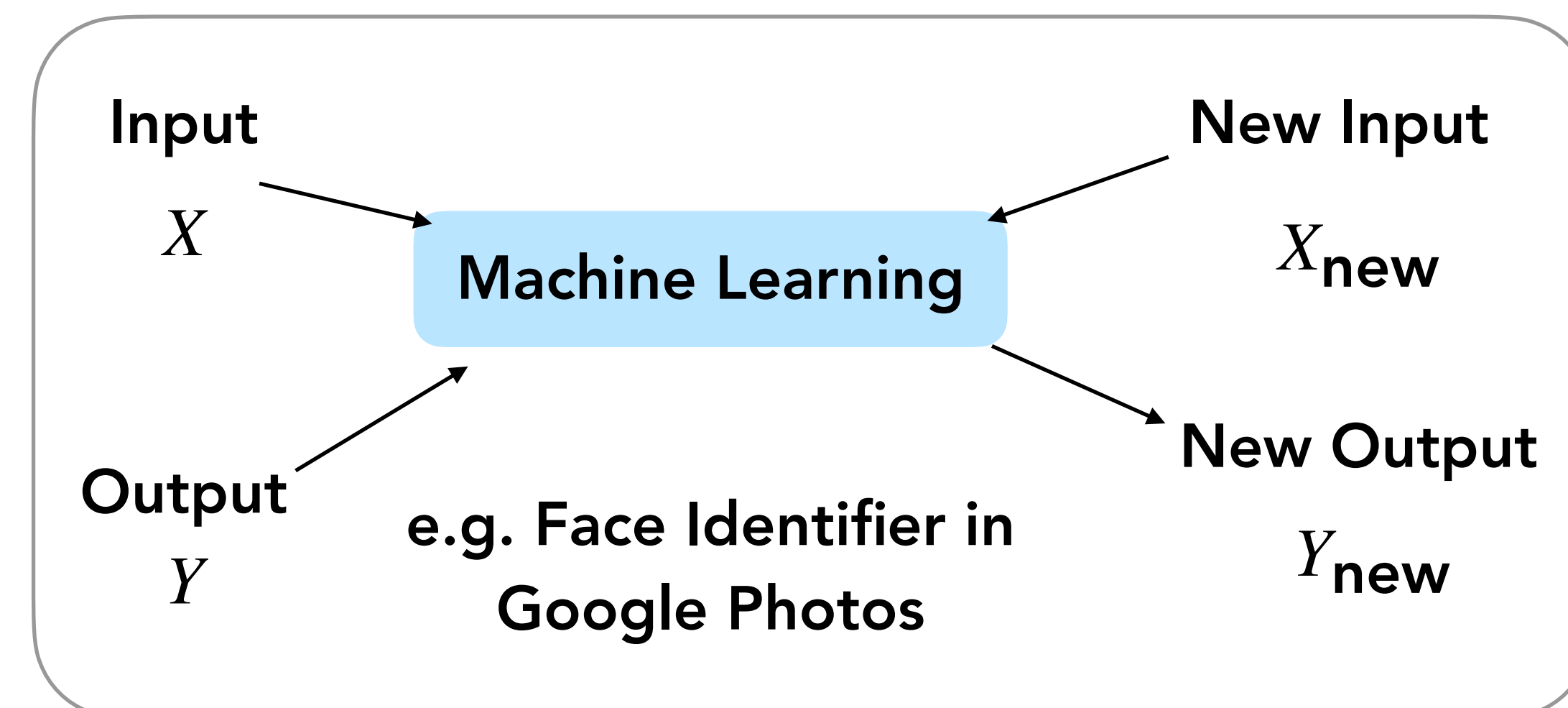
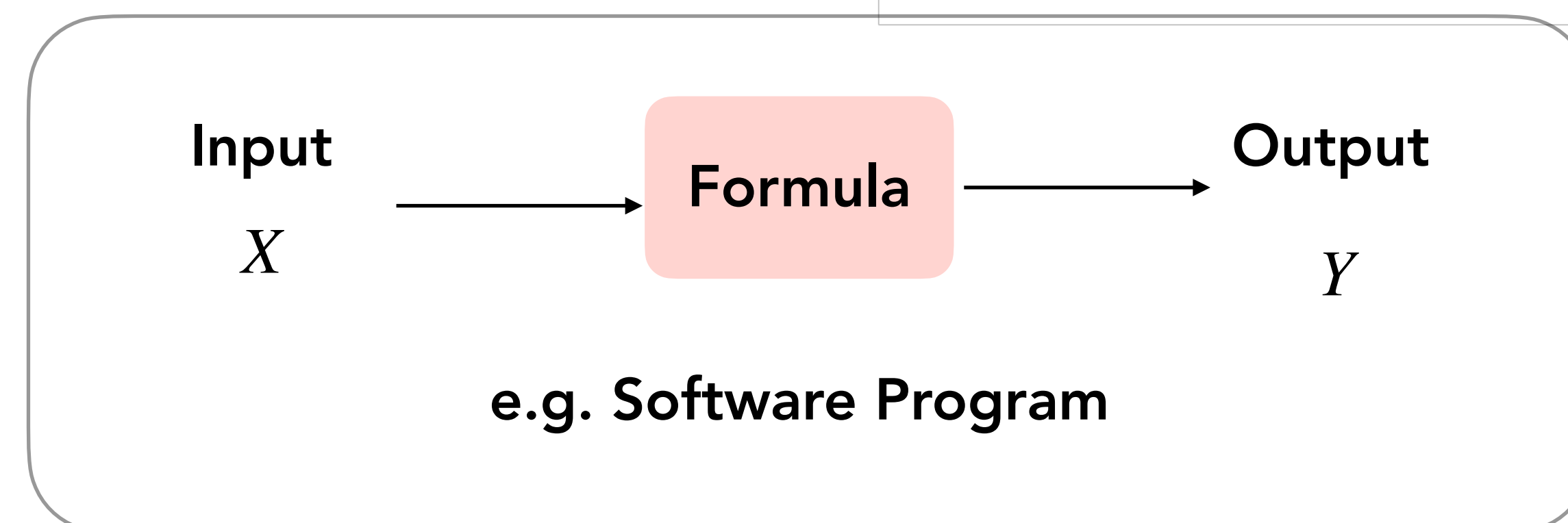
- All about predictions: Input  $X$  and Output  $Y$
- In most real life problems, there is no simple formula to obtain  $Y$  from  $X$
- Machine learning uses statistical analysis to figure out what would be the probability of the output  $Y$ , written as  $p(Y)$ , given the input  $X$
- Statistical Analysis: Lots of data as example pairs of input  $X$  and output  $Y$



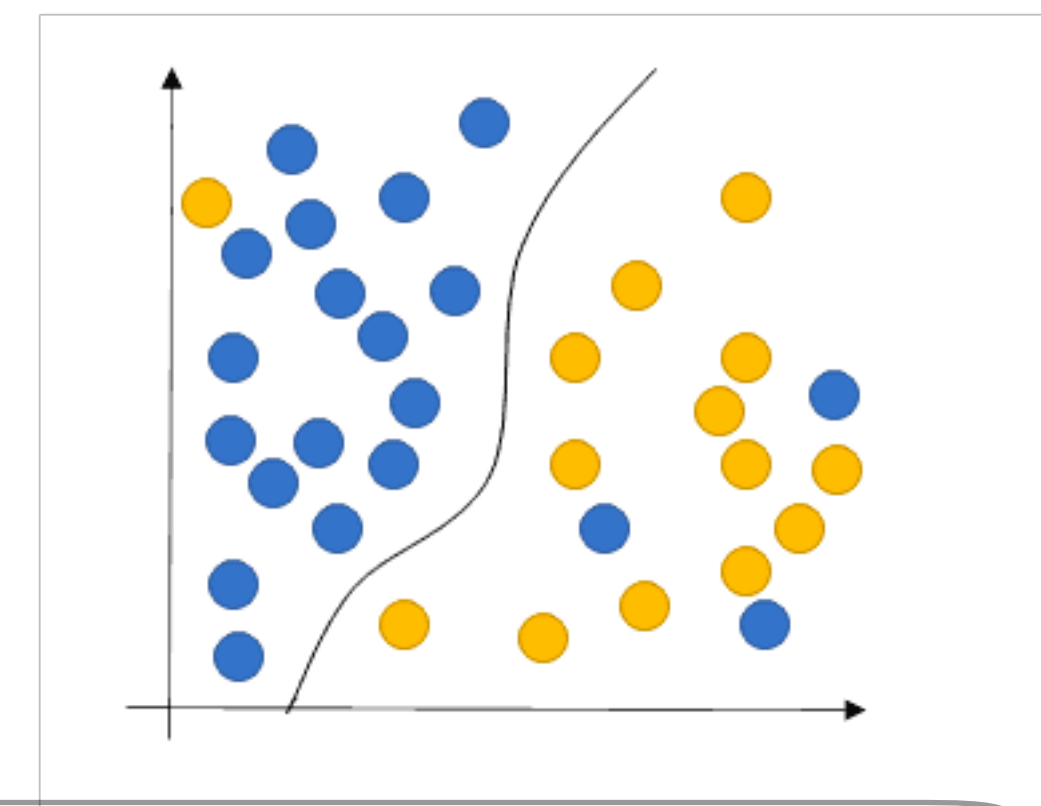
# Machine Learning



- All about predictions: Input  $X$  and Output  $Y$
- In most real life problems, there is no simple formula to obtain  $Y$  from  $X$
- Machine learning uses statistical analysis to figure out what would be the probability of the output  $Y$ , written as  $p(Y)$ , given the input  $X$
- Statistical Analysis: Lots of data as example pairs of input  $X$  and output  $Y$

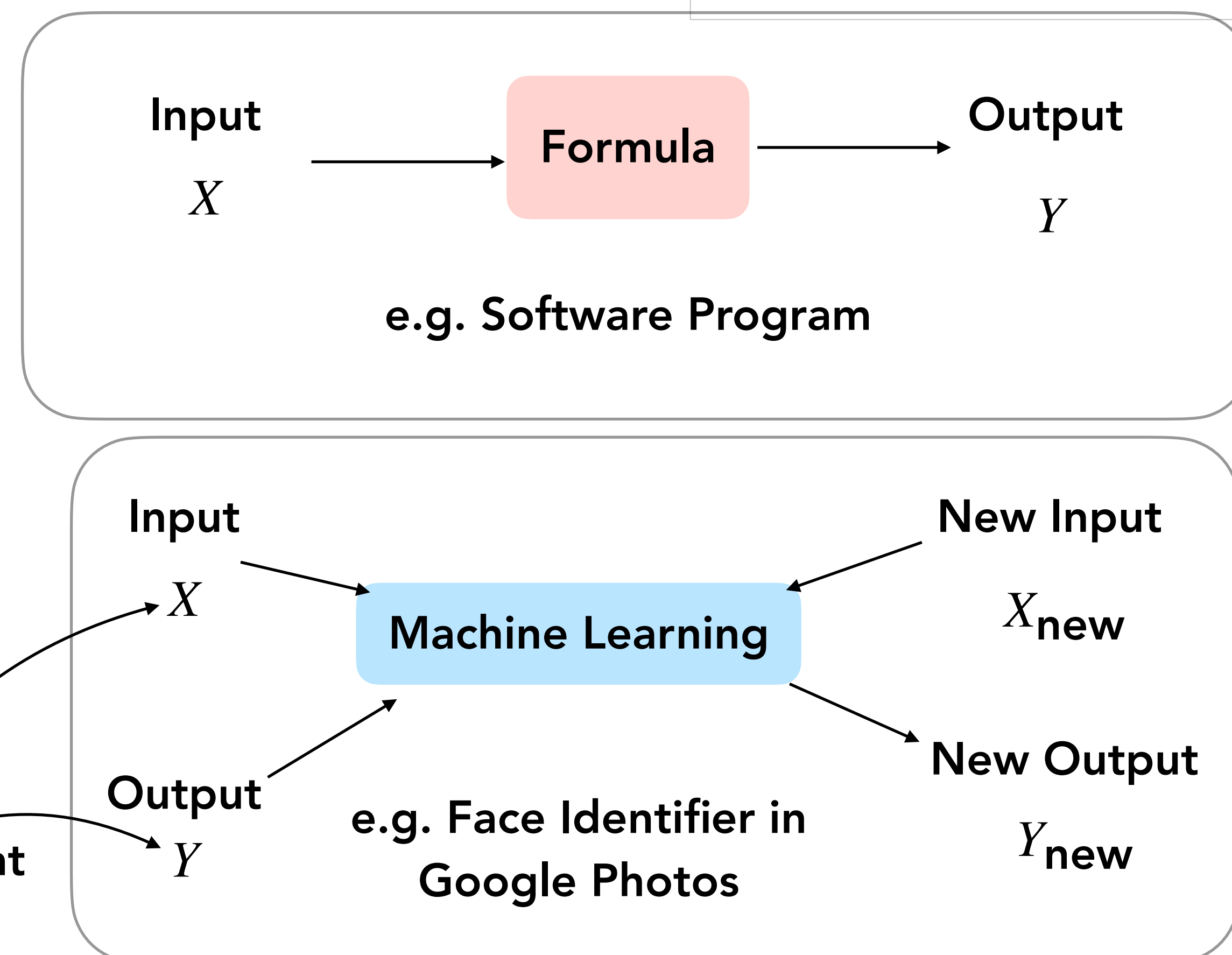


# Machine Learning



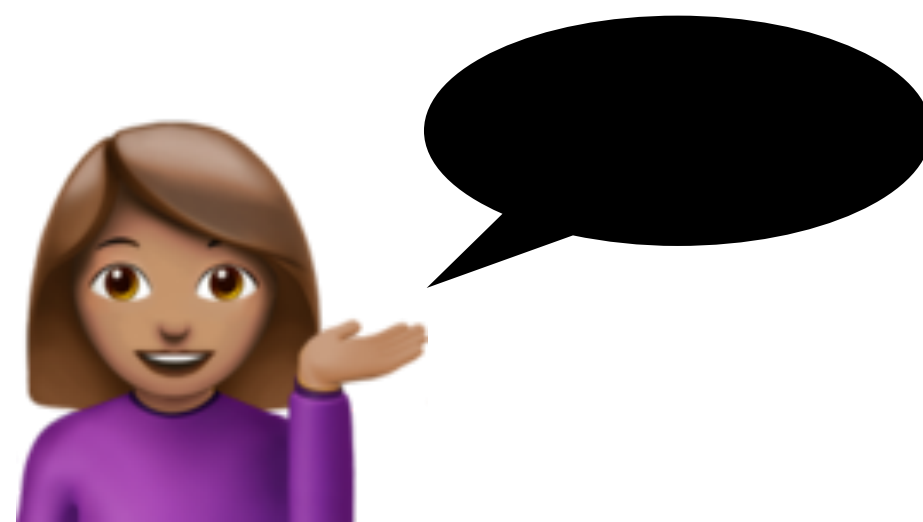
- All about predictions: Input  $X$  and Output  $Y$
- In most real life problems, there is no simple formula to obtain  $Y$  from  $X$
- Machine learning uses statistical analysis to figure out what would be the probability of the output  $Y$ , written as  $p(Y)$ , given the input  $X$
- Statistical Analysis: Lots of data as example pairs of input  $X$  and output  $Y$

Examples of what  
we want to do

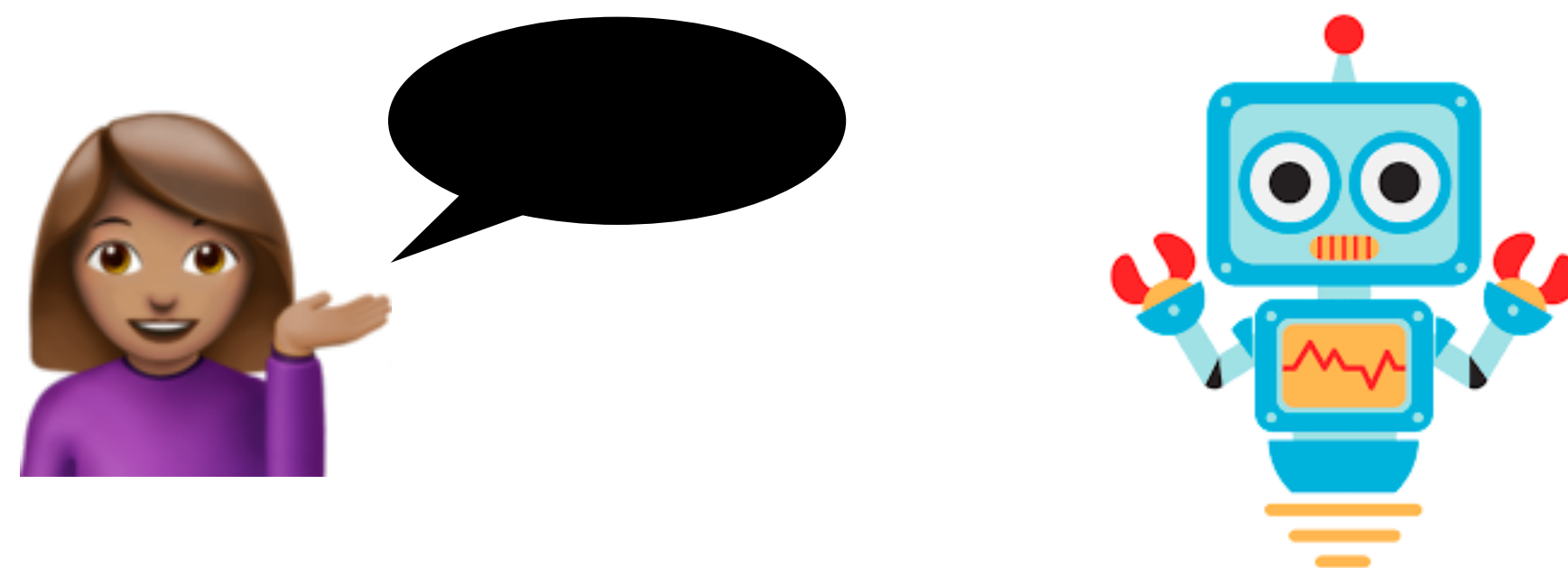




# Natural Language Processing and Machine Learning

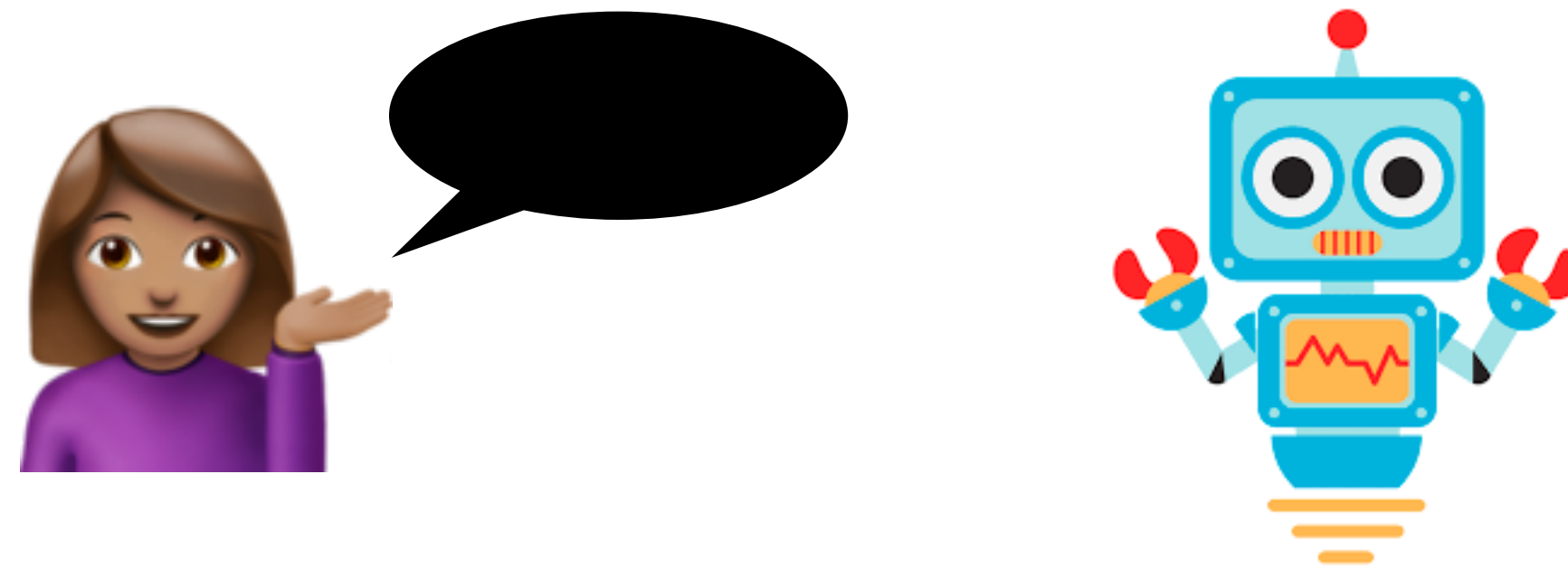


# Natural Language Processing and Machine Learning



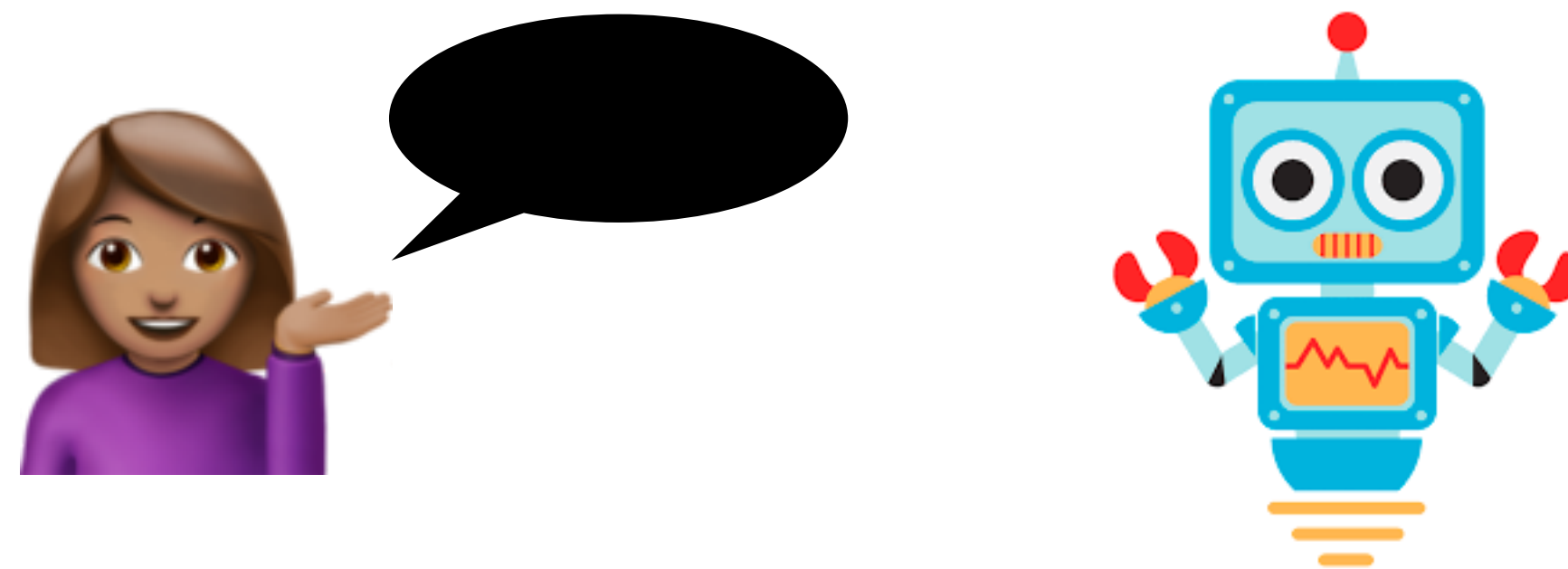
- Natural language processing uses a lot of ideas from machine learning

# Natural Language Processing and Machine Learning



- Natural language processing uses a lot of ideas from machine learning
- Humans are good at understanding language. Computers are bad at it and it's hard to program them.

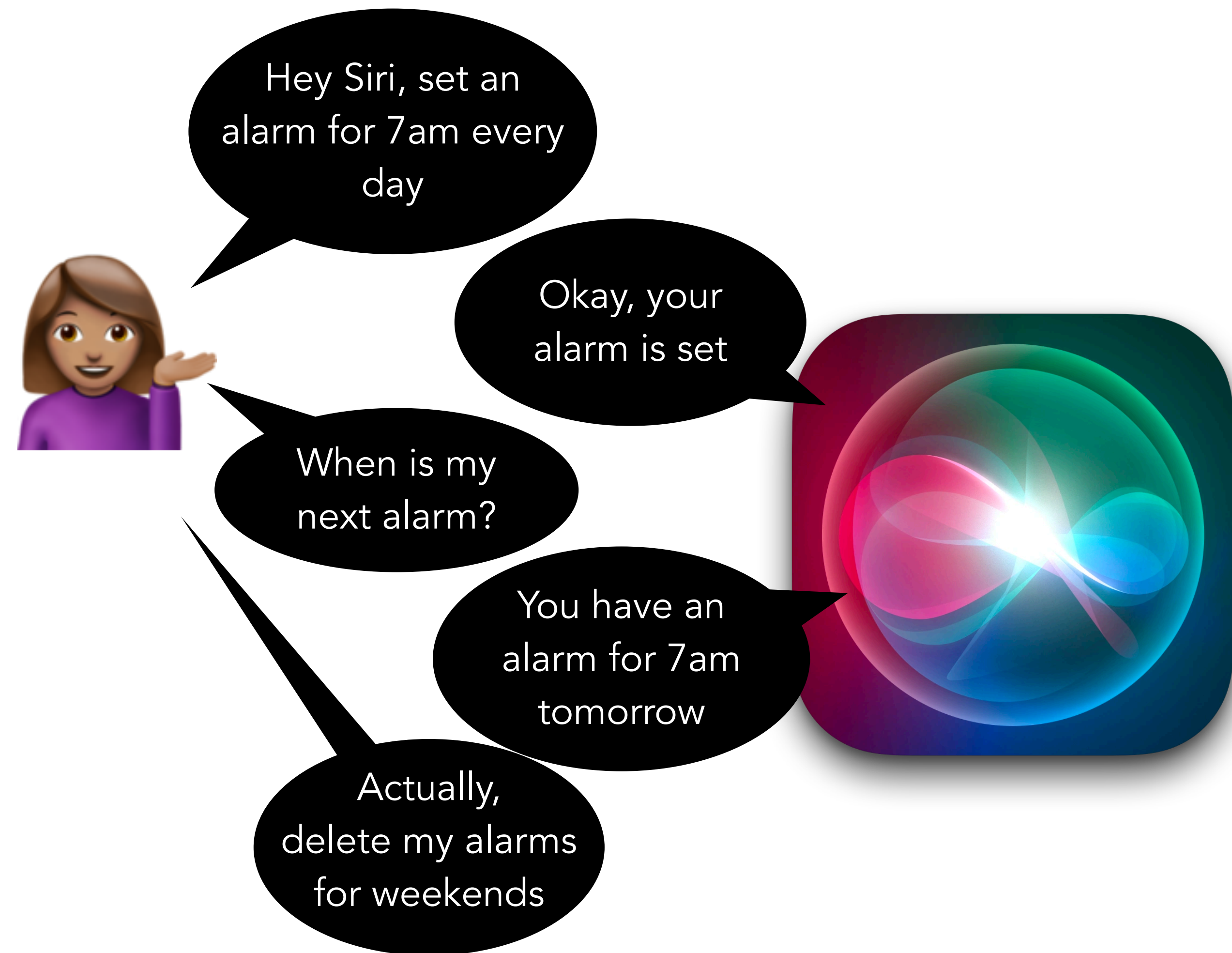
# Natural Language Processing and Machine Learning



- Natural language processing uses a lot of ideas from machine learning
- Humans are good at understanding language. Computers are bad at it and it's hard to program them.
- If we see lots of examples of how humans do a task, can we teach a computer how to do it?

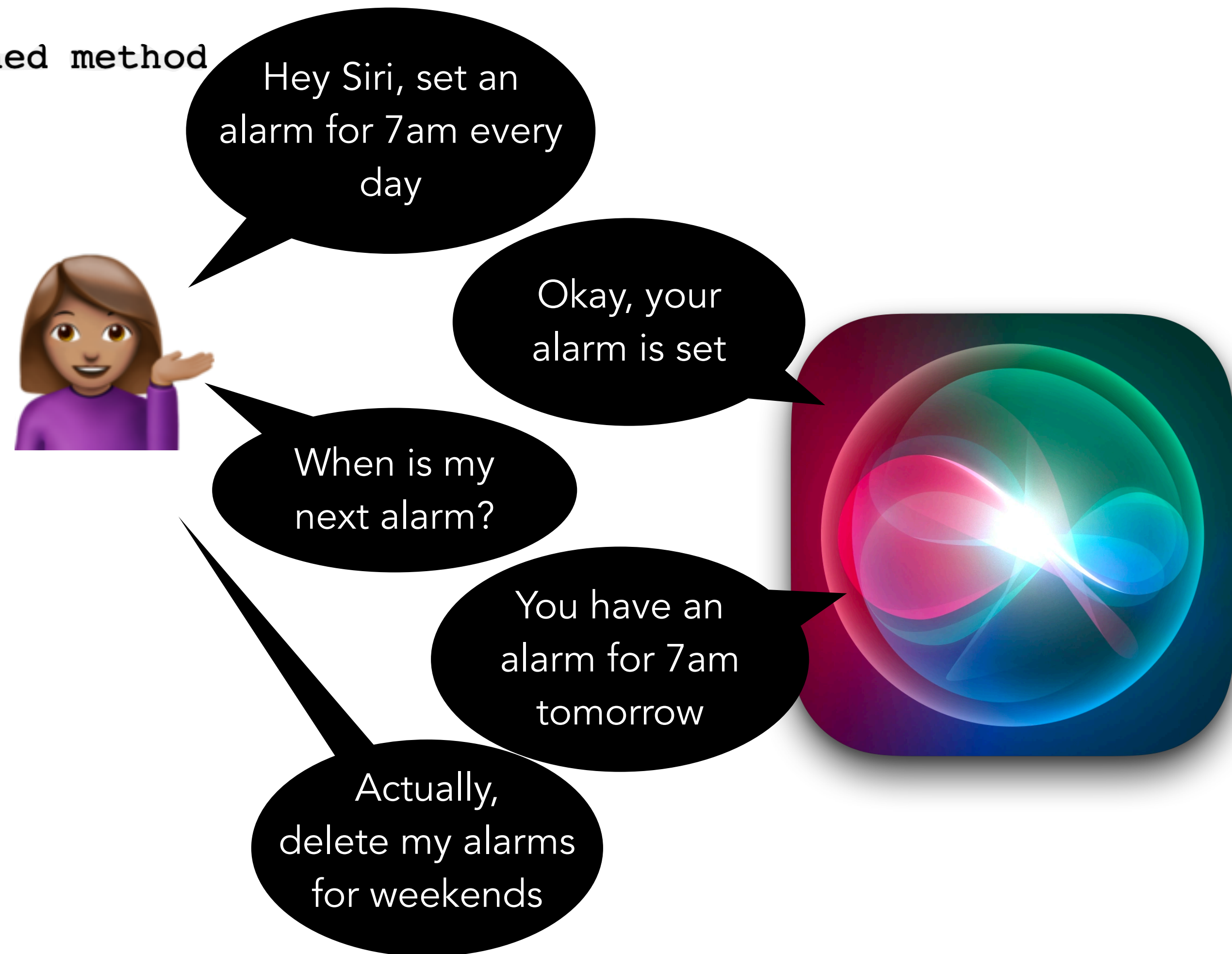


# Building Siri



# Building Siri

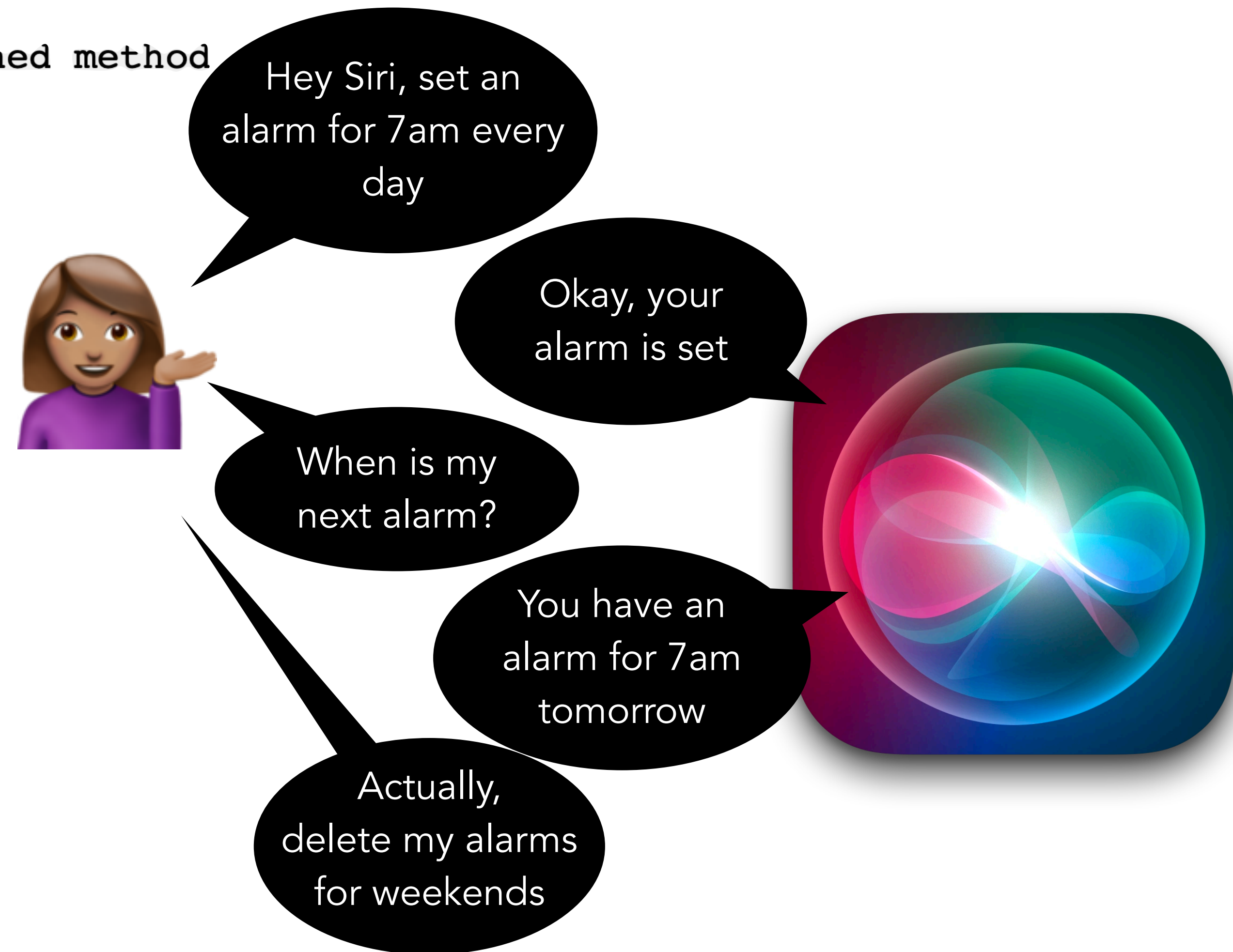
```
// Start by reading the user input with a predefined method
String userStr = readUserInput();
if (userStr.startsWith("set a timer"))
    startTimerDialogue();
else if (userStr.startsWith("set an alarm") ||
        userStr.startsWith("wake me up at"))
    startAlarmDialogue();
else [...]
```



# Building Siri

```
// Start by reading the user input with a predefined method
String userStr = readUserInput();
if (userStr.startsWith("set a timer"))
    startTimerDialogue();
else if (userStr.startsWith("set an alarm") ||
        userStr.startsWith("wake me up at"))
    startAlarmDialogue();
else [...]
```

- Too hard to list every case here!

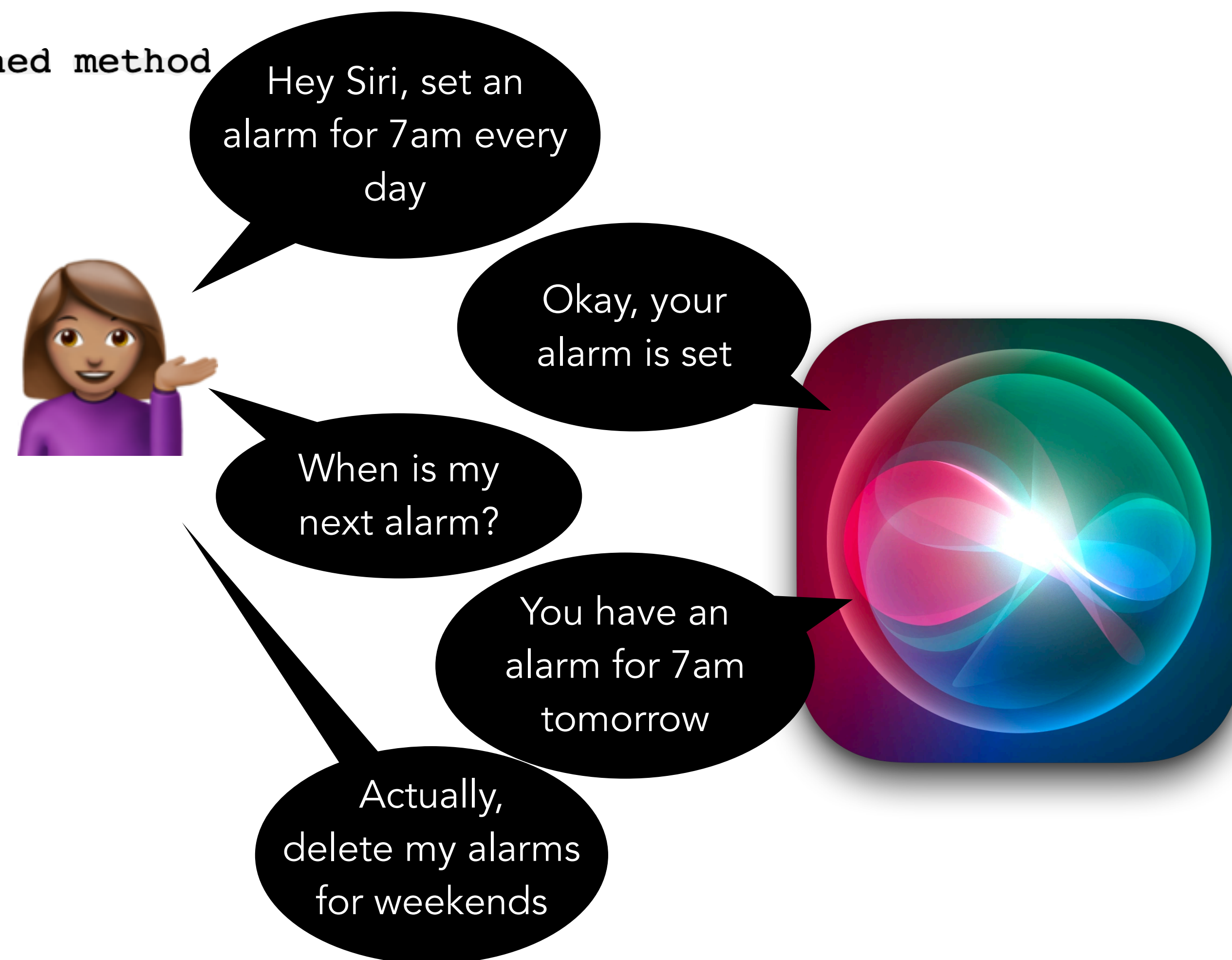




# Building Siri

```
// Start by reading the user input with a predefined method
String userStr = readUserInput();
if (userStr.startsWith("set a timer"))
    startTimerDialogue();
else if (userStr.startsWith("set an alarm") ||
        userStr.startsWith("wake me up at"))
    startAlarmDialogue();
else [...]
```

- Too hard to list every case here!
- This is where machine learning comes in!





# Analyze Movie Review Sentiment

**Spider-Man: Across the Spider-Verse** is an absolute triumph that takes everything we loved about the original film and cranks it up to a whole new level. This stunning sequel is a true testament to the power of animation, storytelling, and the enduring legacy of everyone's favorite web-slinger.



**The Little Mermaid:** To anyone who is planning on seeing this movie, I'd highly recommend to just wait until it comes out on Disney+ or something so you don't waste your money. I only went to see the movie because of my daughter and we can both say that this movie did not live up to our expectations. Furthermore, we both did not enjoy the majority of the movie at all.



**Spider-Man: Across the Spider-Verse** is an absolute triumph that takes everything we loved about the original film and cranks it up to a whole new level. This stunning sequel is a true testament to the power of animation, storytelling, and the enduring legacy of everyone's favorite web-slinger.



**Spider-Man: Across the Spider-Verse** is an absolute triumph that takes everything we loved about the original film and cranks it up to a whole new level. This stunning sequel is a true testament to the power of animation, storytelling, and the enduring legacy of everyone's favorite web-slinger.



- Let's try something simple:
  - (numberOfGoodWords, numberOfBadWords)

**Spider-Man: Across the Spider-Verse** is an absolute triumph that takes everything we loved about the original film and cranks it up to a whole new level. This stunning sequel is a true testament to the power of animation, storytelling, and the enduring legacy of everyone's favorite web-slinger.



- Let's try something simple:
  - (numberOfGoodWords, numberOfBadWords)

```
int numberOfGoodWords = computeNumGoodWords(review);
int numberOfBadWords = computeNumBadWords(review);
if (numberOfGoodWords > 3 && numberOfBadWords < 2)
    return "4 stars";
else if (numberOfGoodWords > 2 && numberOfBadWords < 3)
    return "3 stars";
else [...]
```



**Spider-Man: Across the Spider-Verse** is an absolute triumph that takes everything we loved about the original film and cranks it up to a whole new level. This stunning sequel is a true testament to the power of animation, storytelling, and the enduring legacy of everyone's favorite web-slinger.



- Let's try something simple:
  - (numberOfGoodWords, numberOfBadWords)

```
int numberOfGoodWords = computeNumGoodWords(review);  
int numberOfBadWords = computeNumBadWords(review);  
if (numberOfGoodWords > 3 && numberOfBadWords < 2)  
    return "4 stars";  
else if (numberOfGoodWords > 2 && numberOfBadWords < 3)  
    return "3 stars";  
else [...]
```

We can automatically generate this program!  
(It's called a decision tree)

# Machine Learning Overview

# Machine Learning Overview

- Lots of different models: decision trees, neural networks, Bayes Networks, ...

# Machine Learning Overview

- Lots of different models: decision trees, neural networks, Bayes Networks, ...
- Machine Learning starts with a **feature representation** of this data: how do we represent it to a system?



# Machine Learning Overview

- Lots of different models: decision trees, neural networks, Bayes Networks, ...
- Machine Learning starts with a **feature representation** of this data: how do we represent it to a system?
  - We did for sentiment analysis with our variables, (numberOfGoodWords, numberOfBadWords)

# Machine Learning Overview

- Lots of different models: decision trees, neural networks, Bayes Networks, ...
- Machine Learning starts with a **feature representation** of this data: how do we represent it to a system?
  - We did for sentiment analysis with our variables, (numberOfGoodWords, numberOfBadWords)
- **Neural networks** will view this as thousands of numbers (similar to how computers view programs as boolean codes) associated with each word.

# Machine Learning Overview

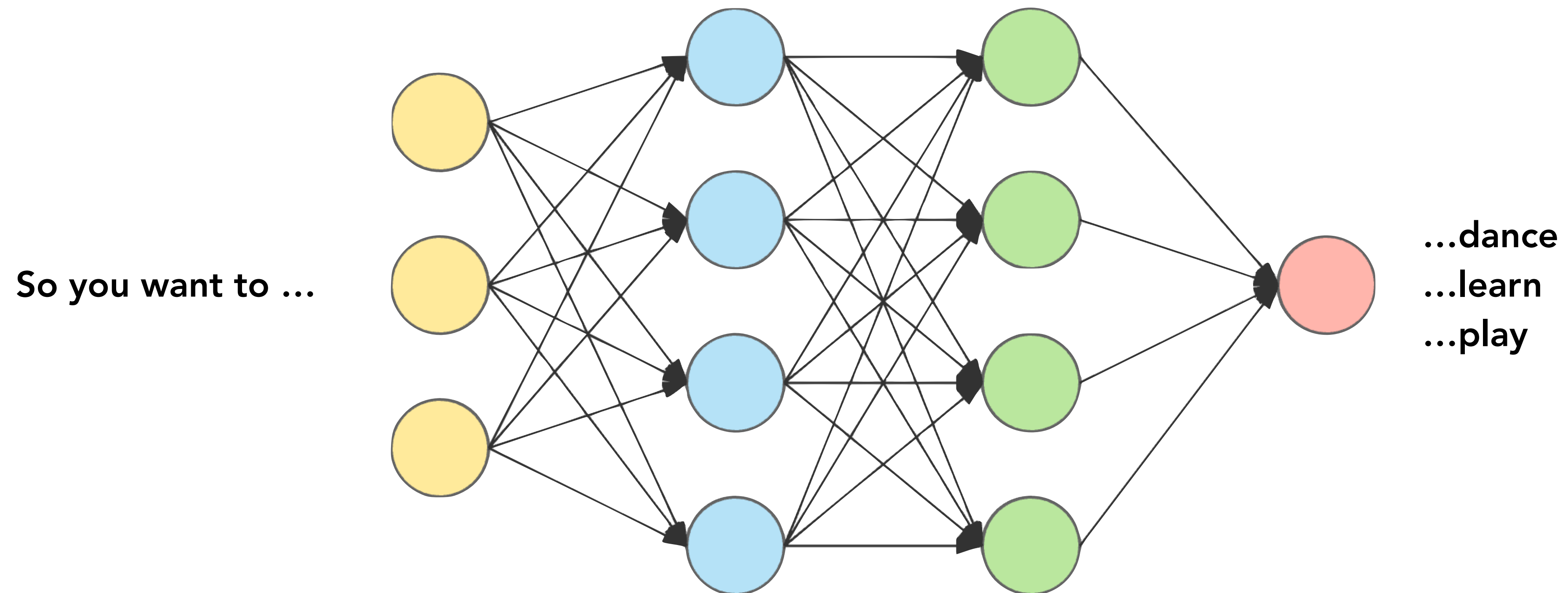
- Lots of different models: decision trees, neural networks, Bayes Networks, ...
- Machine Learning starts with a **feature representation** of this data: how do we represent it to a system?
  - We did for sentiment analysis with our variables, (numberOfGoodWords, numberOfBadWords)
- **Neural networks** will view this as thousands of numbers (similar to how computers view programs as boolean codes) associated with each word.
- Let's use a probabilistic model for language modeling...

# Machine Learning Overview

- Lots of different models: decision trees, neural networks, Bayes Networks, ...
- Machine Learning starts with a **feature representation** of this data: how do we represent it to a system?
  - We did for sentiment analysis with our variables, (numberOfGoodWords, numberOfBadWords)
- **Neural networks** will view this as thousands of numbers (similar to how computers view programs as boolean codes) associated with each word.
- Let's use a probabilistic model for language modeling...
  - Very little math to implement...



# Language Models



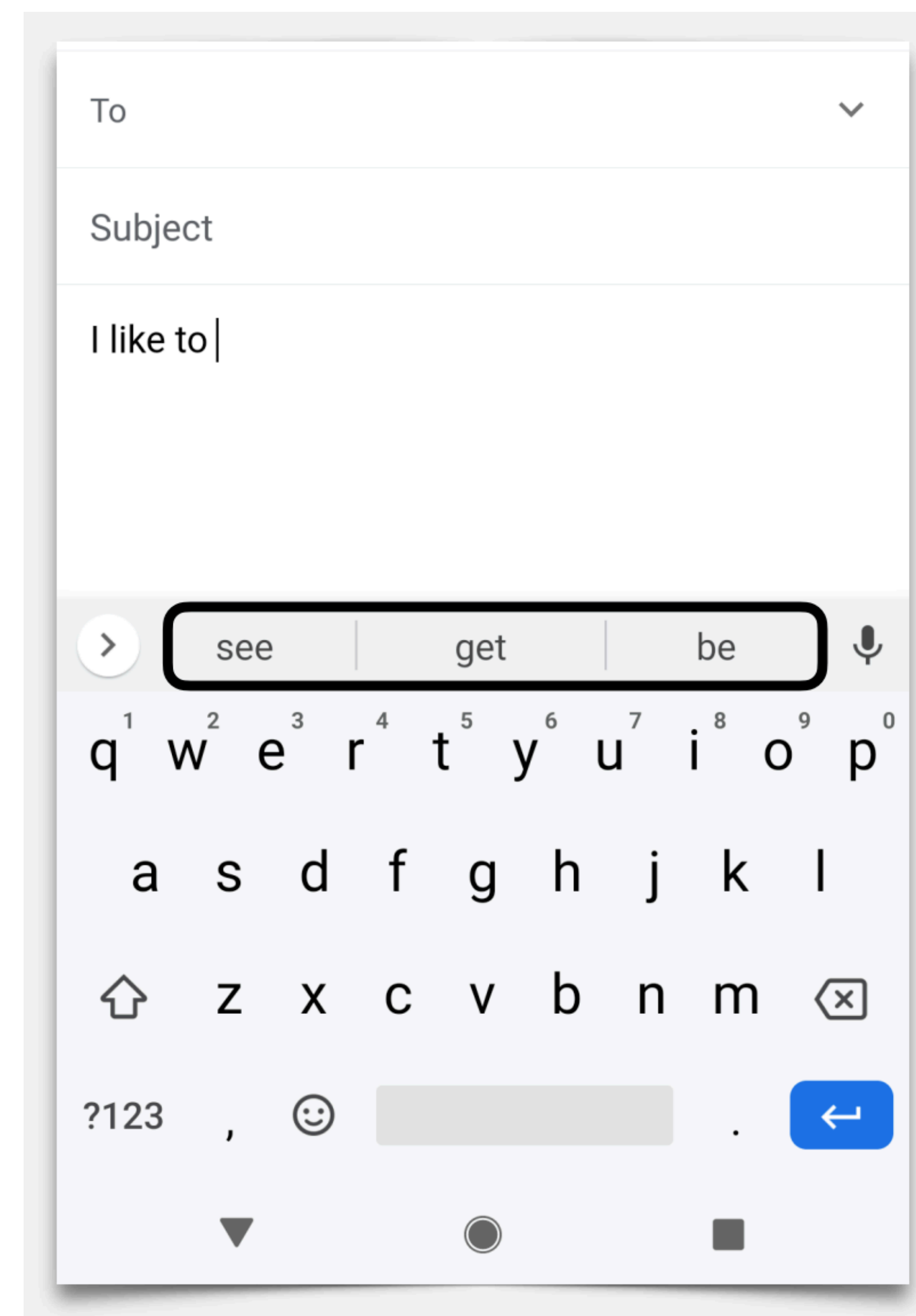
# Language Models

# Language Models


- Task: Given a sequence of words so far (**the context**), predict what comes next.

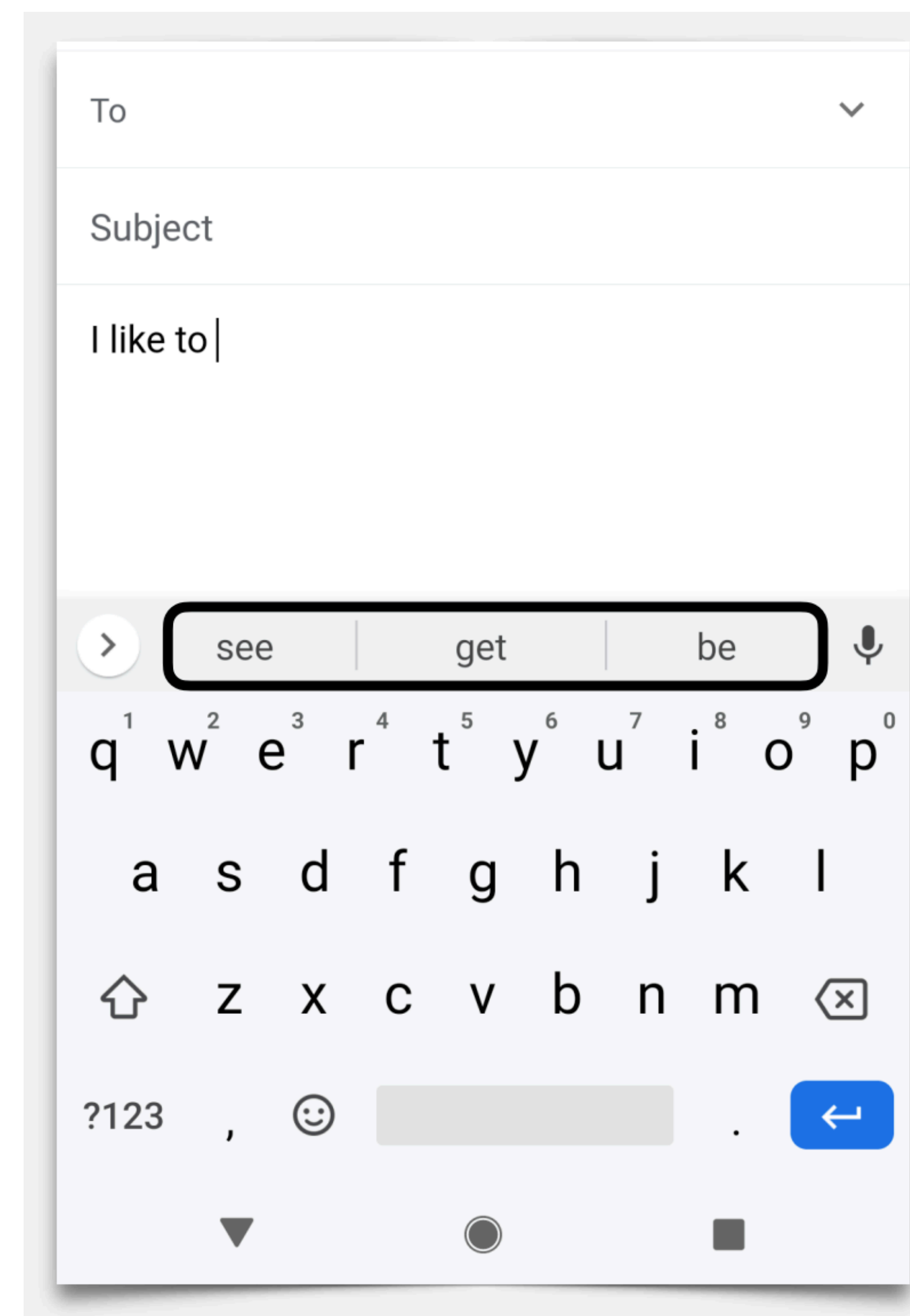
# Language Models

- Task: Given a sequence of words so far (**the context**), predict what comes next.
- Like (a supercharged) autocomplete!
- We never know for sure what comes next, but we can still make good guesses!



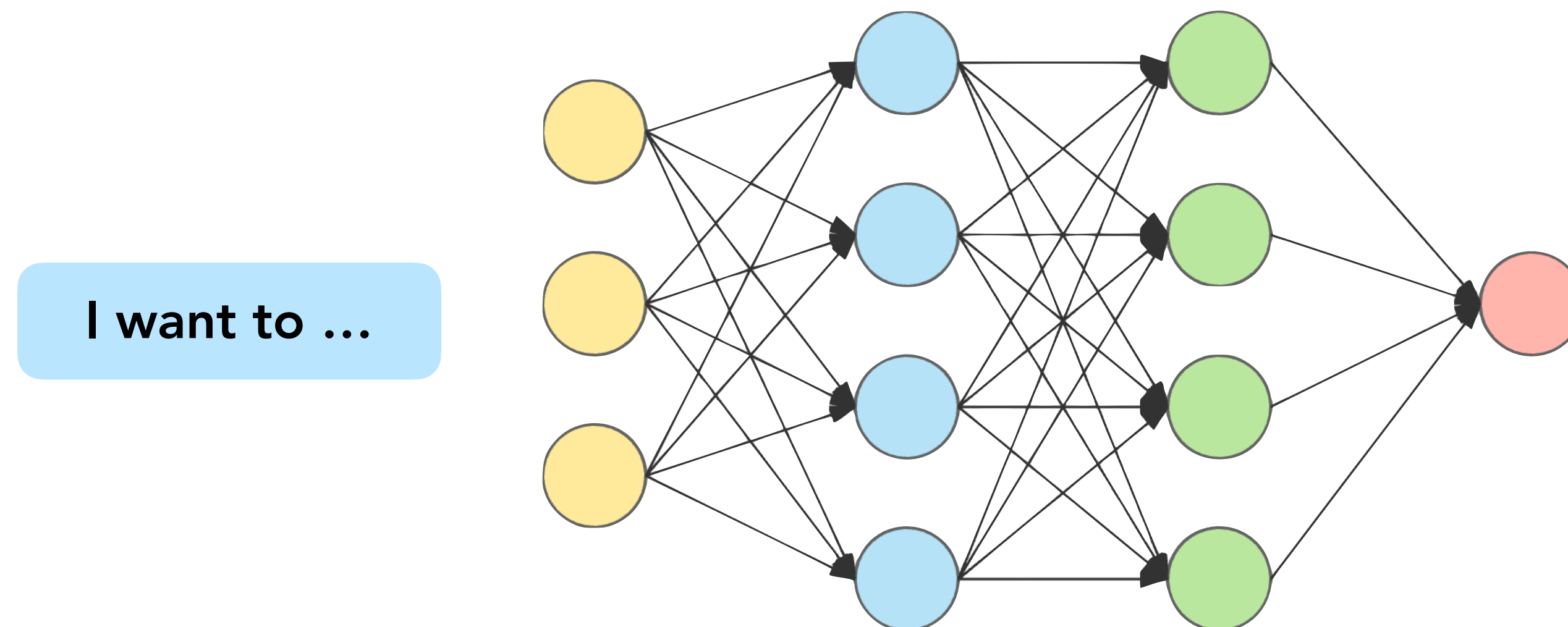
# Language Models

- Task: Given a sequence of words so far (**the context**), predict what comes next.
- Like (a supercharged) autocomplete!
- We never know for sure what comes next, but we can still make good guesses!
- Question: what is  $X$  and what is  $Y$  here? 
  - Input  $X$  = Language and Output  $Y$  = Language





# Building a Language Model

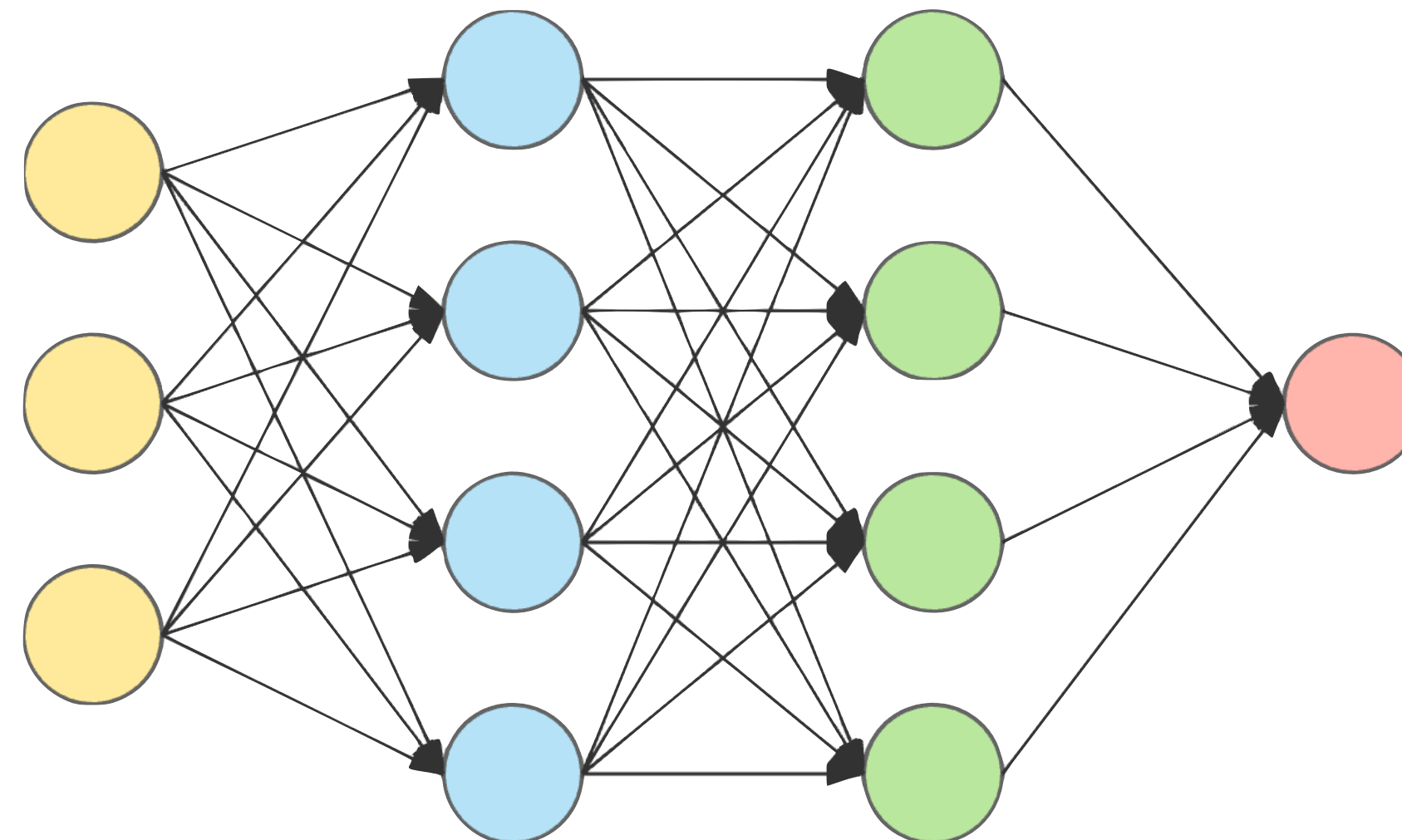


# Building a Language Model



What words can follow this?

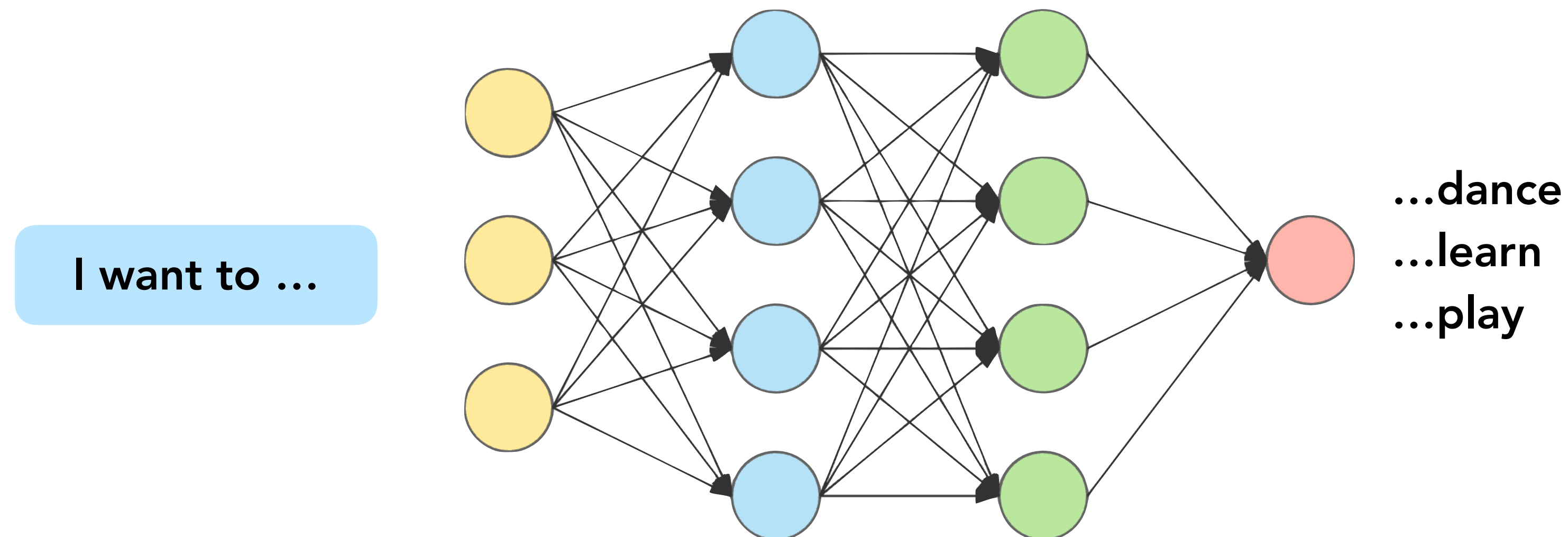
I want to ...



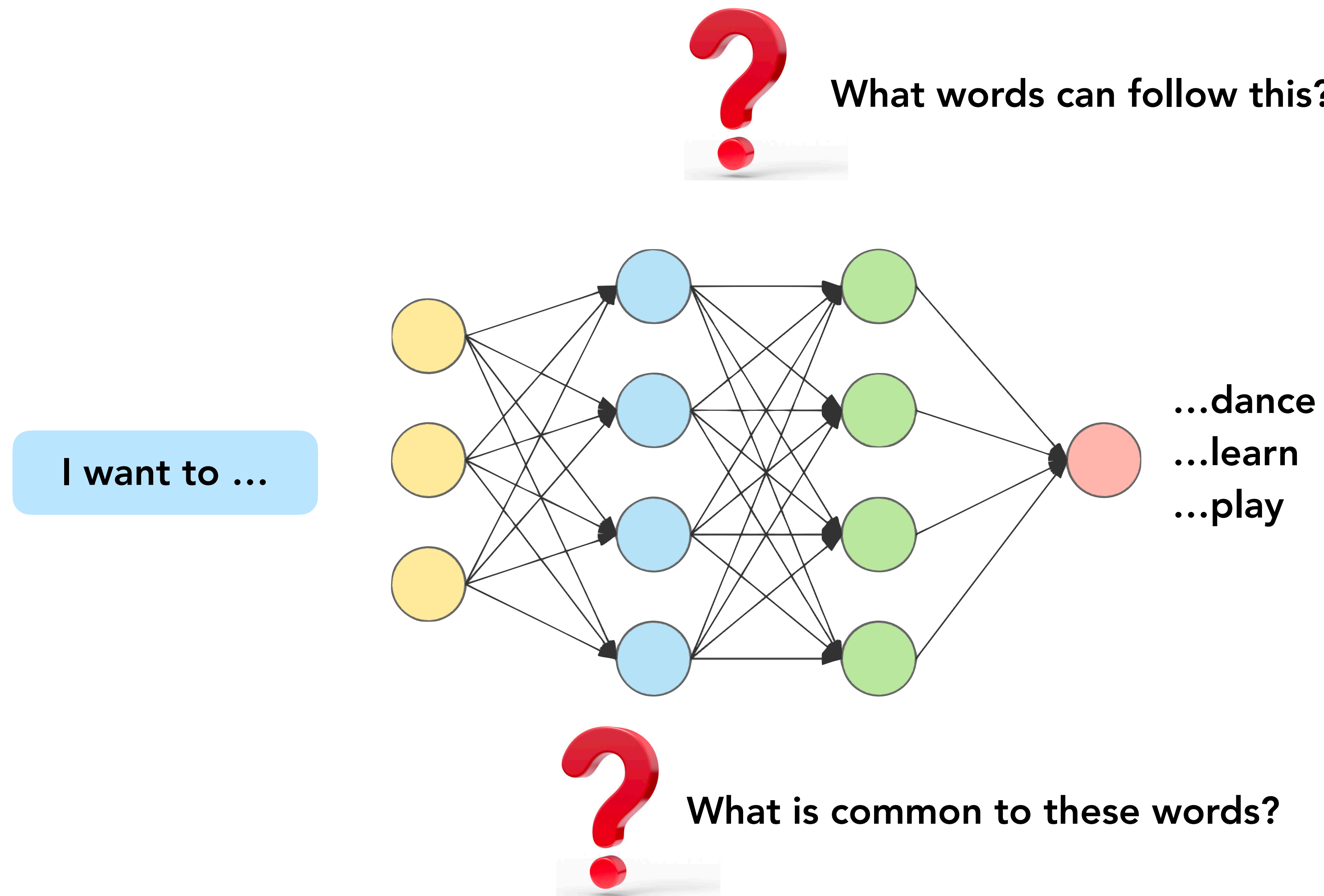
# Building a Language Model



What words can follow this?



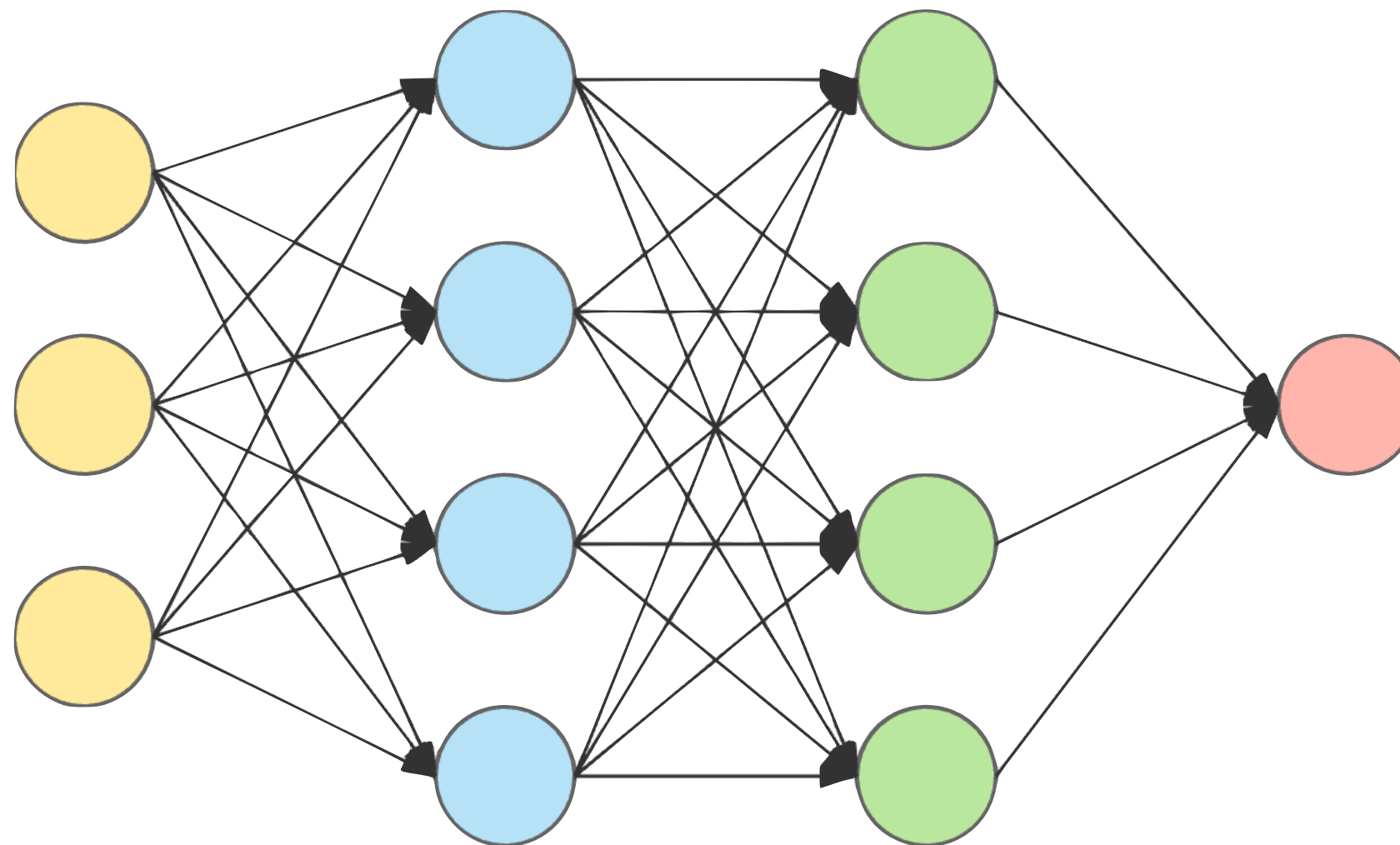
# Building a Language Model





# Building a Language Model

The 44th President of United States was

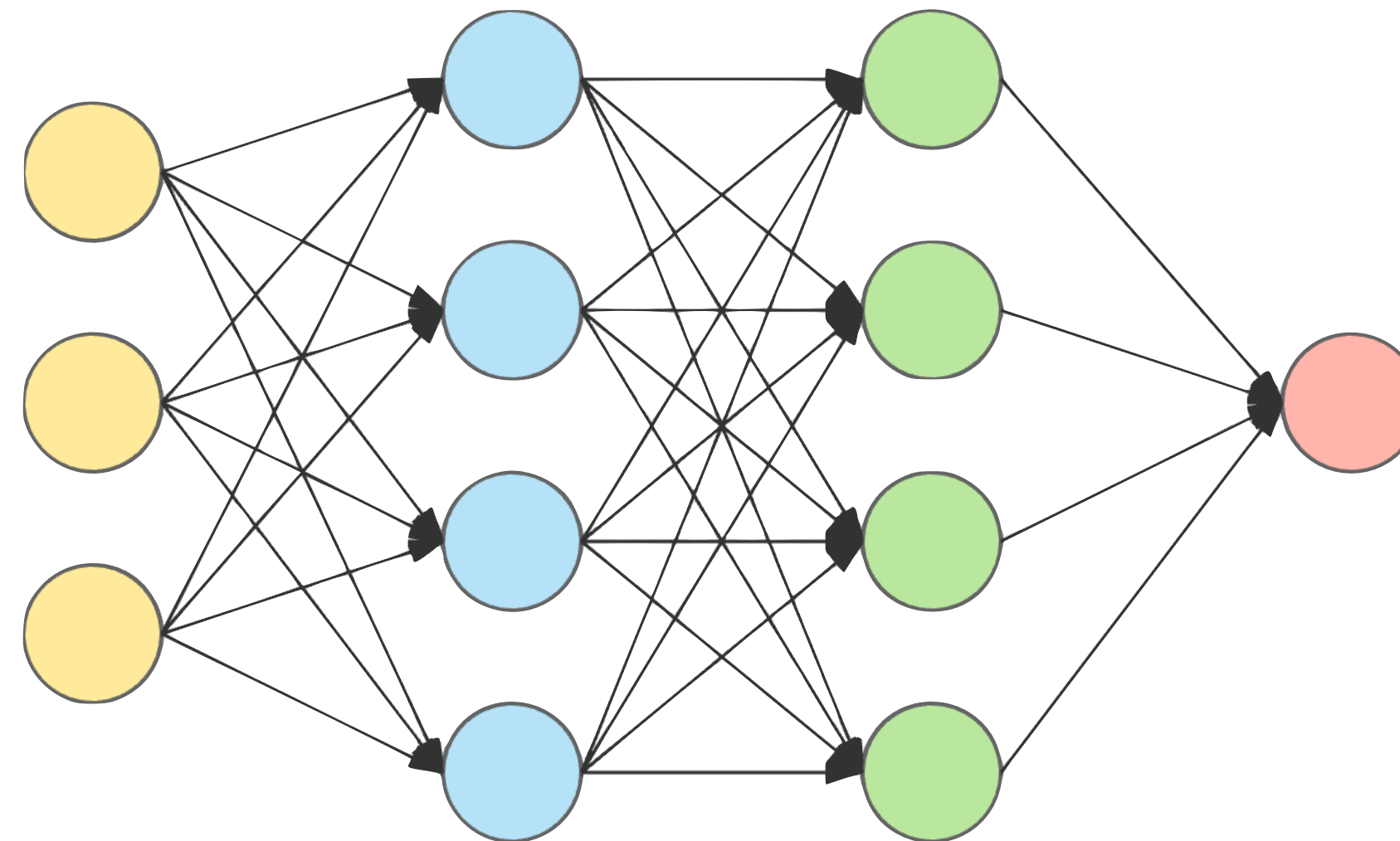


# Building a Language Model



What words can follow this?

The 44th President of United States was

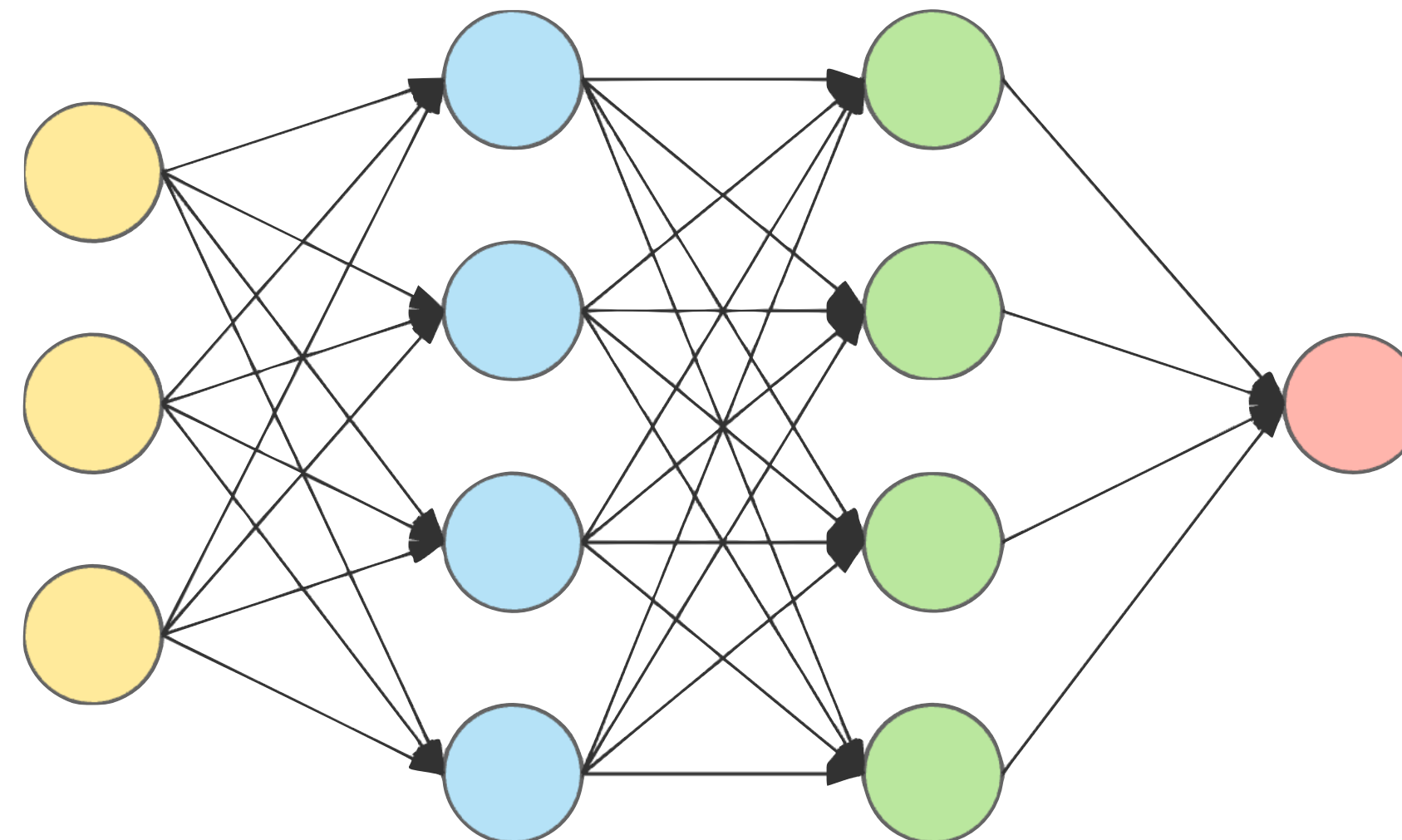


# Building a Language Model



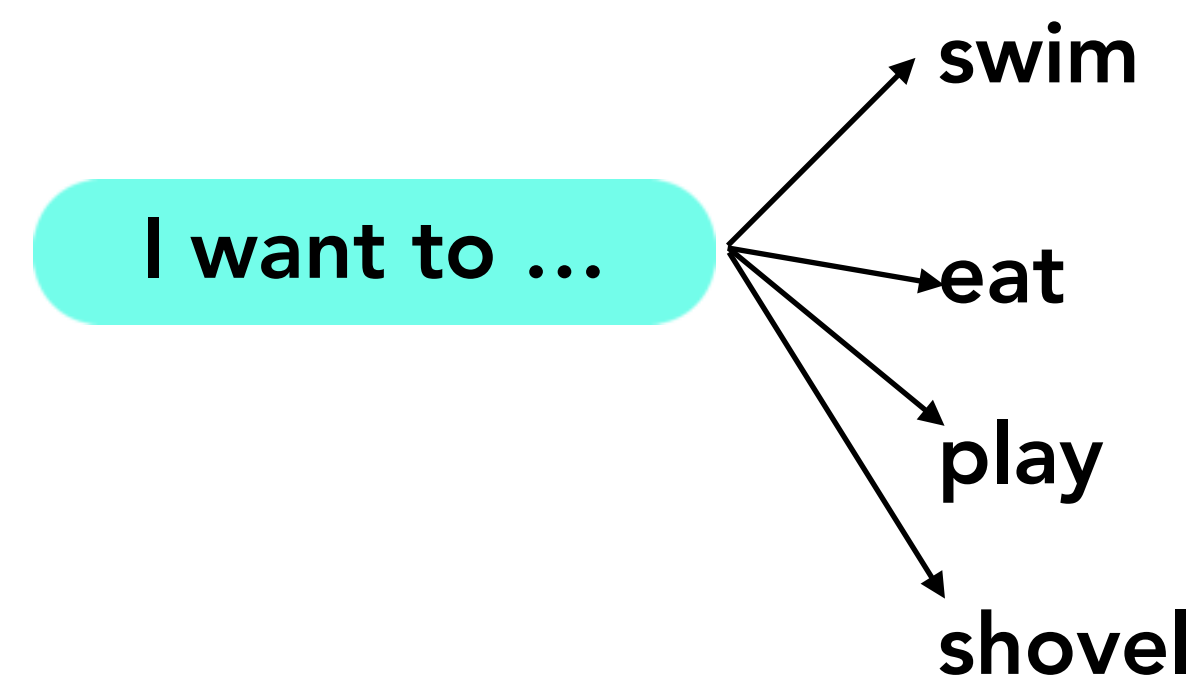
What words can follow this?

The 44th President of United States was

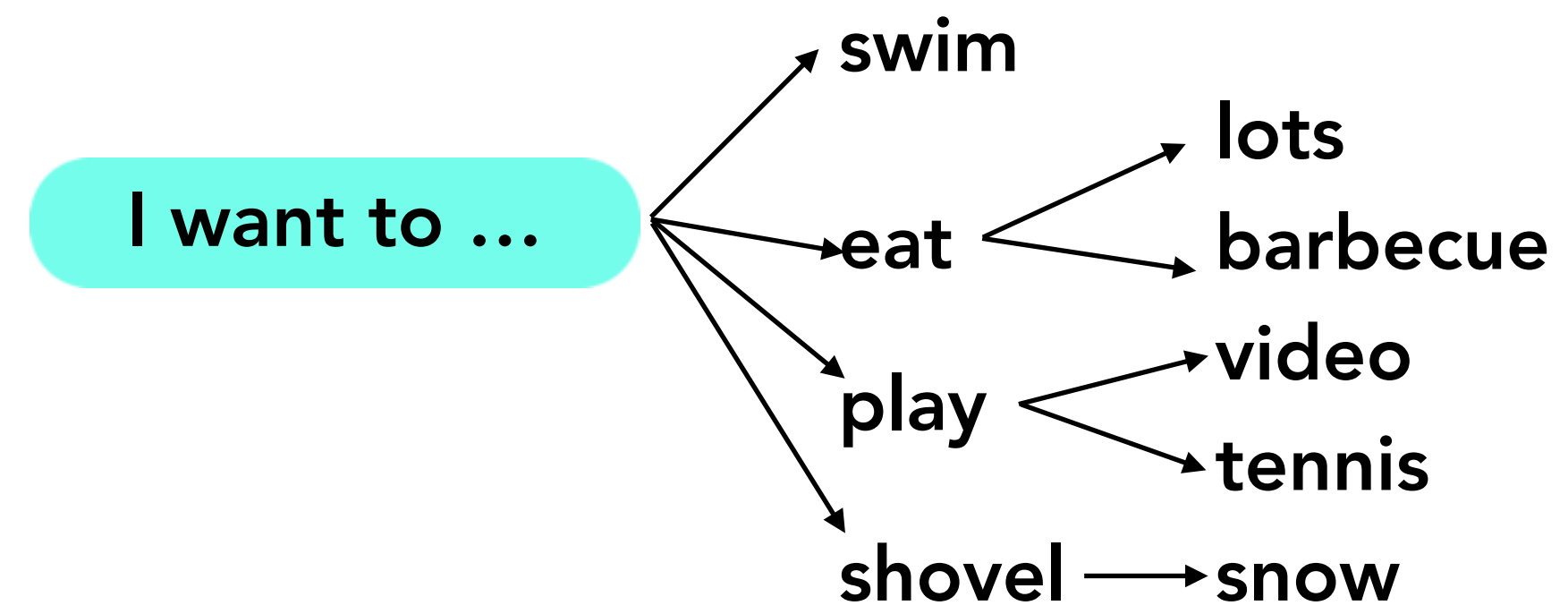


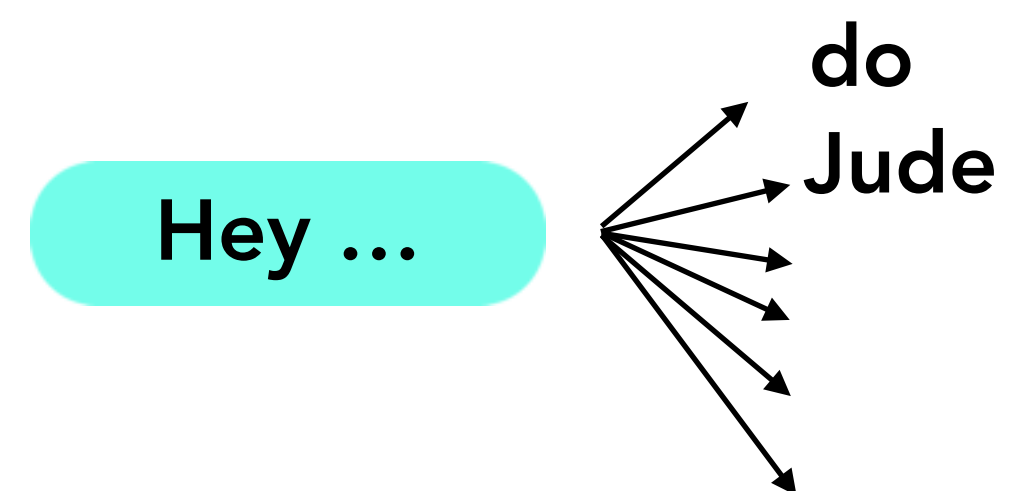
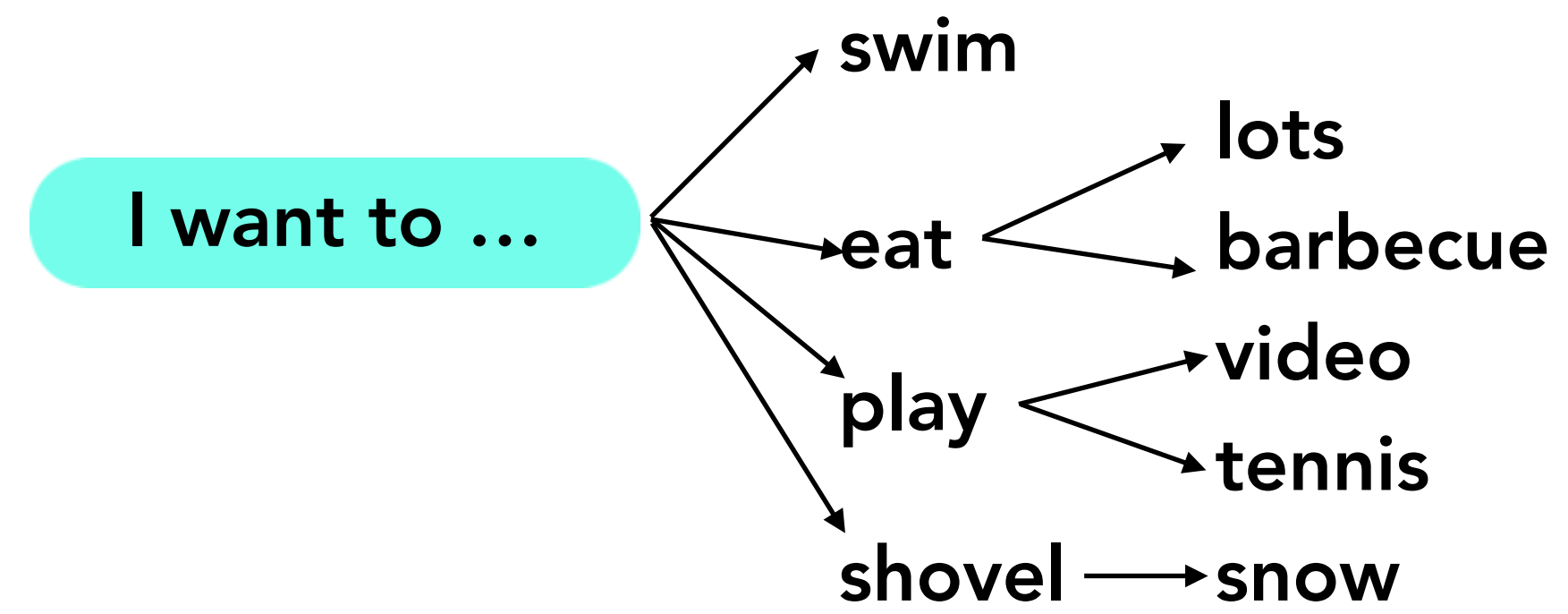
...Barack Obama

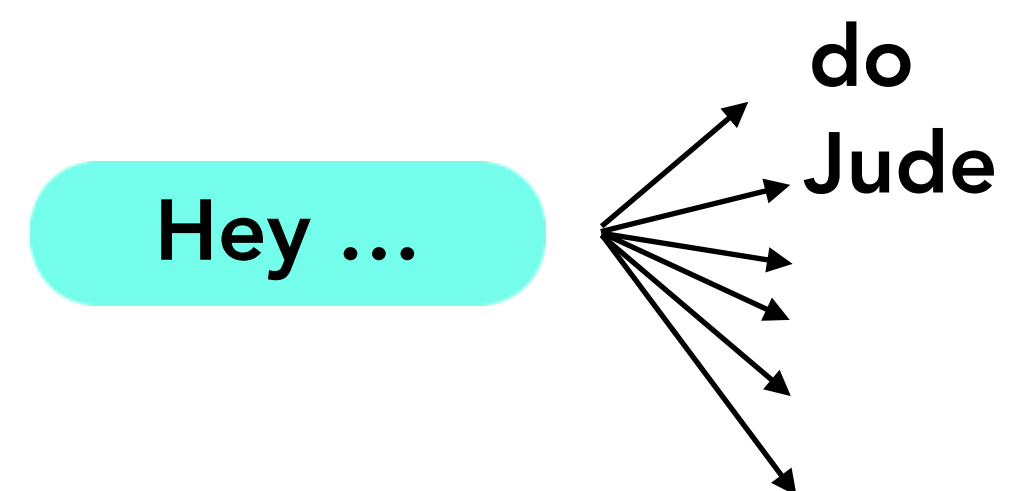
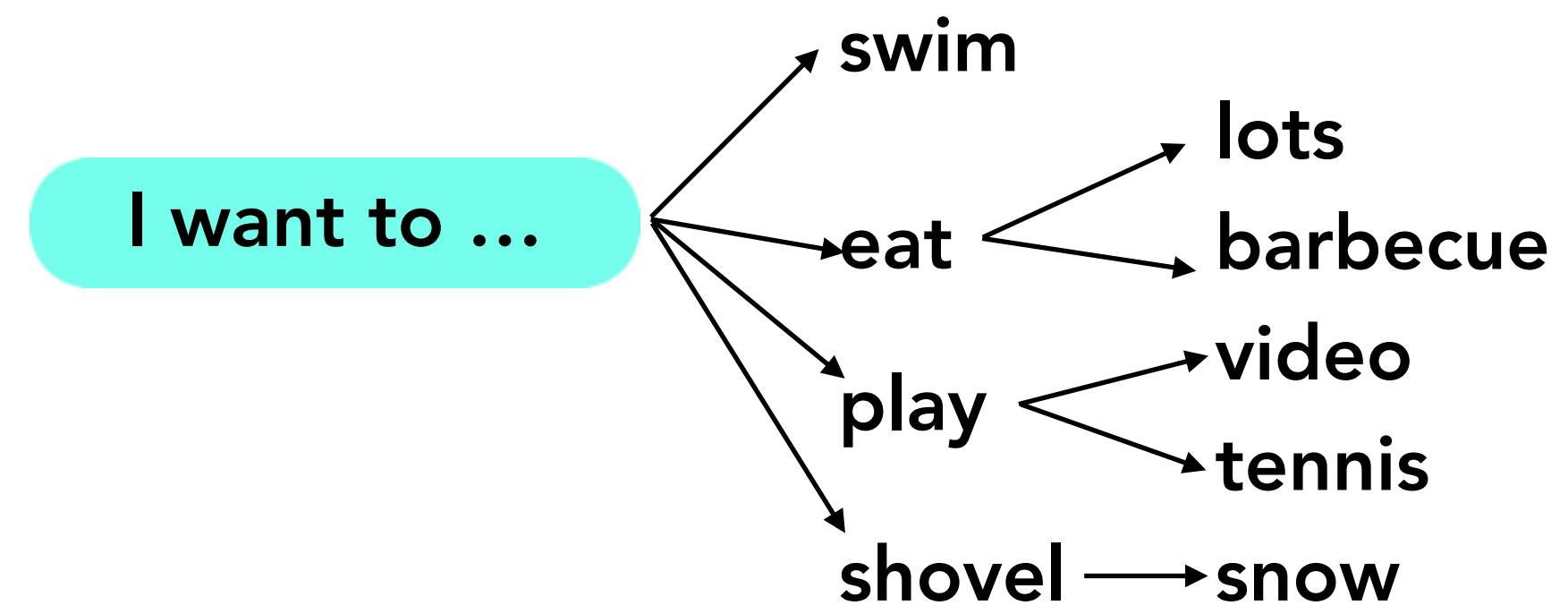
I want to ...



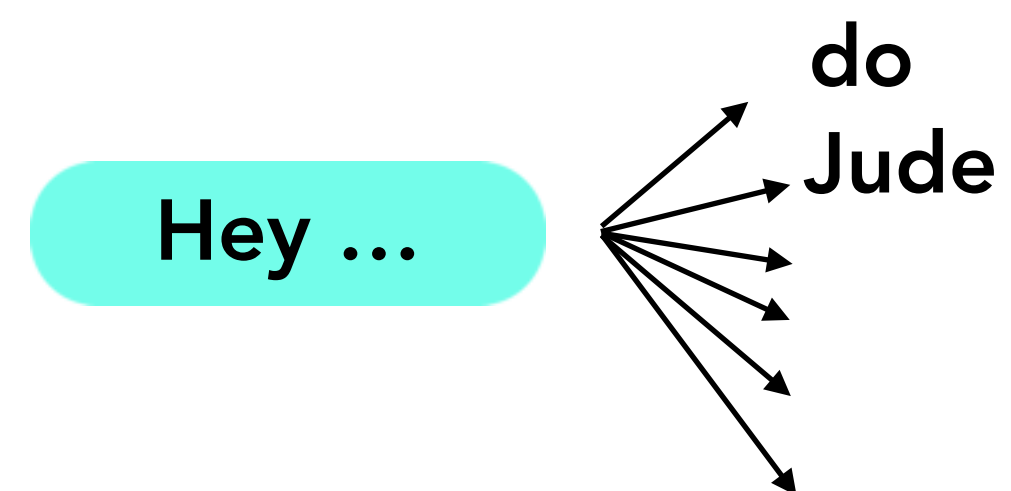
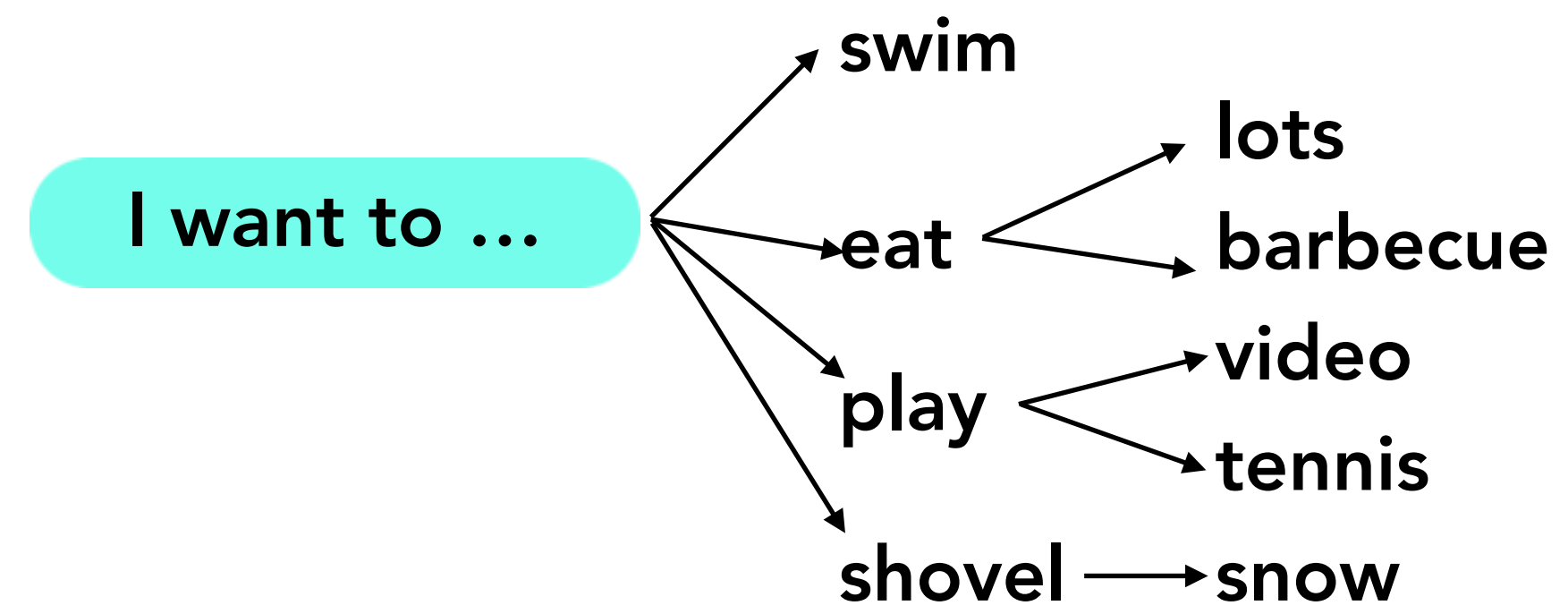








The capital of Nebraska is ... →



# n-gram Language Modeling



# n-gram Language Modeling

- Our focus: build a model that predicts the next word based on the previous one or two words

# n-gram Language Modeling

- Our focus: build a model that predicts the next word based on the previous one or two words
- **n-gram**: a sequence of  $n$  words

# n-gram Language Modeling

- Our focus: build a model that predicts the next word based on the previous one or two words
- **n-gram**: a sequence of n words
  - *I like to* = 3-gram

# n-gram Language Modeling

- Our focus: build a model that predicts the next word based on the previous one or two words
- **n-gram**: a sequence of n words
  - *I like to* = 3-gram
  - *I really want to go* = 5-gram

# n-gram Language Modeling

- Our focus: build a model that predicts the next word based on the previous one or two words
- **n-gram**: a sequence of  $n$  words
  - *I like to* = 3-gram
  - *I really want to go* = 5-gram
- **n-gram language model**: predict the next word based on the previous  $n-1$  words



# n-gram Language Modeling

- Our focus: build a model that predicts the next word based on the previous one or two words

- **n-gram**: a sequence of n words

I want to \_\_\_\_\_

2-gram

- *I like to* = 3-gram

- *I really want to go* = 5-gram

- **n-gram language model**: predict the next word based on the previous n-1 words

# n-gram Language Modeling

- Our focus: build a model that predicts the next word based on the previous one or two words

- **n-gram**: a sequence of n words

I want to \_\_\_\_\_  $\xrightarrow{\text{2-gram}}$  want to

- *I like to* = 3-gram
  - *I really want to go* = 5-gram
- **n-gram language model**: predict the next word based on the previous n-1 words

# n-gram Language Modeling

- Our focus: build a model that predicts the next word based on the previous one or two words

- **n-gram**: a sequence of n words

- *I like to* = 3-gram

- *I really want to go* = 5-gram

- **n-gram language model**: predict the next word based on the previous n-1 words

I want to \_\_\_\_\_  $\xrightarrow{\text{2-gram}}$  want to  
to \_\_\_\_\_

# n-gram Language Modeling

- Our focus: build a model that predicts the next word based on the previous one or two words

- **n-gram**: a sequence of n words

- *I like to* = 3-gram

- *I really want to go* = 5-gram

- **n-gram language model**: predict the next word based on the previous n-1 words

I want to \_\_\_\_\_

2-gram

I want  
want to  
to \_\_\_\_\_

# n-gram Language Modeling

- Our focus: build a model that predicts the next word based on the previous one or two words
- **n-gram**: a sequence of n words
  - *I like to* = 3-gram
  - *I really want to go* = 5-gram
- **n-gram language model**: predict the next word based on the previous n-1 words

I want to \_\_\_\_\_

2-gram

I want  
want to  
to \_\_\_\_\_



How does a bigram context change the words that might follow?



[illegible]

# 2-gram language models

# 2-gram language models

This is a **conditional probability distribution**:

# 2-gram language models

This is a **conditional probability distribution**:

$$P(\text{next word} = y \mid \text{previous word} = x)$$

# 2-gram language models

This is a **conditional probability distribution**:

$$P(\text{next word} = y \mid \text{previous word} = x)$$

“the probability of the next word is  $y$  given that the previous word is  $x$ ”



# 2-gram language models

This is a **conditional probability distribution**:

$$P(\text{next word} = y \mid \text{previous word} = x)$$

“the probability of the next word is  $y$  given that the previous word is  $x$ ”

I want to go **to** \_\_\_\_\_

$$P(\text{next word} = \text{was} \mid \text{previous word} = \text{to}) = 0.0$$

$$P(\text{next word} = \text{LA} \mid \text{previous word} = \text{to}) = 0.2$$

$$P(\text{next word} = \text{Europe} \mid \text{previous word} = \text{to}) = 0.1$$

$$P(\text{next word} = \text{Mexico} \mid \text{previous word} = \text{to}) = 0.1$$

$$P(\text{next word} = \text{eat} \mid \text{previous word} = \text{to}) = 0.1 \dots$$

# 2-gram language models

This is a **conditional probability distribution**:

$$P(\text{next word} = y \mid \text{previous word} = x)$$

“the probability of the next word is  $y$  given that the previous word is  $x$ ”

I want to go **to** \_\_\_\_\_

$$P(\text{next word} = \text{was} \mid \text{previous word} = \text{to}) = 0.0$$

$$P(\text{next word} = \text{LA} \mid \text{previous word} = \text{to}) = 0.2$$

$$P(\text{next word} = \text{Europe} \mid \text{previous word} = \text{to}) = 0.1$$

$$P(\text{next word} = \text{Mexico} \mid \text{previous word} = \text{to}) = 0.1$$

$$P(\text{next word} = \text{eat} \mid \text{previous word} = \text{to}) = 0.1 \dots$$

These have to add up to 1 over the vocabulary (every possible word  $y$  could be) “if we see **to** I think there’s a 20% chance the next word is LA”

# 2-gram language models

This is a **conditional probability distribution**:

$$P(\text{next word} = y \mid \text{previous word} = x)$$

“the probability of the next word is  $y$  given that the previous word is  $x$ ”

I want to go **to** \_\_\_\_\_

$$P(\text{next word} = \text{was} \mid \text{previous word} = \text{to}) = 0.0$$

$$P(\text{next word} = \text{LA} \mid \text{previous word} = \text{to}) = 0.2$$

$$P(\text{next word} = \text{Europe} \mid \text{previous word} = \text{to}) = 0.1$$

$$P(\text{next word} = \text{Mexico} \mid \text{previous word} = \text{to}) = 0.1$$

$$P(\text{next word} = \text{eat} \mid \text{previous word} = \text{to}) = 0.1 \dots$$

These have to add up to 1 over the vocabulary (every possible word  $y$  could be) “if we see **to** I think there’s a 20% chance the next word is LA”

Assume a **fixed vocabulary** of  $\sim 30,000$  words

# 2-gram language models

# 2-gram language models

- If we have these probabilities, we can build our predictive text system:



# 2-gram language models

- If we have these probabilities, we can build our predictive text system:

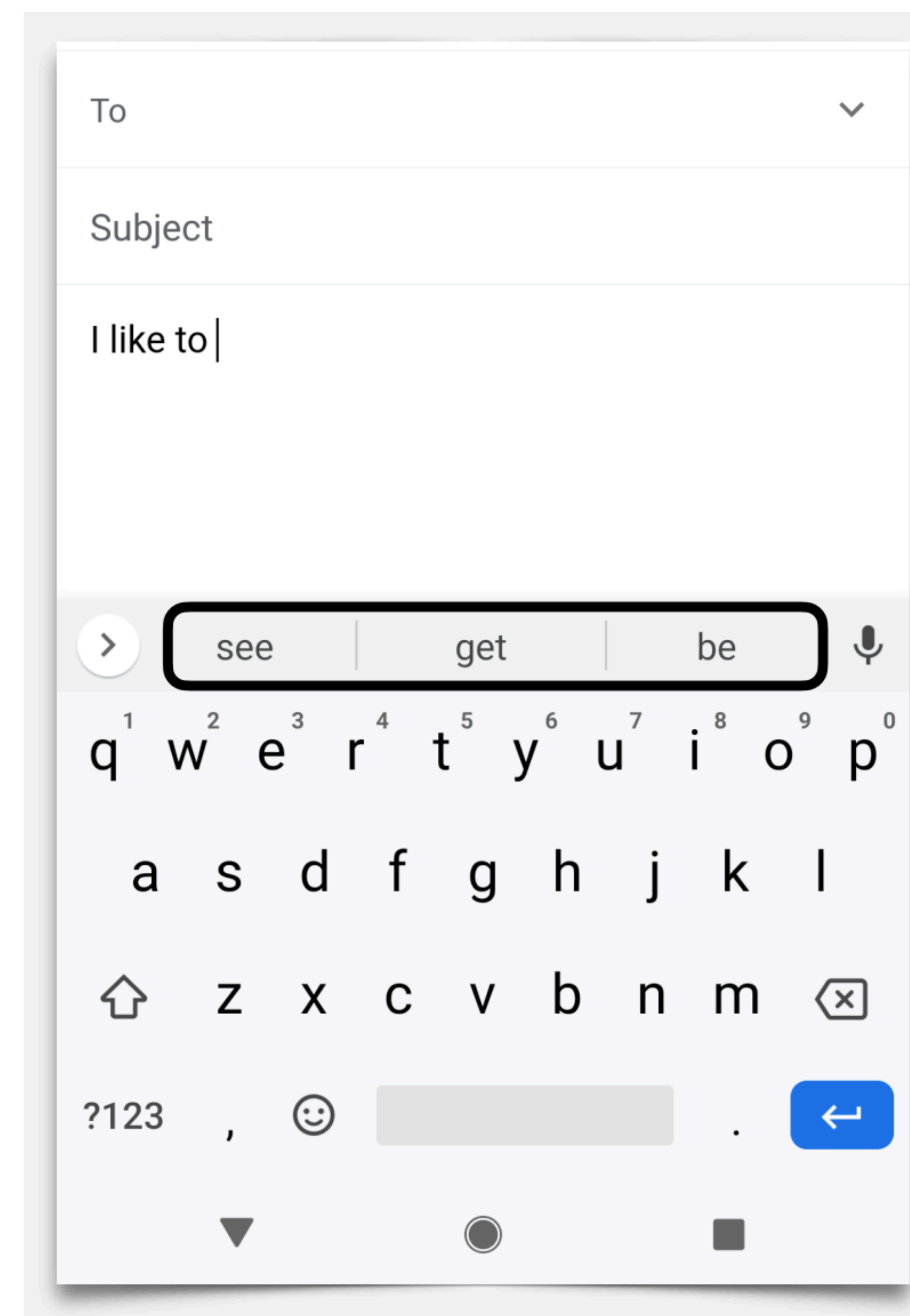
$$P(\text{next word} = \_ | \text{previous word} = \text{to})$$

# 2-gram language models

- If we have these probabilities, we can build our predictive text system:

$$P(\text{next word} = \_ | \text{previous word} = \text{to})$$

Check all the possible words from that list, pick the ones with the highest probability (most likely next words)



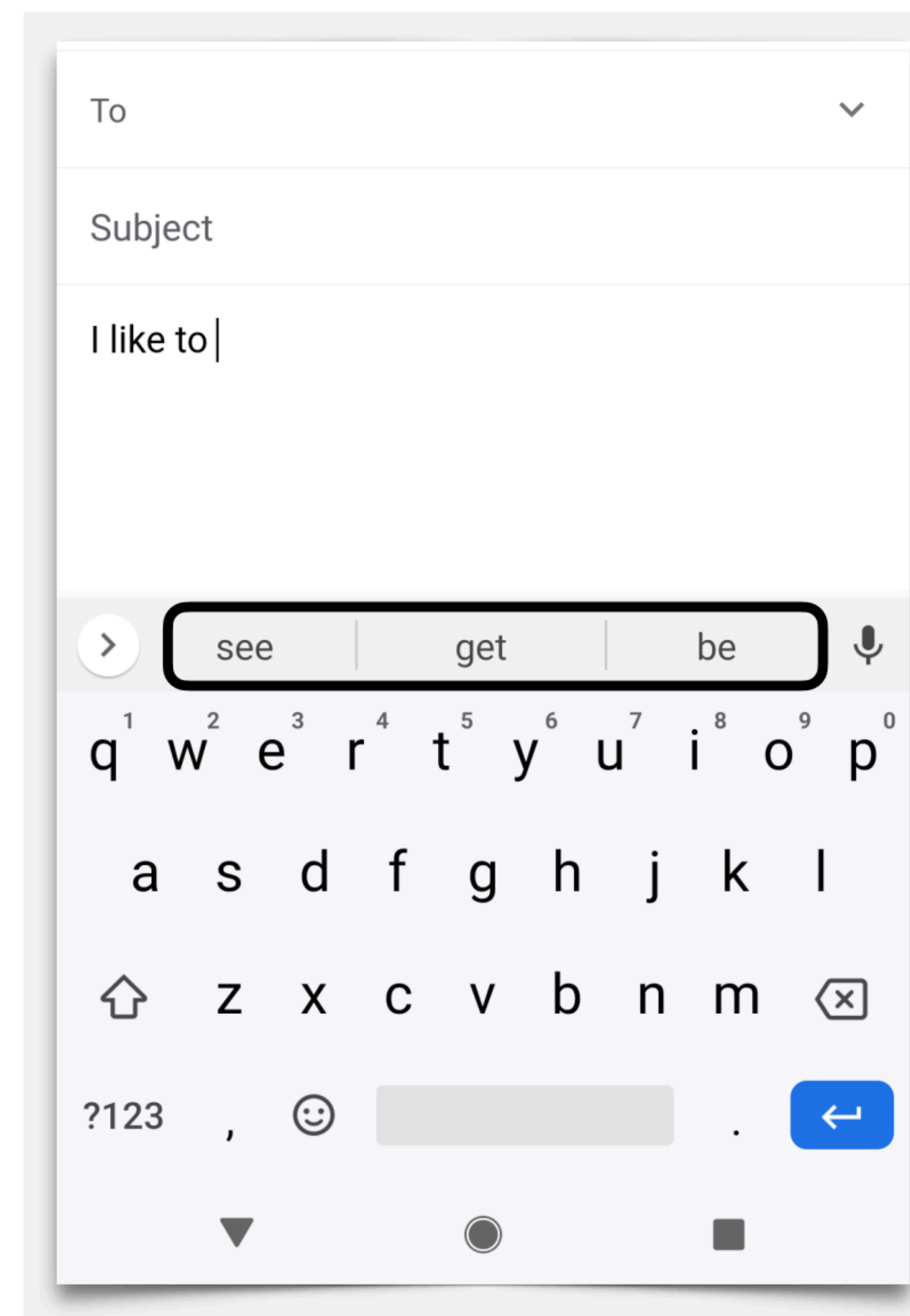
# 2-gram language models

- If we have these probabilities, we can build our predictive text system:

$$P(\text{next word} = \_ | \text{previous word} = \text{to})$$

Check all the possible words from that list, pick the ones with the highest probability (most likely next words)

- Where do these probabilities come from? We're going to **learn them** from a bunch of text data we see





Lots and lots of text data



→  
2-gram LM  
probabilities

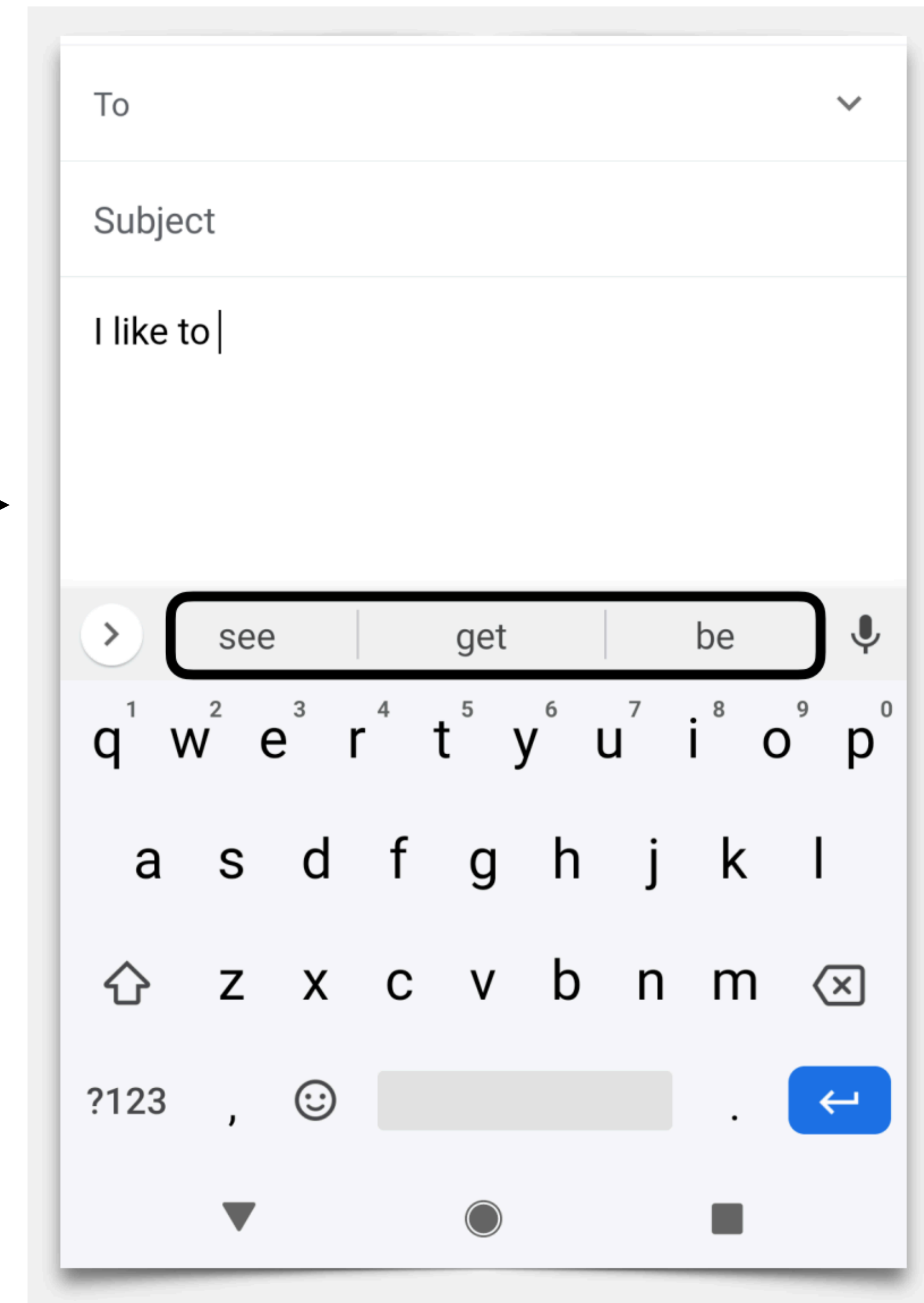
Lots and lots of text data





Lots and lots of text data

→ 2-gram LM probabilities →

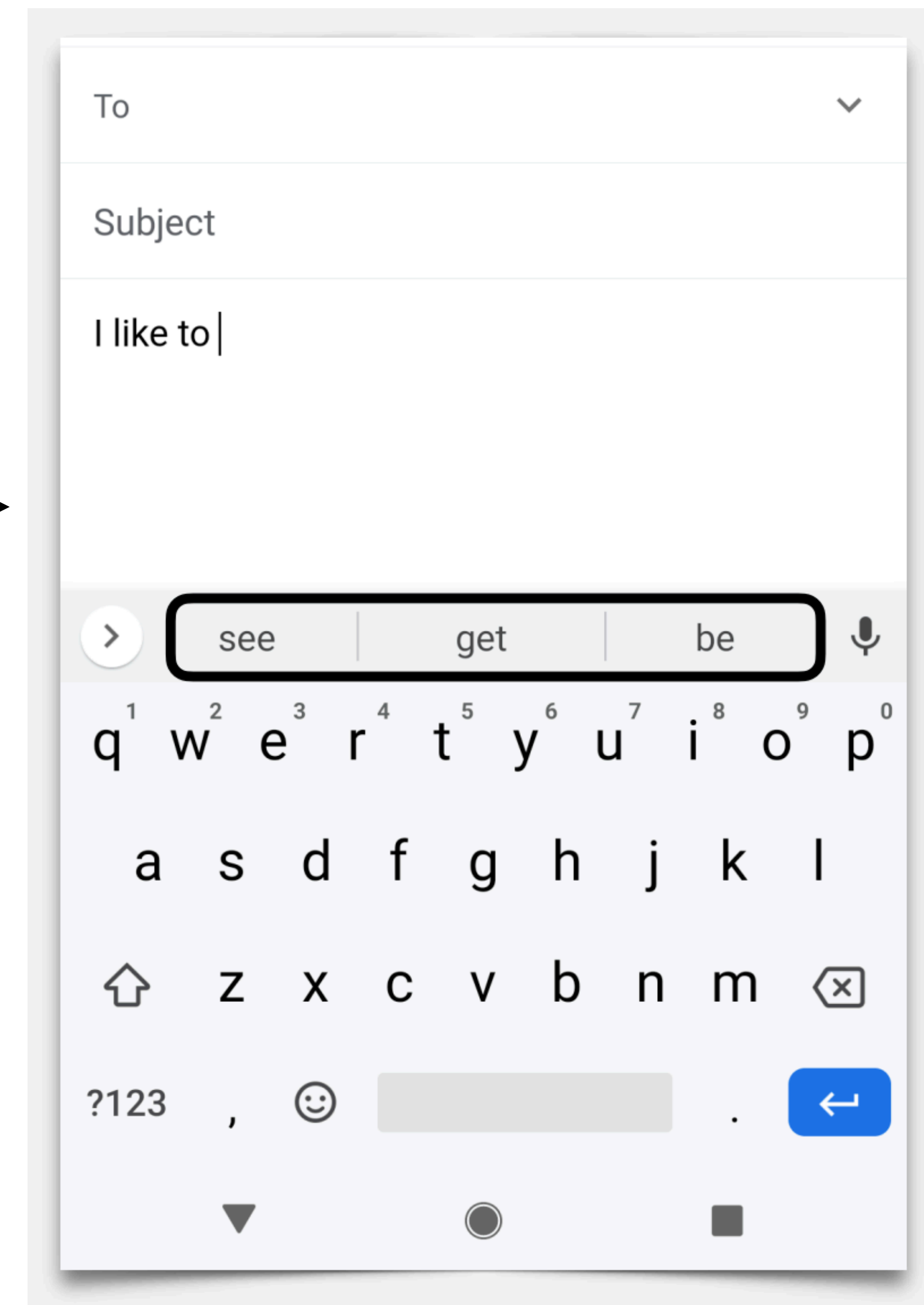




Lots and lots of text data

→ 2-gram LM probabilities →

Probability Estimation  
(Statistical Modeling)



# Probability Estimation (Statistical Modeling)



Suppose we have a biased coin that's heads with probability  $p$ , which is a number between 0 and 1, and for a normal coin,  $p = 0.5$  (equal probability of heads or tails).

Suppose we flip the coin four times and see (H, H, H, T)

1. What do you think the probability  $p$  of heads is with this coin? Take a guess!

# Probability Estimation (Statistical Modeling)



Suppose we have a biased coin that's heads with probability  $p$ , which is a number between 0 and 1, and for a normal coin,  $p = 0.5$  (equal probability of heads or tails).

Suppose we flip the coin four times and see (H, H, H, T)

1. What do you think the probability  $p$  of heads is with this coin? Take a guess!

- We don't know what  $p$  is —  $p$  could be 0.5! But  $p = 3/4 = 0.75$  maximizes the probability of the data. We'll say "this is the most likely value of  $p$ "



# Probability Estimation (Statistical Modeling)



Suppose we have a biased coin that's heads with probability  $p$ , which is a number between 0 and 1, and for a normal coin,  $p = 0.5$  (equal probability of heads or tails).

Suppose we flip the coin four times and see (H, H, H, T)

1. What do you think the probability  $p$  of heads is with this coin? Take a guess!

- We don't know what  $p$  is —  $p$  could be 0.5! But  $p = 3/4 = 0.75$  maximizes the probability of the data. We'll say "this is the most likely value of  $p$ "
- The probability of the data is  $p * p * p * (1 - p)$  — if you've taken calculus, you can take the derivative and set it equal to zero and find  $p = 0.75$

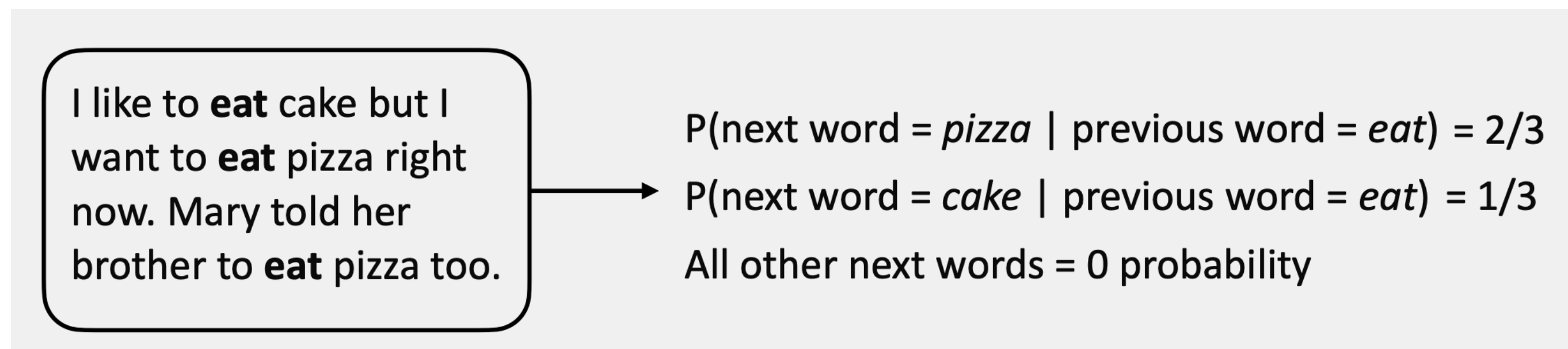


# n-gram Language Model

The decision for what words occur after a word  $w$  is exactly the same as the biased coin, but with 33,000 possible outcomes (different words) instead of 2.

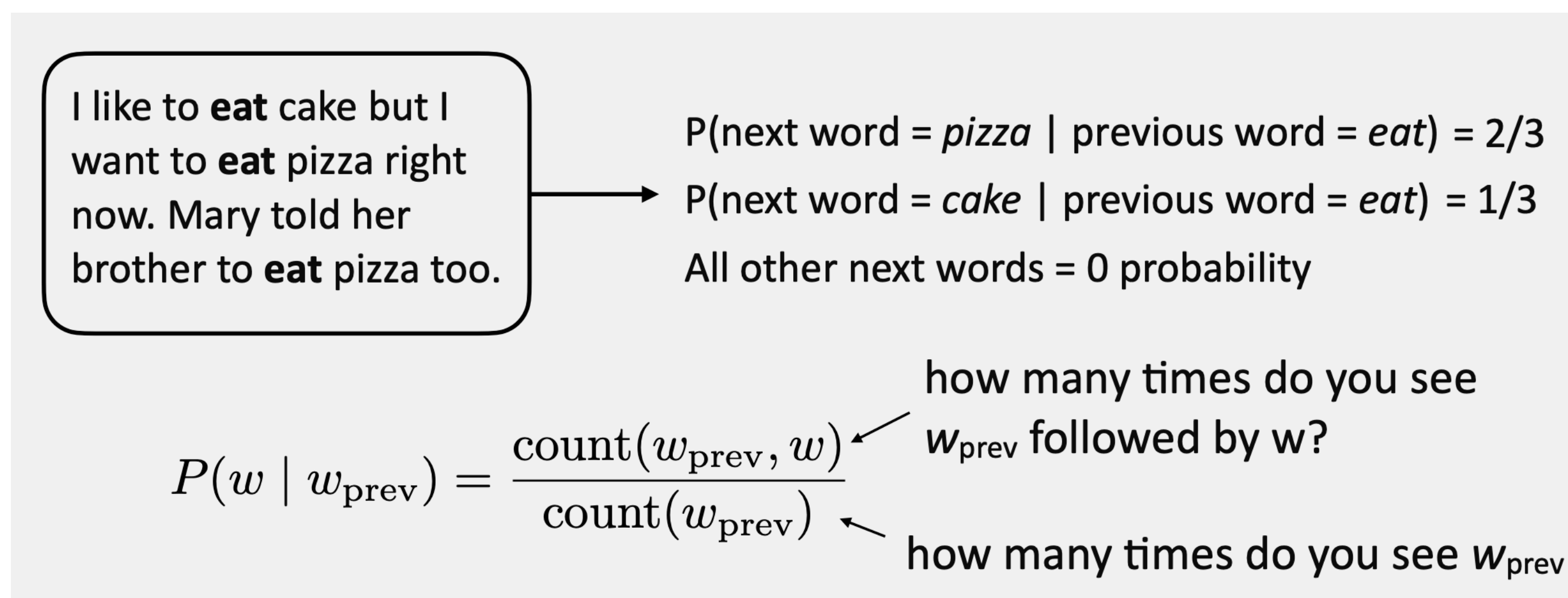
# n-gram Language Model

The decision for what words occur after a word  $w$  is exactly the same as the biased coin, but with 33,000 possible outcomes (different words) instead of 2.



# n-gram Language Model

The decision for what words occur after a word  $w$  is exactly the same as the biased coin, but with 33,000 possible outcomes (different words) instead of 2.



# Smoothing

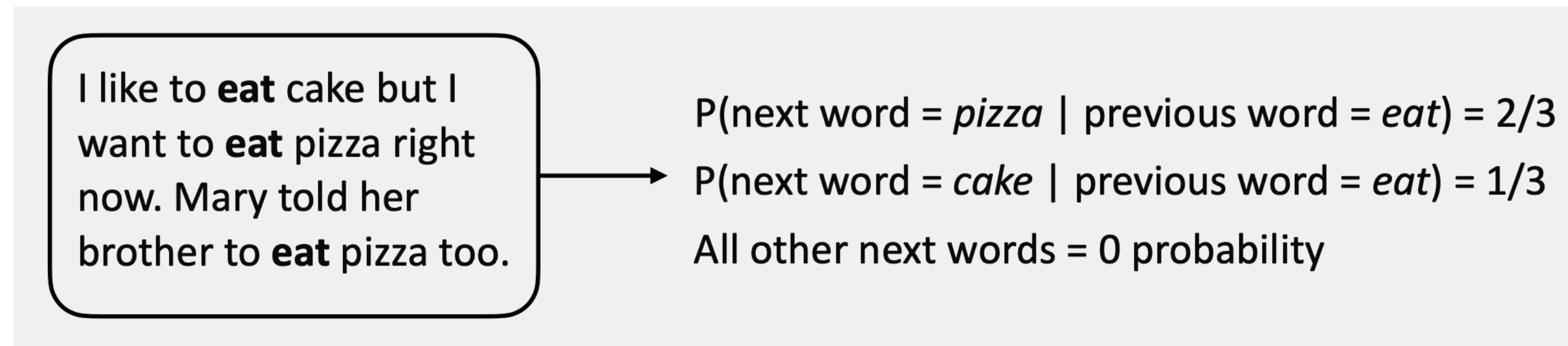
I like to **eat** cake but I want to **eat** pizza right now. Mary told her brother to **eat** pizza too.

$P(\text{next word} = \textit{pizza} \mid \text{previous word} = \textit{eat}) = 2/3$

$P(\text{next word} = \textit{cake} \mid \text{previous word} = \textit{eat}) = 1/3$

All other next words = 0 probability

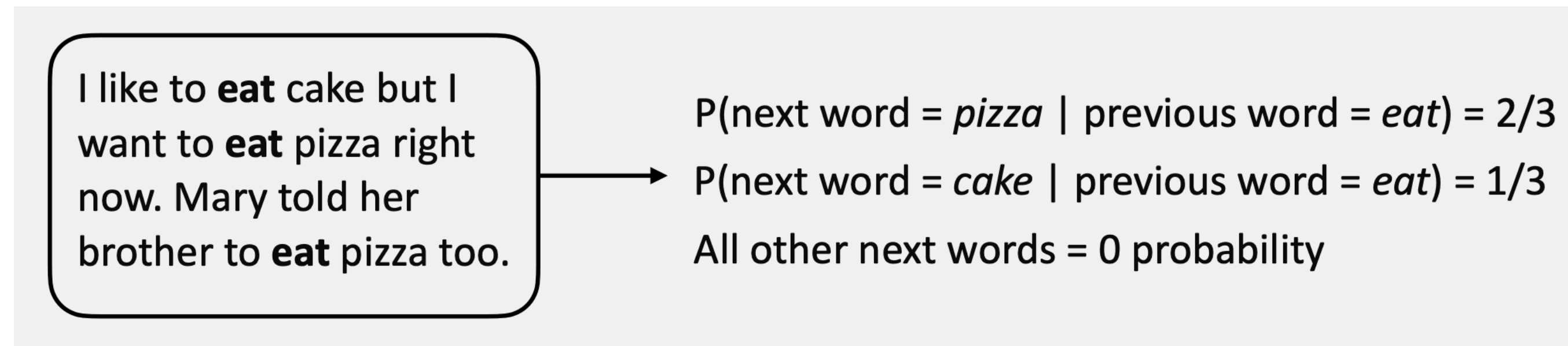
# Smoothing



- All other 29,998 words getting 0 probability just doesn't seem right. We want to assign some probability to other words



# Smoothing



- All other 29,998 words getting 0 probability just doesn't seem right. We want to assign some probability to other words
- We want to smooth the distribution from our counts

# Smoothing

I like to **eat** cake but I want to **eat** pizza right now. Mary told her brother to **eat** pizza too.

$P(\text{next word} = \text{pizza} \mid \text{previous word} = \text{eat}) = 2/3$   
 $P(\text{next word} = \text{cake} \mid \text{previous word} = \text{eat}) = 1/3$   
All other next words = 0 probability

- All other 29,998 words getting 0 probability just doesn't seem right. We want to assign some probability to other words
- We want to smooth the distribution from our counts

$$P(w \mid w_{\text{prev}}) = \lambda \frac{\text{count}(w_{\text{prev}}, w)}{\text{count}(w_{\text{prev}})} + (1 - \lambda) \frac{\text{count}(w)}{\text{total word count}}$$

a number between 0 and 1 (like 0.9)      what we had before      a *unigram* LM

# Neural Network Language Models

Neural networks are function that map  $f(\text{context}) \rightarrow \text{prediction}$

# Neural Network Language Models

Neural networks are function that map  $f(\text{context}) \rightarrow \text{prediction}$

$f$  is very, very complicated!

# Neural Network Language Models



# Neural Network Language Models

Neural networks are function that map  $f(\text{context}) \rightarrow \text{prediction}$

$f$  is very, very complicated!

$f(x) = 2x + 3$  has one input ( $x$ ) and two parameters (2 and 3)

# Neural Network Language Models

Neural networks are function that map  $f(\text{context}) \rightarrow \text{prediction}$

$f$  is very, very complicated!

$f(x) = 2x + 3$  has one input ( $x$ ) and two parameters (2 and 3)

The  $f$  we use in typical LLMs has  $>1000$  inputs and  $>1$  billion parameters!

# Neural Network Language Models

Neural networks are function that map  $f(\text{context}) \rightarrow \text{prediction}$

$f$  is very, very complicated!

$f(x) = 2x + 3$  has one input ( $x$ ) and two parameters (2 and 3)

The  $f$  we use in typical LLMs has  $>1000$  inputs and  $>1$  billion parameters!

These can be learned from data using derivatives from calculus

# Neural Network Language Models

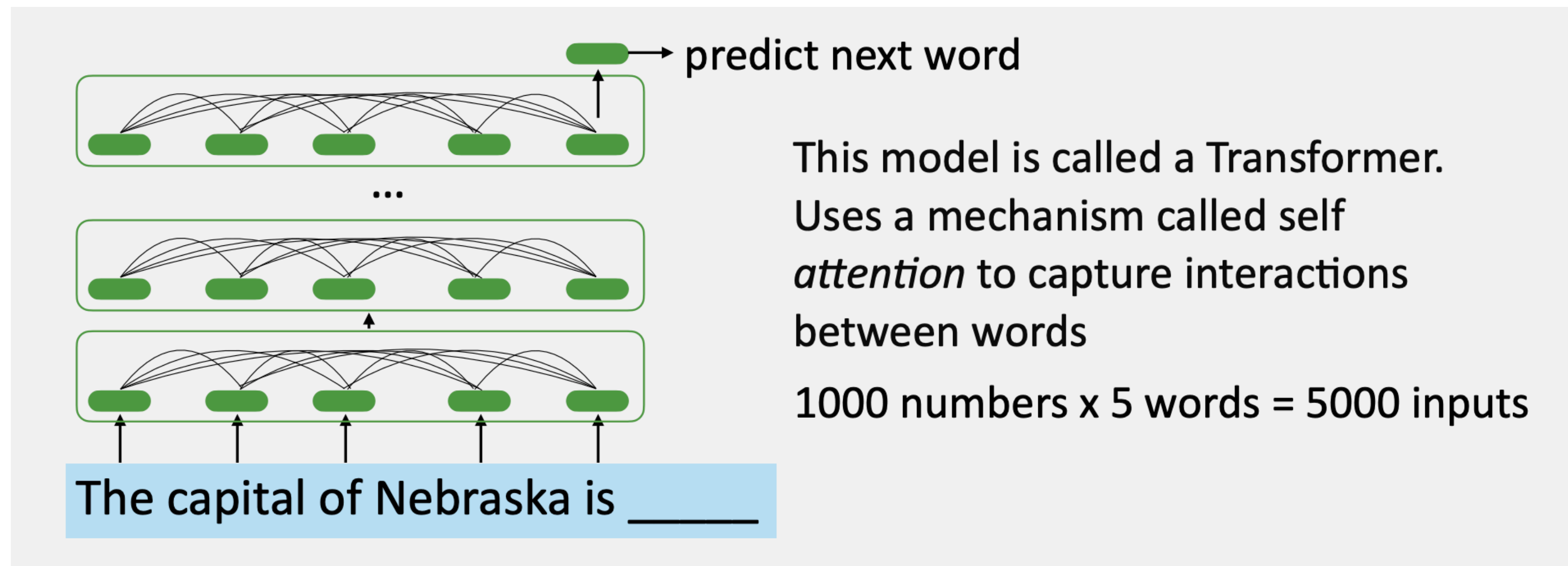
Neural networks are function that map  $f(\text{context}) \rightarrow \text{prediction}$

$f$  is very, very complicated!

$f(x) = 2x + 3$  has one input ( $x$ ) and two parameters (2 and 3)

The  $f$  we use in typical LLMs has  $>1000$  inputs and  $>1$  billion parameters!

These can be learned from data using derivatives from calculus



# Neural Network Language Models

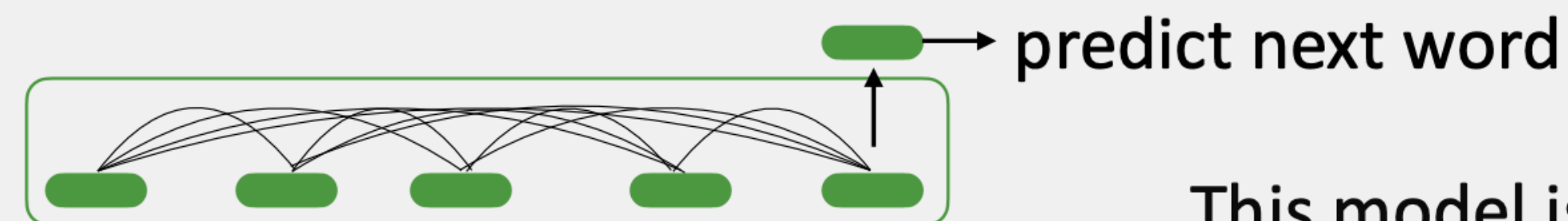
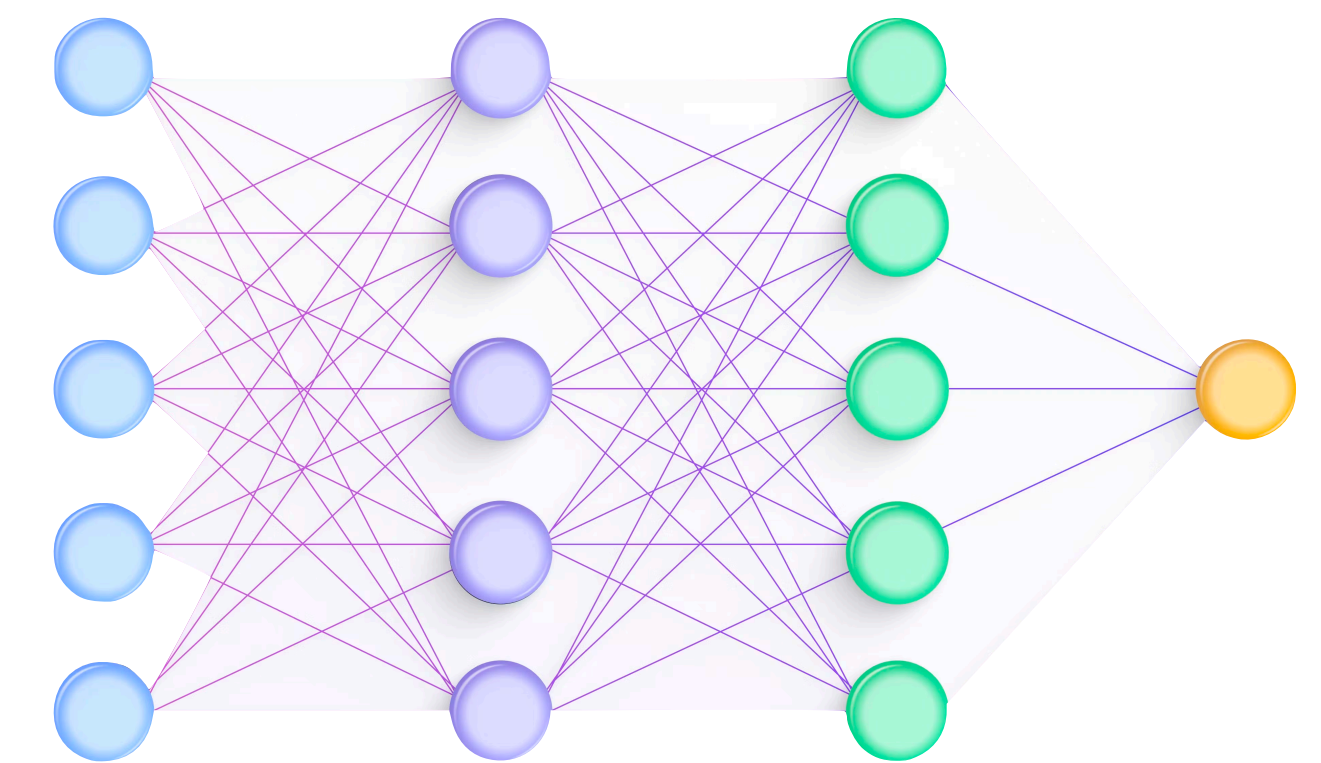
Neural networks are function that map  $f(\text{context}) \rightarrow \text{prediction}$

$f$  is very, very complicated!

$f(x) = 2x + 3$  has one input ( $x$ ) and two parameters (2 and 3)

The  $f$  we use in typical LLMs has  $>1000$  inputs and  $>1$  billion parameters!

These can be learned from data using derivatives from calculus



This model is called a Transformer.  
Uses a mechanism called self  
*attention* to capture interactions  
between words

1000 numbers x 5 words = 5000 inputs

The capital of Nebraska is \_\_\_\_\_



# Large Language Models



# Large Language Model

- Definition: **Large, general-purpose** models that can predict language by **pre-training** and **fine-tuning** for specific purposes

# Large Language Model

- Definition: **Large, general-purpose** models that can predict language by **pre-training** and **fine-tuning** for specific purposes

Pre-training



Lots of data from crawling the internet!

Retail

Entertainment

Finance

# Large Language Model

- Definition: **Large, general-purpose** models that can predict language by **pre-training** and **fine-tuning** for specific purposes

Pre-training



Lots of data from crawling the internet!

Fine-tuning



Retail



Entertainment



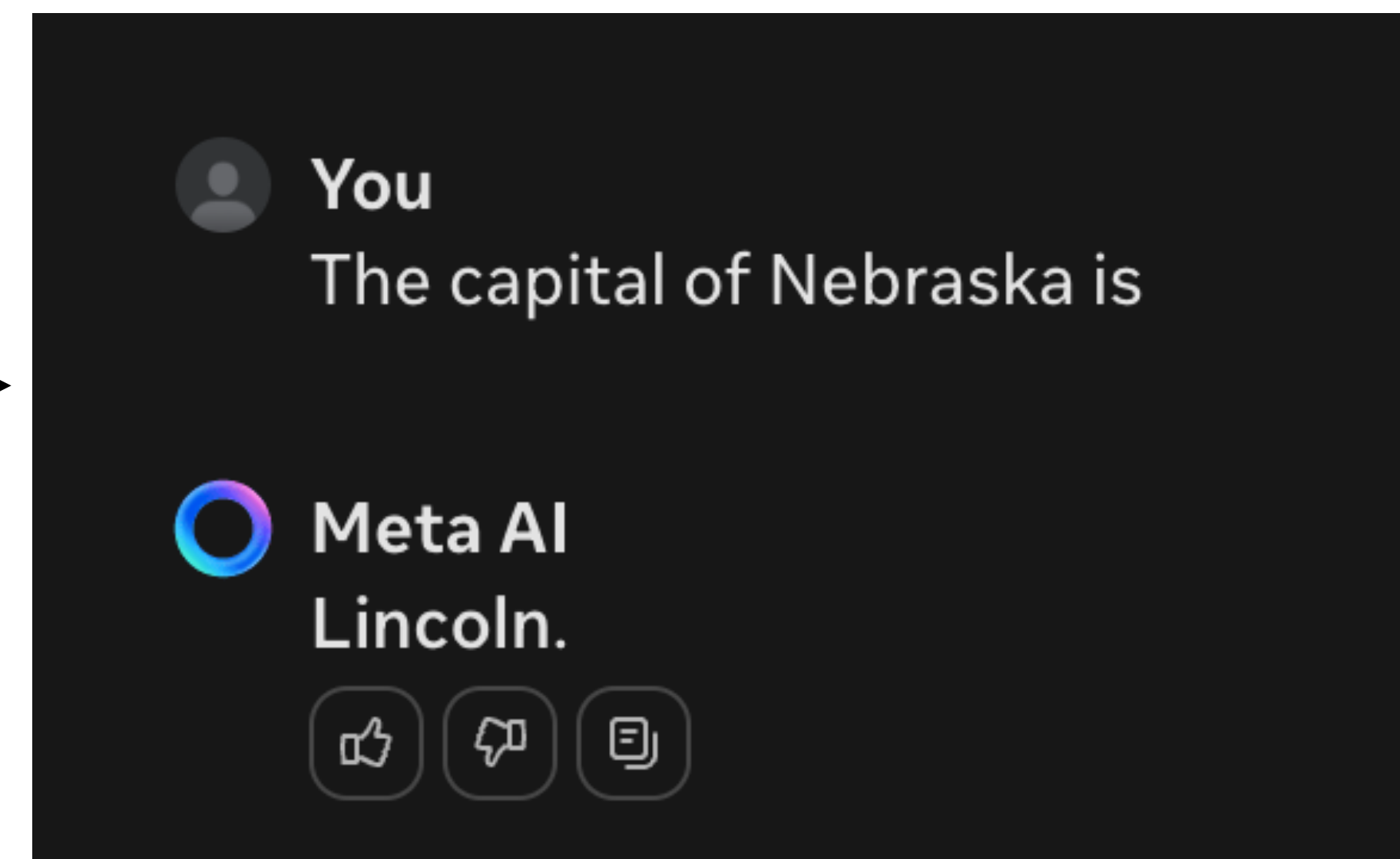
Finance

Small amounts of data!

# Using Large Language Models



**LLaMA**  
by  **Meta**



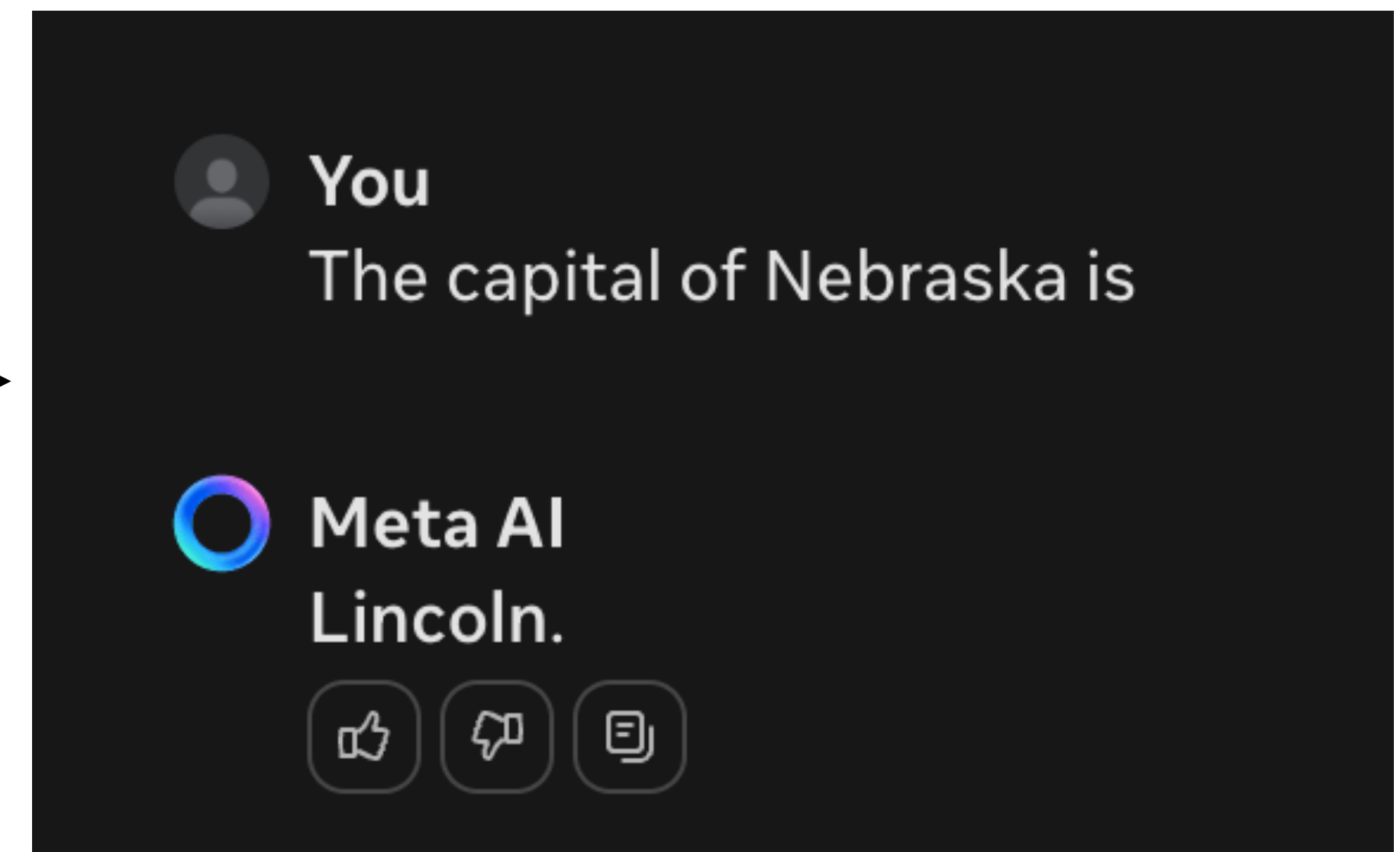


# Using Large Language Models

- These models are trained over a ton of data (a curated scrape of the web). So they will have seen information about Nebraska and Lincoln.



→ **LLaMA**  
by  **Meta**

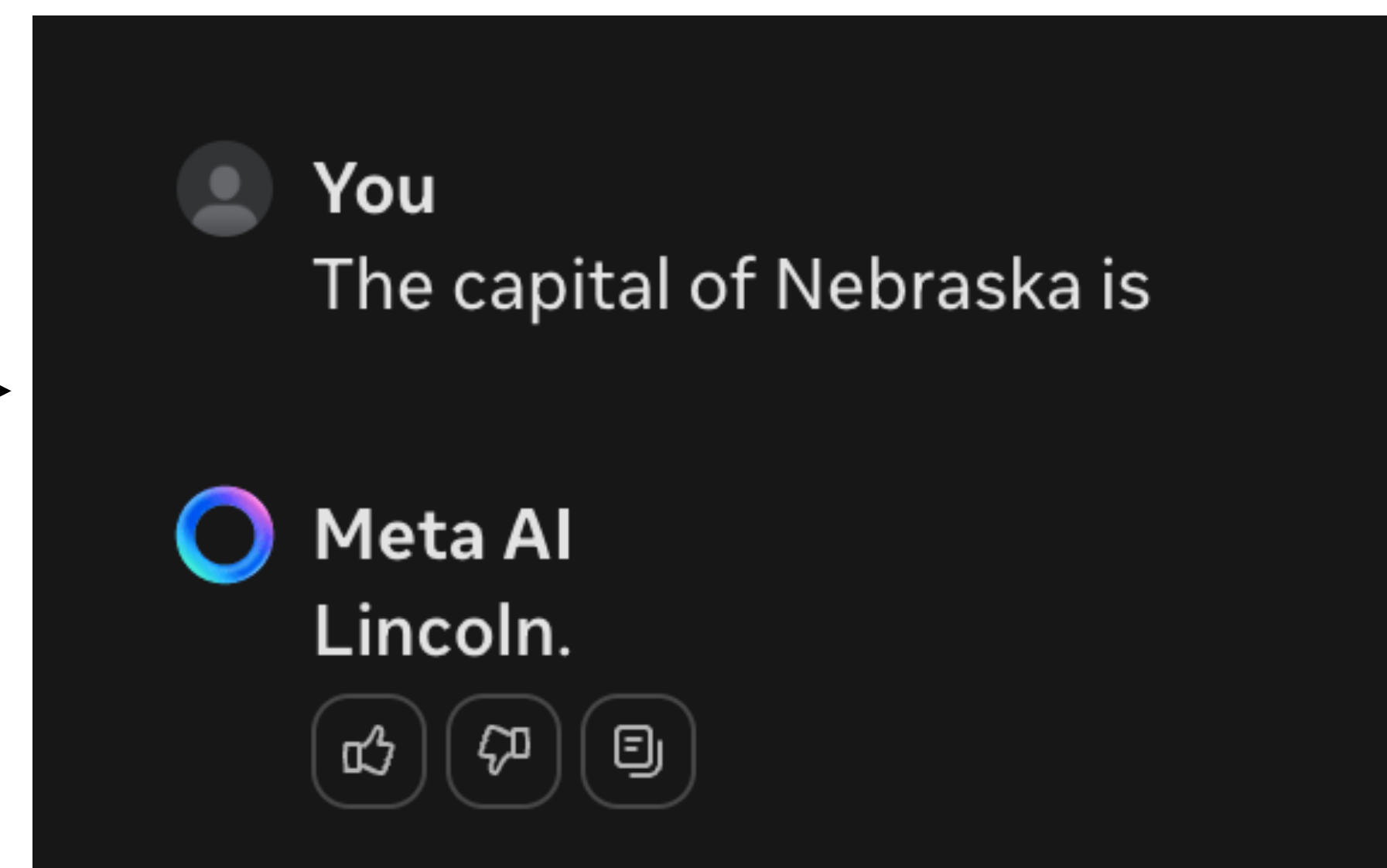


# Using Large Language Models

- These models are trained over a ton of data (a curated scrape of the web). So they will have seen information about Nebraska and Lincoln.
- A big enough model can answer questions even without being trained to do so. What else can we get these models to do?



**LLaMA**  
by  **Meta**



# Why Large Language Models?

# Why Large Language Models?

Summarize articles, podcasts or presentations

Code computer programs!

Script social media posts

Create a title for an article

Draft emails

Play games.

Assist with job searches,  
including writing resumes  
and cover letters.

Ask trivia questions.

Compose music!!!

Discover keywords for search  
engine optimization.

Formulate product  
descriptions.

Describe complex topics more simply.

Solve math problems

Create articles, blog posts  
and quizzes for websites.

Reword existing content for a different medium, such  
as a presentation transcript for a blog post.



# Why Large Language Models?

Summarize articles, podcasts or presentations

Code computer programs!

Script social media posts

Create a title for an article

Draft emails

Play games.

Assist with job searches,  
including writing resumes  
and

Ask trivia questions.

Compose music!!!

Extremely powerful: can in many cases replace  
laborious manual efforts

Describe complex topics more simply.

Generate product  
descriptions.

Solve math problems

Create articles, blog posts  
and quizzes for websites.

Reword existing content for a different medium, such  
as a presentation transcript for a blog post.



# Properties of Large Language Models



# Properties of Large Language Models

- Large



# Properties of Large Language Models

- Large
  - Trained on lots of data



# Properties of Large Language Models

- Large
  - Trained on lots of data
  - Contain lots (hundreds of billions) of parameters



# Properties of Large Language Models

- Large
  - Trained on lots of data
  - Contain lots (hundreds of billions) of parameters
  - Performance keeps growing with data and parameters





# Properties of Large Language Models

- Large
  - Trained on lots of data
  - Contain lots (hundreds of billions) of parameters
  - Performance keeps growing with data and parameters
- General-purpose



# Properties of Large Language Models



- Large
  - Trained on lots of data
  - Contain lots (hundreds of billions) of parameters
  - Performance keeps growing with data and parameters
- General-purpose
  - Can be used for many different tasks, without any modification

# Properties of Large Language Models



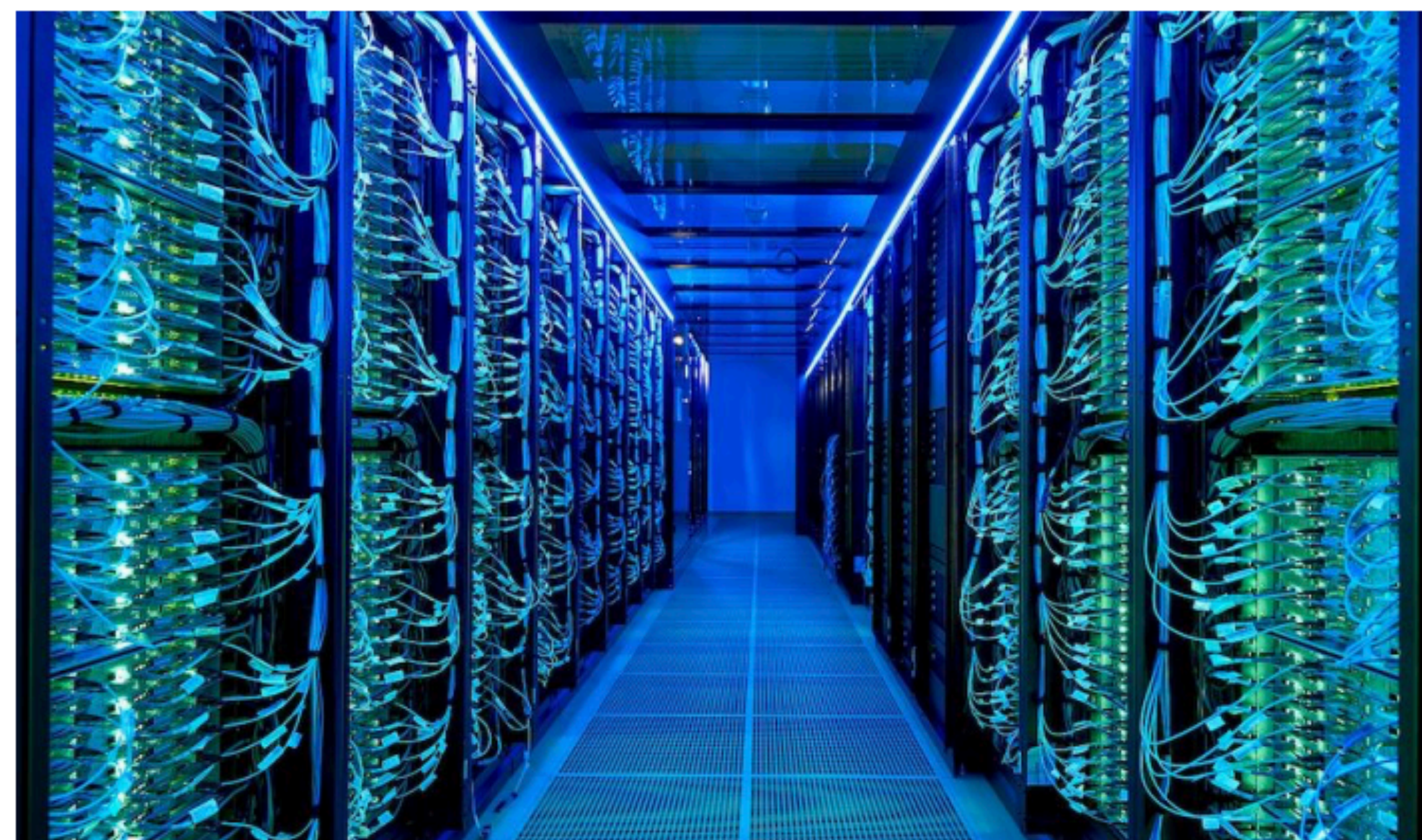
- Large
  - Trained on lots of data
  - Contain lots (hundreds of billions) of parameters
  - Performance keeps growing with data and parameters
- General-purpose
  - Can be used for many different tasks, without any modification
  - Can only be built by those who have the resources (Google, OpenAI, Meta, etc.)



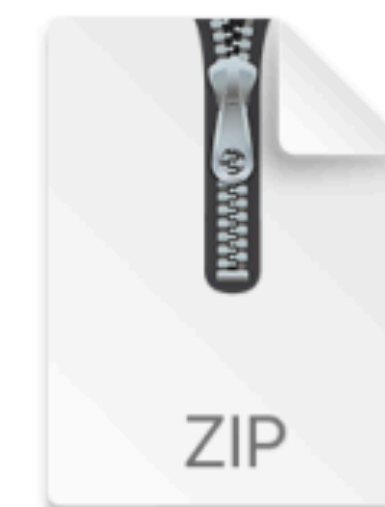
# Pre-training an LLM



Chunk of the internet,  
~10TB of text



6,000 GPUs for 12 days, ~\$2M  
~1e24 FLOPS



parameters.zip

~140GB file

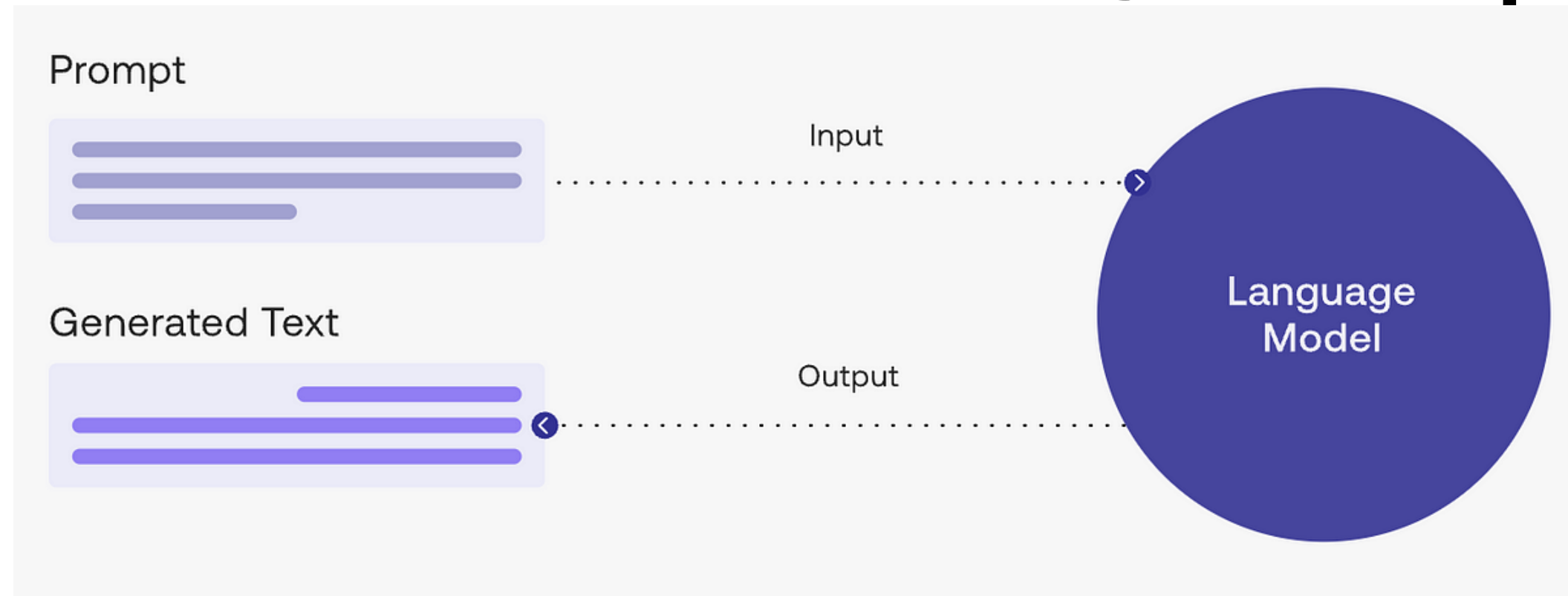
\*numbers for Llama 2 70B



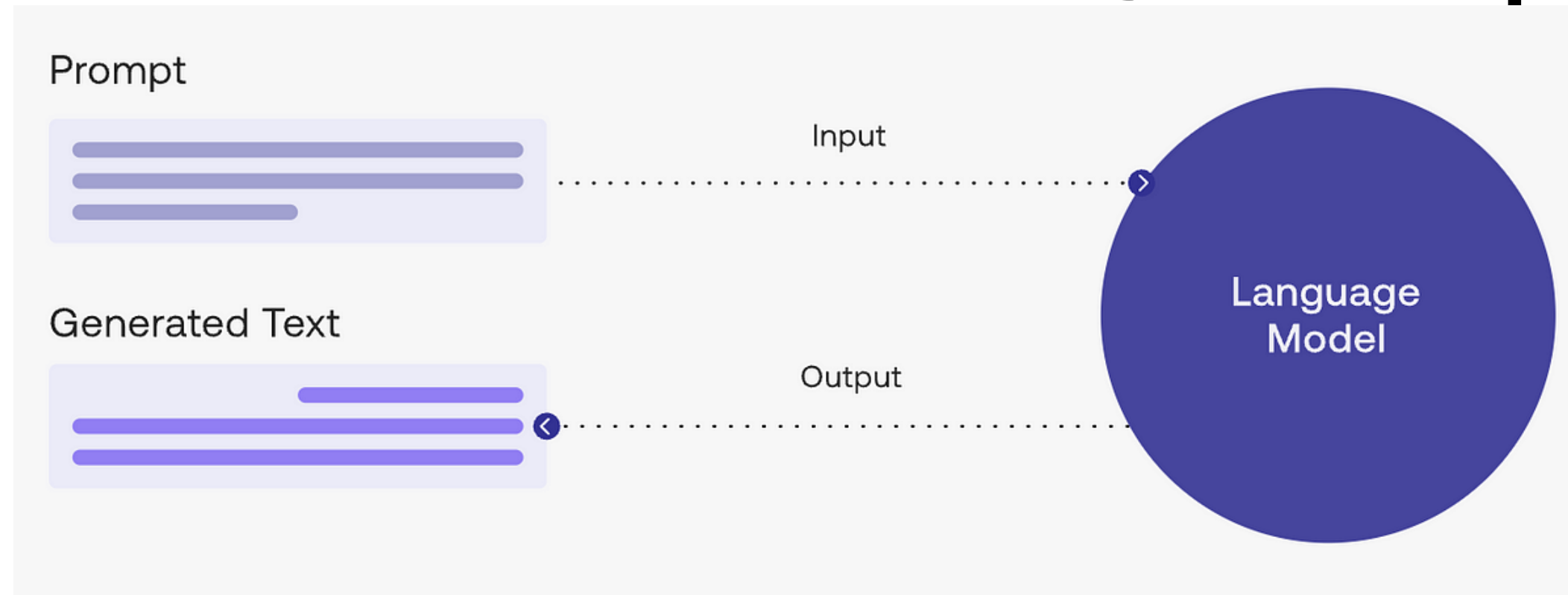
**But LLMs are not  
perfect...**



# LLMs are very dependent on prompting

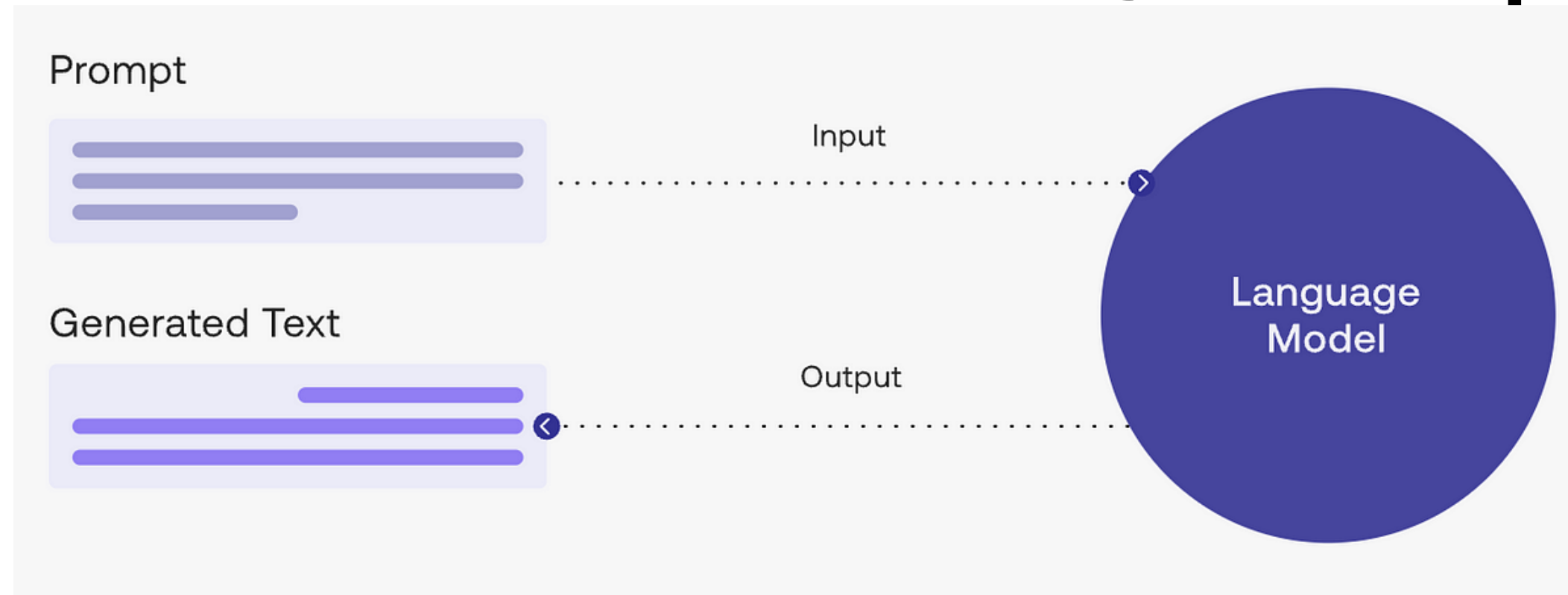


# LLMs are very dependent on prompting



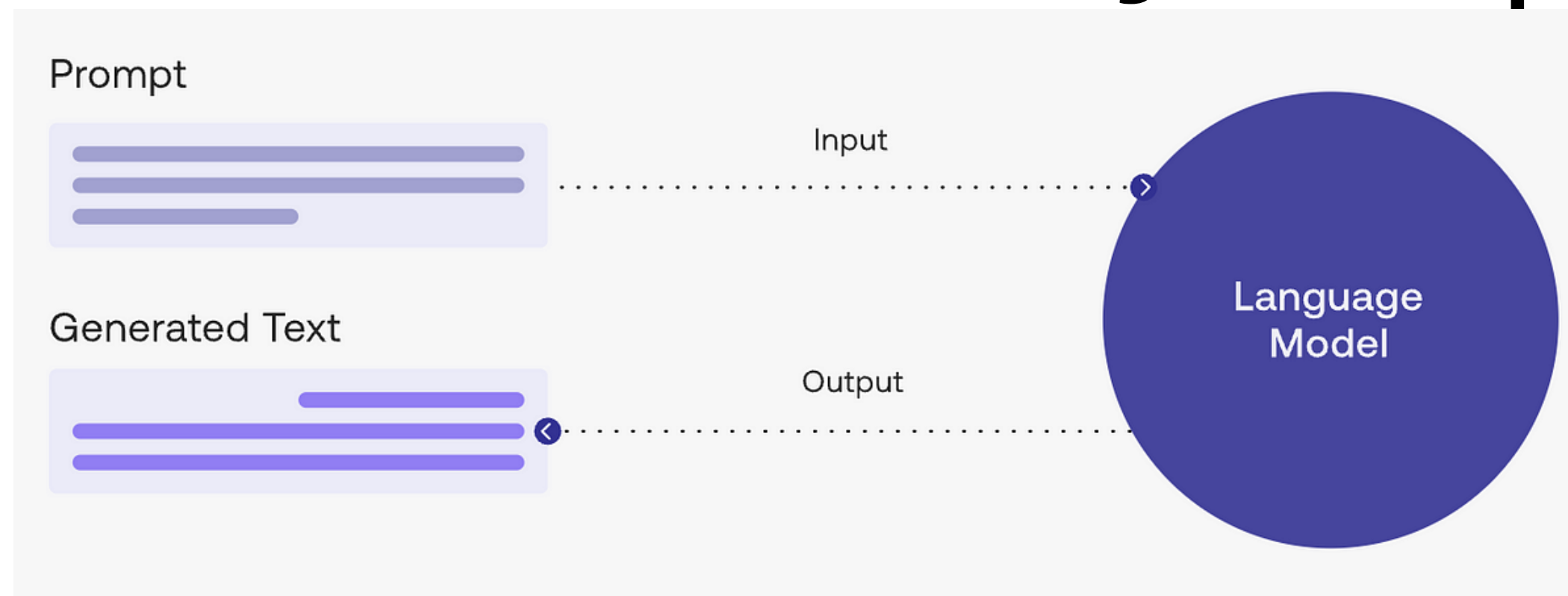
- Prompts: contexts and instruction that elicit the desired response from the language model

# LLMs are very dependent on prompting



- Prompts: contexts and instruction that elicit the desired response from the language model
- Depending on your prompt, the language model can do very well or very poorly.

# LLMs are very dependent on prompting



- Prompts: contexts and instruction that elicit the desired response from the language model
- Depending on your prompt, the language model can do very well or very poorly.

**You**  
Answer this Jeopardy style question. This territory was formally transferred to the U.S. at Castle Hill in Sitka

**Meta AI**  
What is Alaska?

Alaska was formally transferred to the United States from Russia at a ceremony on Castle Hill in Sitka on October 18, 1867<sup>1 2 3 4 5 6</sup>. The Alaska Purchase was negotiated for \$7.2 million earlier that year<sup>3</sup>. The territory became the 49th state on January 3, 1959<sup>2</sup>.

[This territory was formally tran...](#) [View Sources](#)

**You**  
Which territory was formally transferred to the U.S. at Castle Hill in Sitka?

**Meta AI**  
The territory formally transferred to the U.S. at Castle Hill in Sitka was<sup>1 2 3 4 5 6</sup>:  
Russian Alaska  
This transfer happened on October 18, 1867, after the Alaska Purchase was negotiated, in a ceremony in which Russian Commissioner Alexei Pestchouroff transferred Russian America to U.S. Commissioner General Lovell H. Rousseau.

[territory formally transferred to...](#) [View Sources](#)

# Impreciseness and Hallucinations



# Impreciseness and Hallucinations



**You**

What are the best songs written by Britney Spears?





**Meta AI**

Here are some of the best songs written by Britney Spears <sup>1</sup>:


- "Toxic" from her fourth studio album, "In the Zone" (2003)
- "Everytime" from her fourth studio album, "In the Zone" (2003)
- "My Prerogative" from her first compilation album, "Greatest Hits: My Prerogative" (2004)


# Impreciseness and Hallucinations

 **You**  
What are the best songs written by Britney Spears?

 **Meta AI**  
Here are some of the best songs written by Britney Spears <sup>1</sup>:

- "Toxic" from her fourth studio album, "In the Zone" (2003)
- "Everytime" from her fourth studio album, "In the Zone" (2003)
- "My Prerogative" from her first compilation album, "Greatest Hits: My Prerogative" (2004)

 **You**  
What are the best songs written by Miss Britney Spears?

 **Meta AI**  
Here are some of the best songs written by Britney Spears:

## Emotional Ballads

- "Everytime" (2003) - a heartfelt apology to her ex Justin Timberlake
- "Someday (I Will Understand)" (2004) - a poignant song about her miscarriage
- "Lucky" (2000) - a song about a friend's struggles with fame

## Empowering Anthems

- "Stronger" (2000) - a declaration of independence after a breakup
- "Toxic" (2003) - a dark, edgy song about a harmful relationship
- "Womanizer" (2008) - a sassy, upbeat track about a cheating ex

## Dance-Pop Hits

- "Do Somethin'" (2004) - a fun, upbeat song about letting loose

# Impreciseness and Hallucinations

You

What are the best songs written by Britney Spears

Meta AI

Here are some of the best songs written by Britney Spears:

- "Toxic" from her fourth studio album, "In the Zone"
- "Everytime" from her fourth studio album, "In the Zone"
- "My Prerogative" from her first compilation album, "Britney Spears" (2004)

"Toxic"



Single by Britney Spears

from the album *In the Zone*

B-side

"Me Against the Music"

Released

January 12, 2004

Recorded

2003

Studio

Murlyn (Stockholm)  
Record Plant (Los Angeles)

Genre

Dance-pop · techno-pop · breakbeat<sup>[1]</sup> · disco<sup>[2]</sup>

Length

3:19

Label

Jive

Songwriter(s)

Cathy Dennis · Christian Karlsson · Pontus Winnberg · Henrik Jonback

Producer(s)

Bloodshy & Avant

Britney Spears singles chronology

"Me Against the Music" (2003)

"Toxic" (2004)

"Everytime" (2004)

Music video

"Toxic" [on YouTube](#)

You

What are the best songs written by Miss Britney Spears?

Meta AI

Here are some of the best songs written by Britney Spears:

Emotional Ballads

- "Everytime" (2003) - a heartfelt apology to her ex Justin Timberlake
- "Someday (I Will Understand)" (2004) - a poignant song about her miscarriage
- "Lucky" (2000) - a song about a friend's struggles with fame

Empowering Anthems

- "Stronger" (2000) - a declaration of independence after a breakup
- "Toxic" (2003) - a dark, edgy song about a harmful relationship
- "Womanizer" (2008) - a sassy, upbeat track about a cheating ex

Dance-Pop Hits

- "Do Somethin'" (2004) - a fun, upbeat song about letting loose



# Ambiguous Language

## We're Afraid Language Models Aren't Modeling Ambiguity

Alisa Liu<sup>♥</sup> Zhaofeng Wu<sup>♦</sup> Julian Michael<sup>♦</sup> Alane Suhr<sup>♦△</sup> Peter West<sup>♥♦</sup>  
Alexander Koller<sup>♦♥</sup> Swabha Swayamdipta<sup>◇</sup> Noah A. Smith<sup>♥♦</sup> Yejin Choi<sup>♥♦</sup>  
<sup>♥</sup>Paul G. Allen School of Computer Science & Engineering, University of Washington  
<sup>♦</sup>Allen Institute for AI <sup>◇</sup>University of Southern California <sup>△</sup>UC Berkeley  
<sup>♥</sup>Saarland University <sup>♦</sup>New York University <sup>♦</sup>Massachusetts Institute of Technology



The cat was lost after leaving the house.

unable to find  
its own way

unable to  
be found

entails

neutral

The cat could not find its way.



Scott Walker, former  
governor of Wisconsin

Since I took office, Wisconsin now has the 2nd  
highest health care ranking in the country.

now, in contrast  
with before

currently, regardless of  
what it was before

entails

neutral

Wisconsin's health care ranking changed.



# Ambiguous Language

## We're Afraid Language Models Aren't Modeling Ambiguity

Alisa Liu<sup>♥</sup> Zhaofeng Wu<sup>♦</sup> Julian Michael<sup>♦</sup> Alane Suhr<sup>♦△</sup> Peter West<sup>♥♦</sup>  
 Alexander Koller<sup>♦♥</sup> Swabha Swayamdipta<sup>◇</sup> Noah A. Smith<sup>♥♦</sup> Yejin Choi<sup>♥♦</sup>  
<sup>♥</sup>Paul G. Allen School of Computer Science & Engineering, University of Washington  
<sup>♦</sup>Allen Institute for AI <sup>◇</sup>University of Southern California <sup>△</sup>UC Berkeley  
<sup>♥</sup>Saarland University <sup>♦</sup>New York University <sup>♦</sup>Massachusetts Institute of Technology



The cat was lost after leaving the house.

unable to find  
its own way

unable to  
be found

entails

The cat could not find its way.



GPT-4 struggles on  
this task!



Scott Walker, former  
governor of Wisconsin

Since I took office, Wisconsin now has the 2nd  
highest health care ranking in the country.

now, in contrast  
with before

currently, regardless of  
what it was before

entails

Wisconsin's health care ranking changed.





# Ethical Concerns

# Ethical Concerns

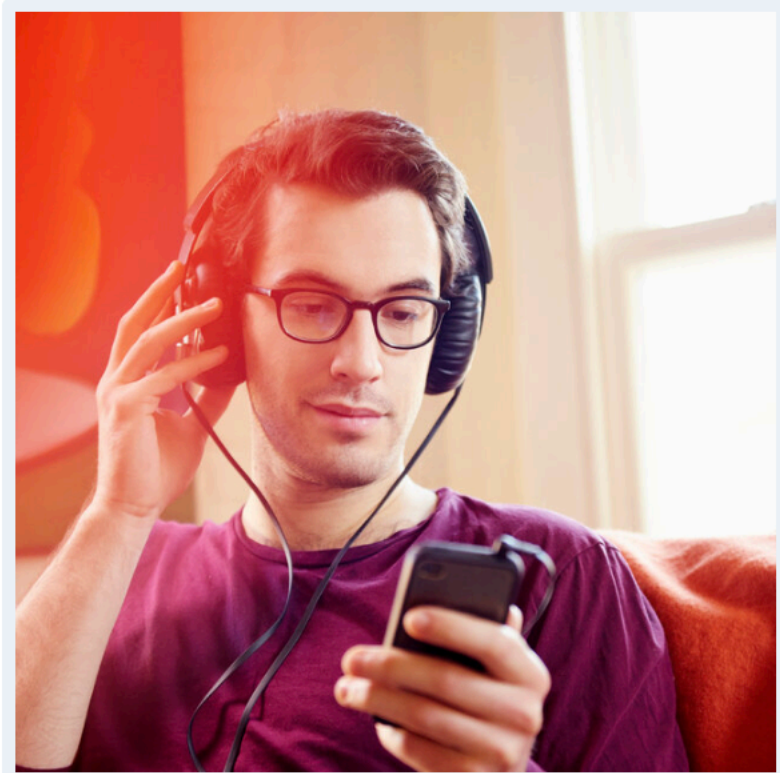
## *Can We No Longer Believe Anything We See?*



By Tiffany Hsu and Steven Lee Myers

April 8, 2023

Which image was created by artificial intelligence? Click on your guess



# Ethical Concerns

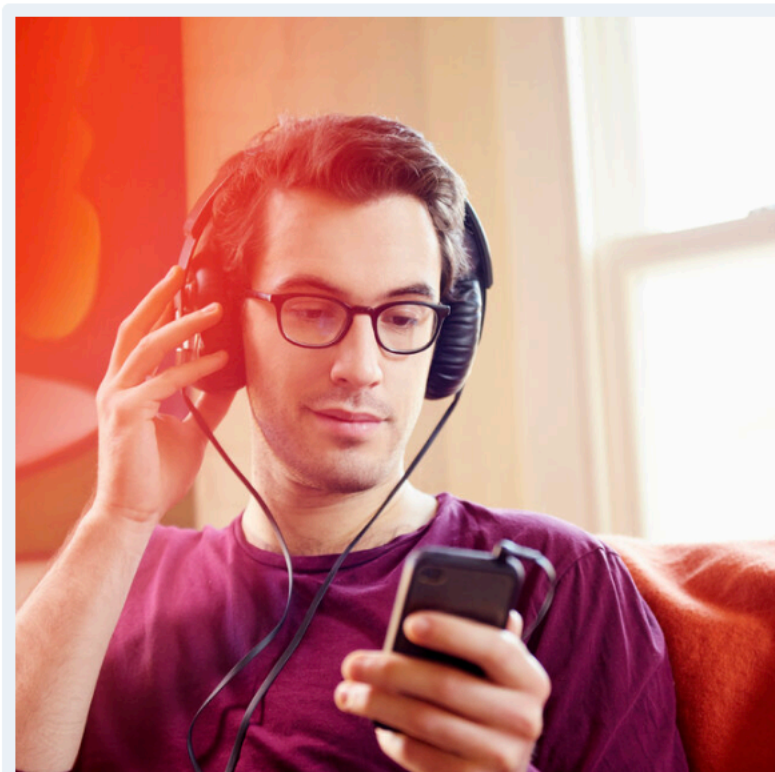
## *Can We No Longer Believe Anything We See?*



By Tiffany Hsu and Steven Lee Myers

April 8, 2023

Which image was created by artificial intelligence? Click on your guess



## *This Tool Could Protect Artists From A.I.-Generated Art That Steals Their Style*

Artists want to be able to post their work online without the fear “of feeding this monster” that could replace them.



# Ethical Concerns

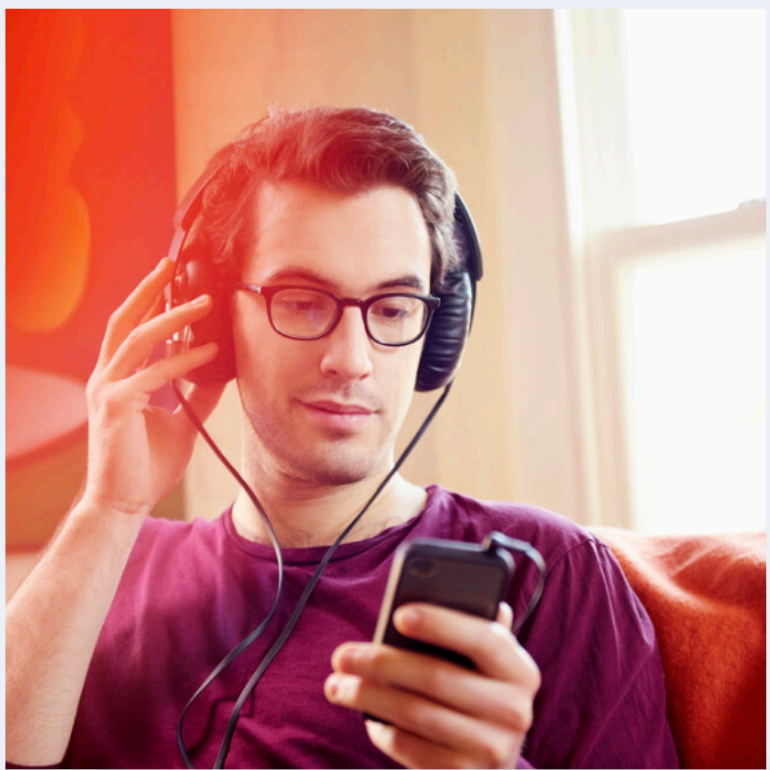
## *Can We No Longer Believe Anything We See?*



By [Tiffany Hsu](#) and [Steven Lee Myers](#)

April 8, 2023

Which image was created by artificial intelligence? Click on your guess



## *This Tool Could Protect Artists From A.I.-Generated Art That Steals Their Style*

Artists want to be able to post their work online without the fear “of feeding this monster” that could replace them.

## *An A.I. Hit of Fake ‘Drake’ and ‘The Weeknd’ Rattles the Music World*

A track like “Heart on My Sleeve,” which went viral before being taken down by streaming services this week, may be a novelty for now. But the legal and creative questions it raises are here to stay.

Give this article 215





# Ethical Concerns

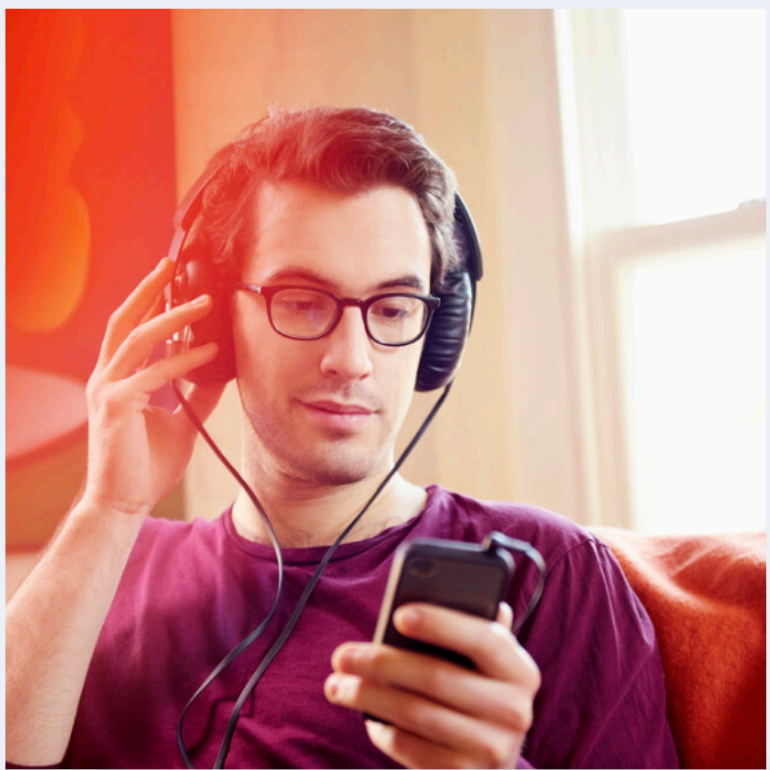
## *Can We No Longer Believe Anything We See?*



By [Tiffany Hsu](#) and [Steven Lee Myers](#)

April 8, 2023

Which image was created by artificial intelligence? Click on your guess



## *This Tool Could Protect Artists From A.I.-Generated Art That Steals Their Style*

Artists want to be able to post their work online without the fear “of feeding this monster” that could replace them.

## *An A.I. Hit of Fake ‘Drake’ and ‘The Weeknd’ Rattles the Music World*

A track like “Heart on My Sleeve,” which went viral before being taken down by streaming services this week, may be a novelty for now. But the legal and creative questions it raises are here to stay.

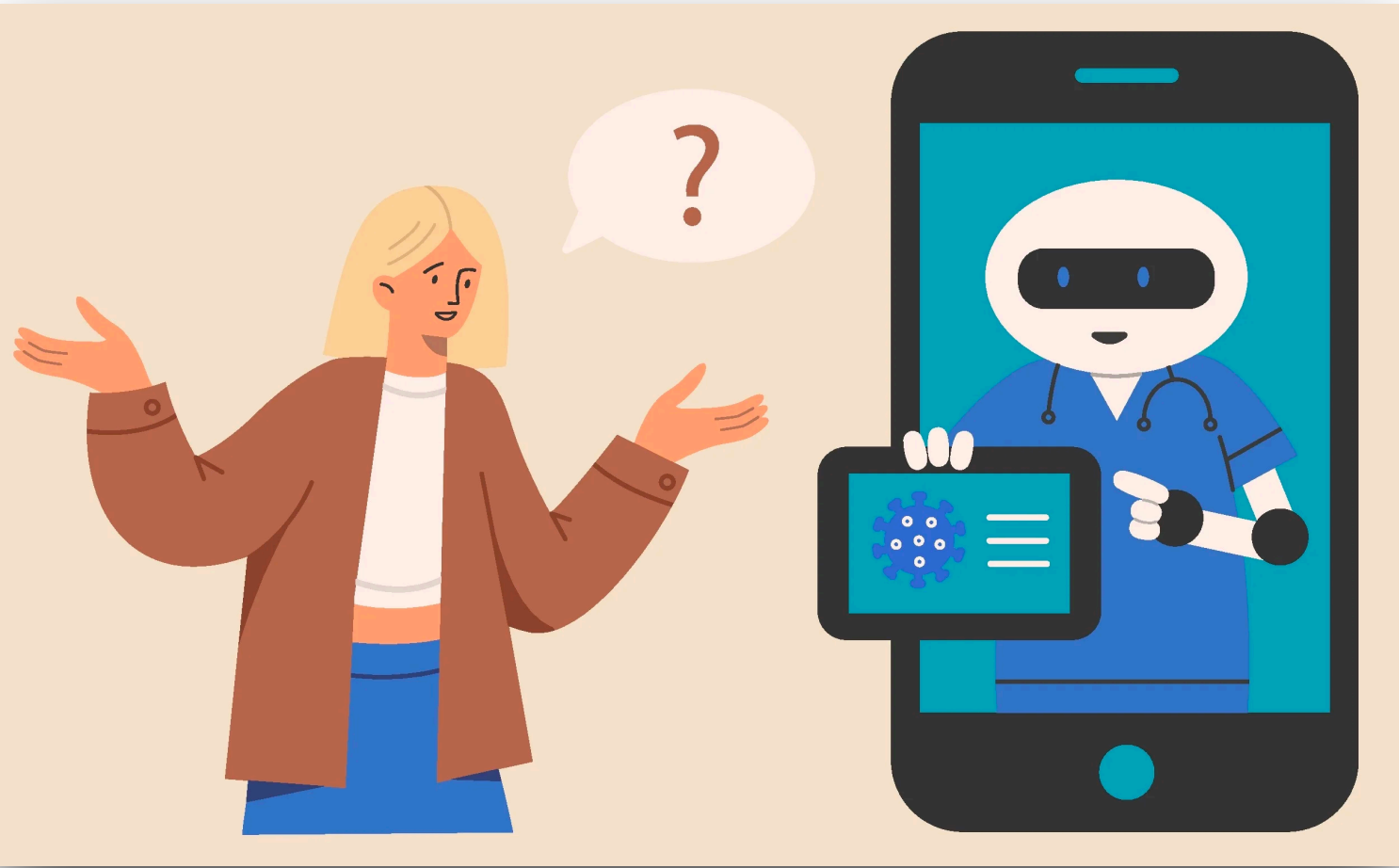
Give this article 215



Mar 8, 2023 - Technology

## Chatbot therapy, despite cautions, finds enthusiasts

Peter Allen Clark





# Ethical Concerns

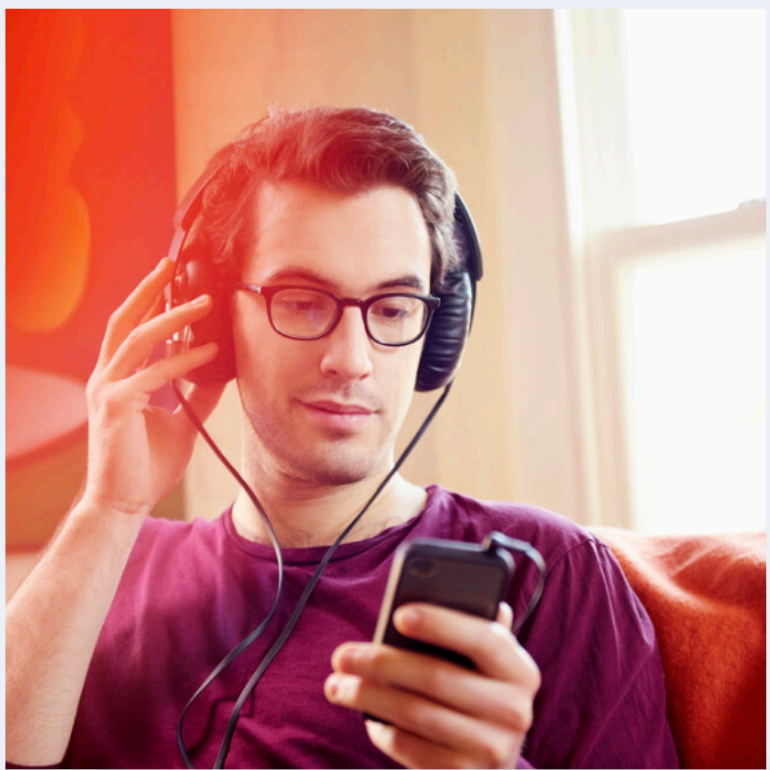
*Can We No Longer Believe Anything We See?*



By [Tiffany Hsu](#) and [Steven Lee Myers](#)

April 8, 2023

Which image was created by artificial intelligence? Click on your guess



*This Tool Could Protect Artists From A.I.-Generated Art That Steals Their Style*

Artists want to be able to post their work online without the fear “of feeding this monster” that could replace them.

## *An A.I. Hit of Fake ‘Drake’ and ‘The Weeknd’ Rattles the Music World*

A track like “Heart on My Sleeve,” which went viral before being taken down by streaming services this week, may be a novelty for now. But the legal and creative questions it raises are here to stay.

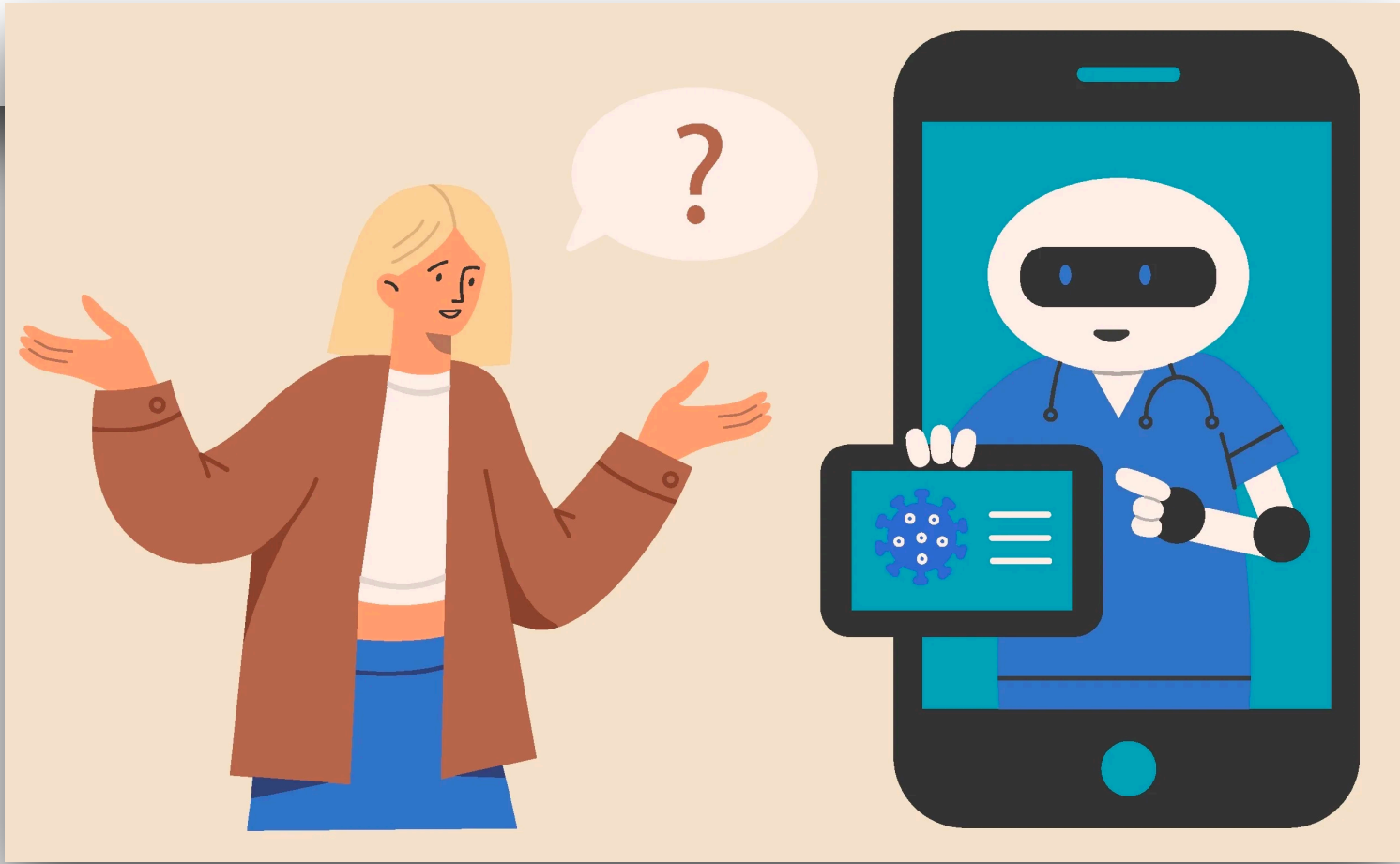
Give this article 215



Mar 8, 2023 - Technology

## Chatbot therapy, despite cautions, finds enthusiasts

Peter Allen Clark



## *A.I.-Generated Content Discovered on News Sites, Content Farms and Product Reviews*

The findings in two new reports raise fresh concerns over how artificial intelligence may transform the misinformation landscape online.



**Key Takeaway:**  
**LLMs are incredibly useful,**  
**but their answers cannot**  
**always be taken at face value**

# What's Next?



# Courses to take

- How to study this more?
- Math to learn: probability, linear algebra
- Machine learning or data science online courses
  - Andrew Ng's Coursera course: <https://www.coursera.org/learn/machine-learning>
  - Introduction to LLMs by Andrej Karpathy: [https://www.youtube.com/watch?v=zjkBMFhNj\\_g](https://www.youtube.com/watch?v=zjkBMFhNj_g)
  - Sentiment Analysis tutorial: <https://realpython.com/sentiment-analysis-python/>
- More programming or software engineering can help
  - Python
- Start exploring the HuggingFace platform, library and tutorials: <https://huggingface.co/>

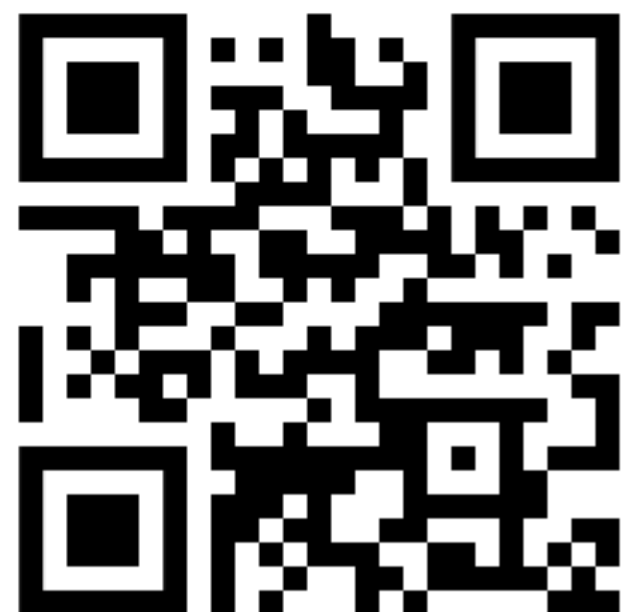
# Further Reading

- Understanding more about neural networks: Chris Olah, Jay Alammar
  - <https://colah.github.io/>
  - <https://jalamar.github.io/>
- Latest big language models:
  - OpenAI's GPT-series: <https://openai.com/index/hello-gpt-4o/>
  - Meta's Llama-series (open-source): <https://www.meta.ai/>
  - Google's Gemini: <https://gemini.google.com/>



# Thank you!

You can find these  
slides here:



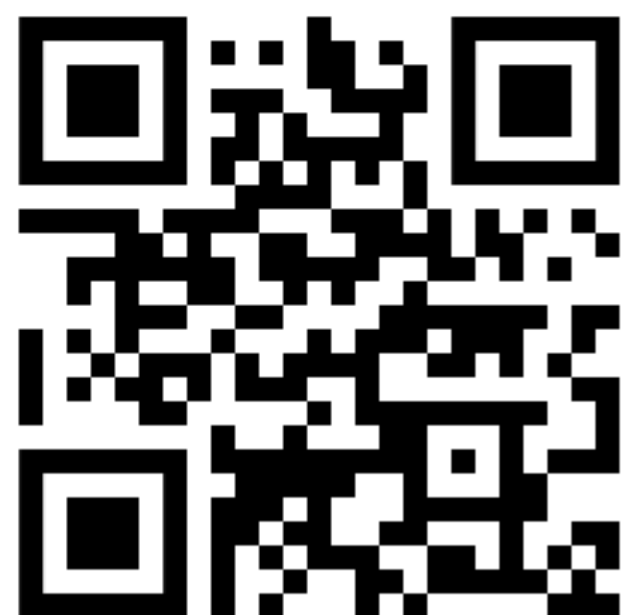
Check out our DILL lab  
and come talk to us!





# Thank you!

You can find these  
slides here:



Check out our DILL lab  
and come talk to us!

