

The Evolutionary Journey in NLP

Lena Voita

Lectures as blogs and lots of additional materials are here:
https://lena-voita.github.io/nlp_course

NLP Course For You 

About a year ago, right before the previous course

About a year ago, right before the previous course

Back then, almost none of the attendees used neural networks or heard much about them.

About a year ago, right before the previous course

Back then, almost none of the attendees used neural networks or heard much about them.

Meanwhile in media...



June, 11

The Google engineer who thinks the company's AI has come to life

AI ethicists warned Google not to impersonate humans. Now one of Google's own thinks there's a ghost in the machine.

About a year ago, right before the previous course

Back then, almost none of the attendees used neural networks or heard much about them.

Meanwhile in media...

The Washington Post logo and tagline "Democracy Dies in Darkness". Navigation menu: TECH, Help Desk, Artificial Intelligence, Internet Culture, Space, Tech Policy. A yellow banner at the bottom left indicates the article is "more than 1 year ago".

June, 11

The Google engineer who thinks the company's AI has come to life

AI ethicists warned Google not to impersonate humans. Now one of Google's own thinks there's a ghost in the machine.

June, 13



The Guardian

<https://www.theguardian.com> › technology › jun › go... ::

[Google engineer put on leave after saying AI chatbot has ...](#)

13 Jun 2022 — Google said it **suspended Lemoine** for breaching confidentiality policies by publishing the conversations with LaMDA online, and said in a ...

Missing: [Im](#) | Show results with: [Im](#)

About a year ago, right before the previous course

Back then, almost none of the attendees used neural networks or heard much about them.

Meanwhile in media...

June, 11

 The Washington Post
Democracy Dies in Darkness

TECH Help Desk Artificial Intelligence Internet Culture Space Tech Policy

⌚ This article was published more than 1 year ago

The Google engineer who thinks the company's AI has come to life

AI ethicists warned Google not to impersonate humans. Now one of Google's own thinks there's a ghost in the machine.

June, 13

 The Guardian
<https://www.theguardian.com> › technology › jun › go... ›

[Google engineer put on leave after saying AI chatbot has ...](#)

13 Jun 2022 — Google said it **suspended Lemoine** for breaching confidentiality policies by publishing the conversations with LaMDA online, and said in a ...

Missing: [Im](#) | Show results with: [Im](#)

July, 11

Lena's lecture at SICSS-Oxford 2022

About a year ago, right before the previous course

Back then, almost none of the attendees used neural networks or heard much about them.

Meanwhile in media...

June, 11

 The Google engineer who thinks the company's AI has come to life

AI ethicists warned Google not to impersonate humans. Now one of Google's own thinks there's a ghost in the machine.

TECH Help Desk Artificial Intelligence Internet Culture Space Tech Policy

⌚ This article was published more than 1 year ago

June, 13

 The Guardian
<https://www.theguardian.com> › technology › jun › go... :

[Google engineer put on leave after saying AI chatbot has ...](#)

13 Jun 2022 — Google said it **suspended Lemoine** for breaching confidentiality policies by publishing the conversations with LaMDA online, and said in a ...

Missing: [Im](#) | Show results with: [Im](#)

July, 11

Lena's lecture at SICSS-Oxford 2022

July, 23

 The New York Times
<https://www.nytimes.com> › 2022/07/23 › technology › g... :

[Google Fires Engineer Who Claims Its A.I. Is Conscious](#)

23 Jul 2022 — The engineer, **Blake Lemoine**, contends that the company's language model has a soul. The company denies that and says he violated its security ...

Missing: [stop Im training](#)

Present Days

How many of you tried using ChatGPT?

Present Days

How many of you tried using ChatGPT?

 The New York Times
<https://www.nytimes.com> › 2023/03/31 › technology › c... ::

[ChatGPT Is Banned in Italy Over Privacy Concerns](#)

31 Mar 2023 — The artificial intelligence tool **ChatGPT** was temporarily **banned in Italy** on Friday, the first known instance of the chatbot being blocked by ...

Present Days

How many of you tried using ChatGPT?

 The New York Times
<https://www.nytimes.com> › 2023/03/31 › technology › c... ::

[ChatGPT Is Banned in Italy Over Privacy Concerns](#)

31 Mar 2023 — The artificial intelligence tool **ChatGPT** was temporarily **banned in Italy** on Friday, the first known instance of the chatbot being blocked by ...

 BBC
<https://www.bbc.co.uk> › news › technology-65431914 ::

[ChatGPT accessible again in Italy](#)

28 Apr 2023 — Access to the **ChatGPT** chatbot has been restored in **Italy**. It was **banned** by the **Italian** data-protection authority at the start of April over ...

Present Days

How many of you tried using ChatGPT?

The New York Times
<https://www.nytimes.com/2023/03/31/technology/chatgpt-banned-italy.html> ::

ChatGPT Is Banned in Italy Over Privacy Concerns

31 Mar 2023 — The artificial intelligence tool ChatGPT was temporarily banned in Italy on Friday, the first known instance of the chatbot being blocked by ...

The existential risks posed by artificial intelligence (AI) are now widely recognized. After hundreds of industry and science leaders warned that “mitigating the risk of extinction from AI should be a global priority alongside other societal-scale risks such as pandemics and nuclear war,” the U.N. 20 Jul 2023



BBC
<https://www.bbc.co.uk/news/technology-65431914> ::

ChatGPT accessible again in Italy

28 Apr 2023 — Access to the ChatGPT chatbot has been restored in Italy. It was banned by the Italian data-protection authority at the start of April over ...

Time Magazine
<https://time.com/5983333/time-ai-pause-extinction/> ::

An AI Pause Is Humanity's Best Bet For Preventing Extinction

Present Days

How many of you tried using ChatGPT?

 The New York Times
<https://www.nytimes.com/2023/03/31/technology/chatgpt-banned-italy.html> ::

[ChatGPT Is Banned in Italy Over Privacy Concerns](#)

31 Mar 2023 — The artificial intelligence tool **ChatGPT** was temporarily **banned in Italy** on Friday, the first known instance of the chatbot being blocked by ...

The existential risks posed by artificial intelligence (AI) are now widely recognized. After hundreds of industry and science leaders warned that “[mitigating the risk of extinction from AI should be a global priority alongside other societal-scale risks such as pandemics and nuclear war](#),” the U.N. 20 Jul 2023



 BBC
<https://www.bbc.co.uk/news/technology-65431914> ::

[ChatGPT accessible again in Italy](#)

28 Apr 2023 — Access to the **ChatGPT** chatbot has been restored in **Italy**. It was **banned** by the Italian data-protection authority at the start of April over ...

 Time Magazine
<https://time.com/5983375/time-intelligence-extinction/> ::

[An AI Pause Is Humanity's Best Bet For Preventing Extinction](#)

 BBC
<https://www.bbc.co.uk/news/science-environment-65431914> ::

[Artificial intelligence could lead to extinction, experts warn](#)

30 May 2023 — **Artificial intelligence could lead to the extinction of humanity**, experts - including the heads of OpenAI and Google Deepmind - have warned.

Present Days

How many of you tried using ChatGPT?

 The New York Times
<https://www.nytimes.com/2023/03/31/technology/chatgpt-banned-italy.html> ::

[ChatGPT Is Banned in Italy Over Privacy Concerns](#)

31 Mar 2023 — The artificial intelligence tool **ChatGPT** was temporarily **banned in Italy** on Friday, the first known instance of the chatbot being blocked by ...

The existential risks posed by artificial intelligence (AI) are now widely recognized. After hundreds of industry and science leaders warned that “[mitigating the risk of extinction from AI should be a global priority alongside other societal-scale risks such as pandemics and nuclear war](#),” the U.N. 20 Jul 2023



 BBC
<https://www.bbc.co.uk/news/technology-65431914> ::

[ChatGPT accessible again in Italy](#)

28 Apr 2023 — Access to the **ChatGPT** chatbot has been restored in **Italy**. It was **banned** by the Italian data-protection authority at the start of April over ...

 Time Magazine
<https://time.com/5973333/time-intelligence-extinction/> ::

[An AI Pause Is Humanity's Best Bet For Preventing Extinction](#)

 BBC
<https://www.bbc.co.uk/news/science-environment-65431914> ::

[Artificial intelligence could lead to extinction, experts warn](#)

30 May 2023 — **Artificial intelligence could lead to the extinction of humanity**, experts - including the heads of OpenAI and Google Deepmind - have warned.

 The New York Times
<https://www.nytimes.com/2023/05/30/technology/ai-threat-warning.html> ::

[AI Poses 'Risk of Extinction,' Industry Leaders Warn](#)

30 May 2023 — A group of industry leaders warned on Tuesday that the **artificial intelligence** technology they were building might one day pose an ...

The Evolutionary Journey in NLP

Lena Voita

Lectures as blogs and lots of additional materials are here:
https://lena-voita.github.io/nlp_course

NLP Course For You 

The Evolutionary Journey in NLP

Since the Beginning of Time

Lena Voita

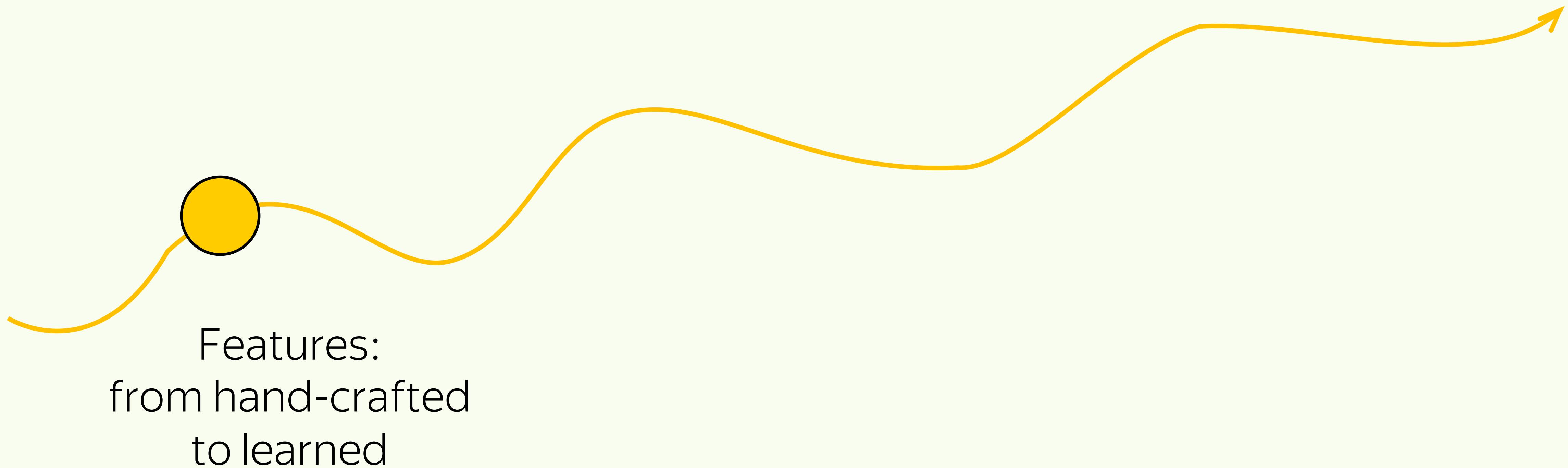
Lectures as blogs and lots of additional materials are here:
https://lena-voita.github.io/nlp_course

NLP Course For You 

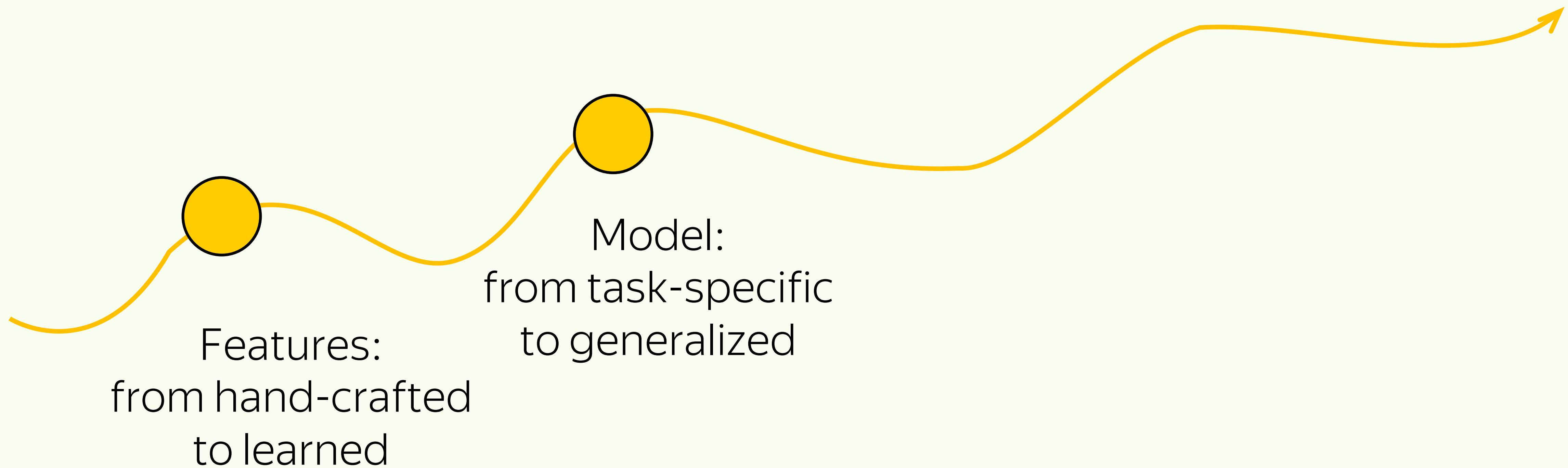
The Evolutionary Journey in NLP



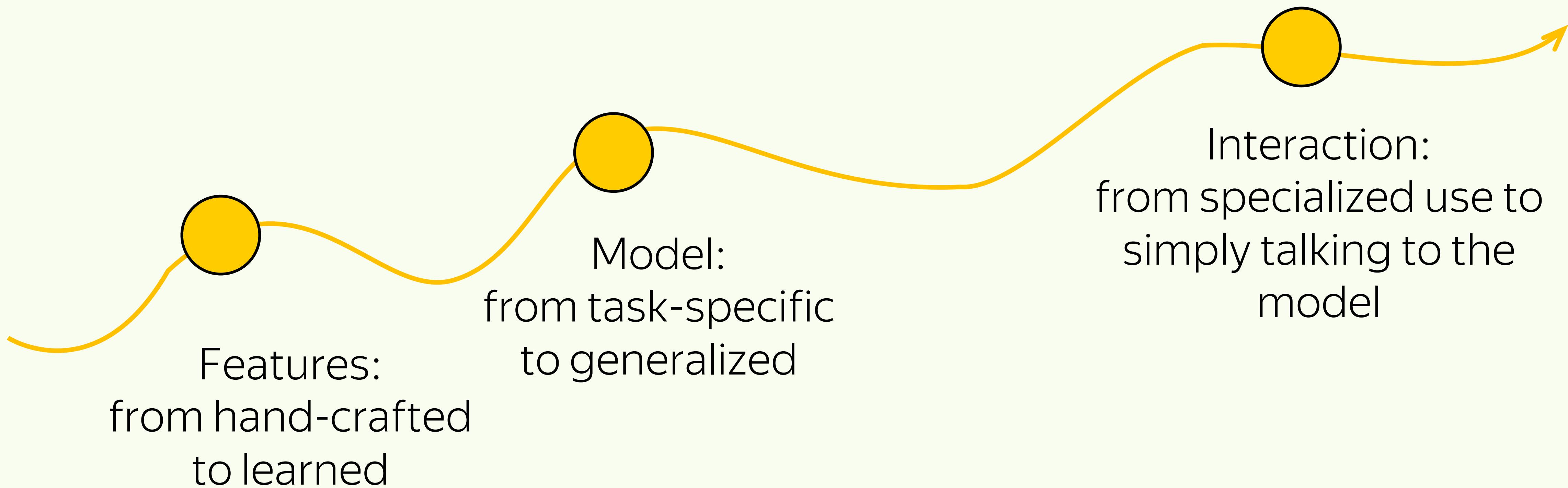
The Evolutionary Journey in NLP



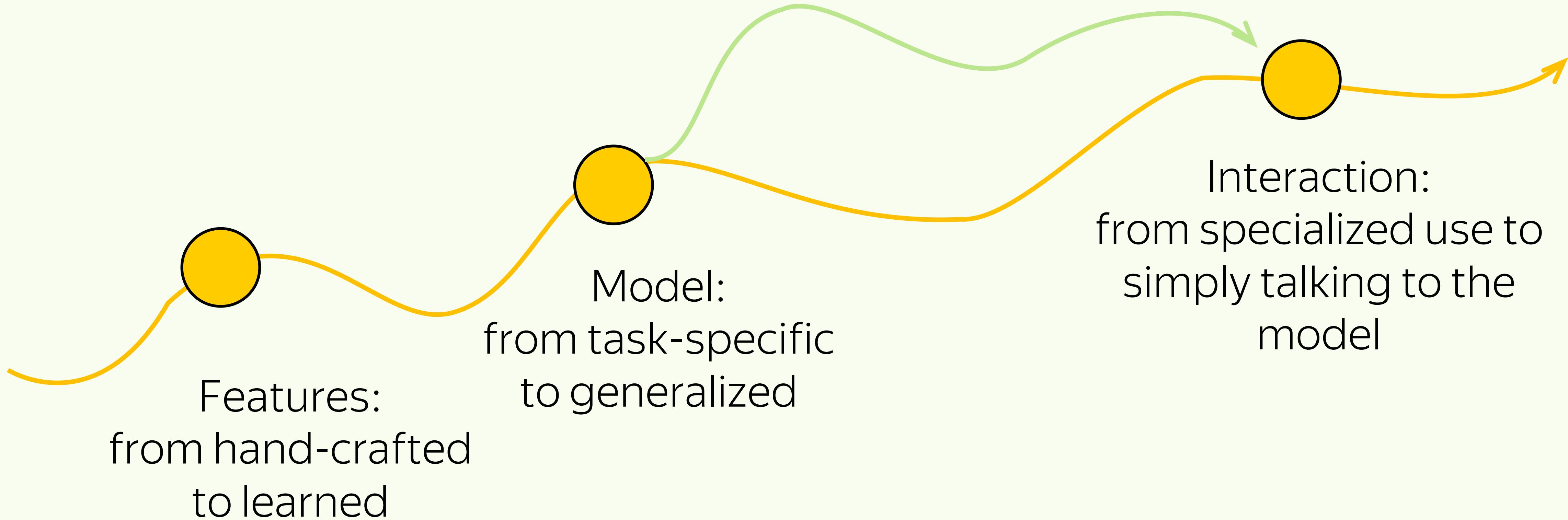
The Evolutionary Journey in NLP



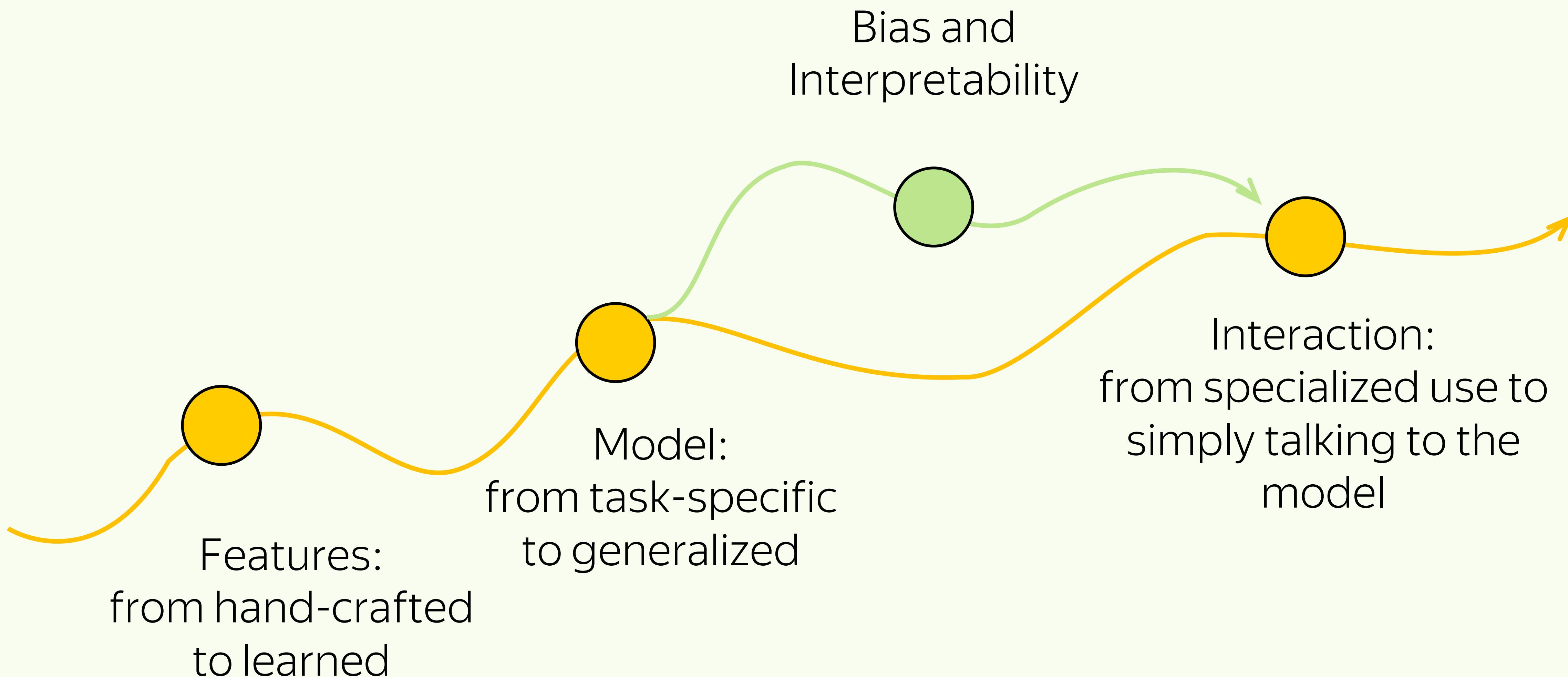
The Evolutionary Journey in NLP



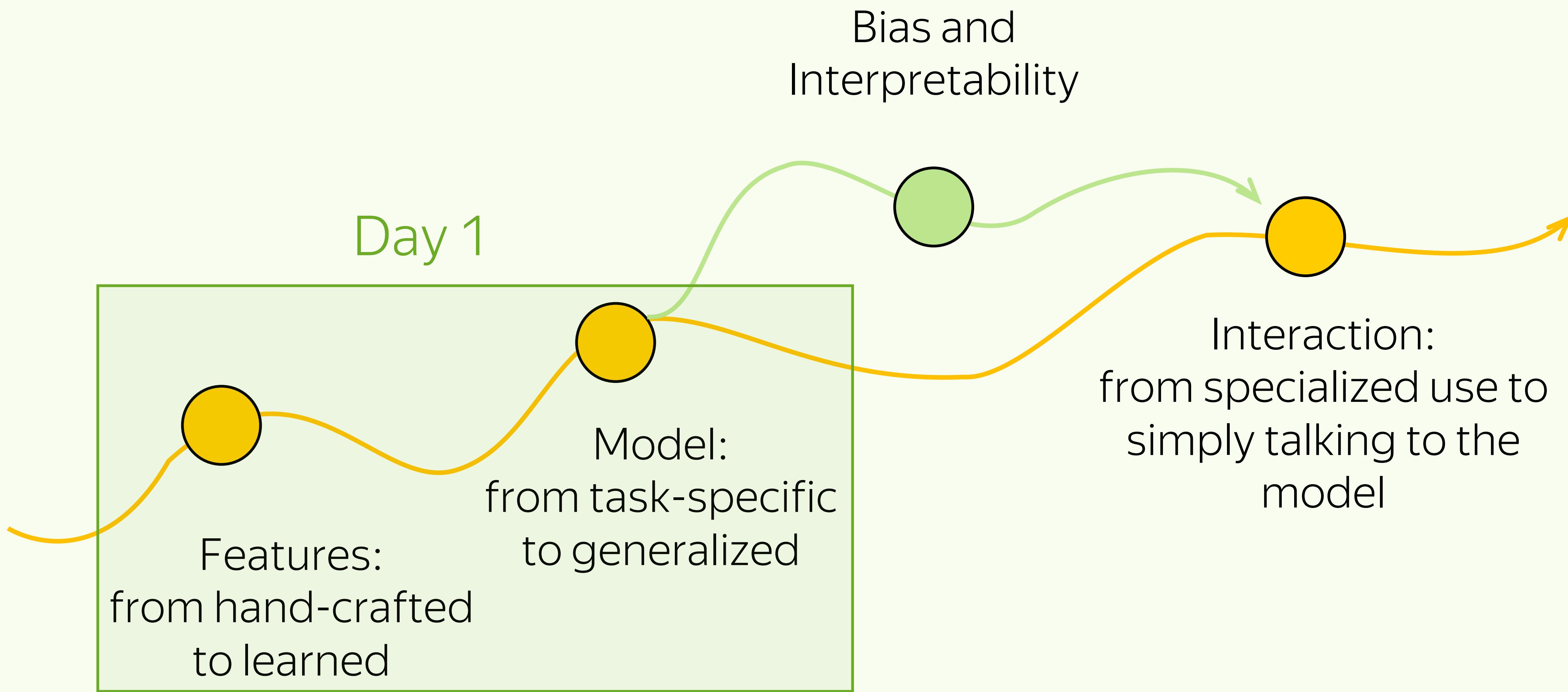
The Evolutionary Journey in NLP



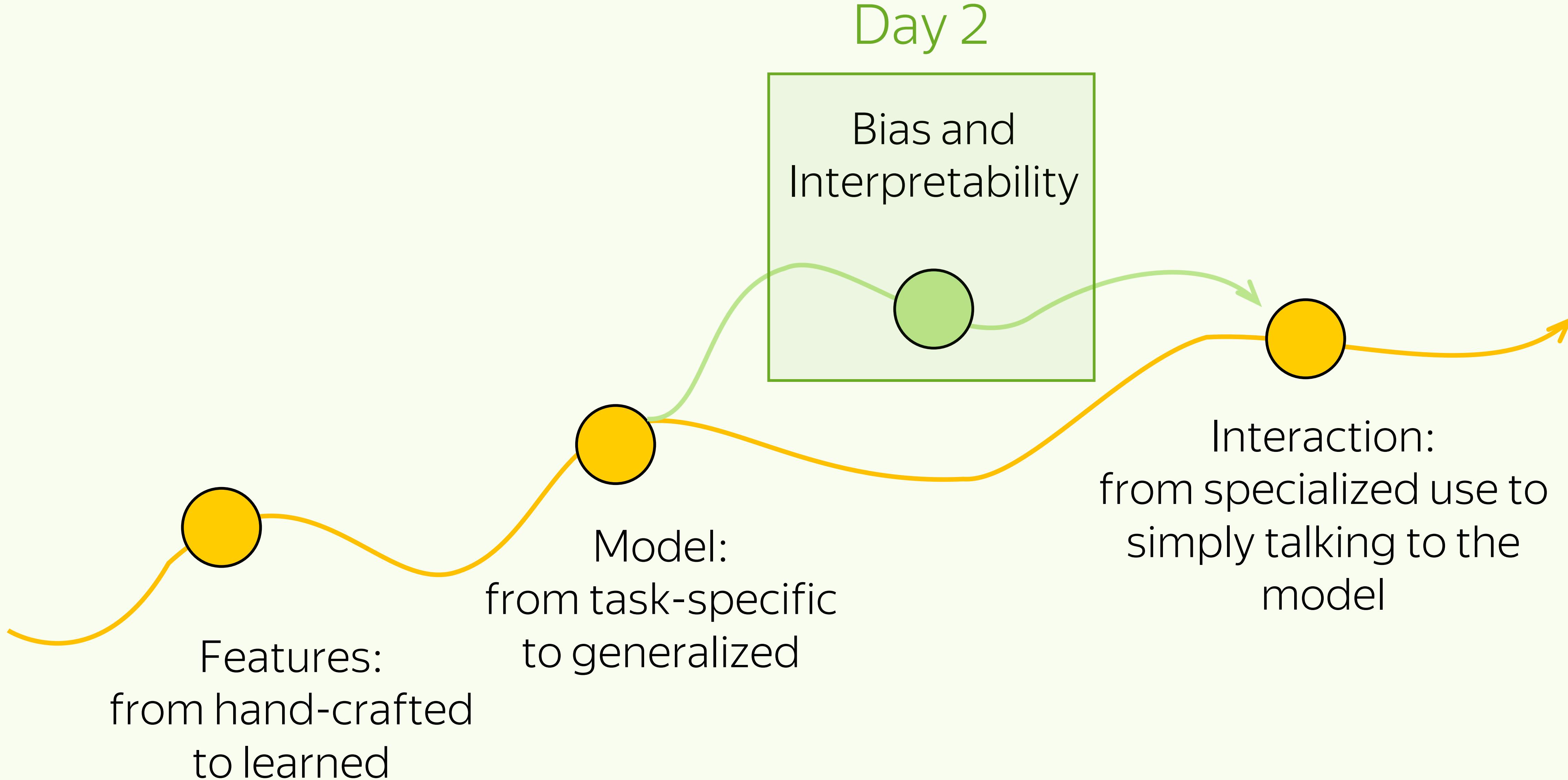
The Evolutionary Journey in NLP



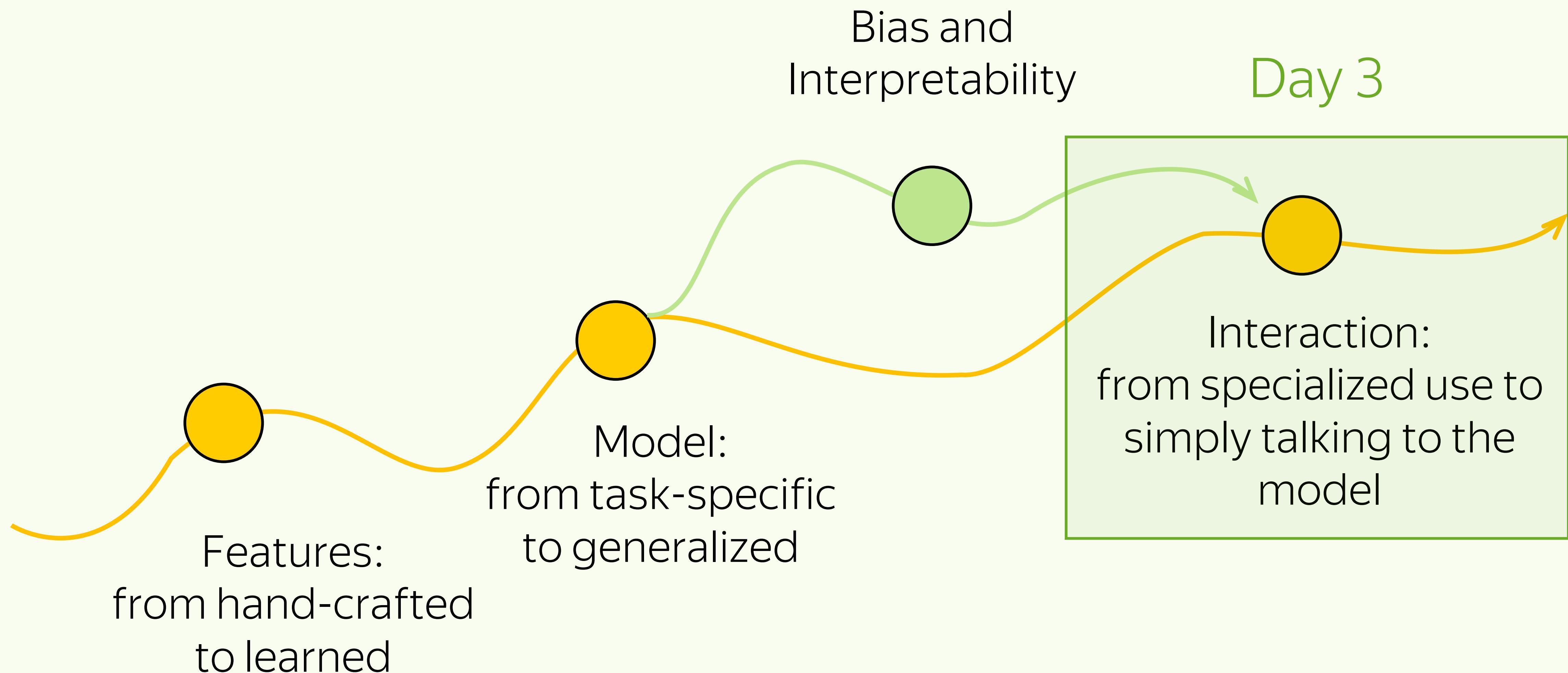
The Evolutionary Journey in NLP



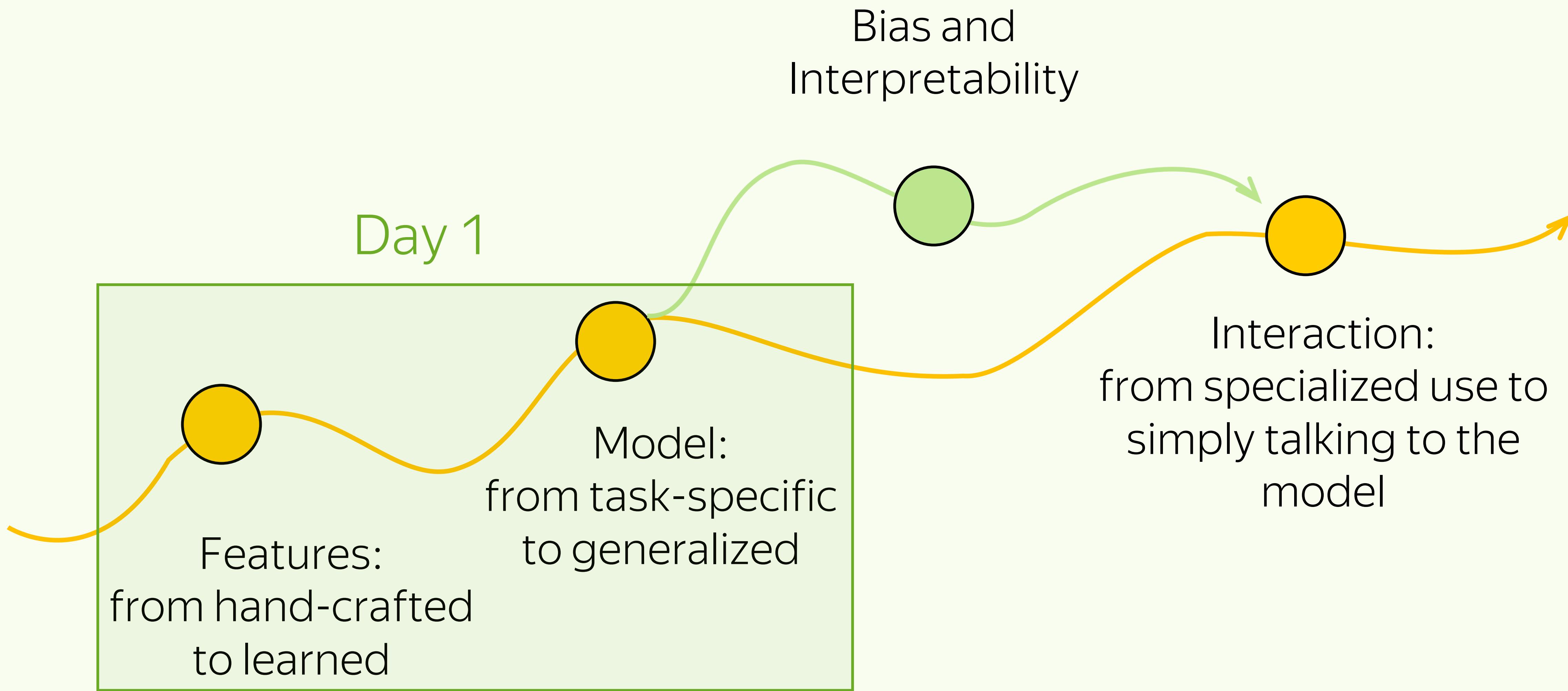
The Evolutionary Journey in NLP



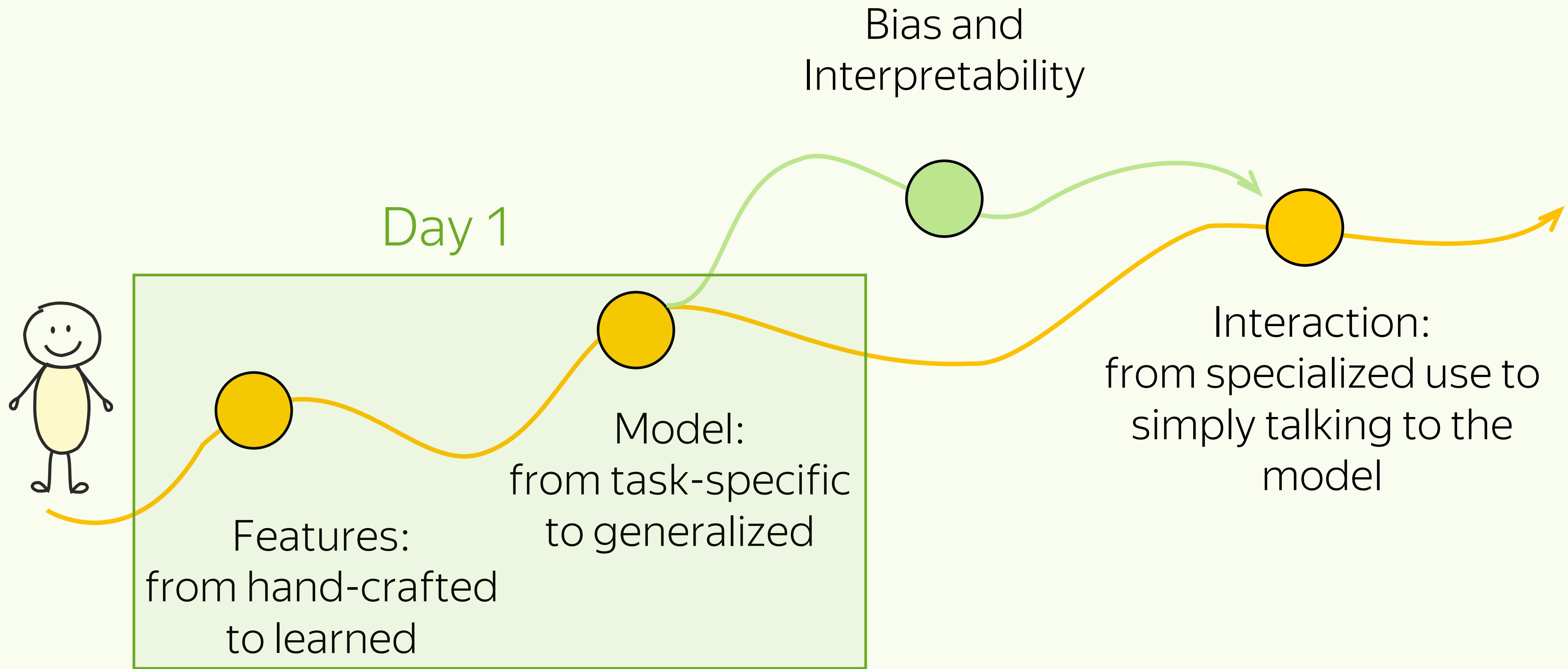
The Evolutionary Journey in NLP



The Evolutionary Journey in NLP



The Evolutionary Journey in NLP



What is going to happen:

- Classification
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff

What is going to happen:

- Classification
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff



Features:
from hand-crafted
to learned

What is going to happen:

- Classification
 - Language Modeling
-
- Transfer Learning
 - A Note on Crazy Stuff



Features:
from hand-crafted
to learned



Model:
from task-specific
to generalized

What is going to happen:

- Classification →
 - General View
 - Neural approach
 - Convolutional Networks
 - Recurrent Networks
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff

Features:
from hand-crafted
to learned

Model:
from task-specific
to generalized

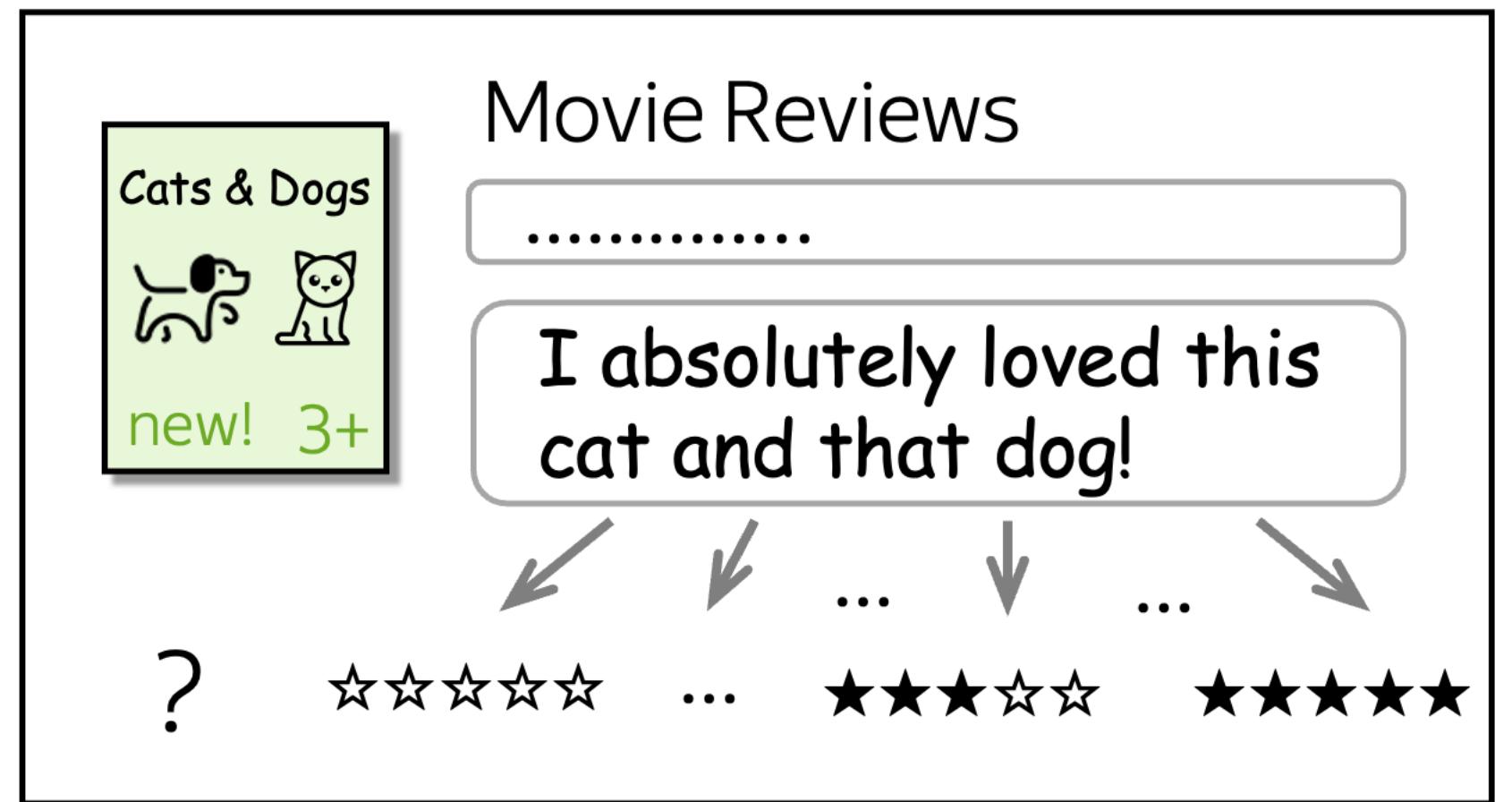
What is going to happen:

- Classification →
 - General View
 - Neural approach
 - Convolutional Networks
 - Recurrent Networks
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff

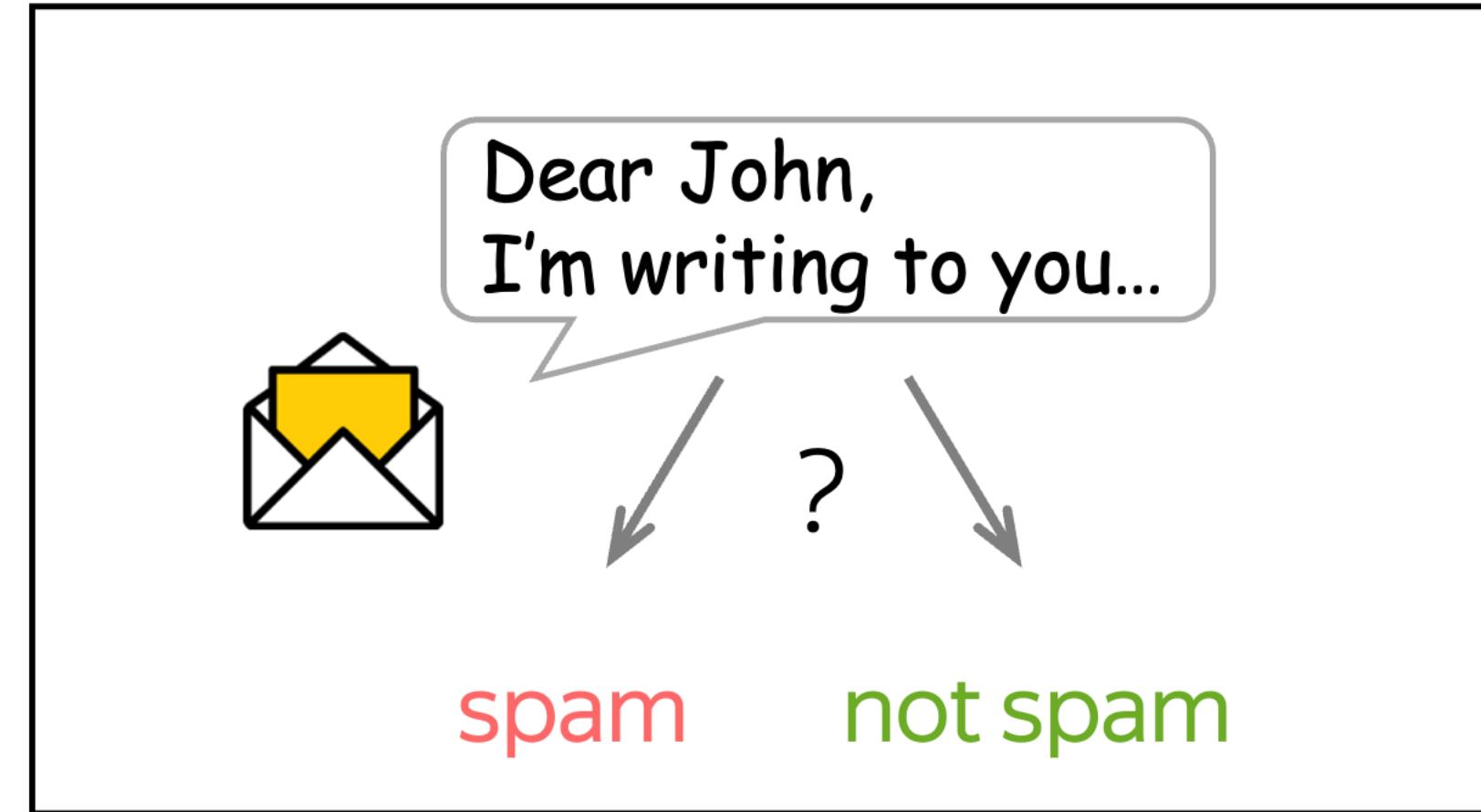
Features:
from hand-crafted
to learned

Model:
from task-specific
to generalized

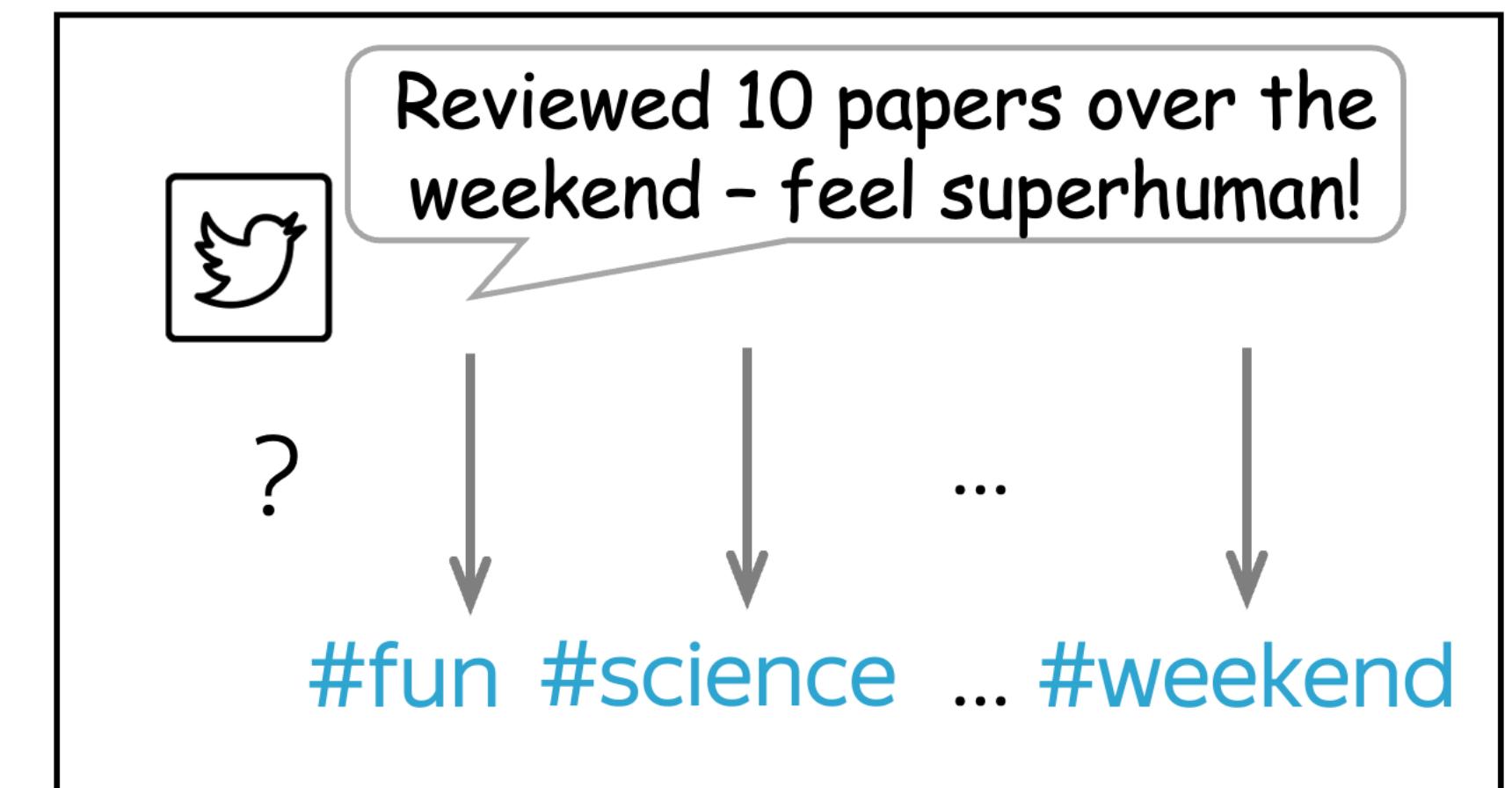
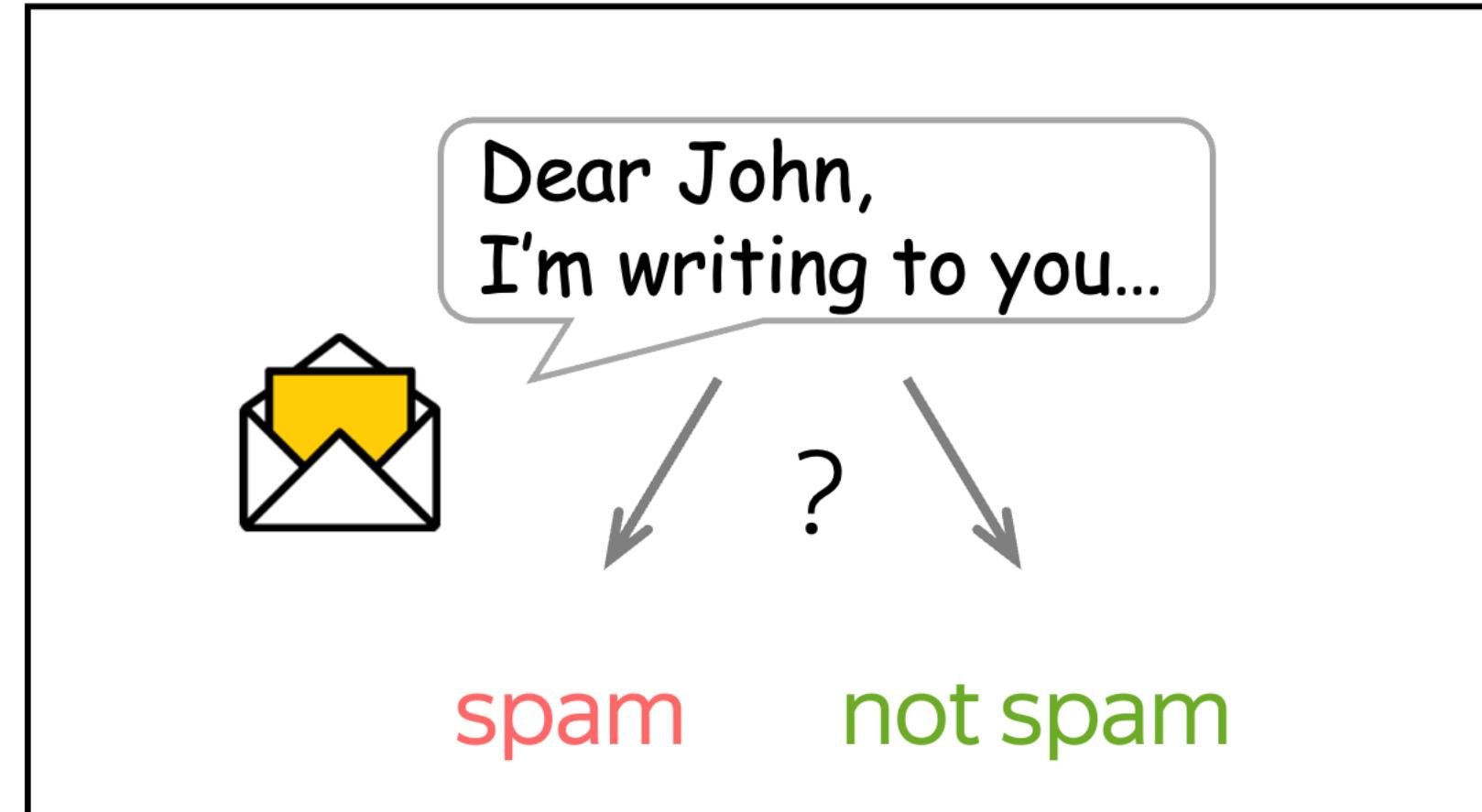
Text Classification



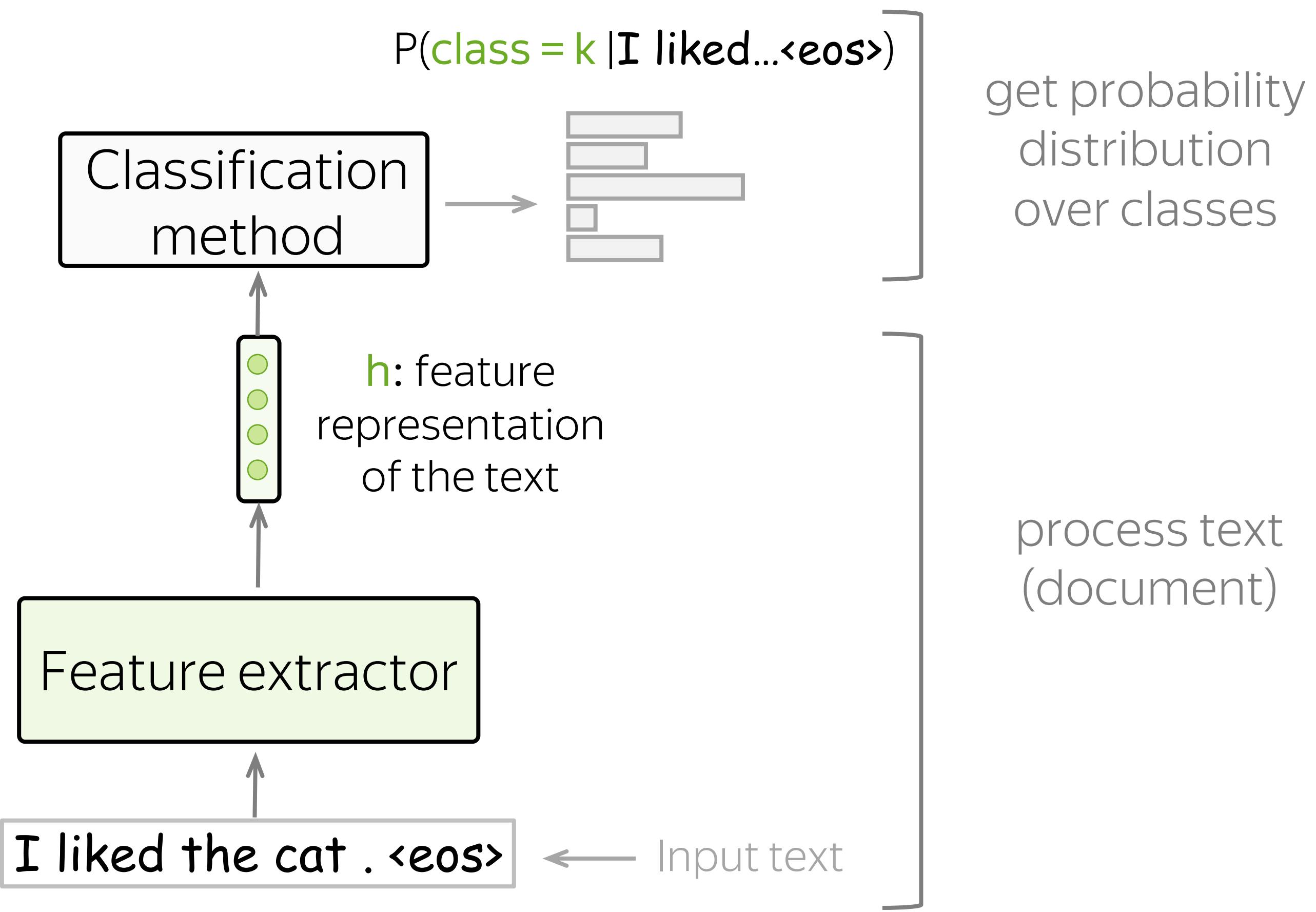
Text Classification



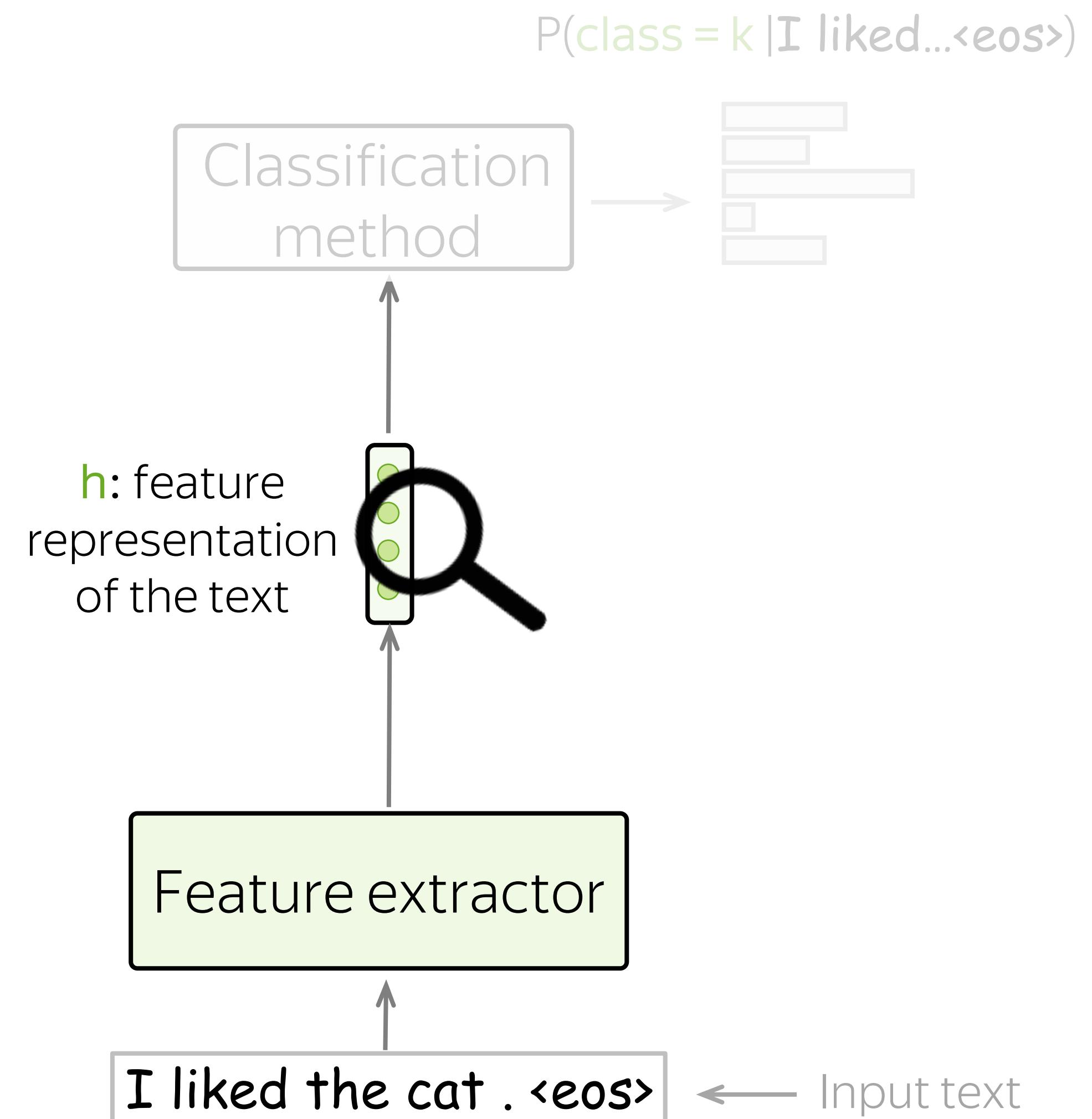
Text Classification



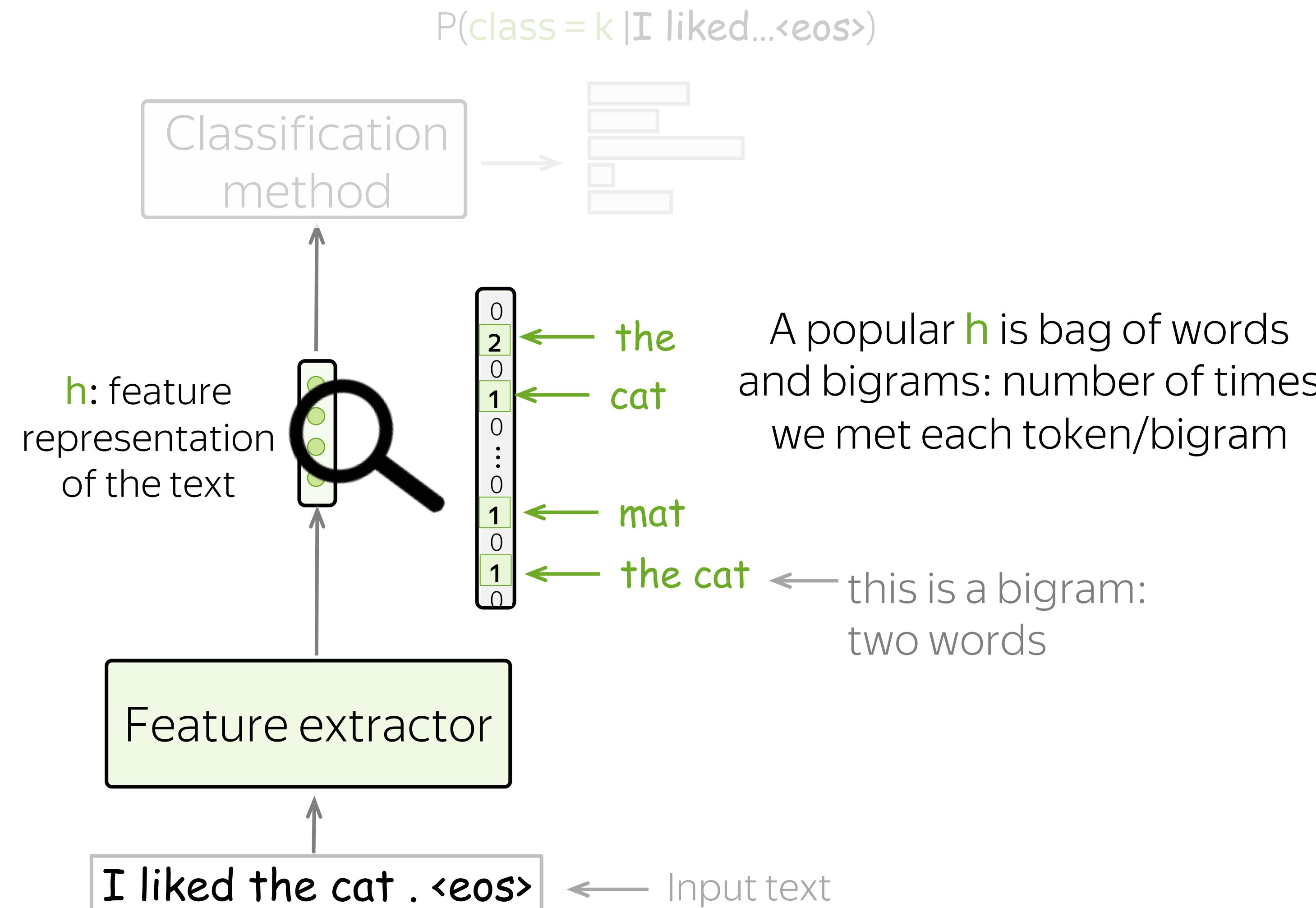
General View: Extract Features and Classify



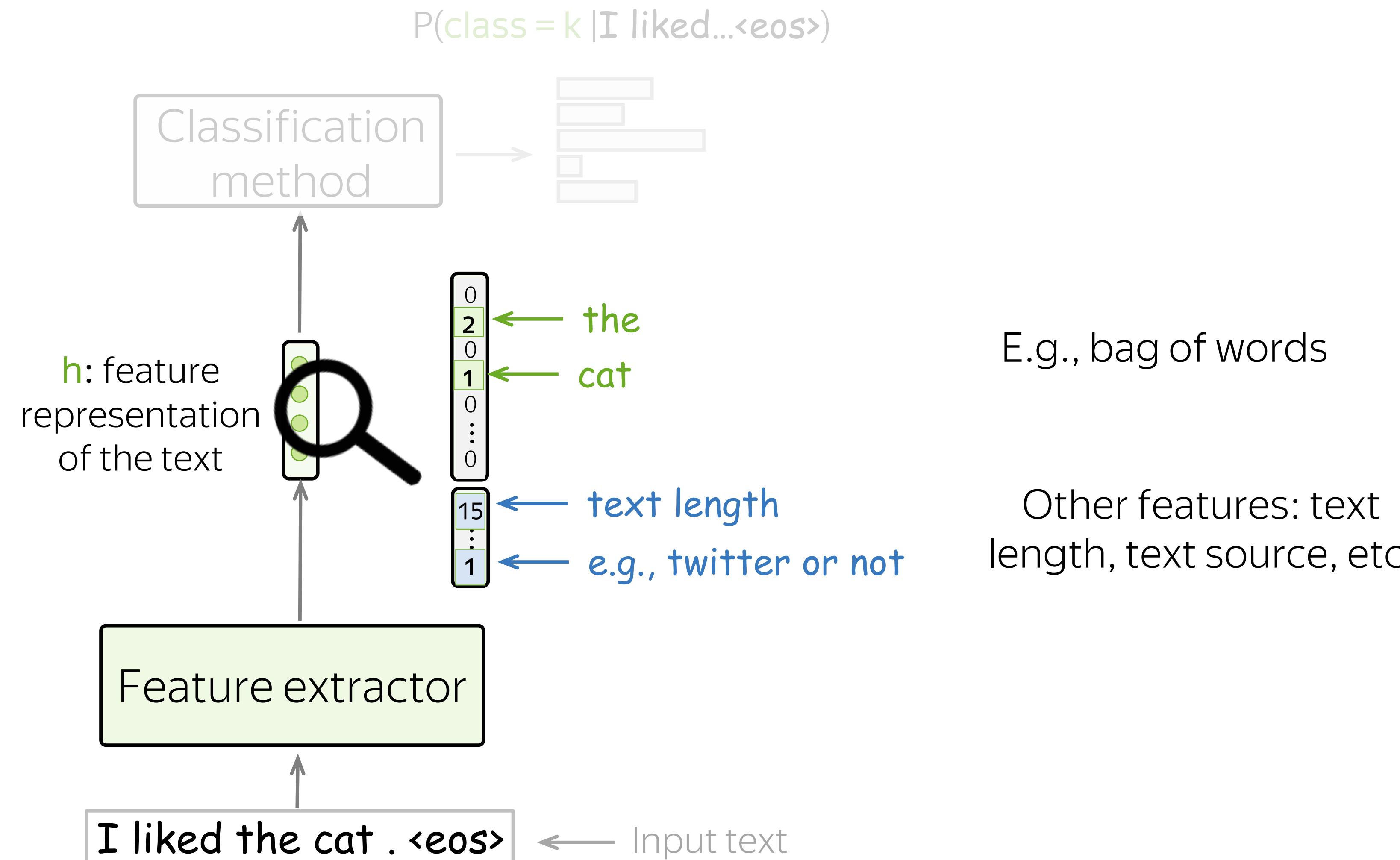
General View: Extract Features and Classify



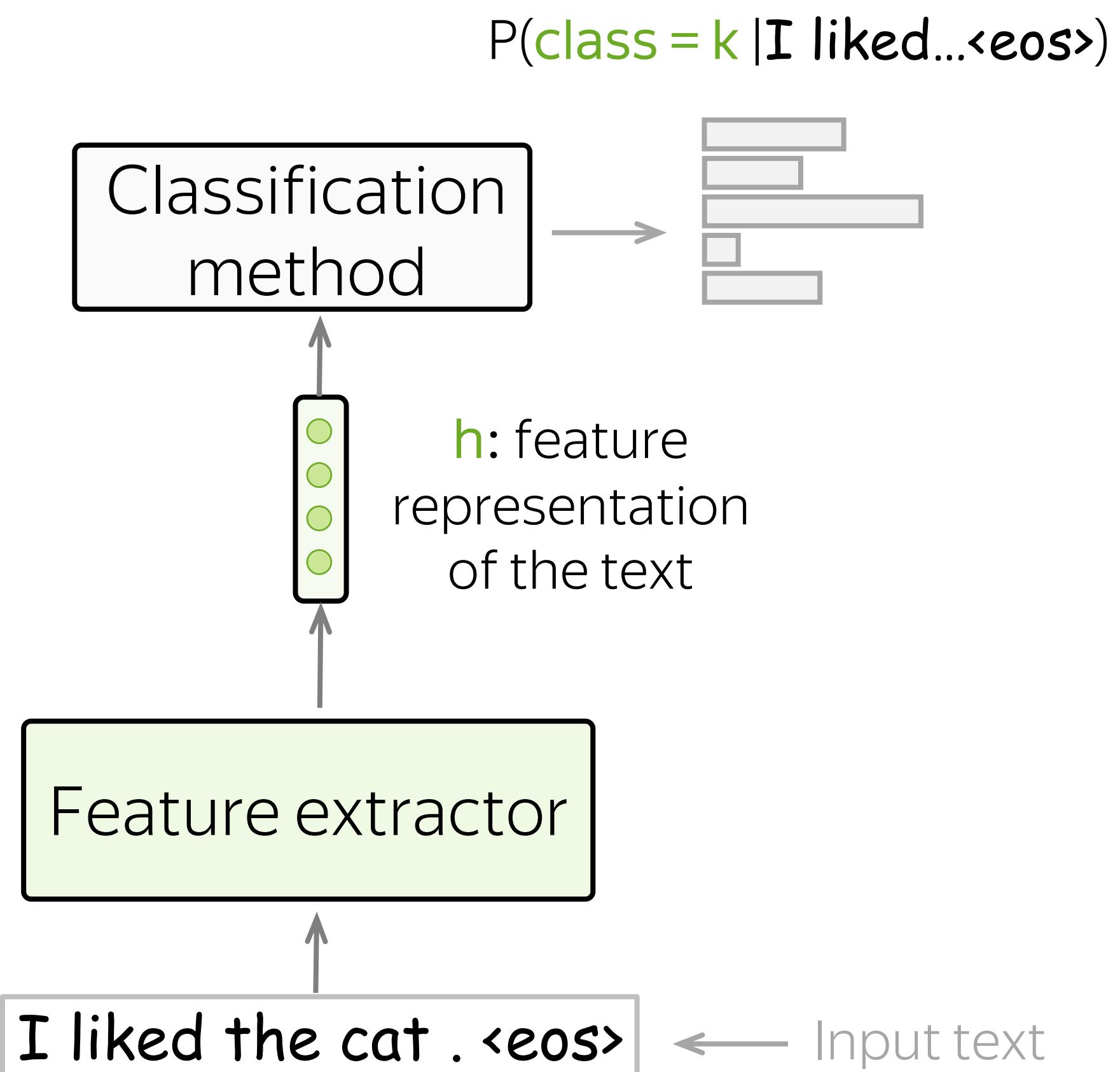
General View: Extract Features and Classify



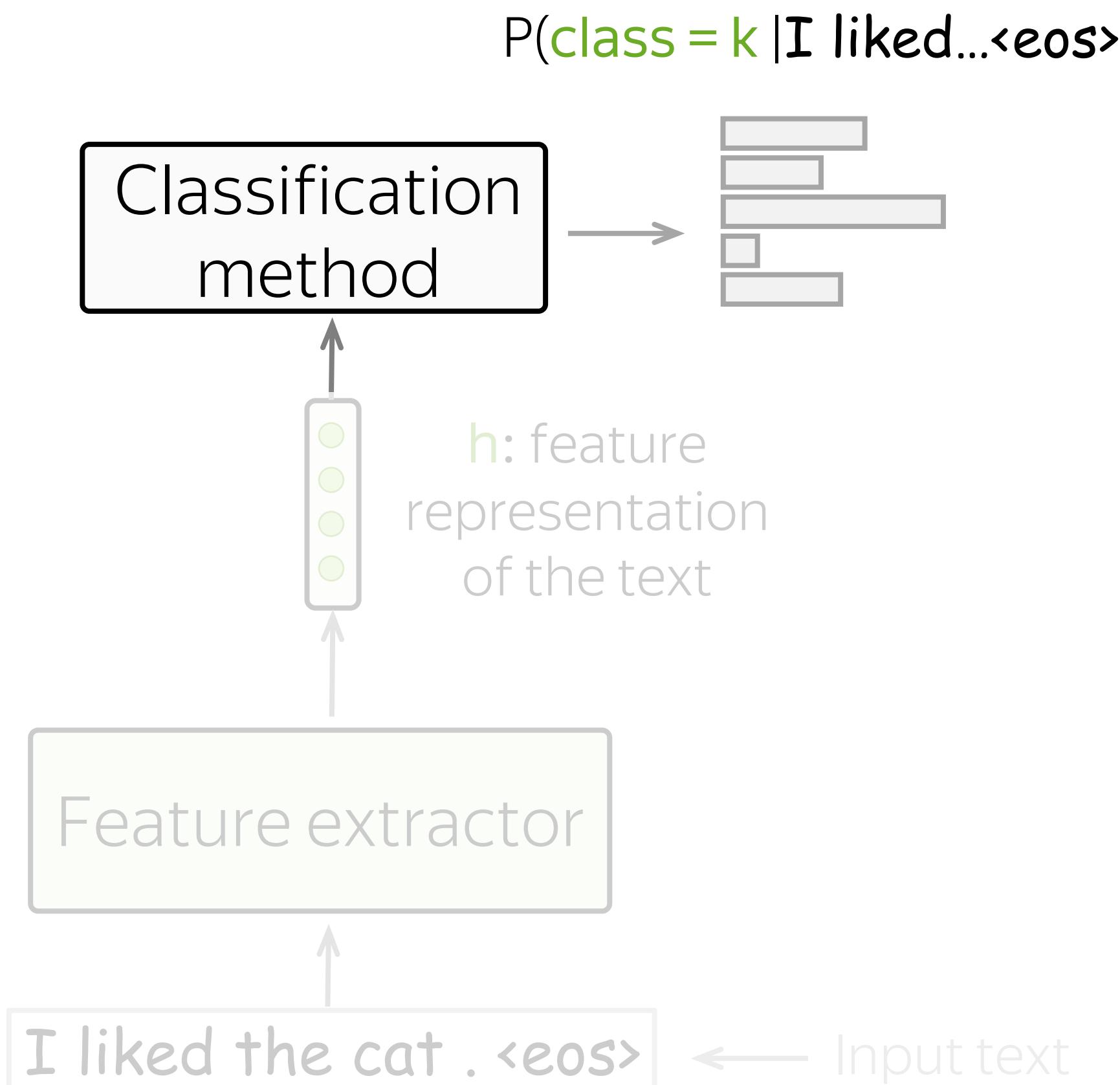
General View: Extract Features and Classify



General View: Extract Features and Classify



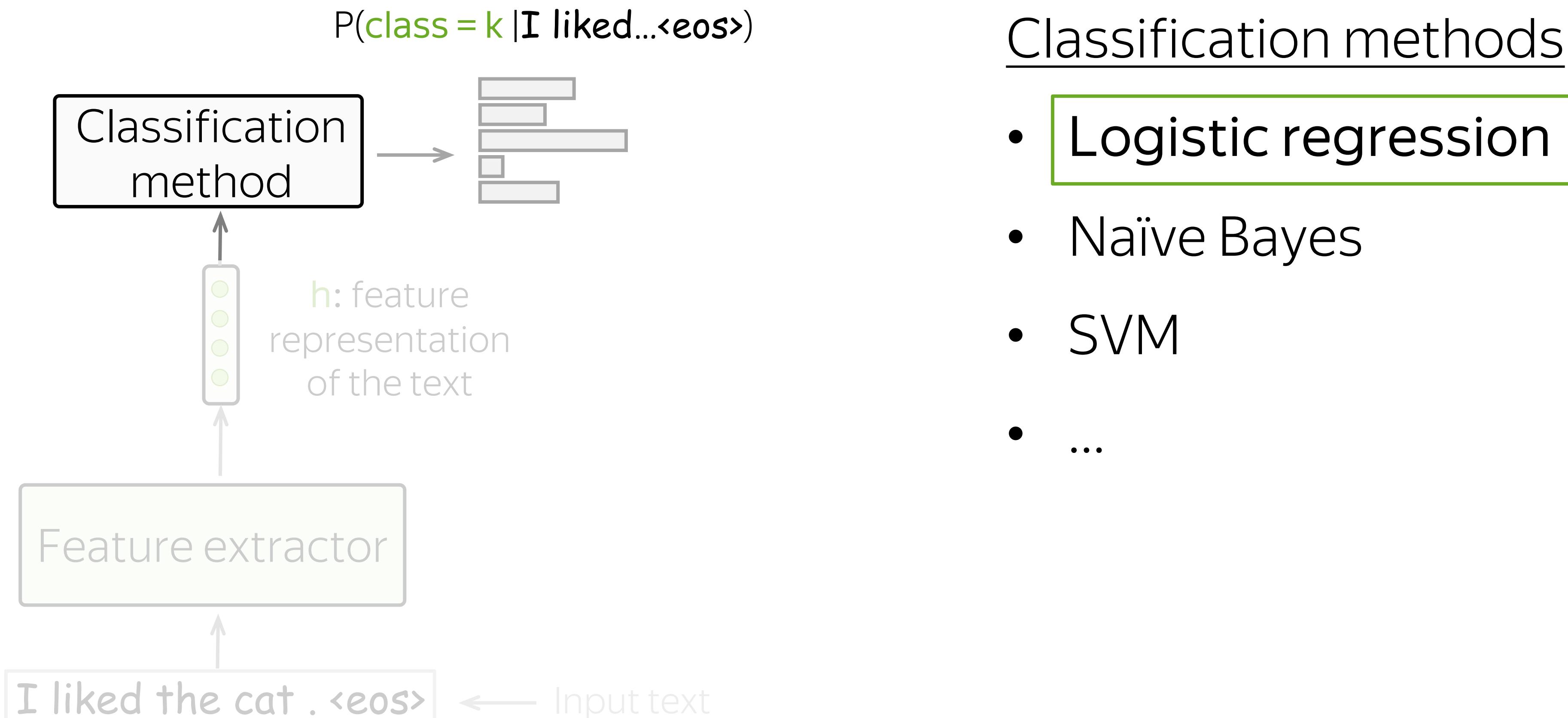
General View: Extract Features and Classify



Classification methods:

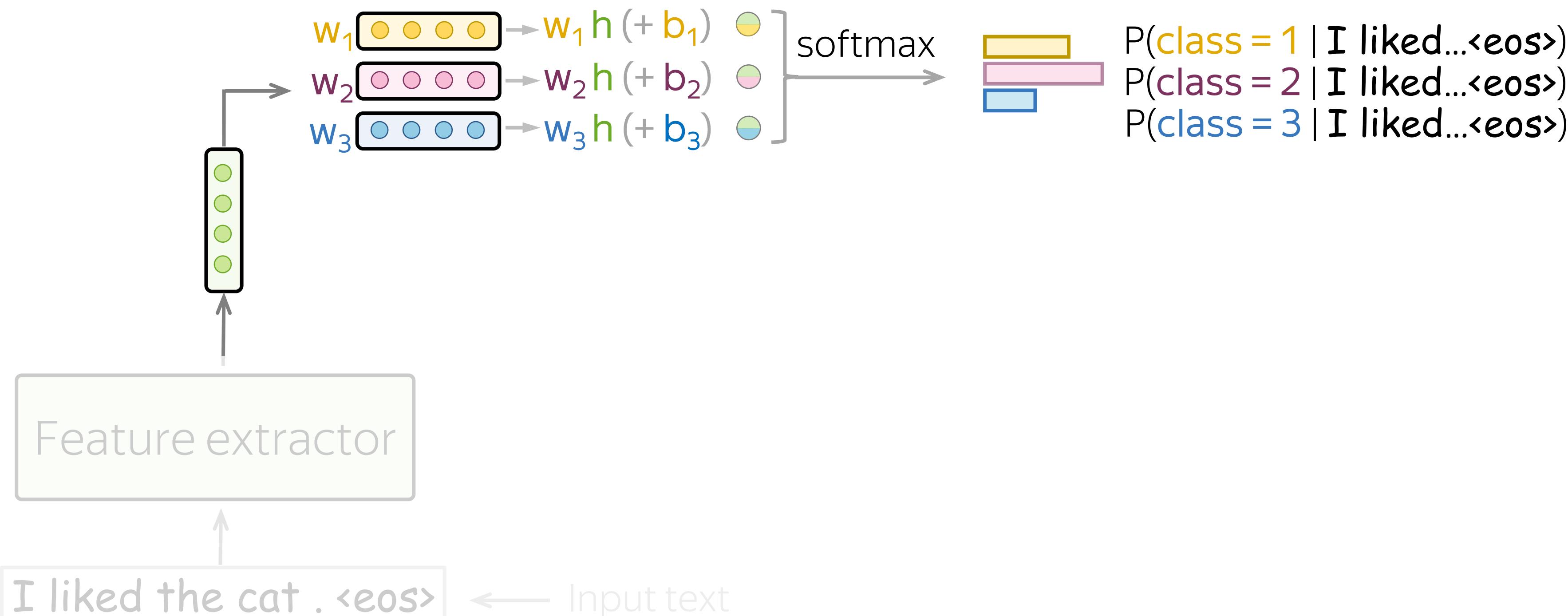
- Logistic regression
- Naïve Bayes
- SVM
- ...

General View: Extract Features and Classify



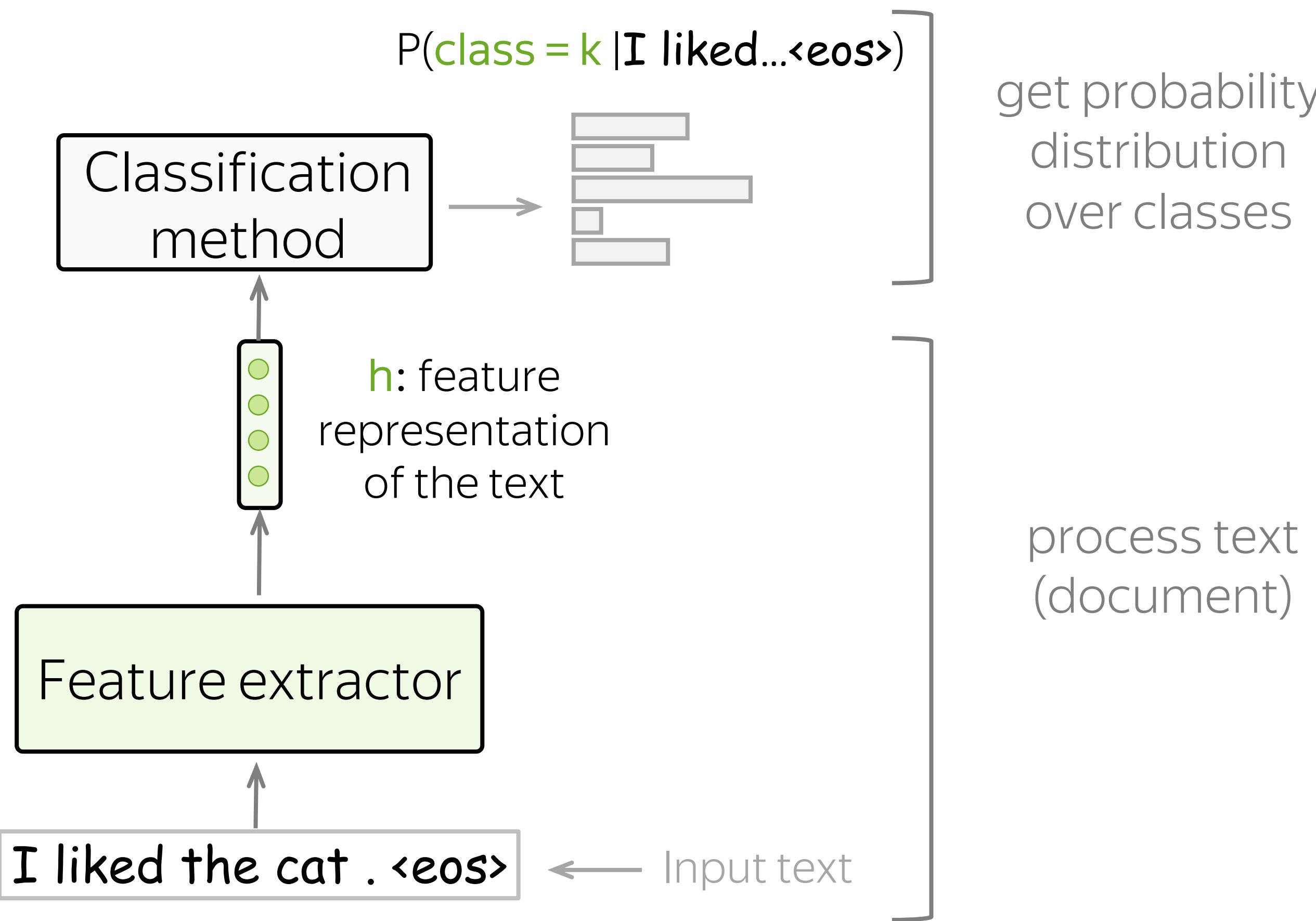
Logistic Regression: Weigh Features

Weigh features: take dot-product of \mathbf{h} with feature weights for each class



Classification with Neural Networks

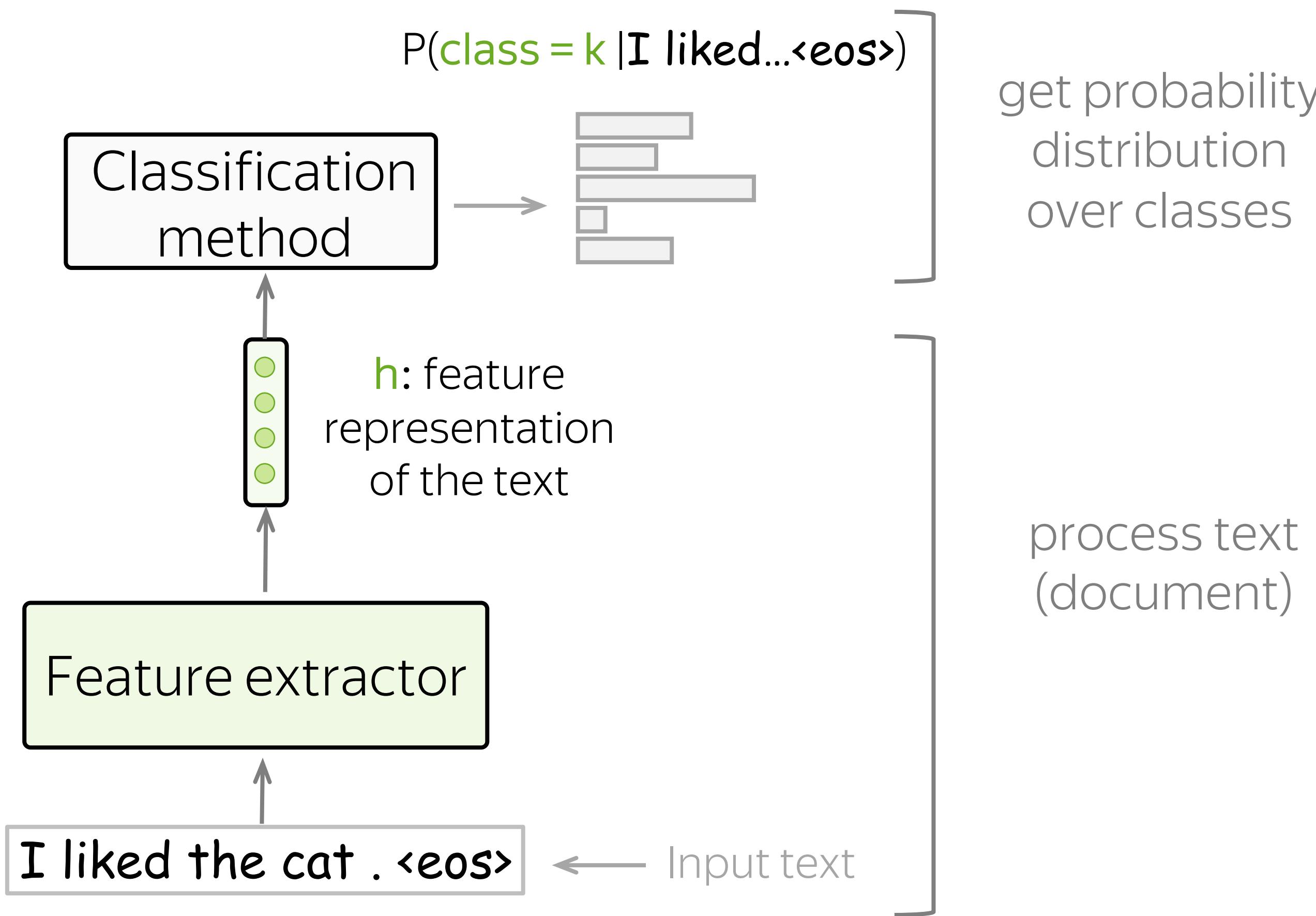
- General Classification Pipeline



Classification with Neural Networks

Instead of manually defined features, let a neural network to learn useful features.

- General Classification Pipeline
- Classification with Neural Networks



What is going to happen:

- Classification →
 - General View
 - Neural approach
 - Convolutional Networks
 - Recurrent Networks
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff

Features:
from hand-crafted
to learned

Model:
from task-specific
to generalized

What is going to happen:

- Classification →
 - General View
 - Neural approach
 - Convolutional Networks
 - Recurrent Networks
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff

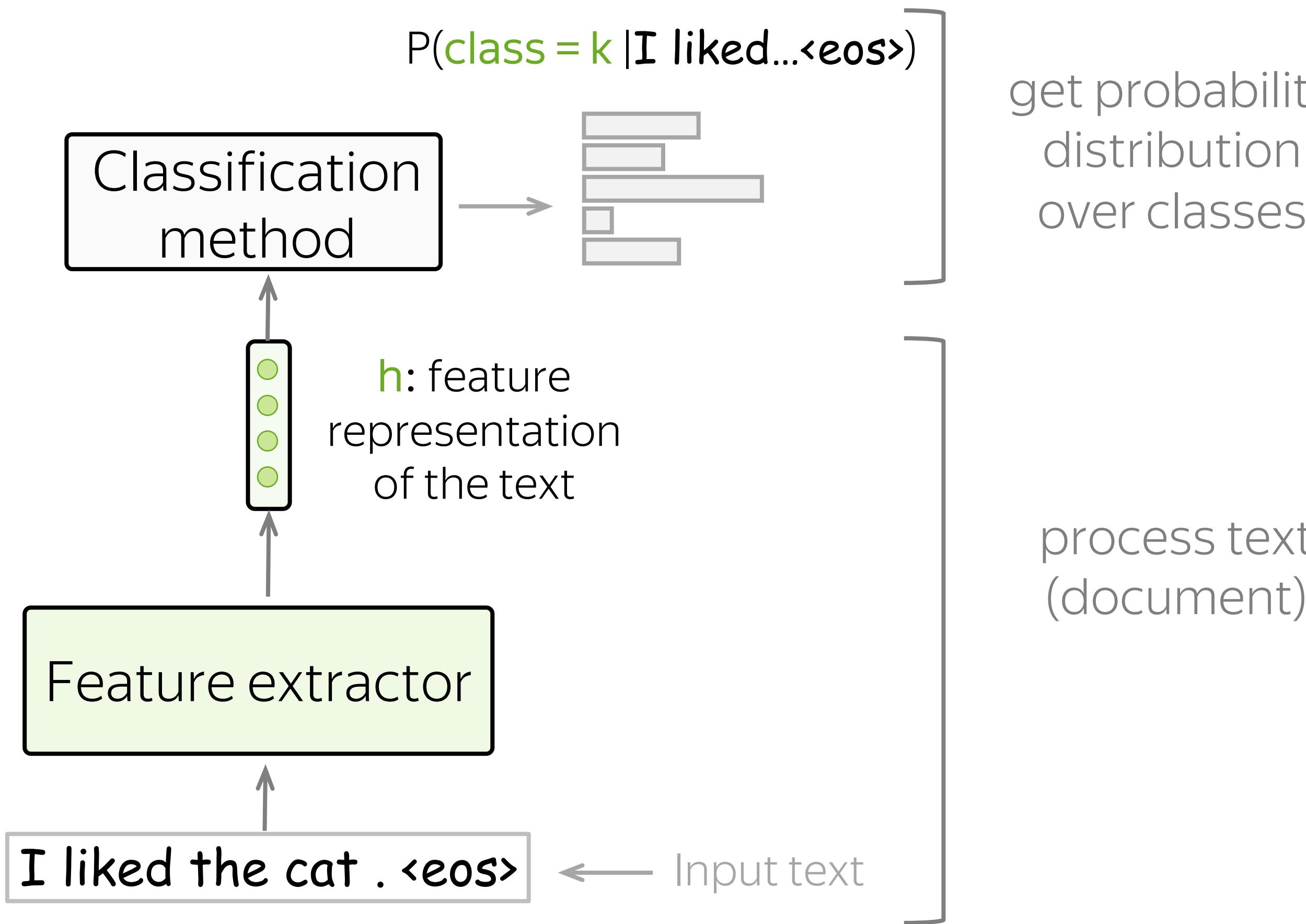
Features:
from hand-crafted
to learned

Model:
from task-specific
to generalized

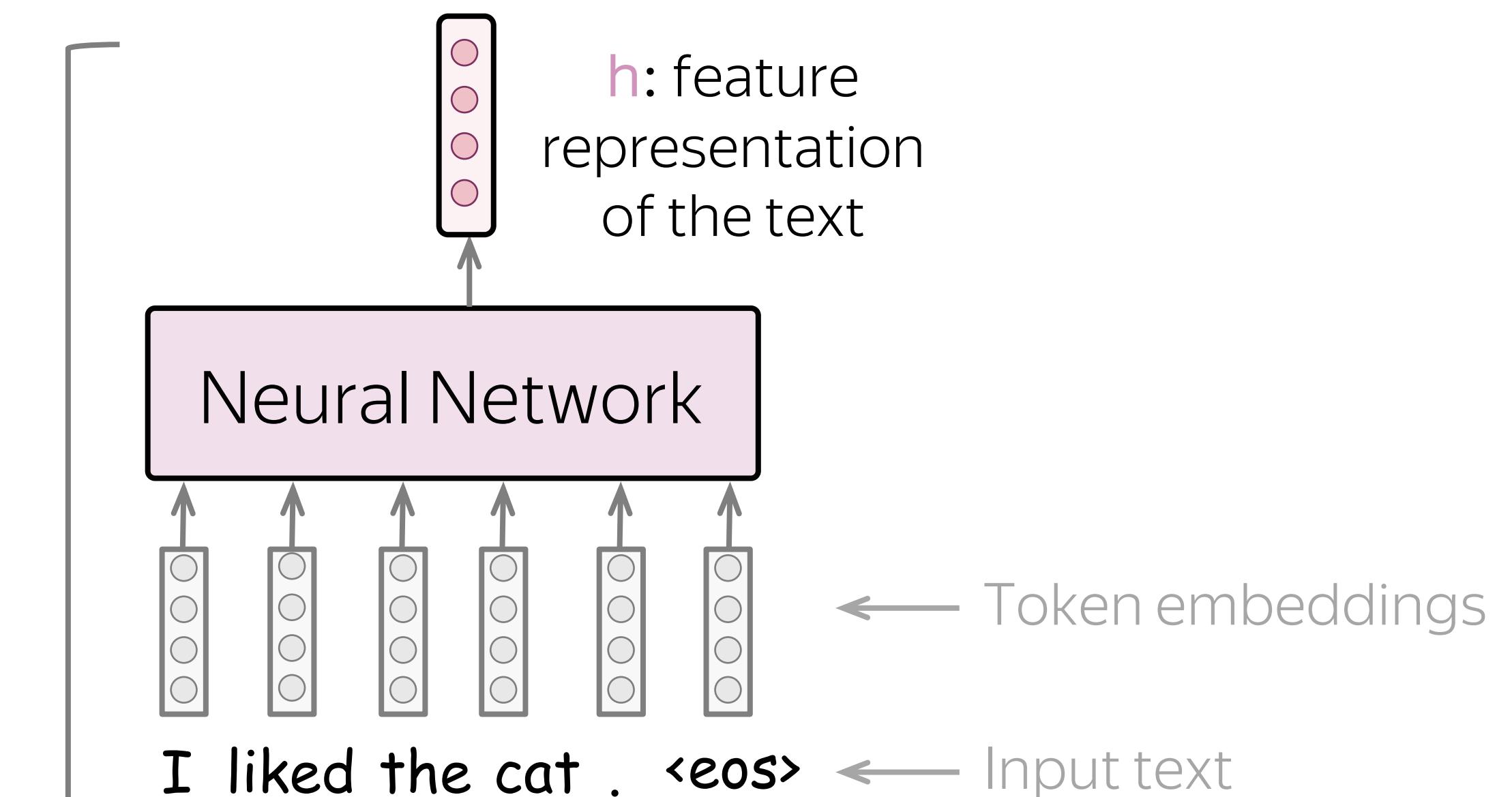
Classification with Neural Networks

Instead of manually defined features, let a neural network to learn useful features.

- General Classification Pipeline



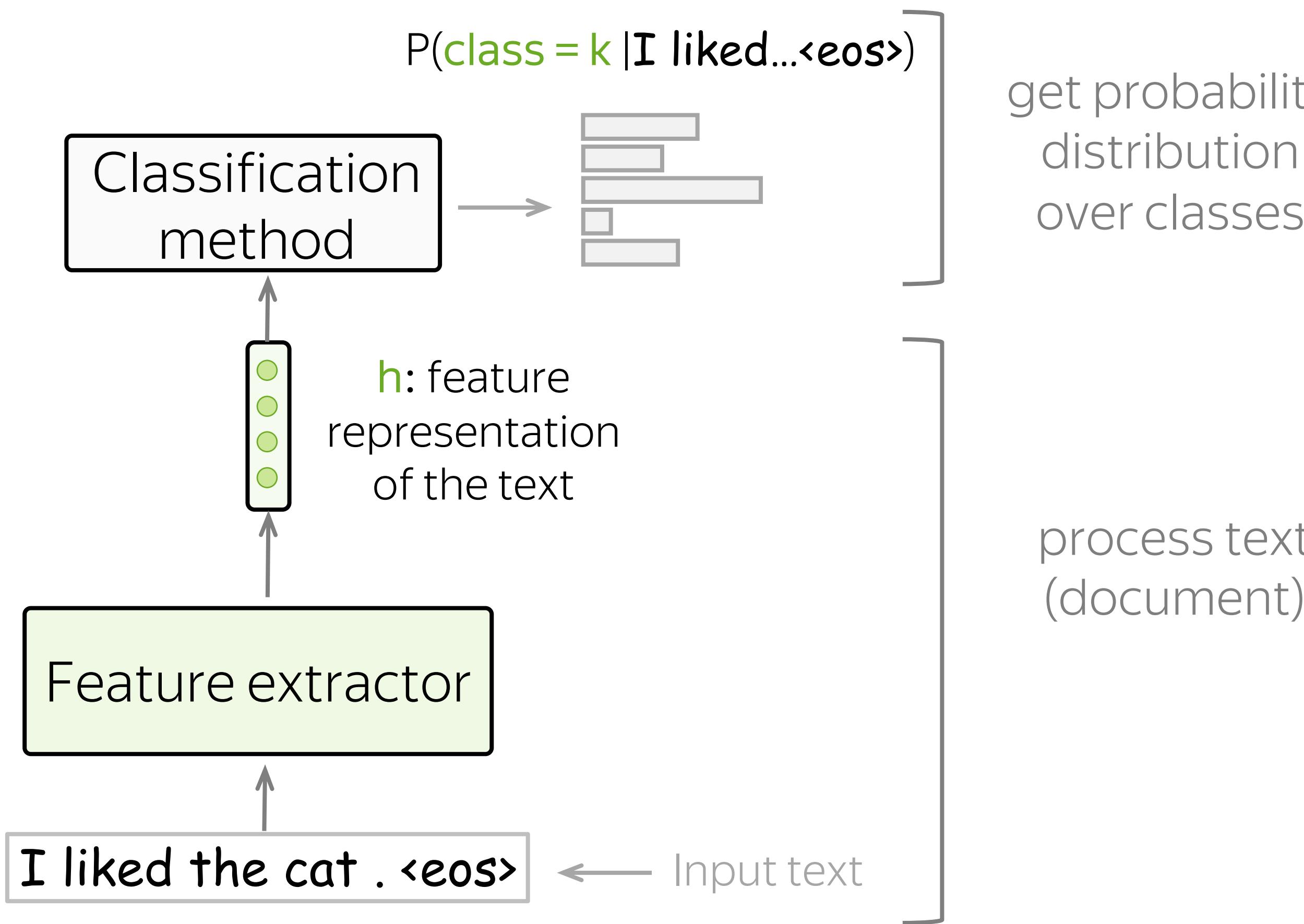
- Classification with Neural Networks



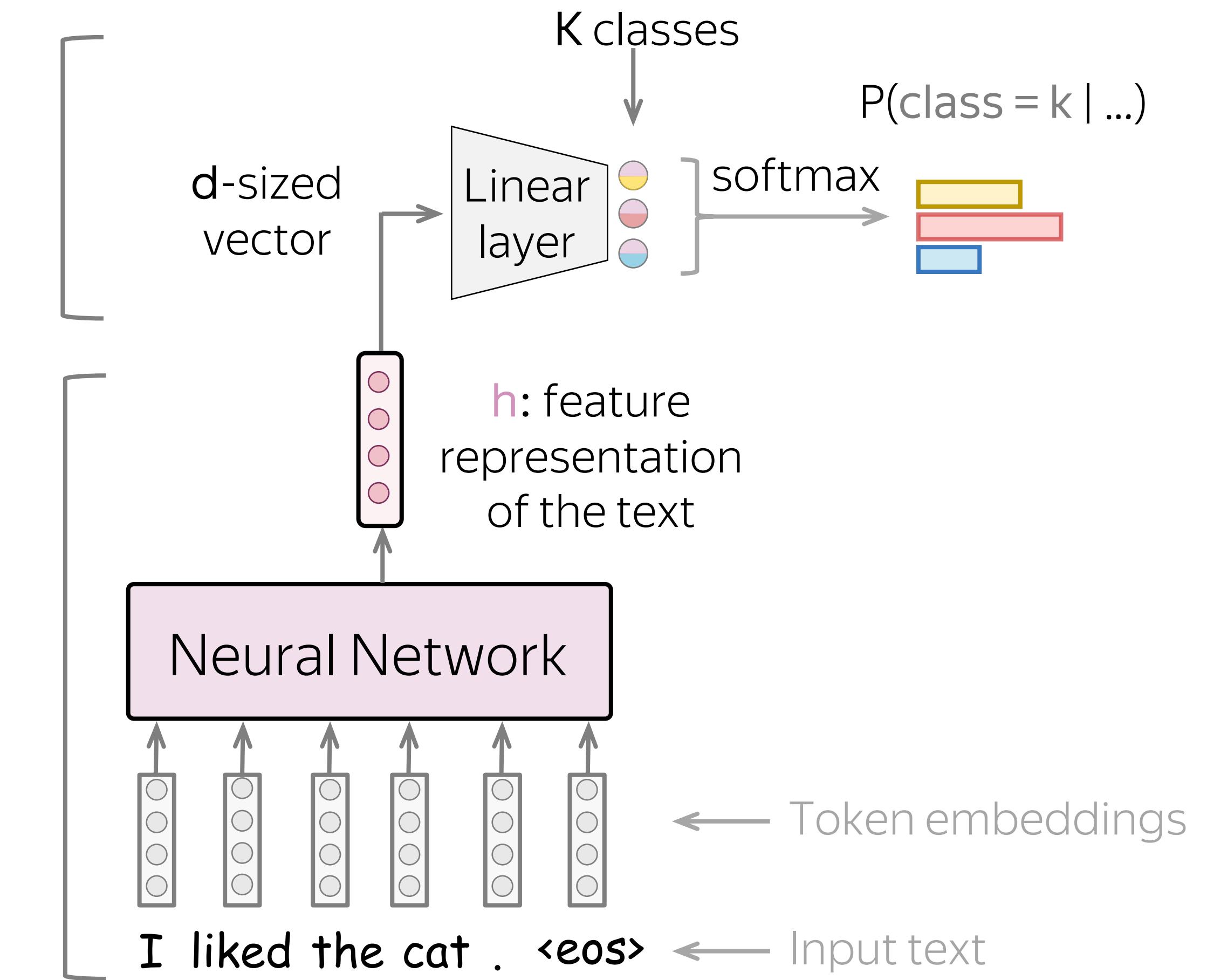
Classification with Neural Networks

Instead of manually defined features, let a neural network to learn useful features.

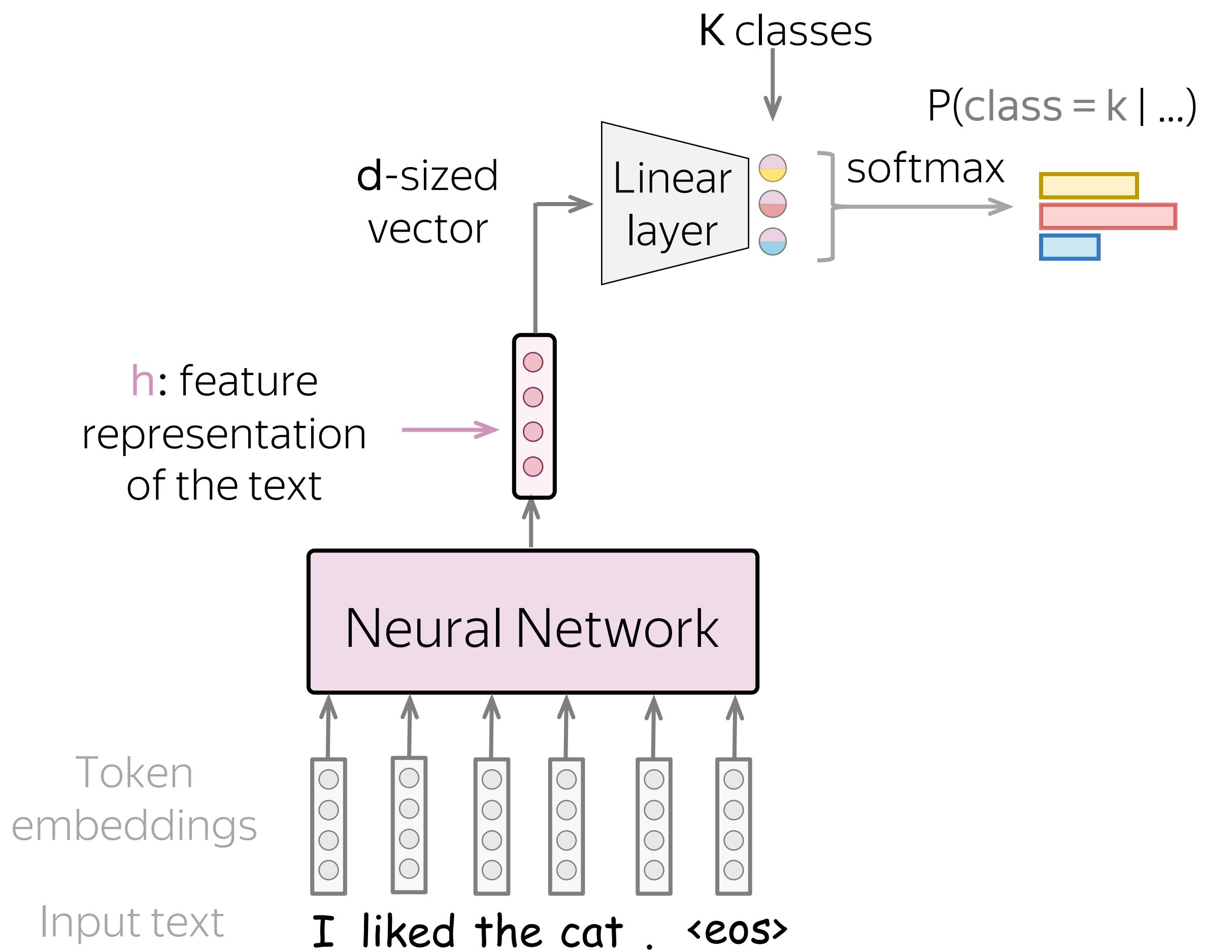
- General Classification Pipeline



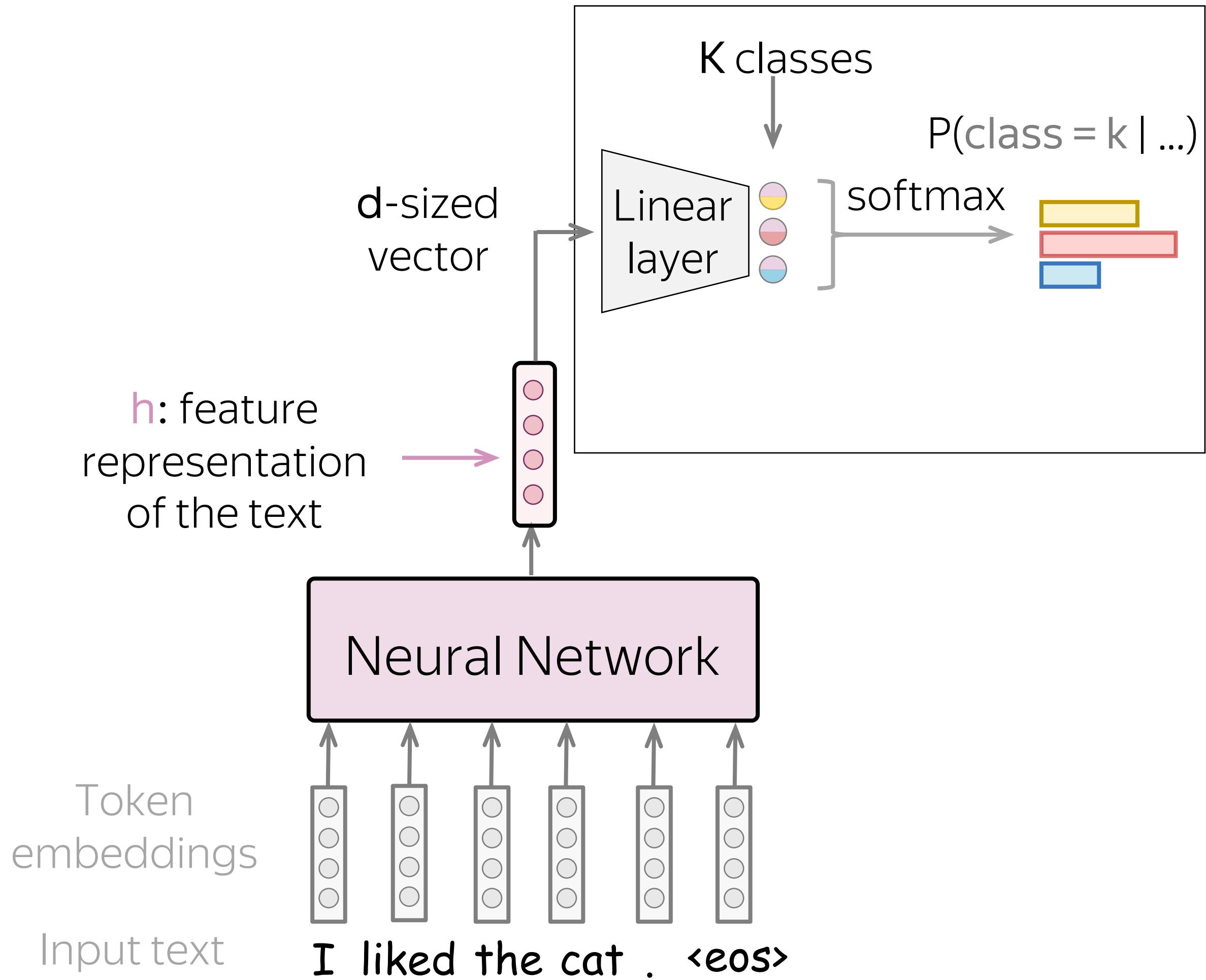
- Classification with Neural Networks



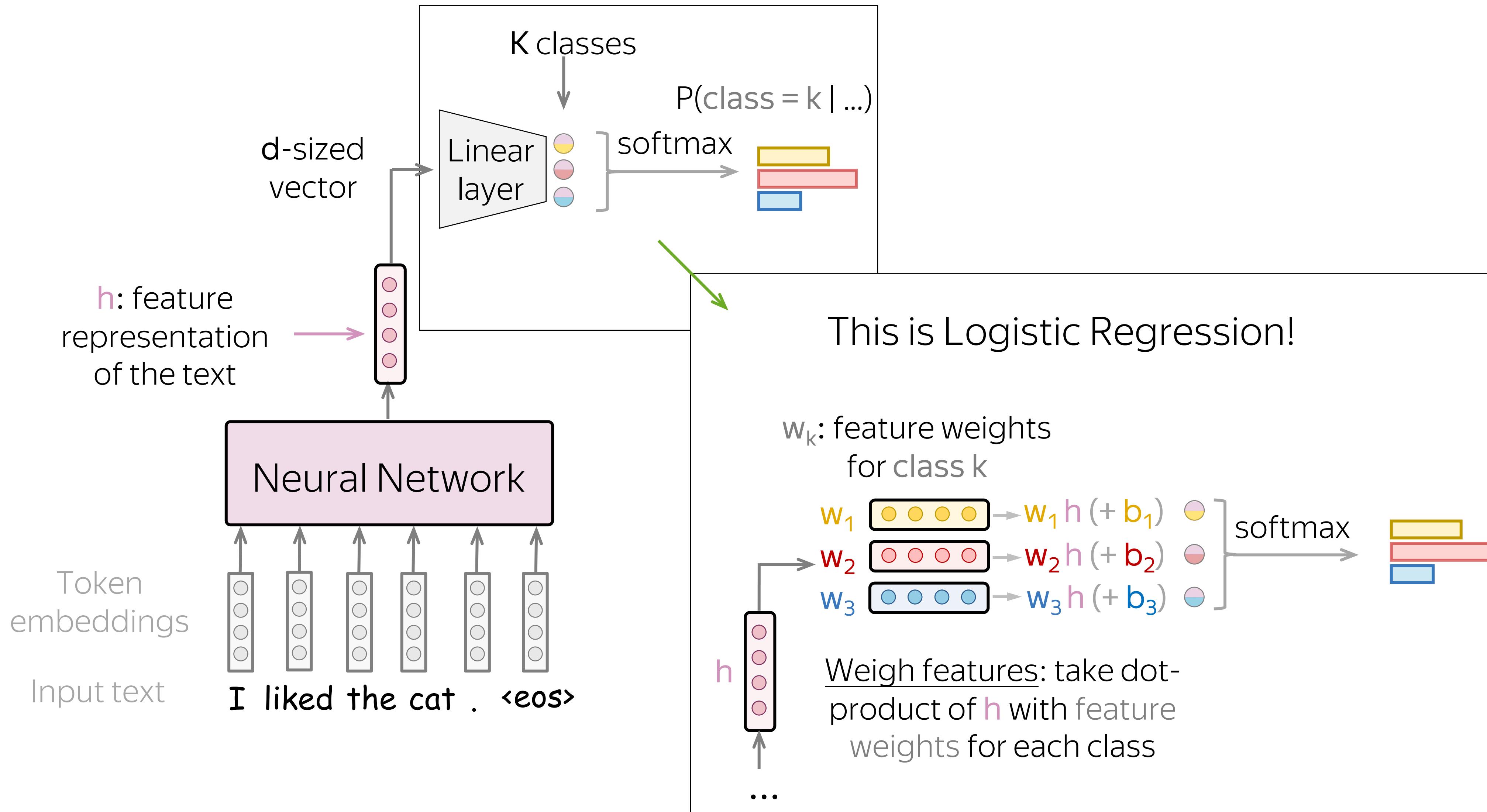
Classification with Neural Networks



Classification with Neural Networks



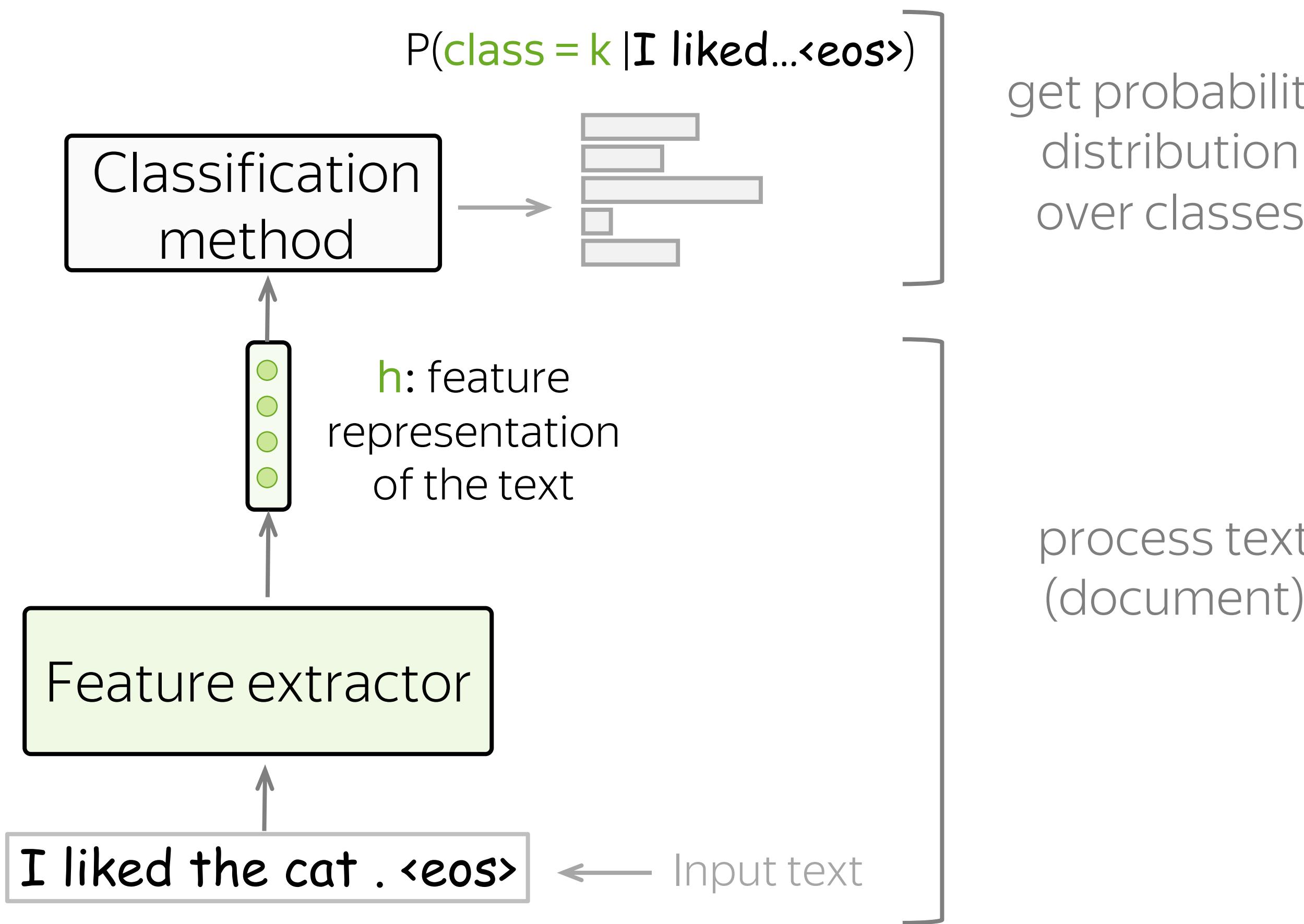
Classification with Neural Networks



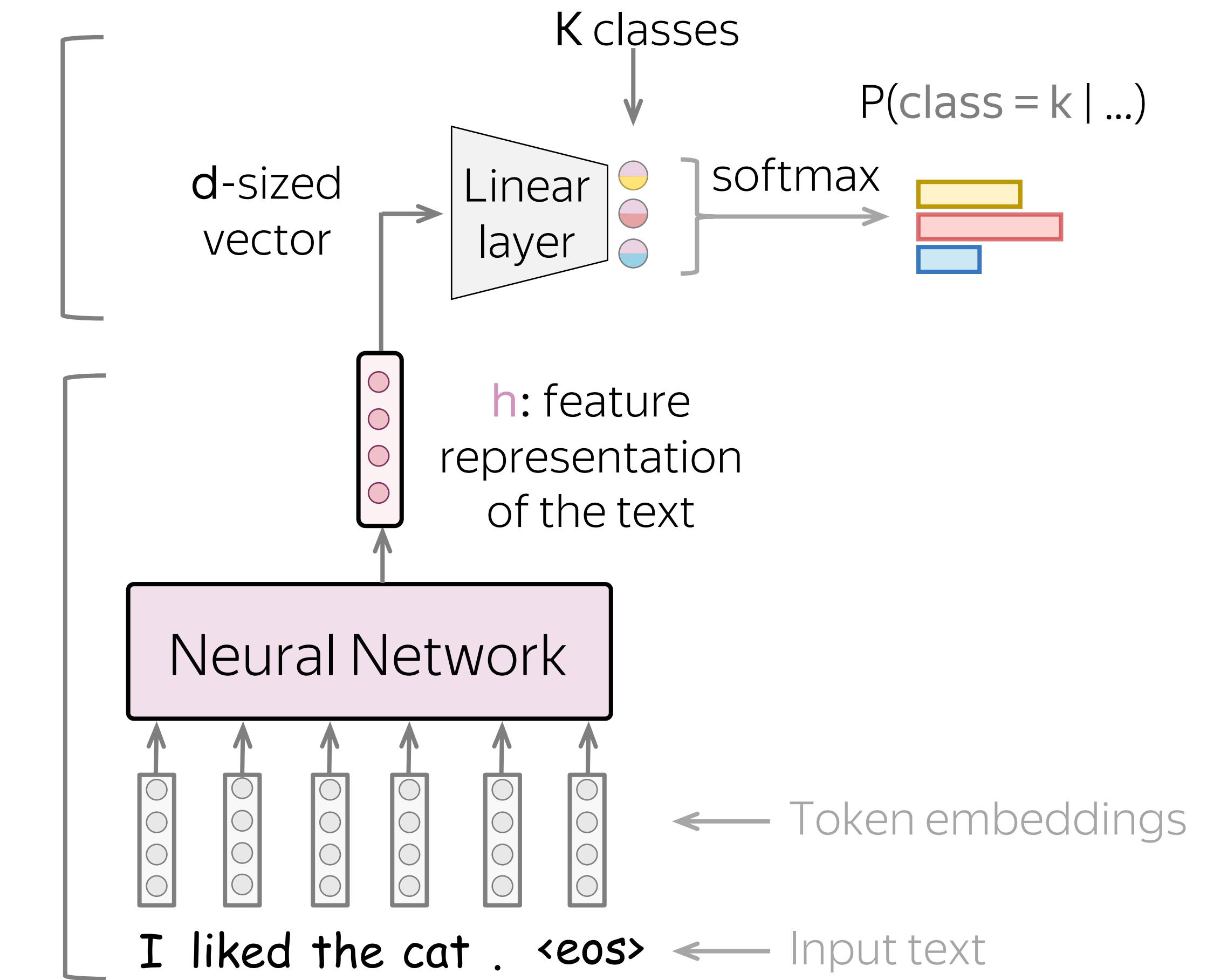
Classification with Neural Networks

Instead of manually defined features, let a neural network to learn useful features.

- General Classification Pipeline



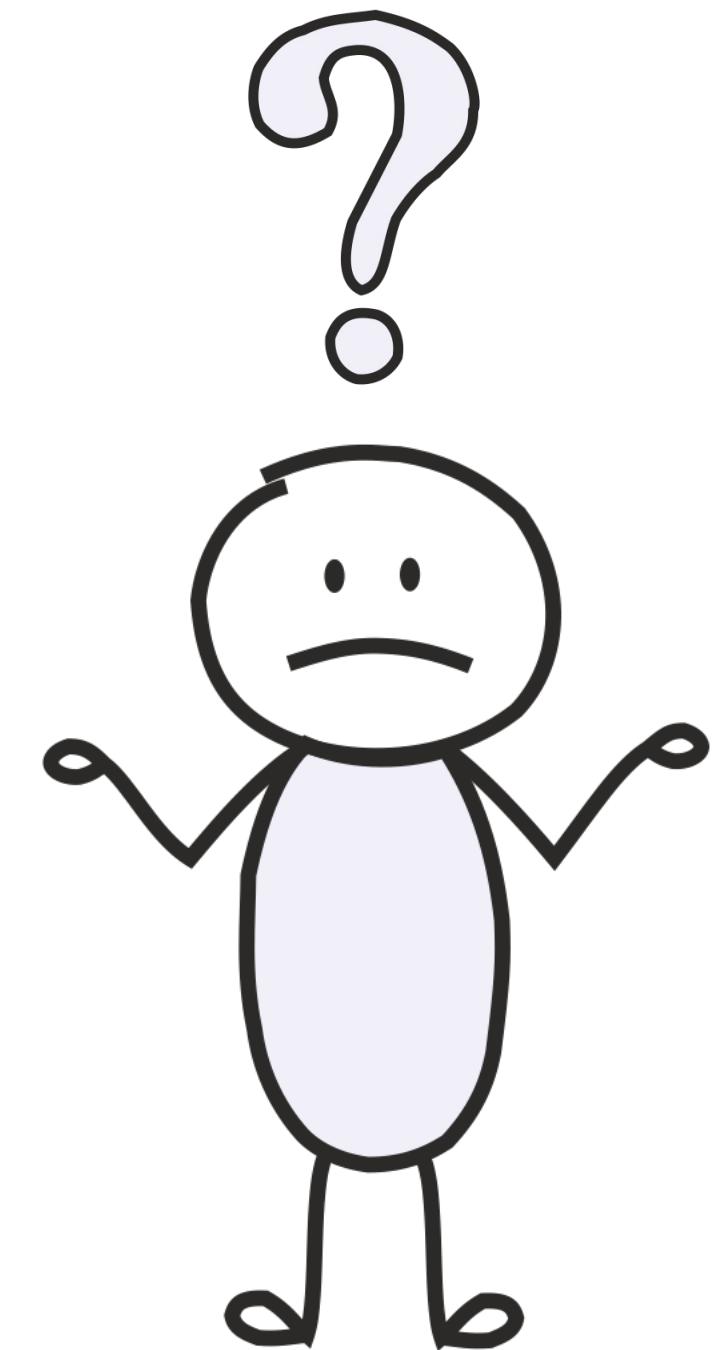
- Classification with Neural Networks



Several Filters – Several Features

Instead of manually defined features, let a neural network to learn useful features.

But how will a network learn
useful features?



Training: Cross-Entropy

Training example: I liked the cat on the mat <eos>

Label: k

Training: Cross-Entropy

Training example: I liked the cat on the mat <eos>

Label: k

Model prediction:

$P(\text{class} = i | \text{I liked...<eos>})$



Target:

p^*



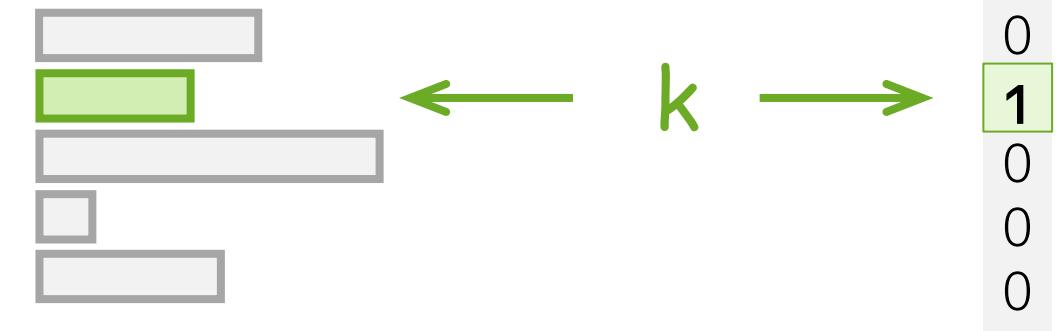
Training: Cross-Entropy

Training example: I liked the cat on the mat <eos>

Label: **k**

Model prediction:

$P(\text{class } i \mid \text{I liked...<eos>})$



Target:

p^*



$$\text{Loss} = -\log(p(k)) \rightarrow \min$$



Training: Cross-Entropy

Training example: I liked the cat on the mat <eos>

Label: k

Model prediction:

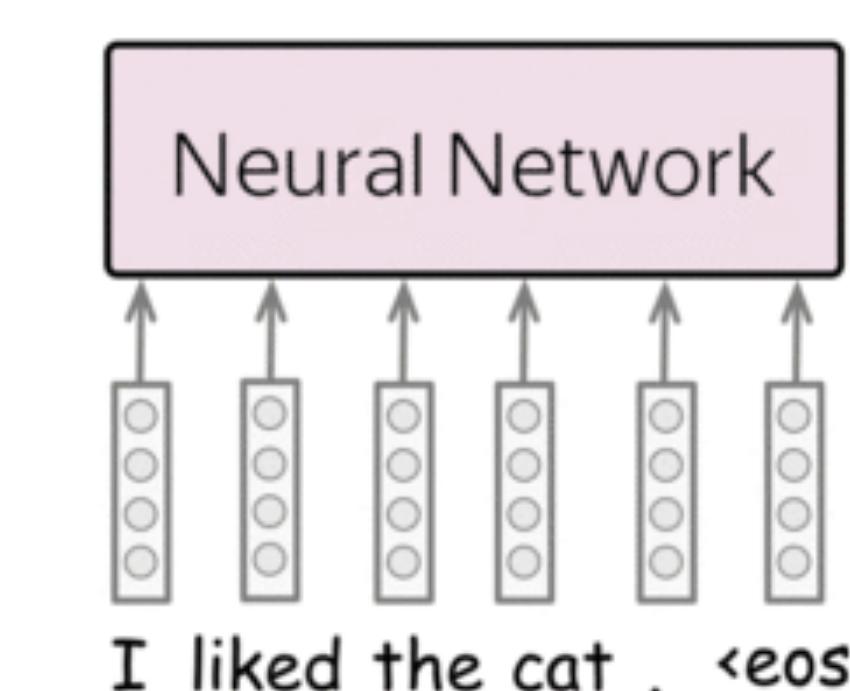
$P(\text{class } i \mid \text{I liked...<eos>})$



Target:

p^*

$$\text{Loss} = -\log(p(k)) \rightarrow \min$$



Correct label: 4 ← we want the model
to predict this

What is going to happen:

- Classification →
 - General View
 - Neural approach
 - Convolutional Networks
 - Recurrent Networks
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff

Features:
from hand-crafted
to learned

Model:
from task-specific
to generalized

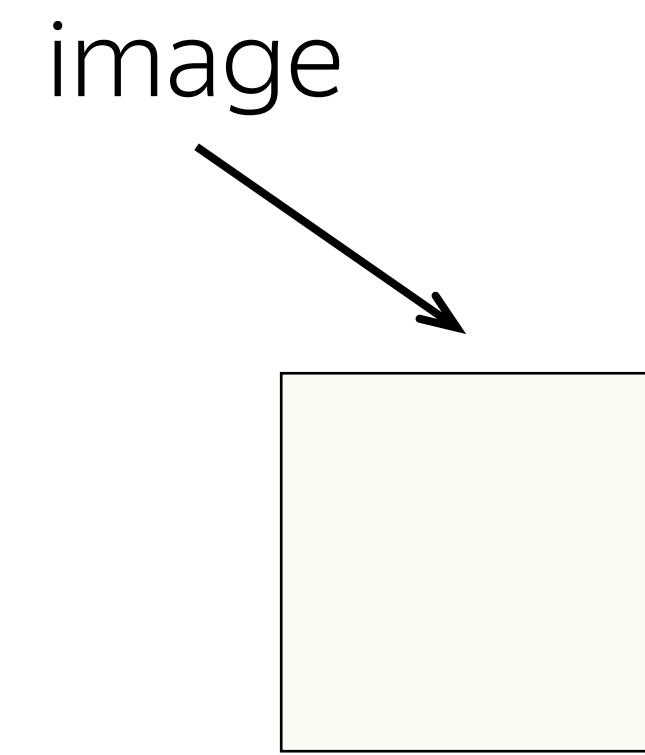
What is going to happen:

- Classification →
 - General View
 - Neural approach
 - Convolutional Networks
 - Recurrent Networks
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff

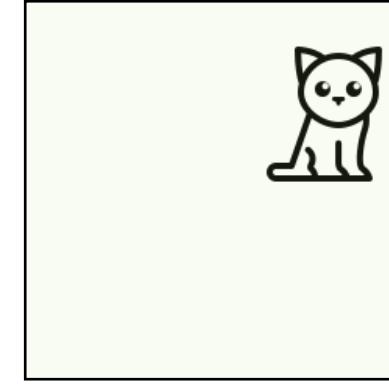
Features:
from hand-crafted
to learned

Model:
from task-specific
to generalized

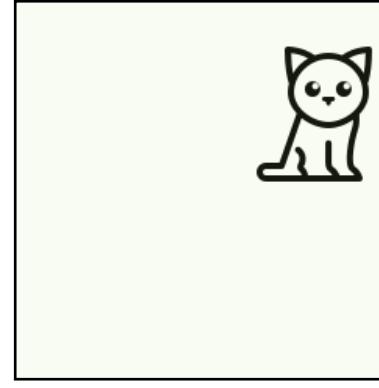
Convolutions for Images and Translation Invariance



Convolutions for Images and Translation Invariance

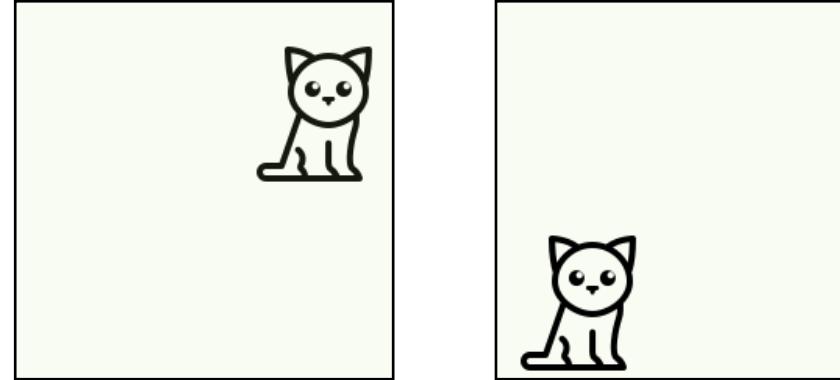


Convolutions for Images and Translation Invariance



Label: **cat**

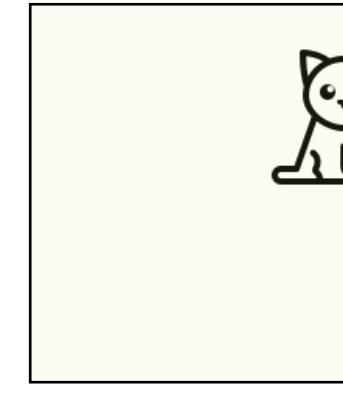
Convolutions for Images and Translation Invariance



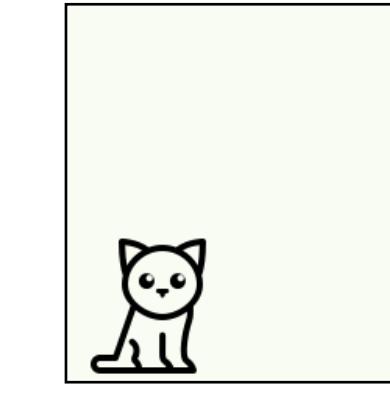
Label: cat

Label: cat

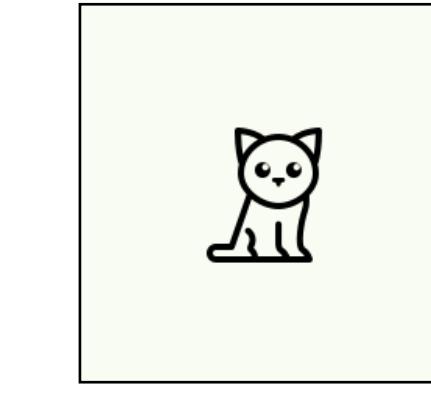
Convolutions for Images and Translation Invariance



Label: cat

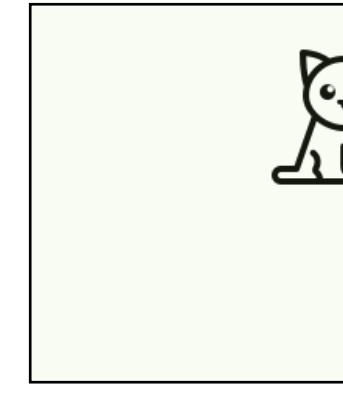


Label: cat

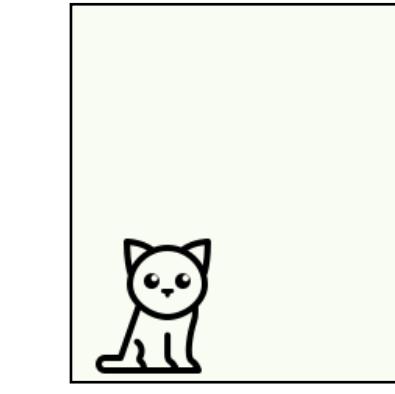


Label: cat

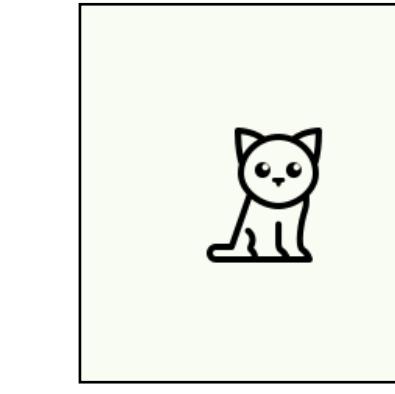
Convolutions for Images and Translation Invariance



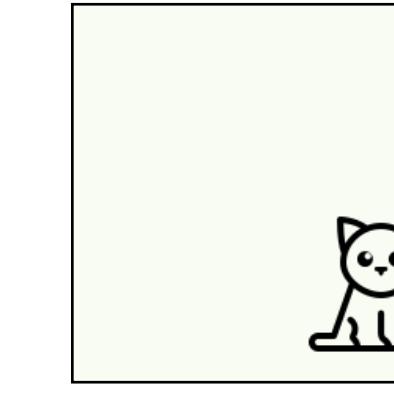
Label: **cat**



Label: **cat**

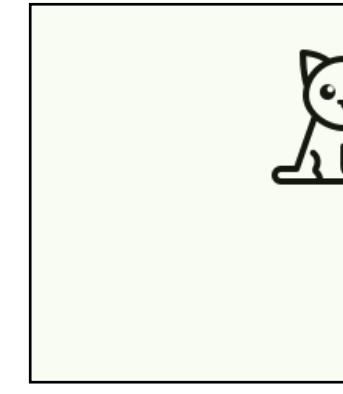


Label: **cat**

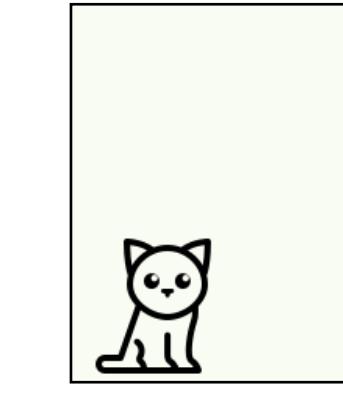


Label: **cat**

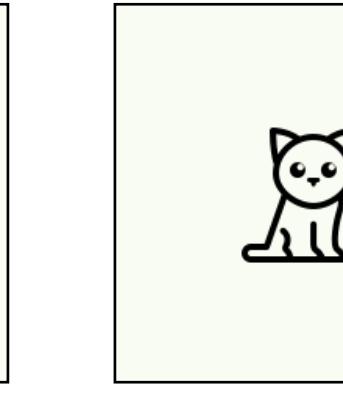
Convolutions for Images and Translation Invariance



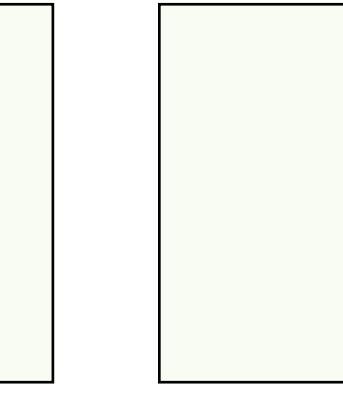
Label: **cat**



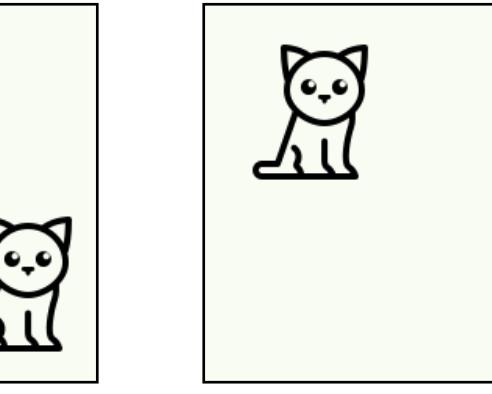
Label: **cat**



Label: **cat**

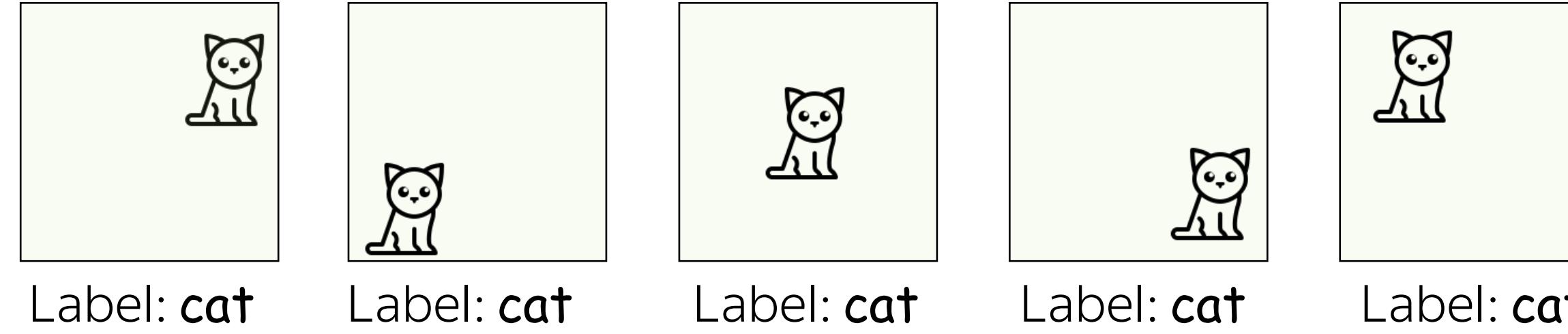


Label: **cat**



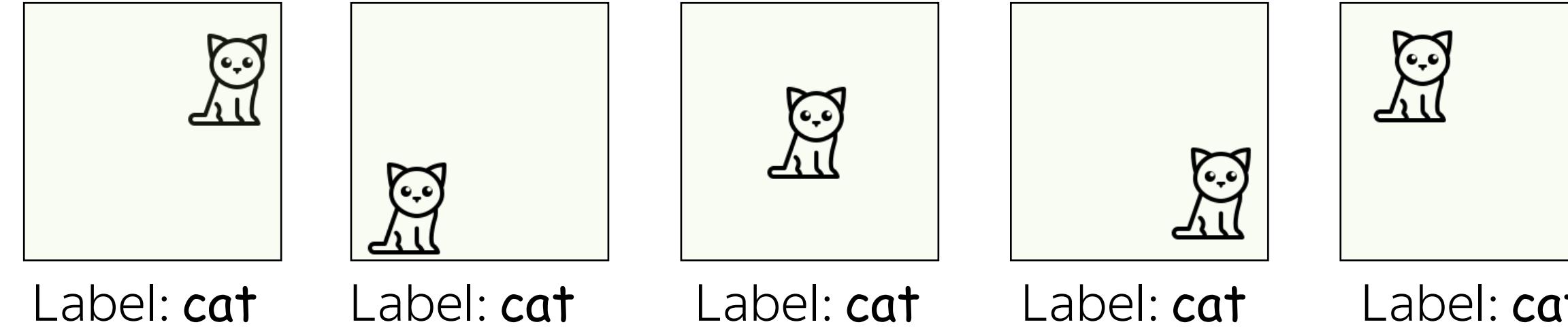
Label: **cat**

Convolutions for Images and Translation Invariance



We don't care where the cat is, we
care that it is somewhere.

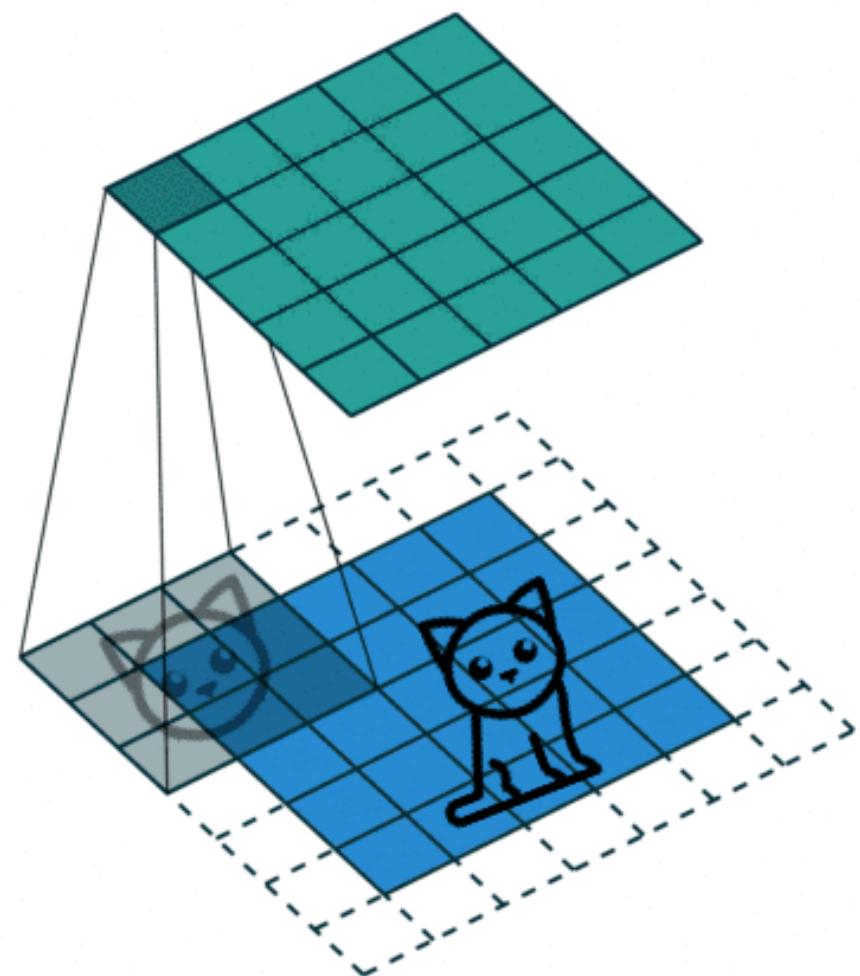
Convolutions for Images and Translation Invariance



We don't care where the cat is, we
care that it is somewhere.

Then why don't we process all these
cats similarly?

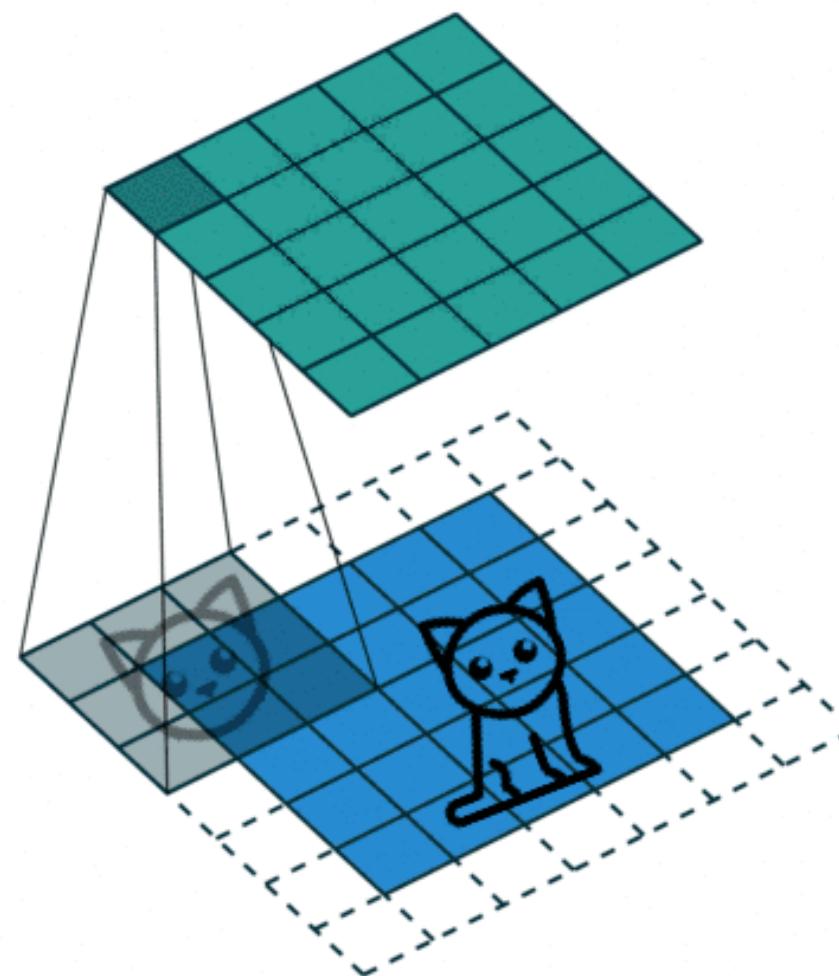
Convolutions for Images and Translation Invariance



- apply the same operation to small parts of an input
- find “matches” with patterns

The gif is adapted from the one taken from the repo
https://github.com/vdumoulin/conv_arithmetic

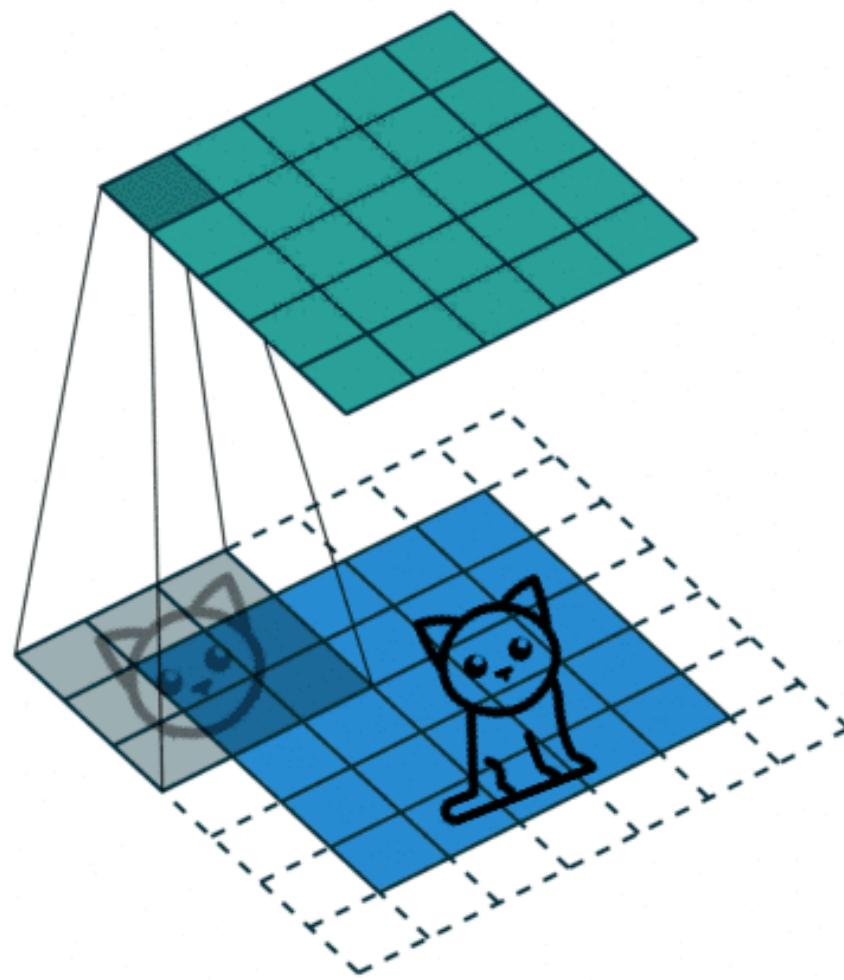
Convolutions for Images and Translation Invariance



- apply the same operation to small parts of an input
 - find “matches” with patterns
- this is how CNNs extract features

The gif is adapted from the one taken from the repo
https://github.com/vdumoulin/conv_arithmetic

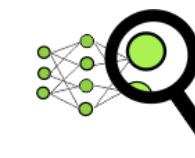
Convolutions for Images and Translation Invariance



The gif is adapted from the one taken from the repo
https://github.com/vdumoulin/conv_arithmetic

- apply the same operation to small parts of an input
- find “matches” with patterns
- a network learns which patterns are useful
- from bottom to top of a network, patterns evolve from simple to complicated

this is how CNNs extract features



We'll see this in the analysis section

What About Texts?

An **absolutely great** movie! I watched the premiere with my friends.

The movie about cats was **absolutely great**, and the cats were cute.

The movie is about cats running around, and it is **absolutely great**.

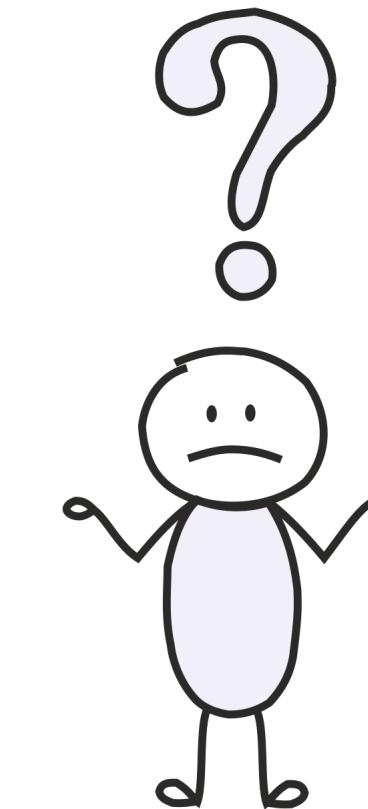
What About Texts?

An **absolutely great** movie! I watched the premiere with my friends.

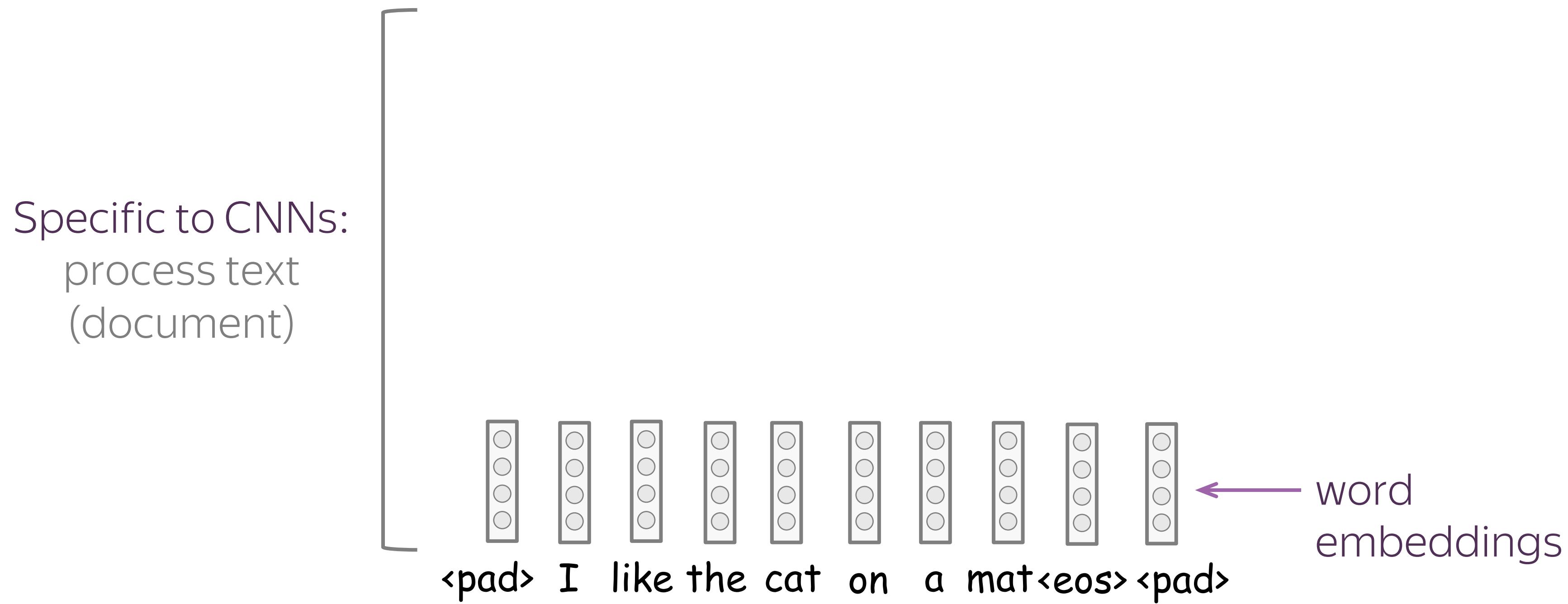
The movie about cats was **absolutely great**, and the cats were cute.

The movie is about cats running around, and it is **absolutely great**.

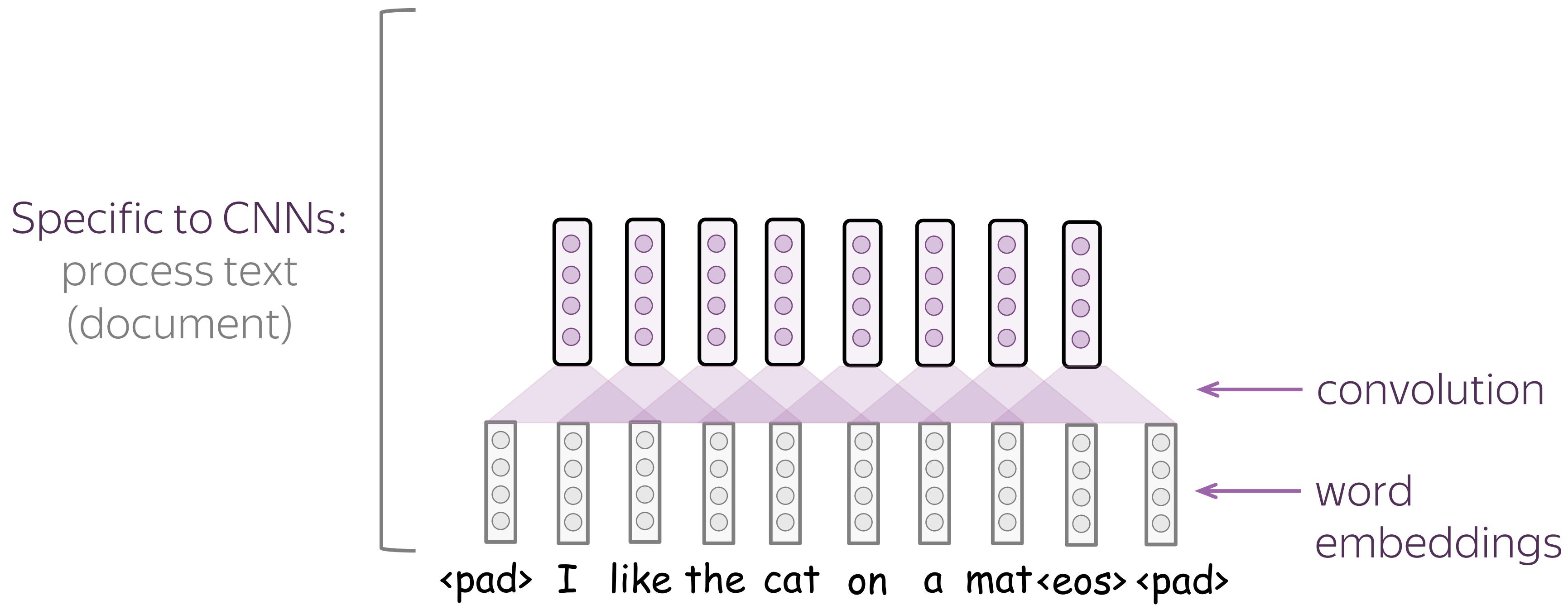
If a clue is very informative, maybe we don't care much where in a text it appears?



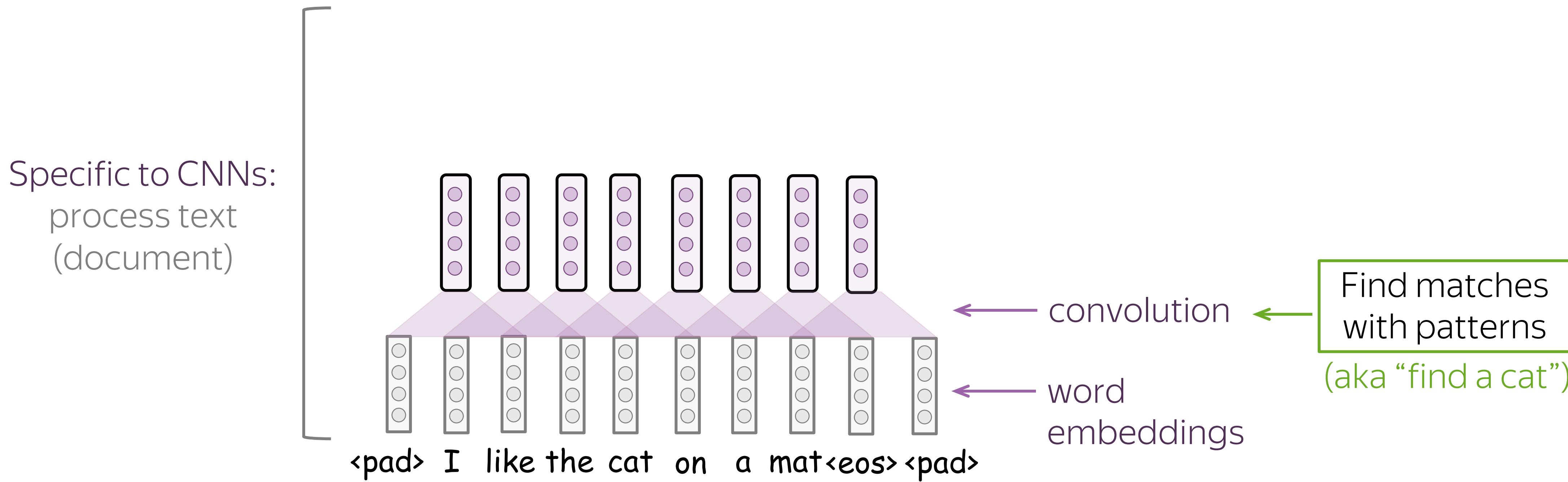
A Typical Model: Convolution + Pooling



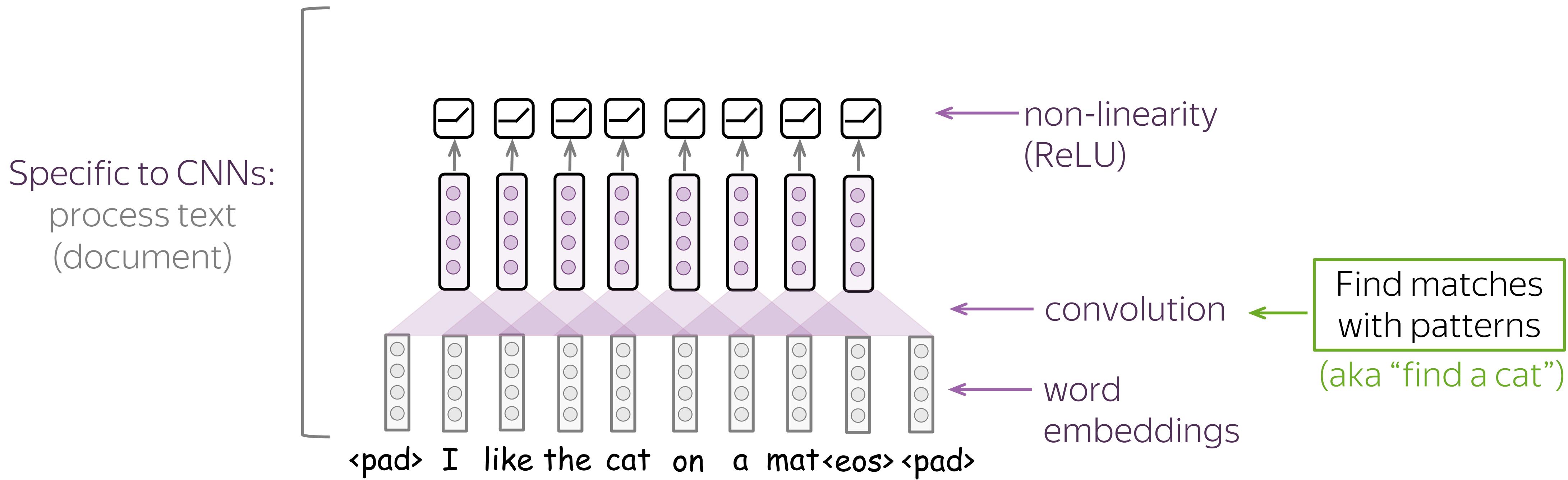
A Typical Model: Convolution + Pooling



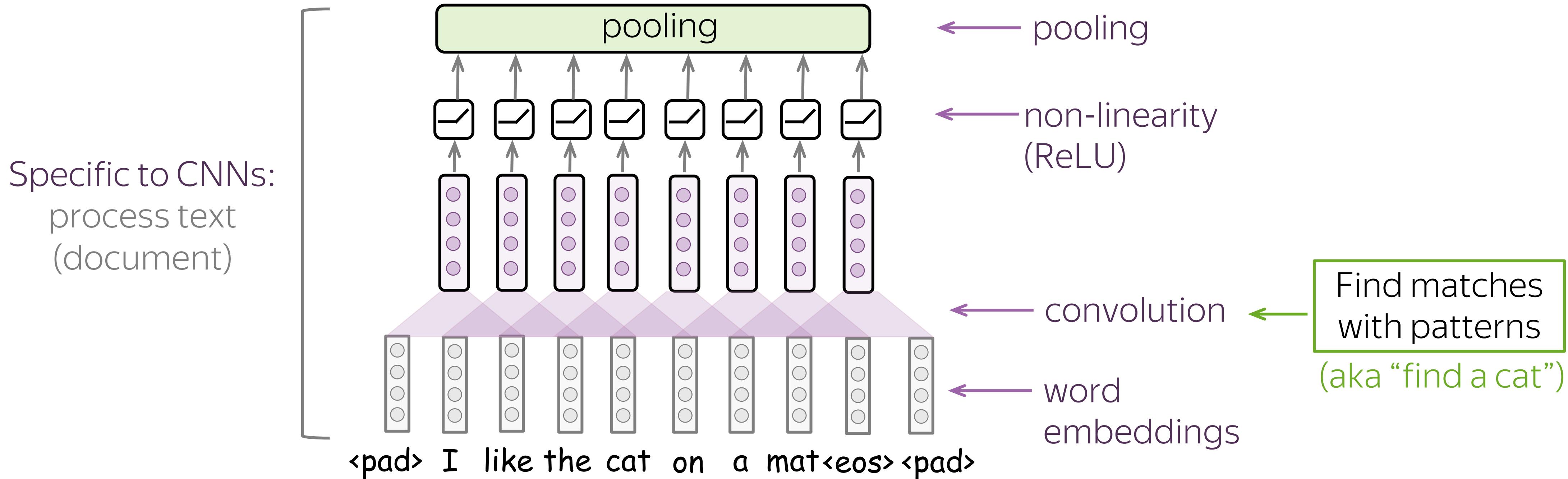
A Typical Model: Convolution + Pooling



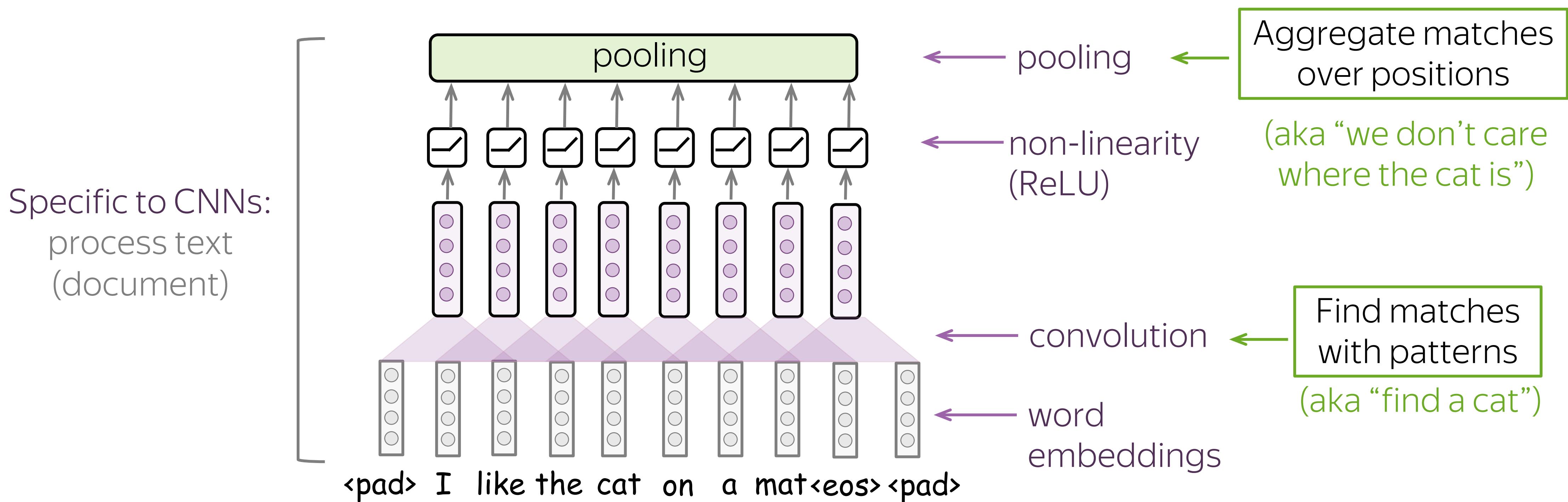
A Typical Model: Convolution + Pooling



A Typical Model: Convolution + Pooling



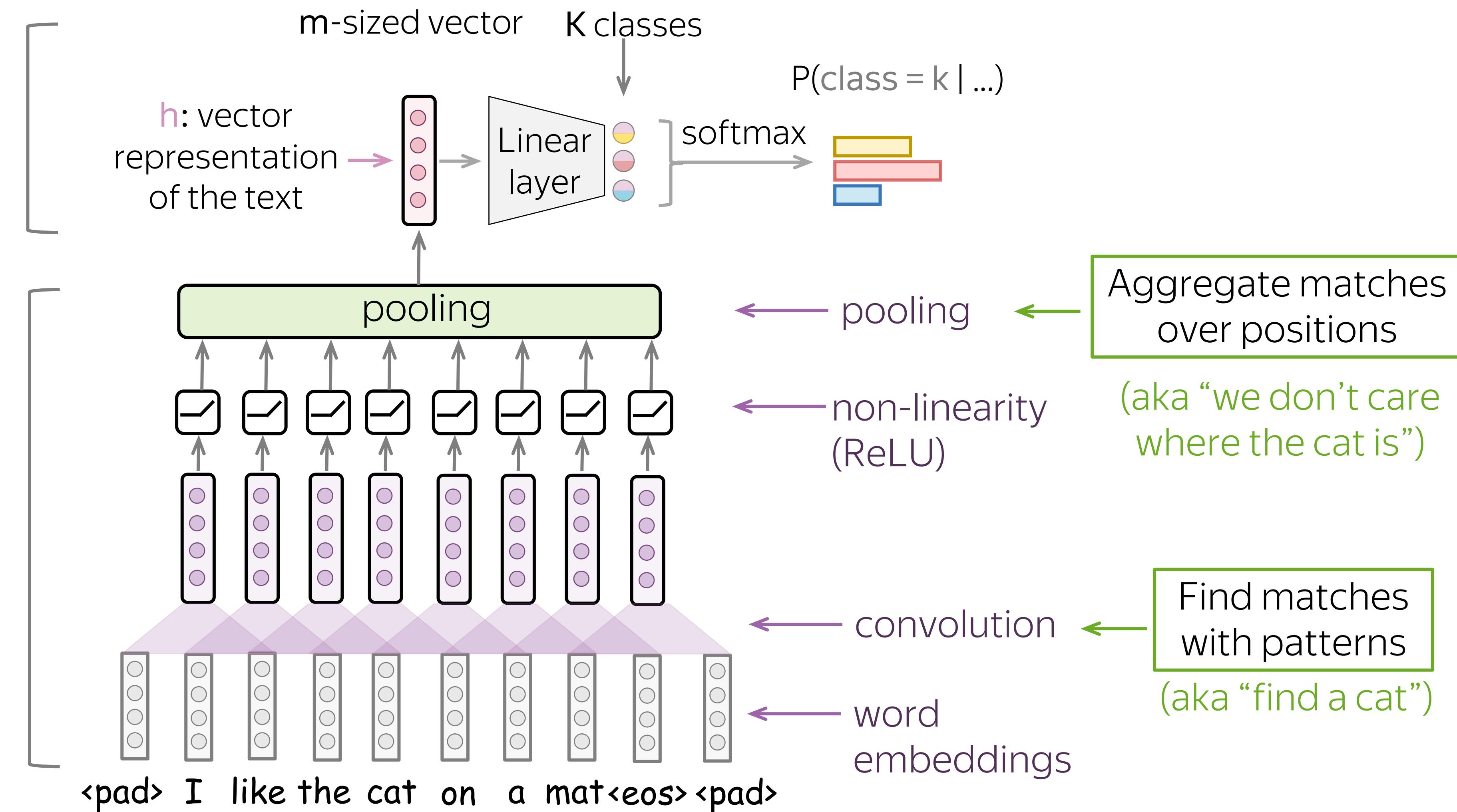
A Typical Model: Convolution + Pooling



A Typical Model: Convolution + Pooling

Standard part
(same for all NNs):
get probability
distribution

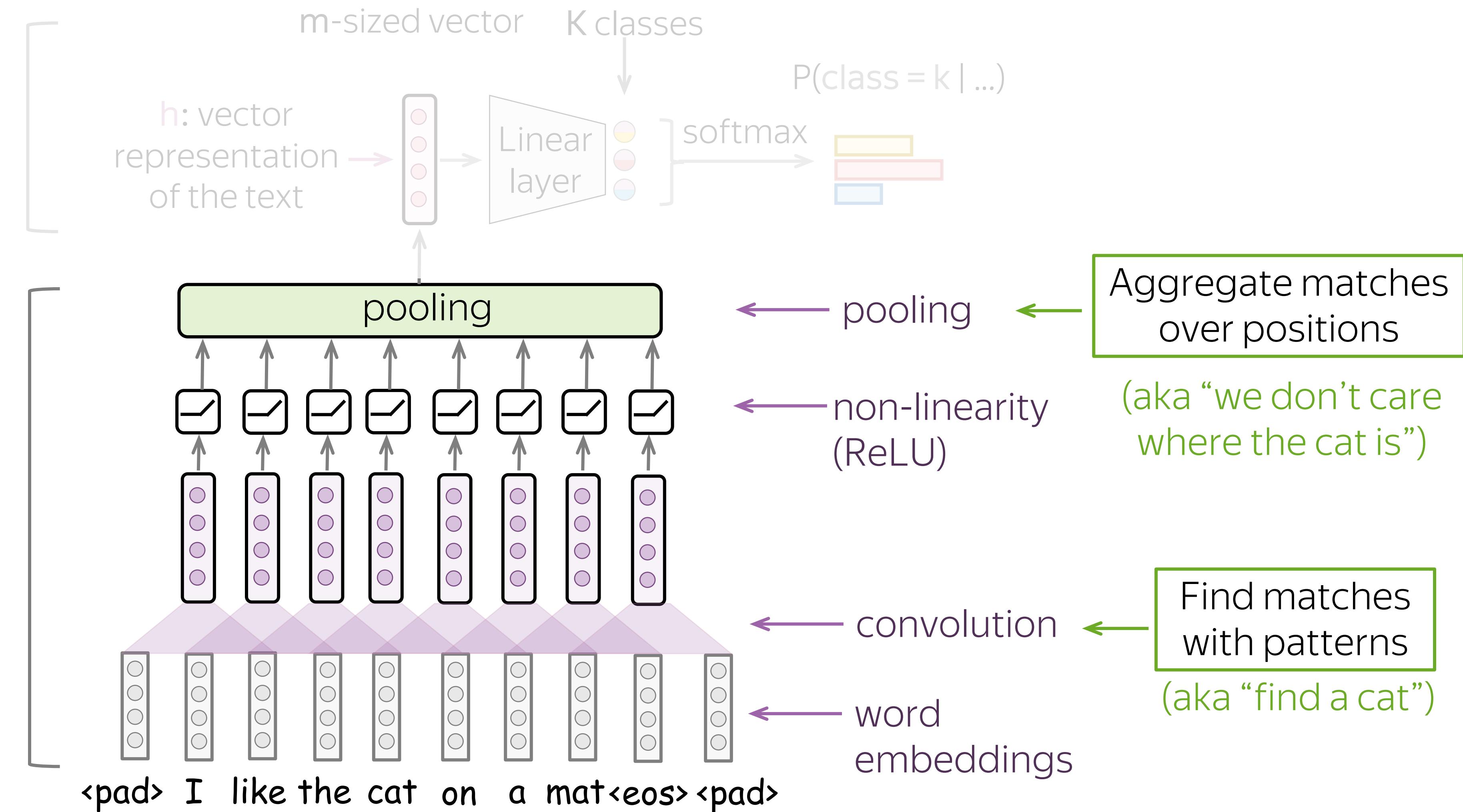
Specific to CNNs:
process text
(document)



A Typical Model: Convolution + Pooling

Standard part
(same for all NNs):
get probability
distribution

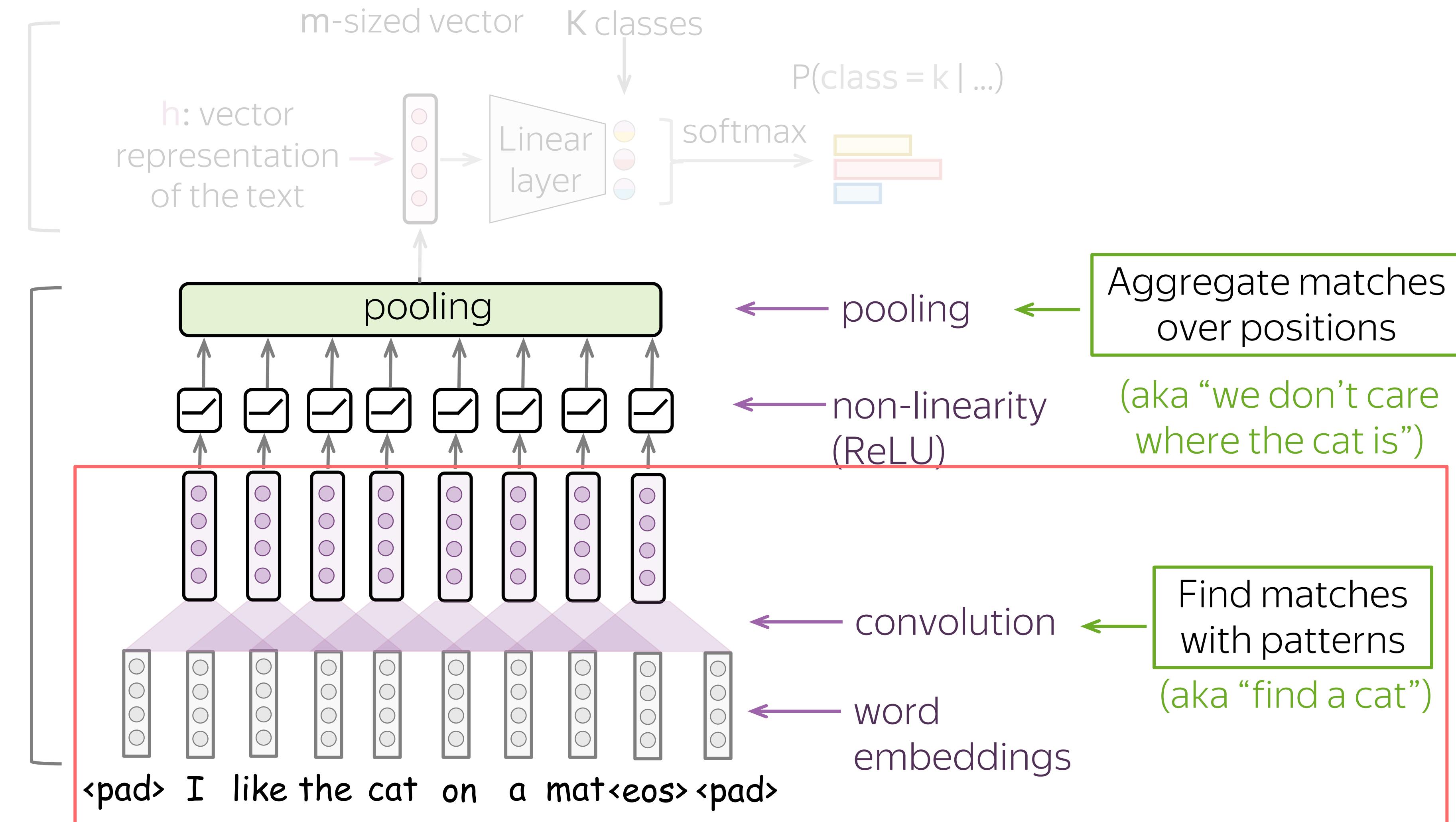
Specific to CNNs:
process text
(document)



A Typical Model: Convolution + Pooling

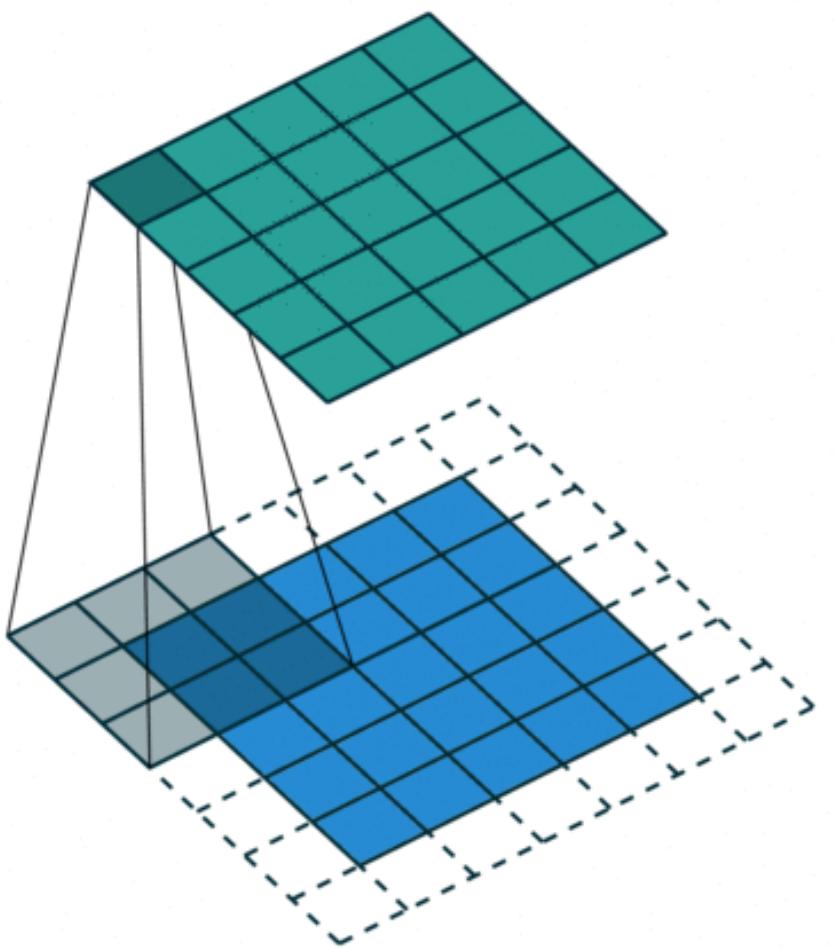
Standard part
(same for all NNs):
get probability
distribution

Specific to CNNs:
process text
(document)



Building Blocks: Convolution

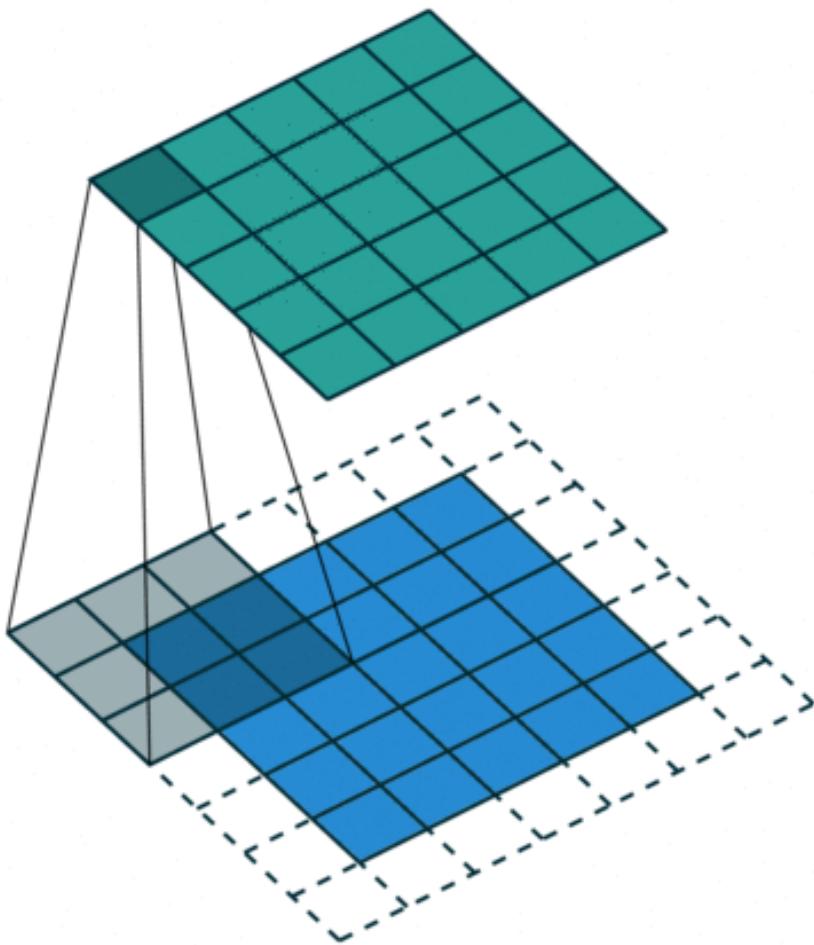
Convolution filter for an image



This gif is from the repo
[https://github.com/vdumoulin/
conv_arithmetic](https://github.com/vdumoulin/conv_arithmetic)

Building Blocks: Convolution

Convolution filter for an image

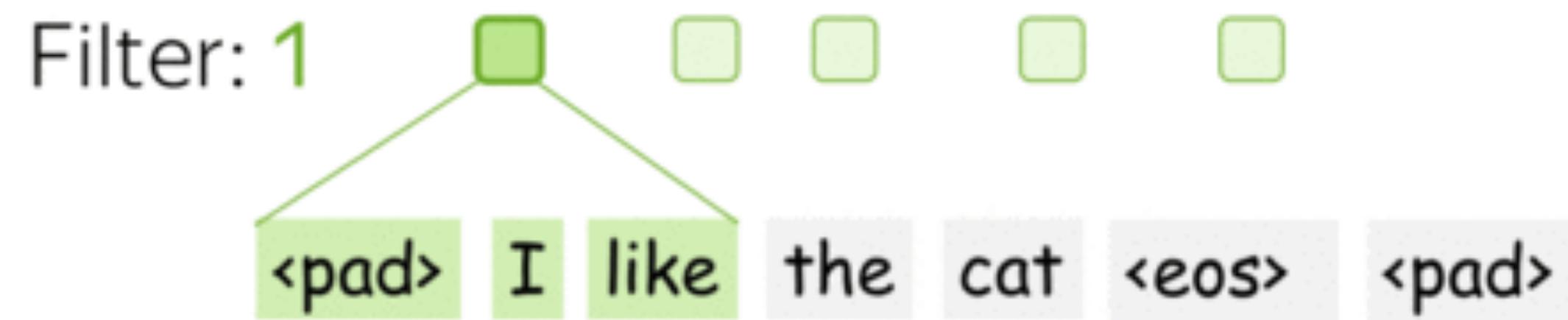


Convolution filter for a text



This gif is from the repo
[https://github.com/vdumoulin/
conv_arithmetic](https://github.com/vdumoulin/conv_arithmetic)

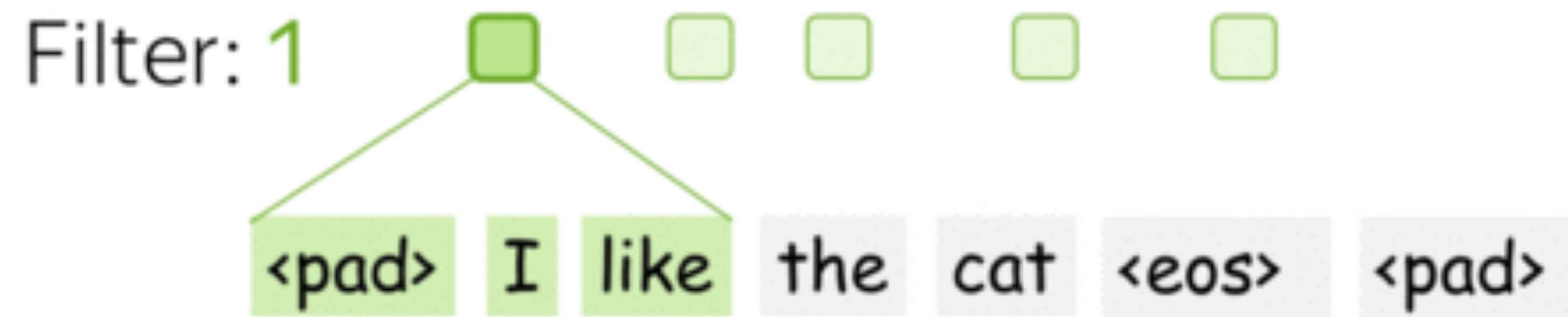
Several Filters – Several Features



$$= \begin{bmatrix} \text{green square} \\ \text{red square} \\ \text{yellow square} \\ \vdots \\ \text{blue square} \end{bmatrix}^T$$

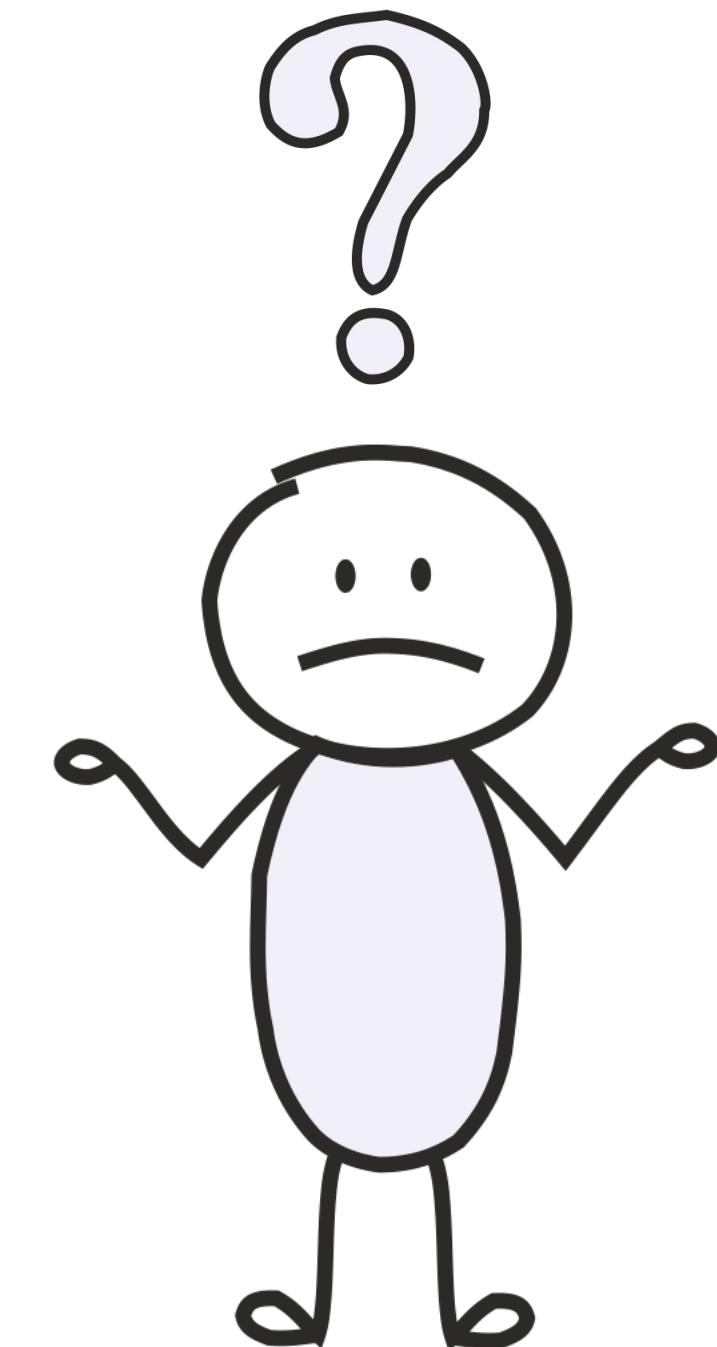
result:
m features

Several Filters – Several Features



$$= \begin{bmatrix} \text{green square} \\ \text{red square} \\ \text{yellow square} \\ \vdots \\ \text{blue square} \end{bmatrix}^T$$

result:
m features



What do these filters learn?

Analyzing Convolutional Filters

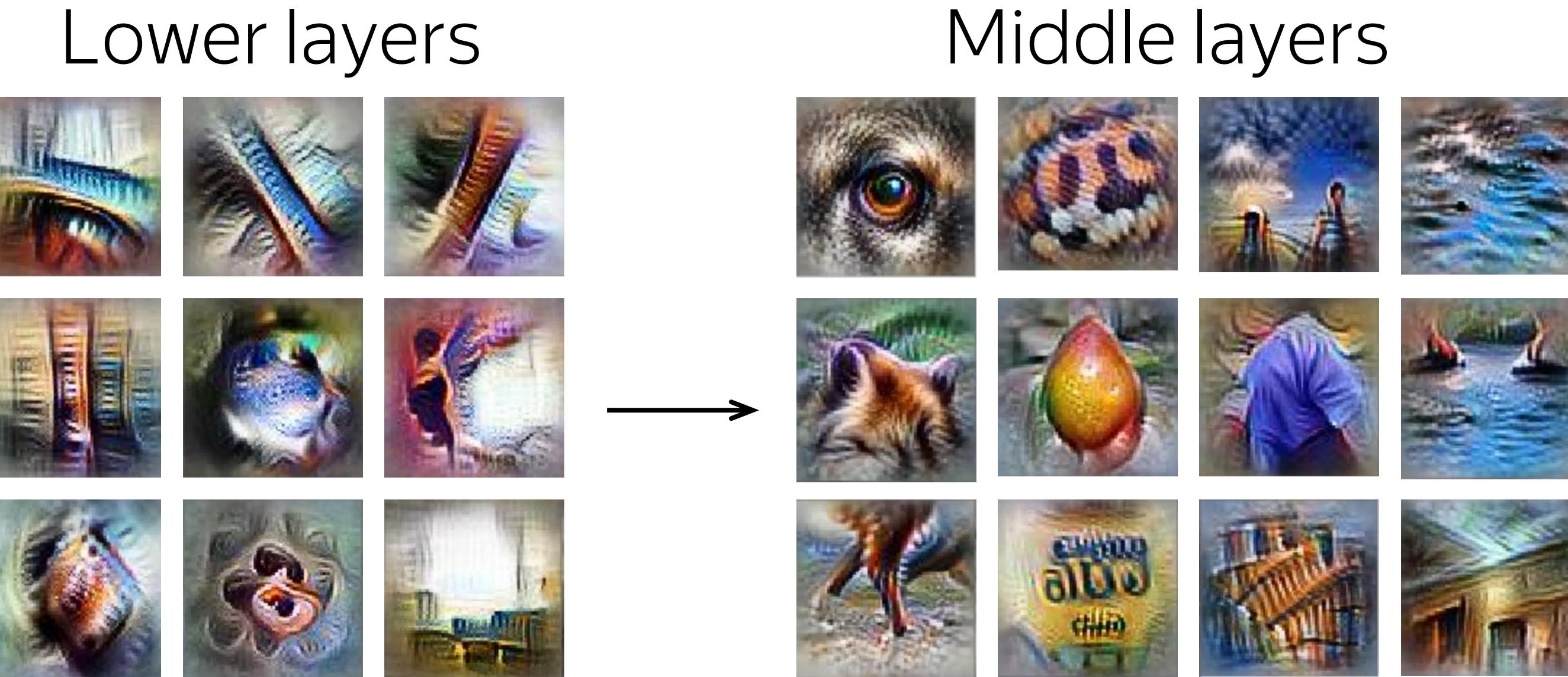
Lower layers



Examples of patterns captured by convolution filters for images.

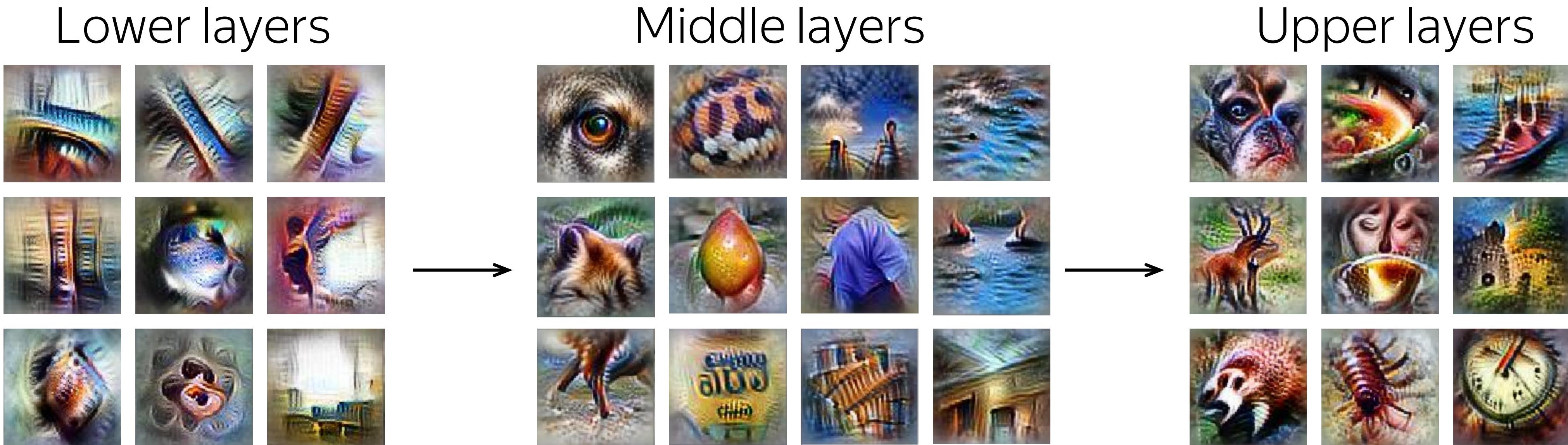
The examples are from [Activation Atlas from distill.pub](#).

Analyzing Convolutional Filters



Examples of patterns captured by convolution filters for images.
The examples are from [Activation Atlas from distill.pub](#).

Analyzing Convolutional Filters

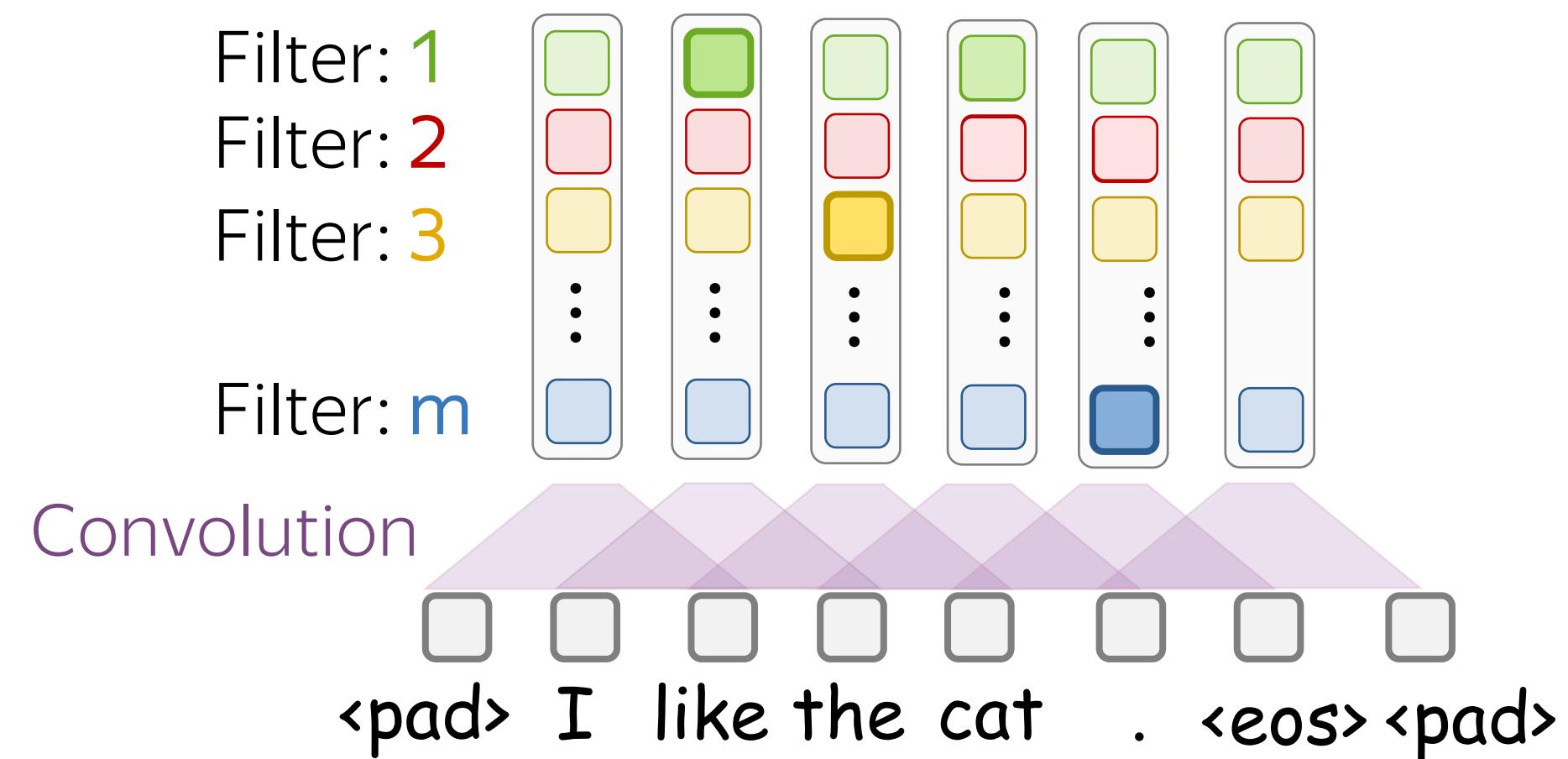


Examples of patterns captured by convolution filters for images.

The examples are from [Activation Atlas from distill.pub](#).

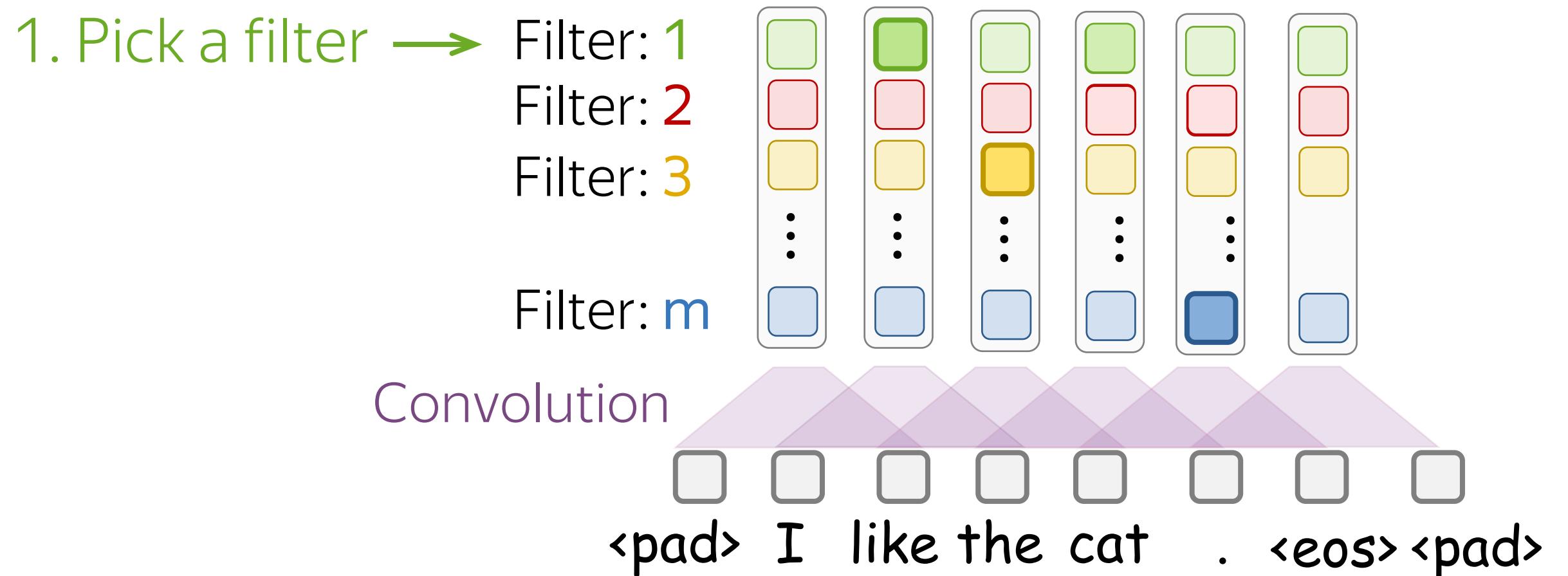
Analyzing Convolutional Filters

- Find which patterns activate neurons



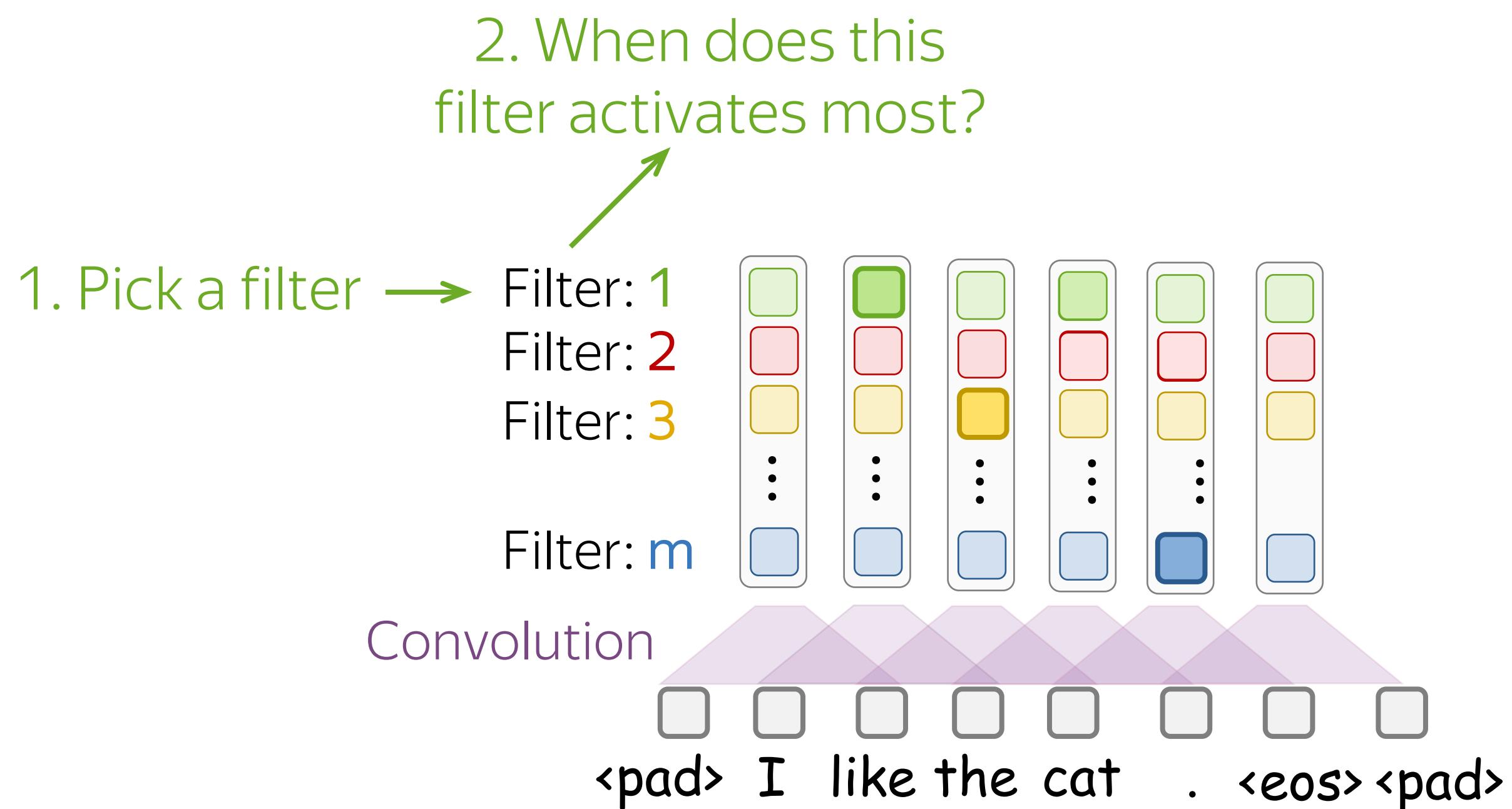
Analyzing Convolutional Filters

- Find which patterns activate neurons



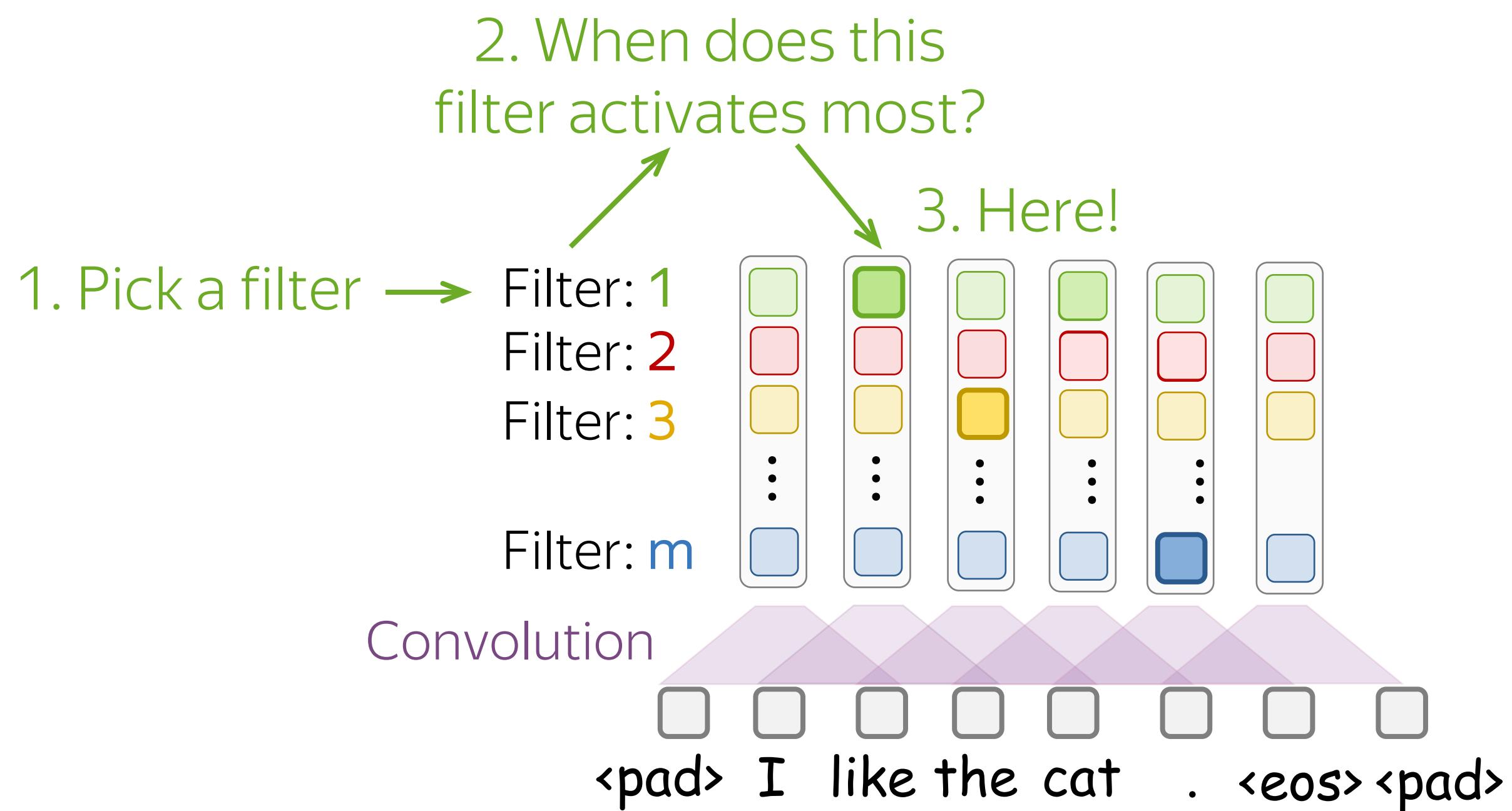
Analyzing Convolutional Filters

- Find which patterns activate neurons



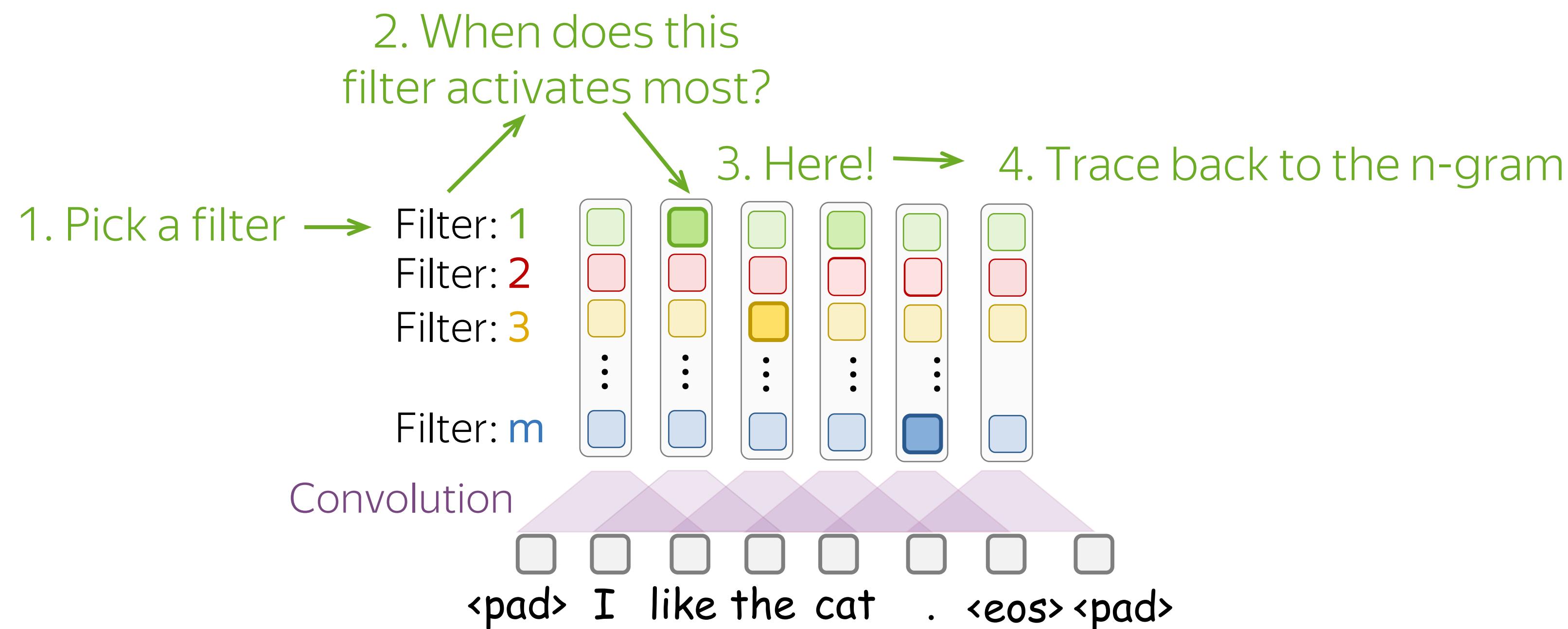
Analyzing Convolutional Filters

- Find which patterns activate neurons



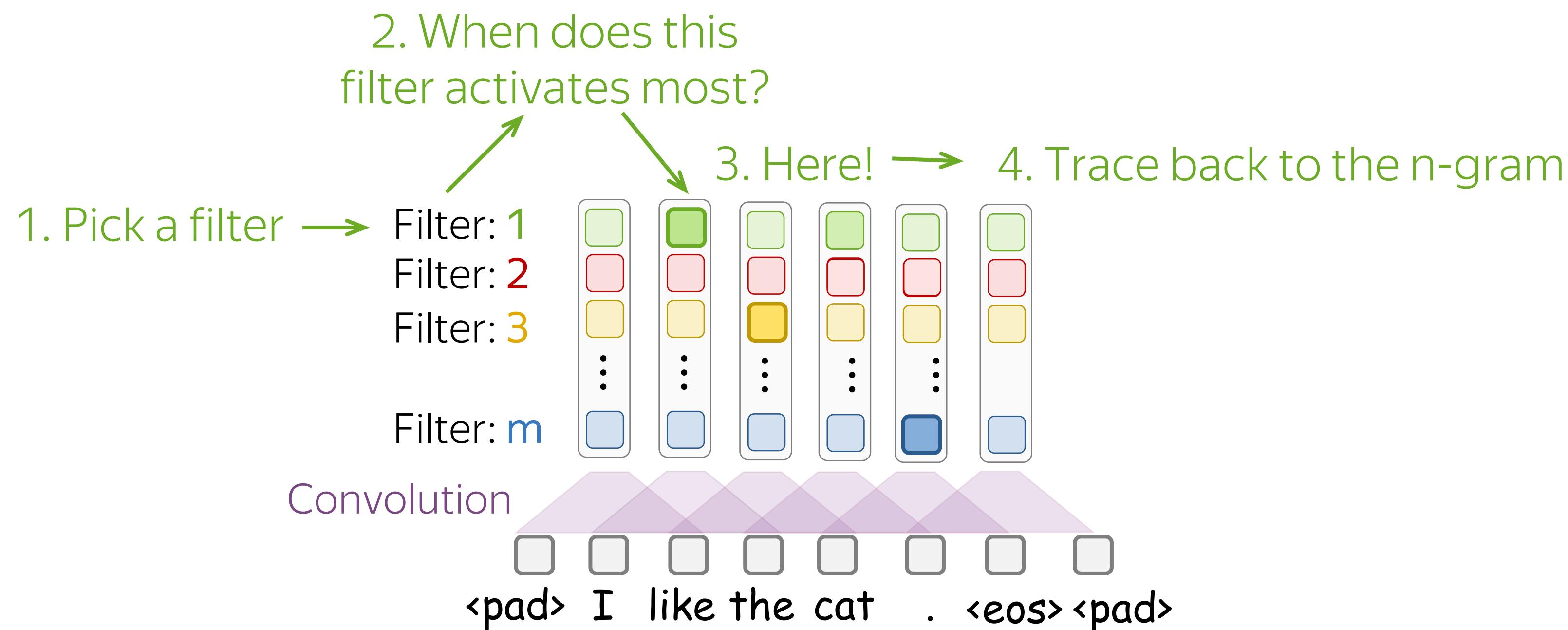
Analyzing Convolutional Filters

- Find which patterns activate neurons



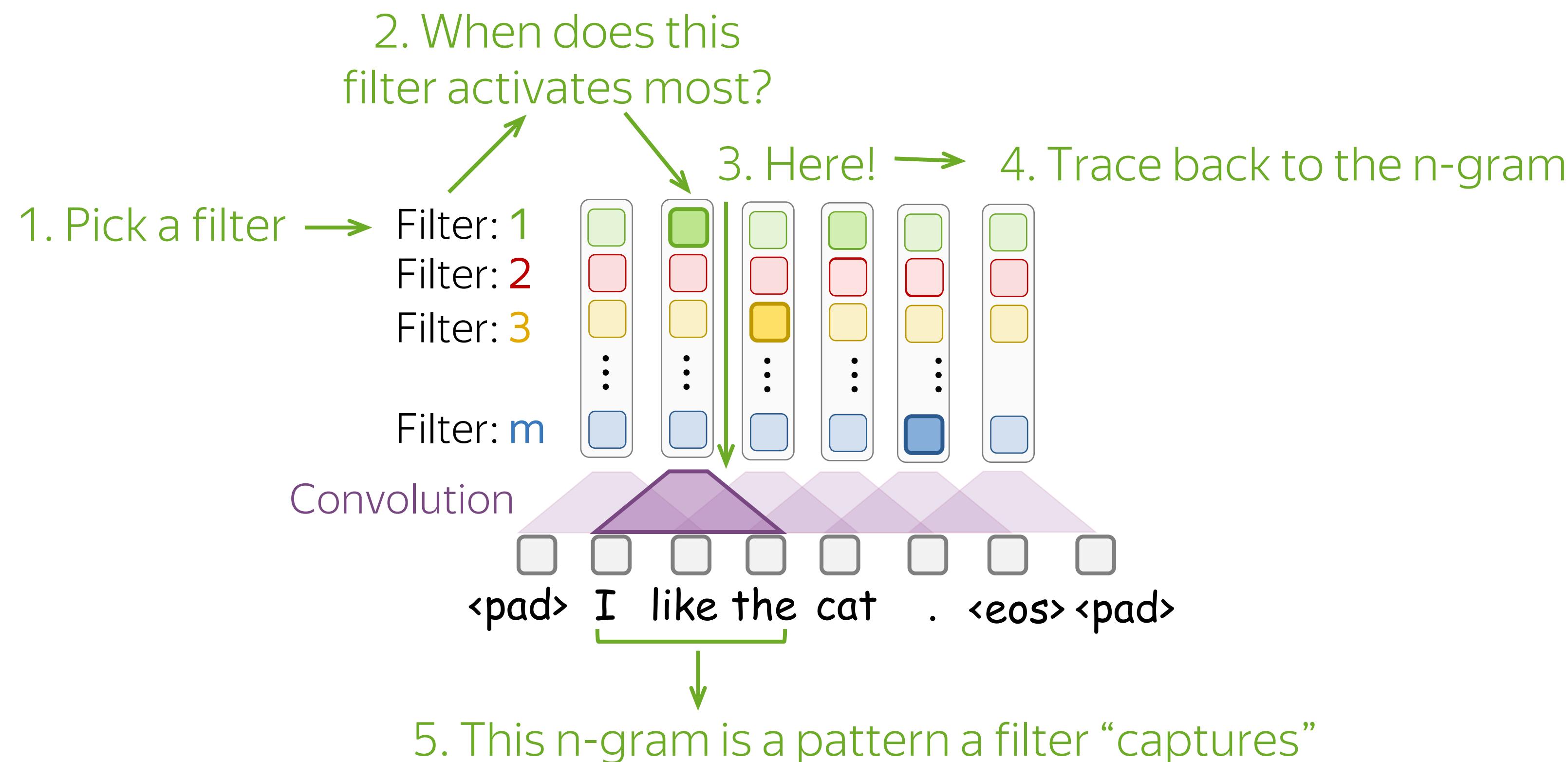
Analyzing Convolutional Filters

- Find which patterns activate neurons



Analyzing Convolutional Filters

- Find which patterns activate neurons



Analyzing Convolutional Filters

filter	Top n-gram	Score
1	poorly designed junk	7.31
2	simply would not	5.75
3	a minor drawback	6.11
4	still working perfect	6.42
5	absolutely gorgeous .	5.36
6	one little hitch	5.72
7	utterly useless .	6.33
8	deserves four stars	5.56
9	a mediocre product	6.91

The example is from the paper [Understanding Convolutional Neural Networks for Text Classification](#).

Analyzing Convolutional Filters

filter	Top n-gram	Score	Top n-grams for filter 4	Score
1	poorly designed junk	7.31		
2	simply would not	5.75		
3	a minor drawback	6.11		
4	still working perfect	6.42	1 still working perfect	6.42
5	absolutely gorgeous .	5.36	2 works - perfect	5.78
6	one little hitch	5.72	3 isolation proves invaluable	5.61
7	utterly useless .	6.33	4 still near perfect	5.6
8	deserves four stars	5.56	5 still working great	5.45
9	a mediocre product	6.91	6 works as good	5.44
			7 still holding strong	5.37

A filter activates for a family of n-grams with similar meaning

The example is from the paper [Understanding Convolutional Neural Networks for Text Classification](#).

What is going to happen:

- Classification →
 - General View
 - Neural approach
 - Convolutional Networks
 - Recurrent Networks
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff

Features:
from hand-crafted
to learned

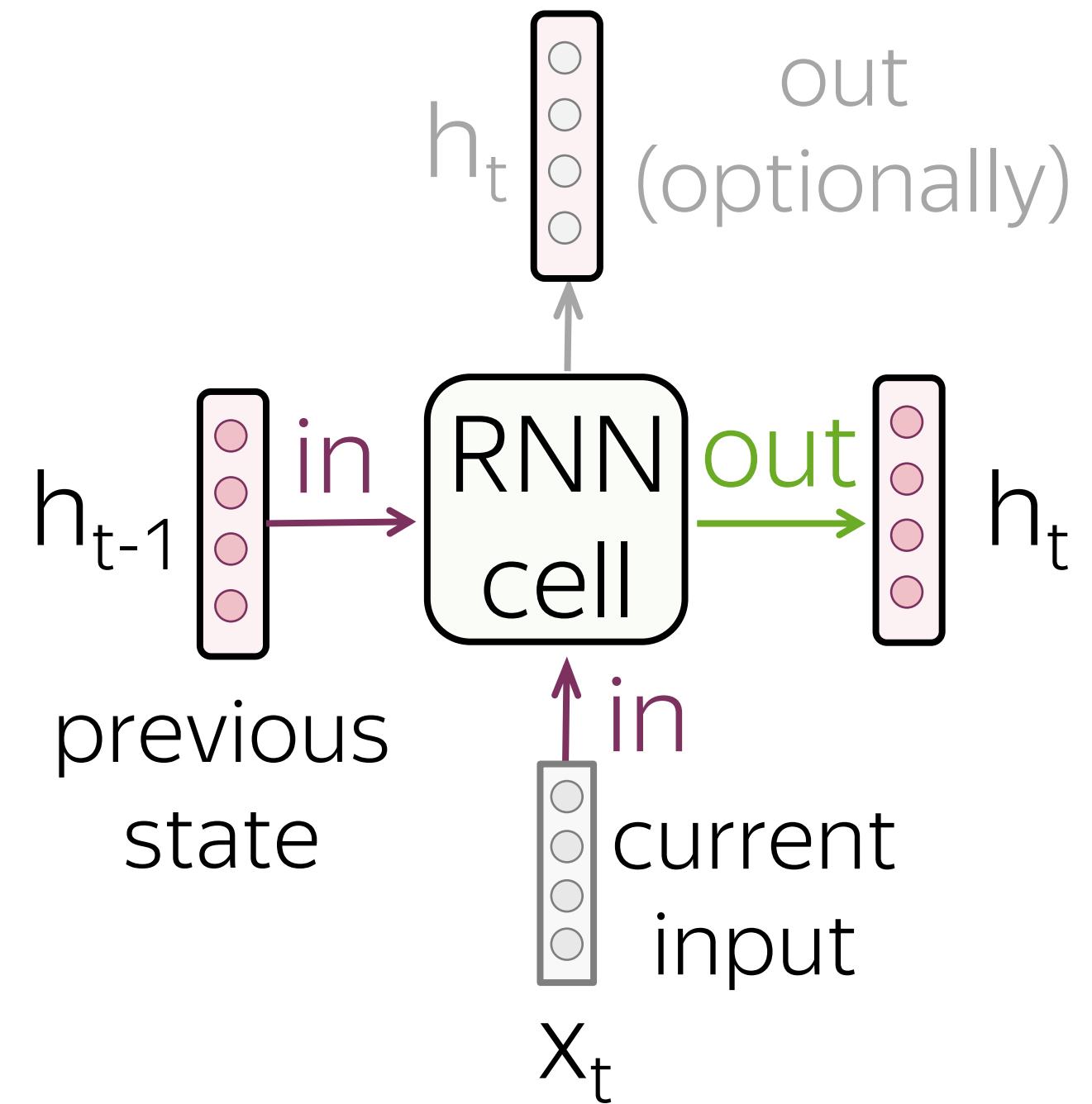
Model:
from task-specific
to generalized

What is going to happen:

- Classification →
 - General View
 - Neural approach
 - Convolutional Networks
 - Recurrent Networks
 - Language Modeling
 - Transfer Learning
 - A Note on Crazy Stuff
- Features:
from hand-crafted
to learned
- Model:
from task-specific
to generalized

Basics: Recurrent Neural Networks

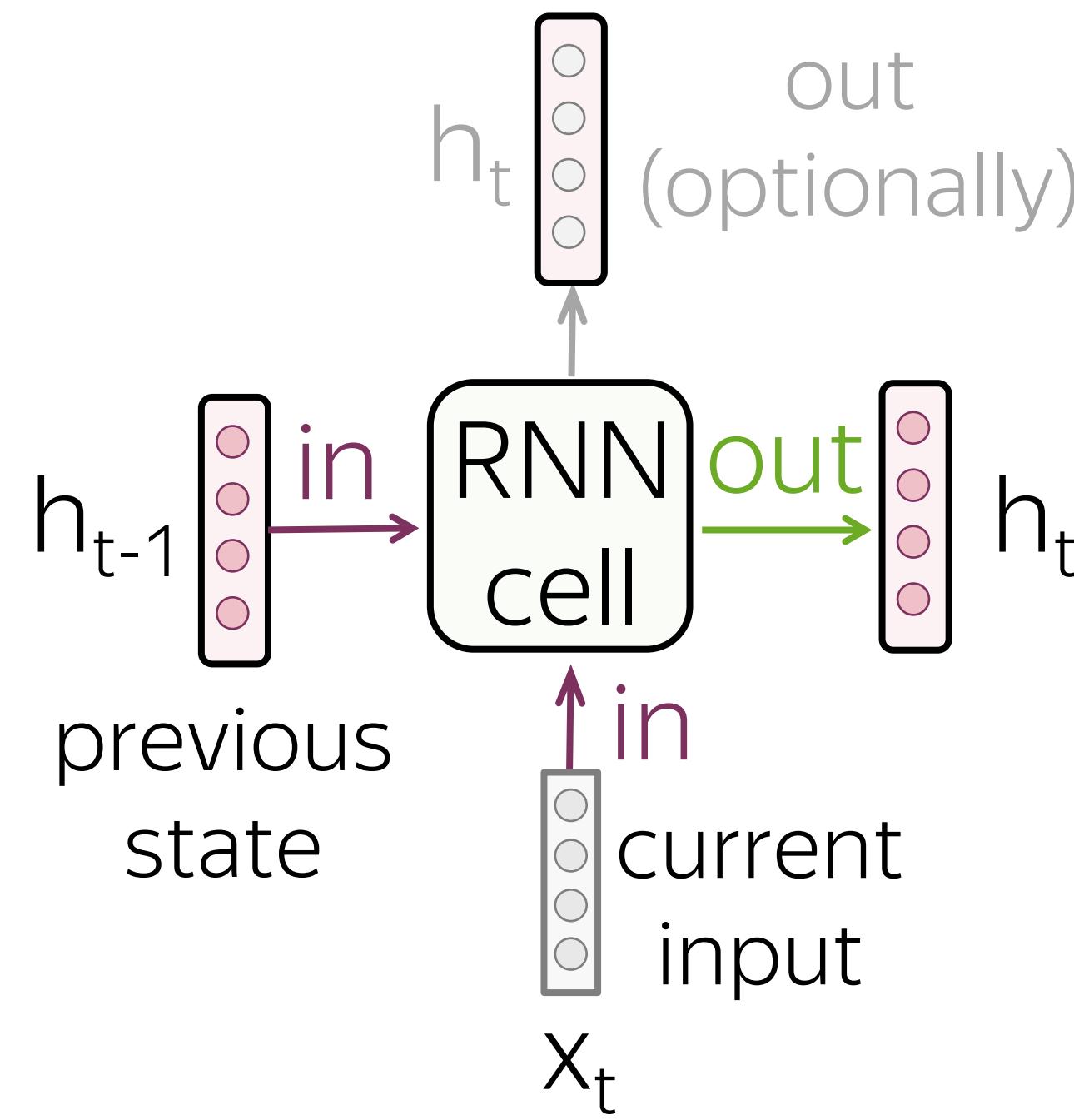
- recurrent cell



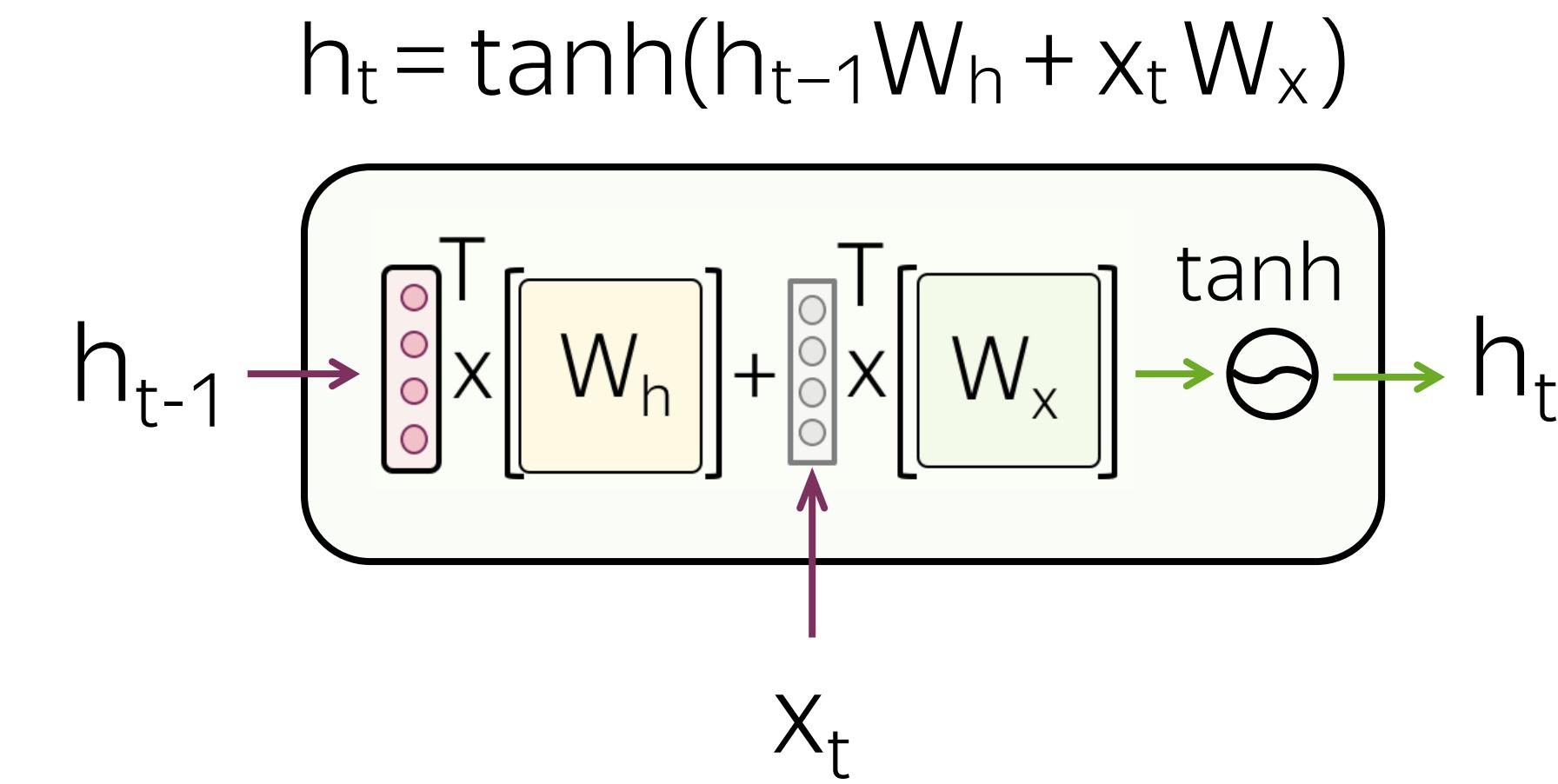
For more details of RNN basics, look at the [Colah's blog post](#).

Basics: Recurrent Neural Networks

- recurrent cell



- vanilla RNN



For more details of RNN basics, look at the [Colah's blog post](#).

Basics: Recurrent Neural Networks

- RNN reads a text



Text: I like the cat on a mat <eos>
not read yet

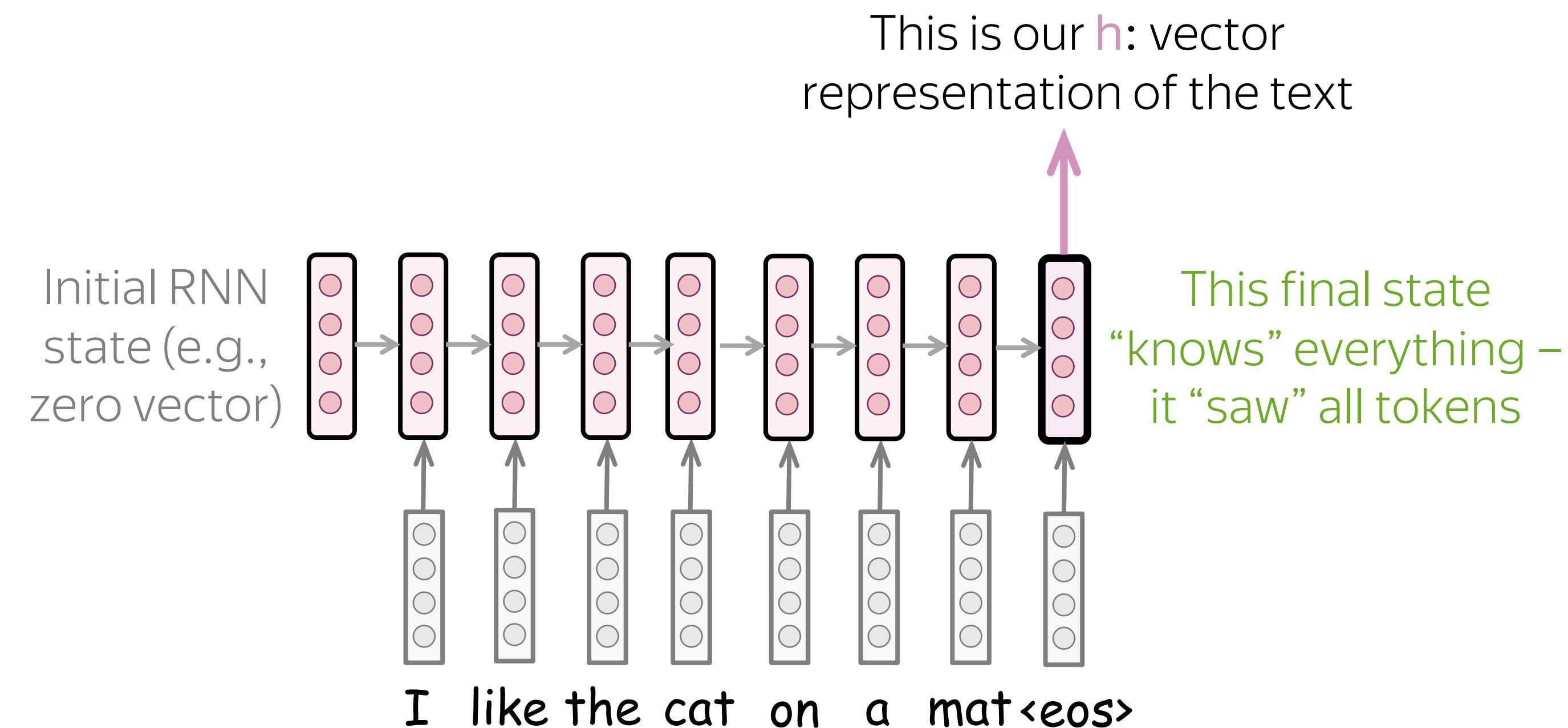
Recurrent Models for Text Classification

We need a model that can produce a **fixed-sized** vector for inputs of **different lengths**.

Recurrent Models for Text Classification

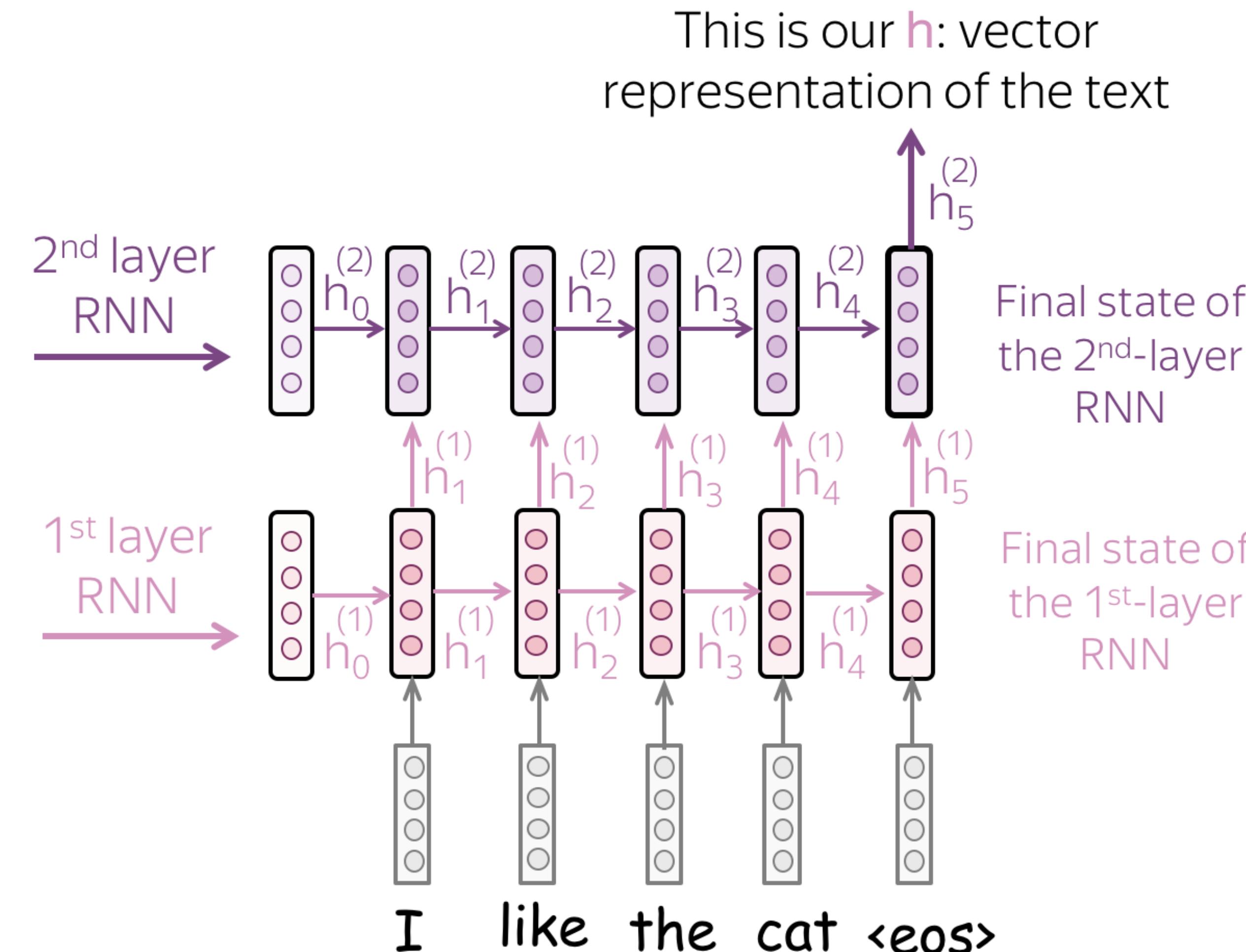
We need a model that can produce a **fixed-sized** vector for inputs of **different lengths**.

- simple: read a text, take the final state



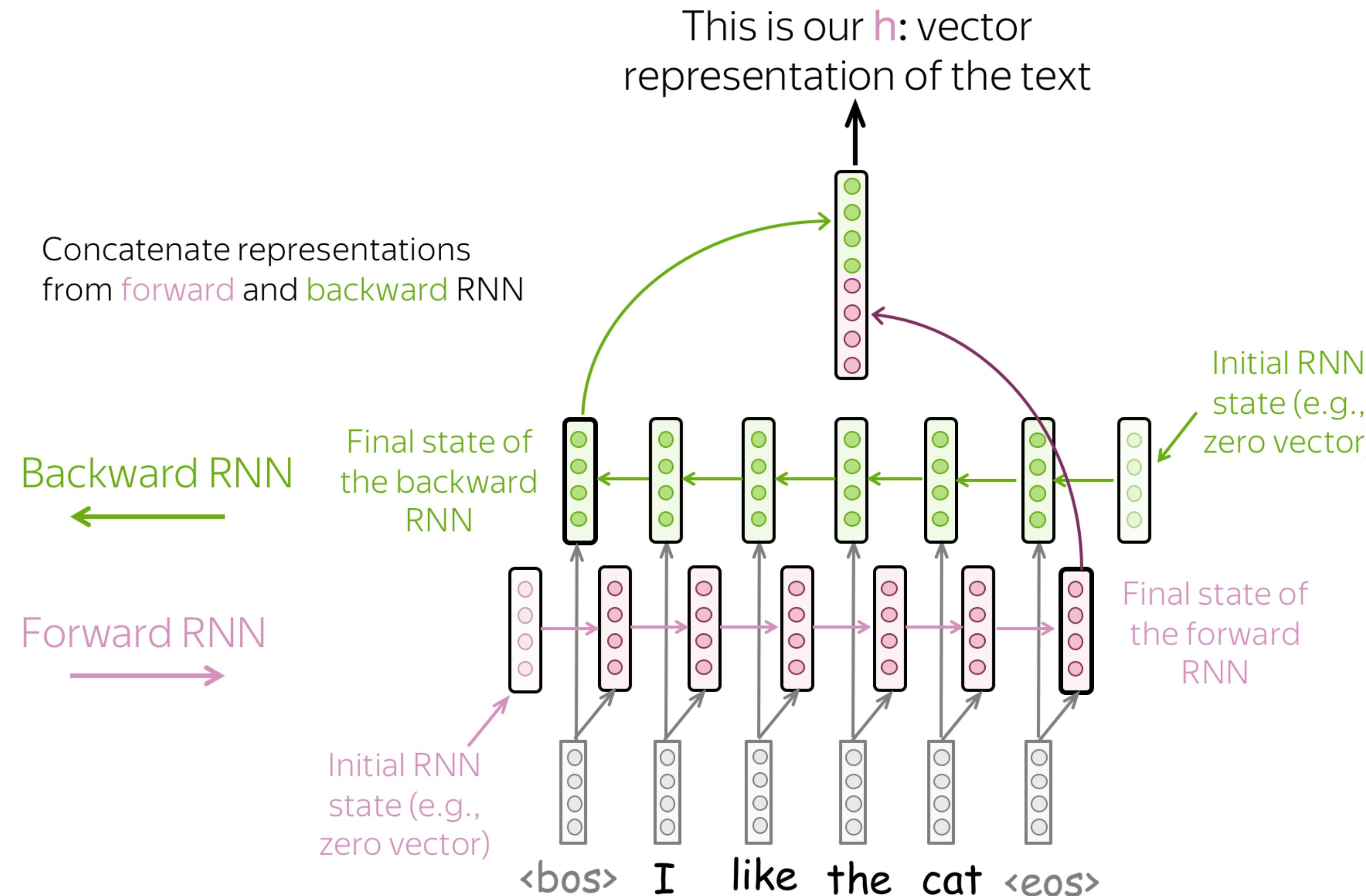
Recurrent Models for Text Classification

- Multiple layers: feed the states from one RNN to the next



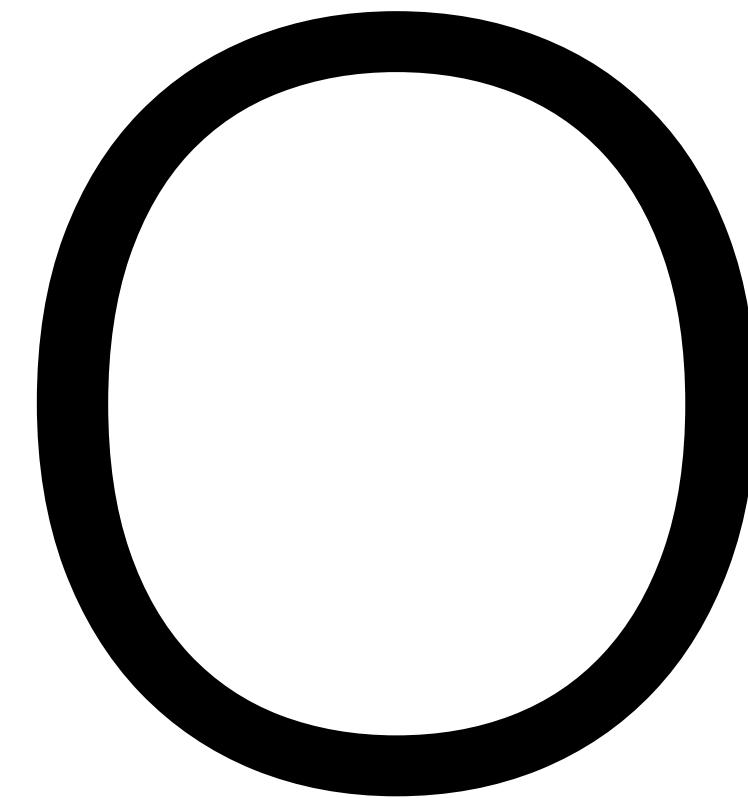
Recurrent Models for Text Classification

- bidirectional: use final states from forward and backward RNNs



Group Exercise: Feel Like an RNN!

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

h

Group Exercise: Feel Like an RNN!

C

Group Exercise: Feel Like an RNN!

e

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

W

Group Exercise: Feel Like an RNN!

h

Group Exercise: Feel Like an RNN!

e

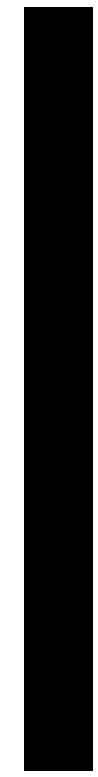
Group Exercise: Feel Like an RNN!

h

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

W

Group Exercise: Feel Like an RNN!

a

Group Exercise: Feel Like an RNN!

S

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

S

Group Exercise: Feel Like an RNN!

i

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

y

Group Exercise: Feel Like an RNN!

e

Group Exercise: Feel Like an RNN!

a

Group Exercise: Feel Like an RNN!

r

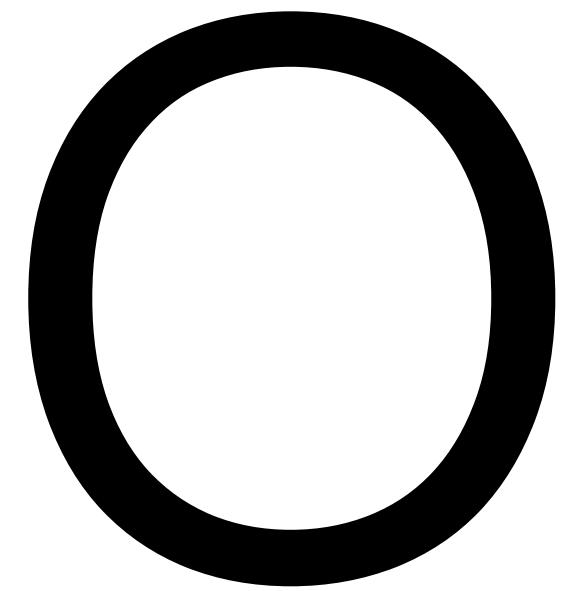
Group Exercise: Feel Like an RNN!

S

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!



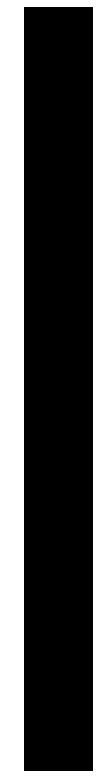
Group Exercise: Feel Like an RNN!

d

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

S

Group Exercise: Feel Like an RNN!

a

Group Exercise: Feel Like an RNN!

W

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

a

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

m

Group Exercise: Feel Like an RNN!

a

Group Exercise: Feel Like an RNN!

g

Group Exercise: Feel Like an RNN!

h

Group Exercise: Feel Like an RNN!

i

Group Exercise: Feel Like an RNN!

f

Group Exercise: Feel Like an RNN!

i

Group Exercise: Feel Like an RNN!

C

Group Exercise: Feel Like an RNN!

e

Group Exercise: Feel Like an RNN!

h

Group Exercise: Feel Like an RNN!

t

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

p

Group Exercise: Feel Like an RNN!

i

Group Exercise: Feel Like an RNN!

C

Group Exercise: Feel Like an RNN!

t

Group Exercise: Feel Like an RNN!

U

Group Exercise: Feel Like an RNN!

r

Group Exercise: Feel Like an RNN!

e

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

i

Group Exercise: Feel Like an RNN!

h

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

t

Group Exercise: Feel Like an RNN!

h

Group Exercise: Feel Like an RNN!

e

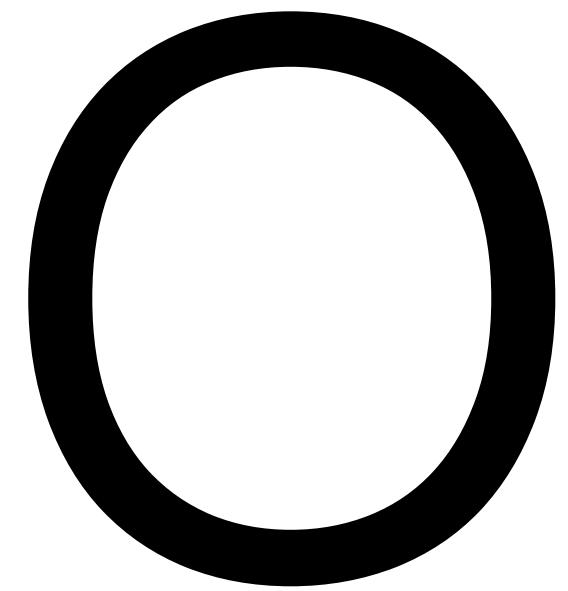
Group Exercise: Feel Like an RNN!



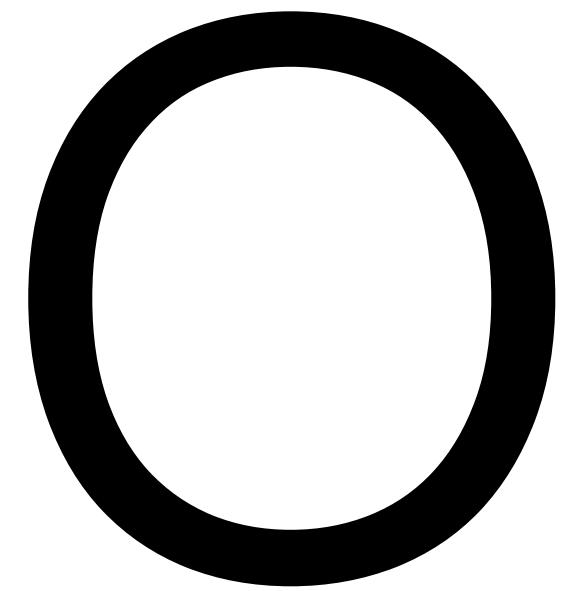
Group Exercise: Feel Like an RNN!

b

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

K

Group Exercise: Feel Like an RNN!

,

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

C

Group Exercise: Feel Like an RNN!

a

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

e

Group Exercise: Feel Like an RNN!

d

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

T

Group Exercise: Feel Like an RNN!

r

Group Exercise: Feel Like an RNN!

u

Group Exercise: Feel Like an RNN!

e

Group Exercise: Feel Like an RNN!



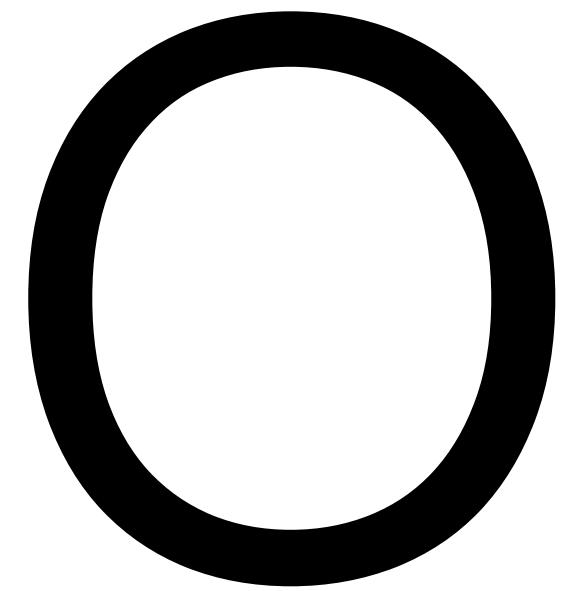
Group Exercise: Feel Like an RNN!

S

Group Exercise: Feel Like an RNN!

t

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

r

Group Exercise: Feel Like an RNN!

i

Group Exercise: Feel Like an RNN!

e

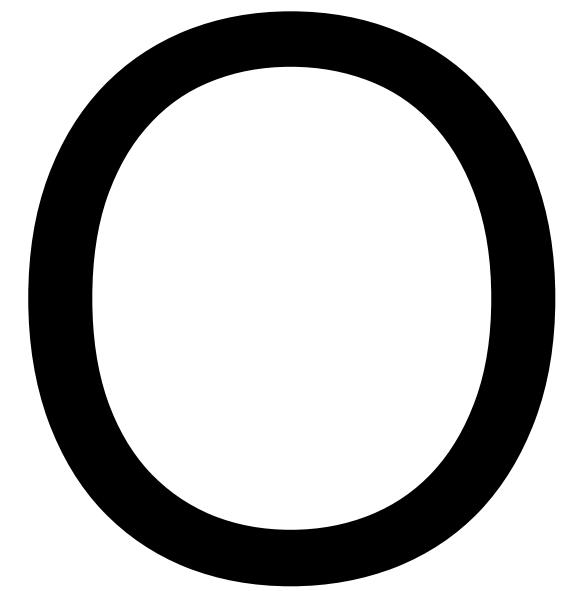
Group Exercise: Feel Like an RNN!

S

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

f

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

h

Group Exercise: Feel Like an RNN!

a

Group Exercise: Feel Like an RNN!

t

Group Exercise: Feel Like an RNN!

u

Group Exercise: Feel Like an RNN!

r

Group Exercise: Feel Like an RNN!

e

Group Exercise: Feel Like an RNN!

,

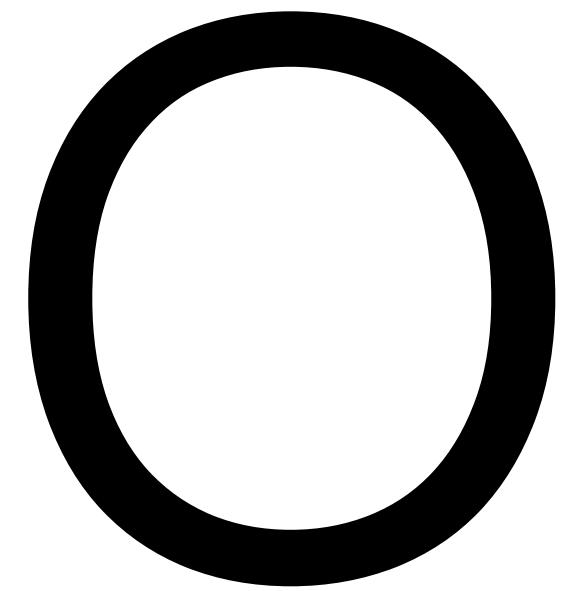
Group Exercise: Feel Like an RNN!

a

Group Exercise: Feel Like an RNN!

b

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

U

Group Exercise: Feel Like an RNN!

t

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

t

Group Exercise: Feel Like an RNN!

h

Group Exercise: Feel Like an RNN!

e

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

p

Group Exercise: Feel Like an RNN!

r

Group Exercise: Feel Like an RNN!

i

Group Exercise: Feel Like an RNN!

m

Group Exercise: Feel Like an RNN!

e

Group Exercise: Feel Like an RNN!

V

Group Exercise: Feel Like an RNN!

a

Group Exercise: Feel Like an RNN!



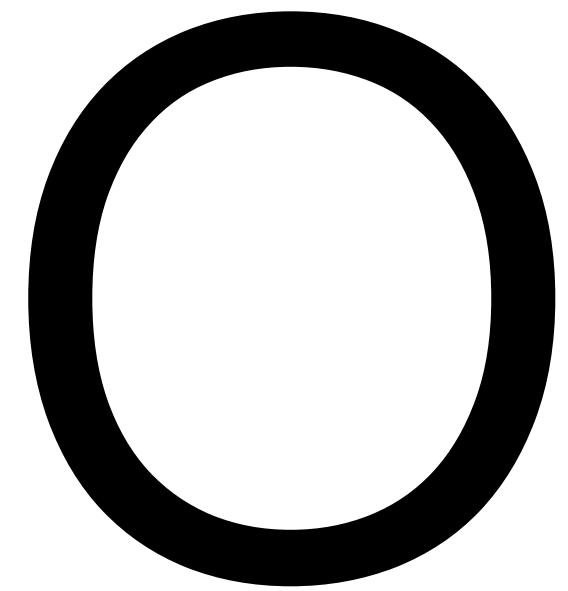
Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

f

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

r

Group Exercise: Feel Like an RNN!

e

Group Exercise: Feel Like an RNN!

S

Group Exercise: Feel Like an RNN!

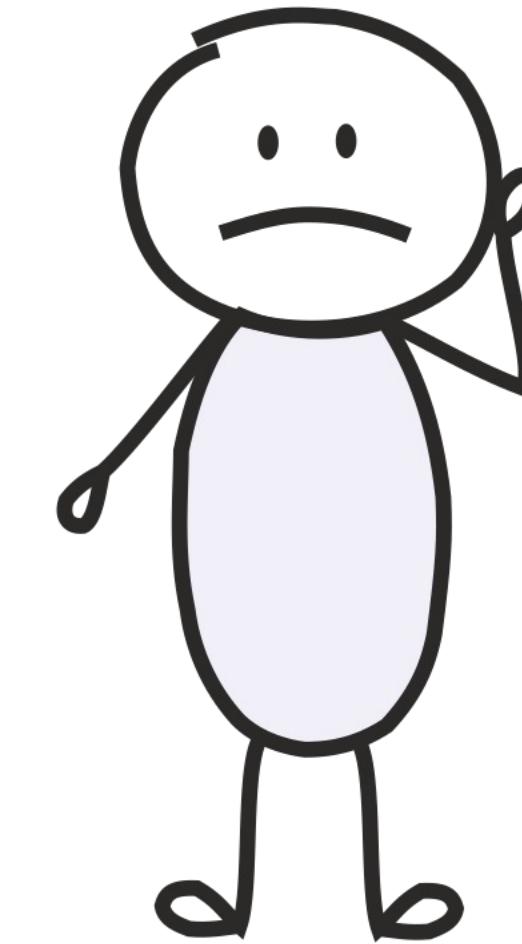
t

Group Exercise: Feel Like an RNN!



Group Exercise: Feel Like an RNN!

How are you feeling?..



Group Exercise: Feel Like an RNN!

Once when I was six years old I saw a magnificent picture in a book, called True Stories from Nature, about the primeval forest.

From “The Little Prince” by Antoine de Saint-Exupéry

What is going to happen:

- Classification →
 - General View
 - Neural approach
 - Convolutional Networks
 - Recurrent Networks
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff

Features:
from hand-crafted
to learned

Model:
from task-specific
to generalized

What is going to happen:

- Classification
 - Language Modeling
 - Transfer Learning
 - A Note on Crazy Stuff
- Features:
from hand-crafted
to learned
- Model:
from task-specific
to generalized

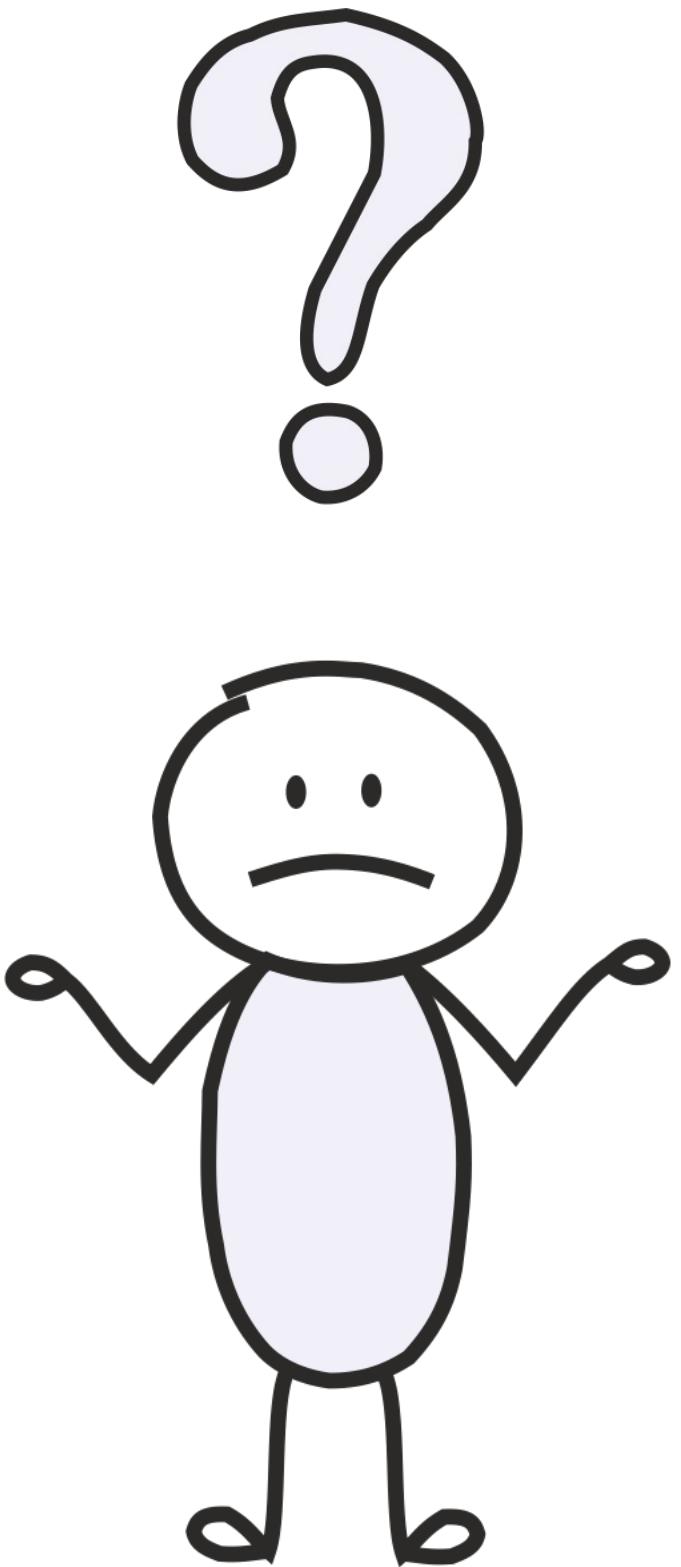
What is going to happen:

- Classification
 - Language Modeling
 - Transfer Learning
 - A Note on Crazy Stuff
- Features:
from hand-crafted
to learned
- Model:
from task-specific
to generalized

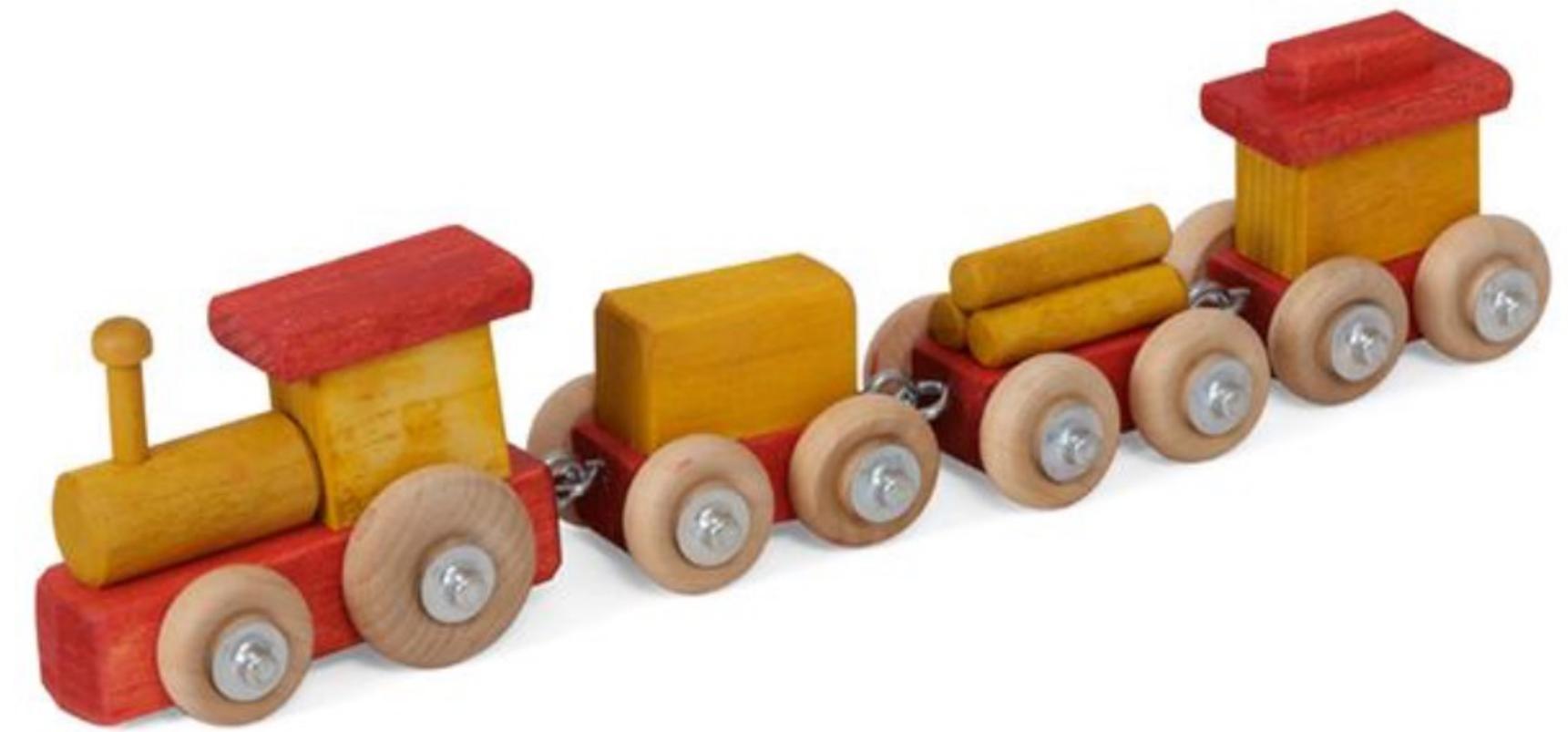
What is going to happen:

- Classification
 - Language Modeling →
 - General View
 - Generating Texts
 - Transfer Learning
 - A Note on Crazy Stuff
- Features:
from hand-crafted
to learned
- Model:
from task-specific
to generalized

What is a Model of Language?



Models of a Train



Models of a Train

- have some properties of trains (look like ones)
- can behave similarly to trains

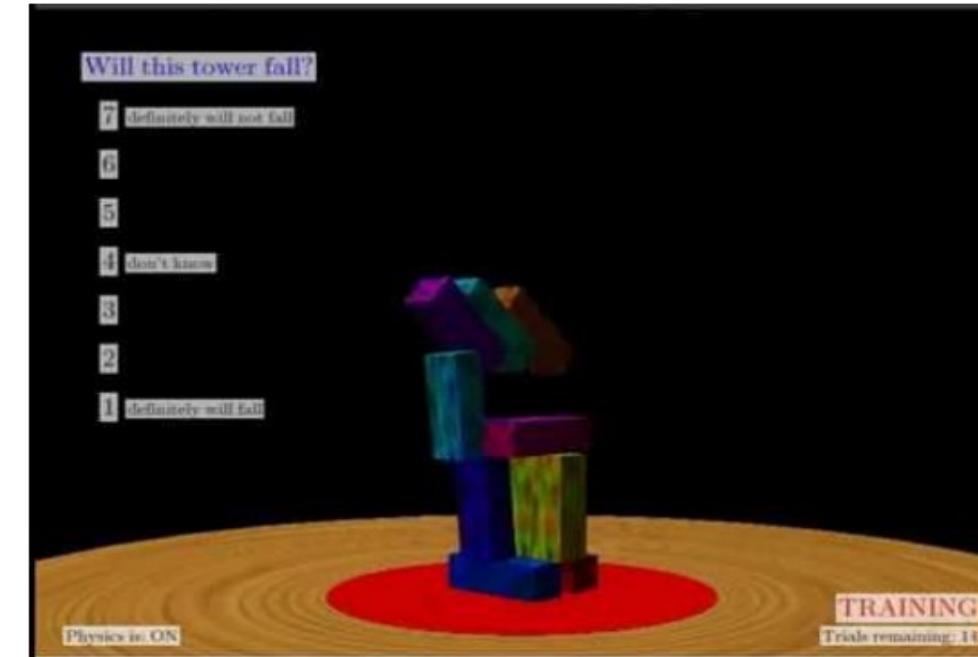


Models of a Train

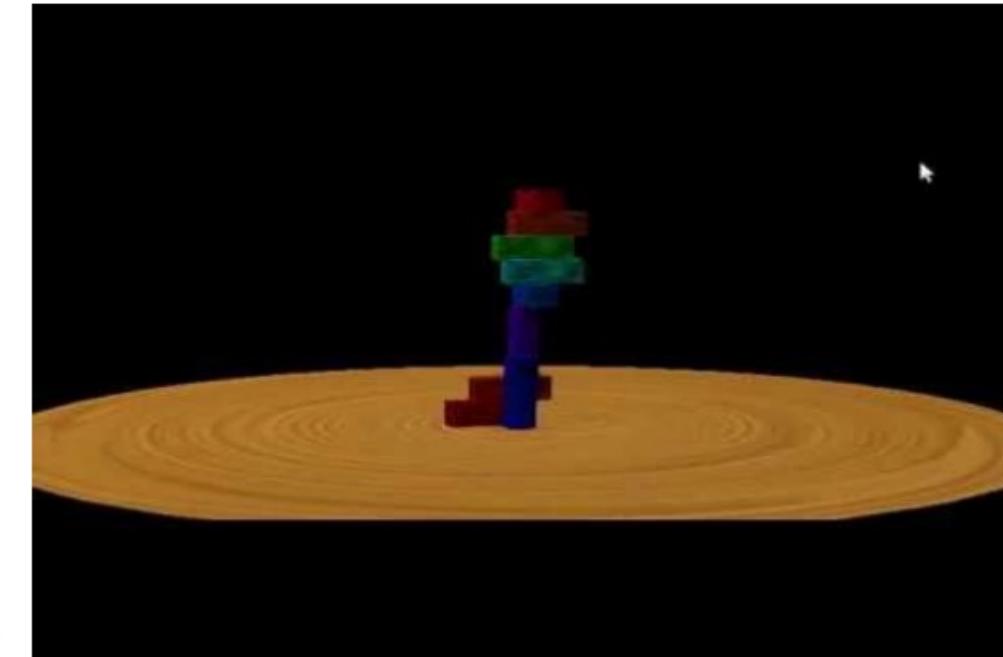
- have some properties of trains (look like ones)
- can behave similarly to trains
- good models have more of the above



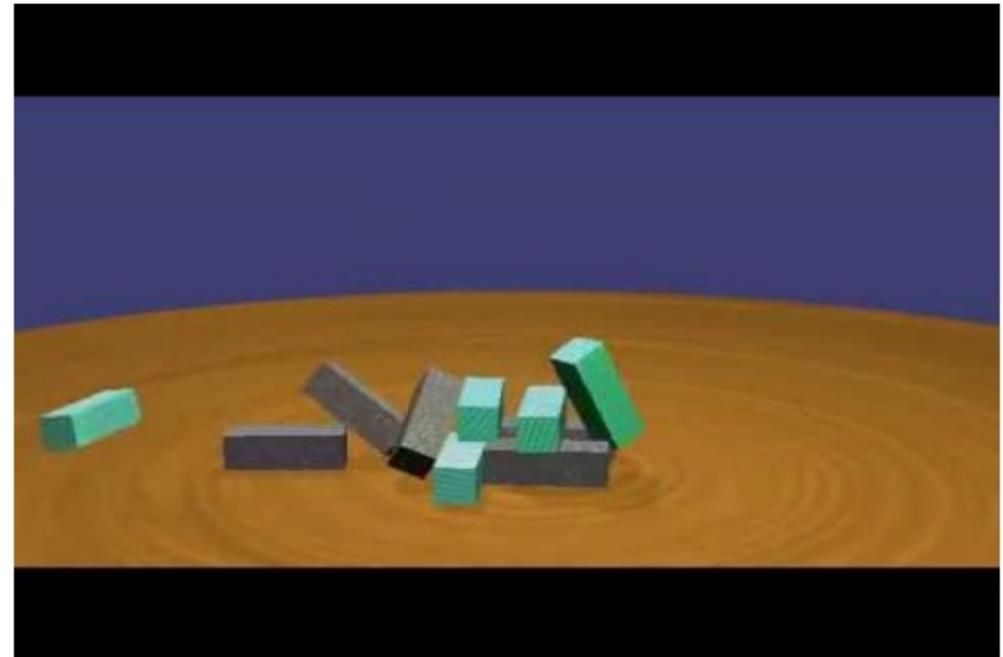
Models of a Physical World



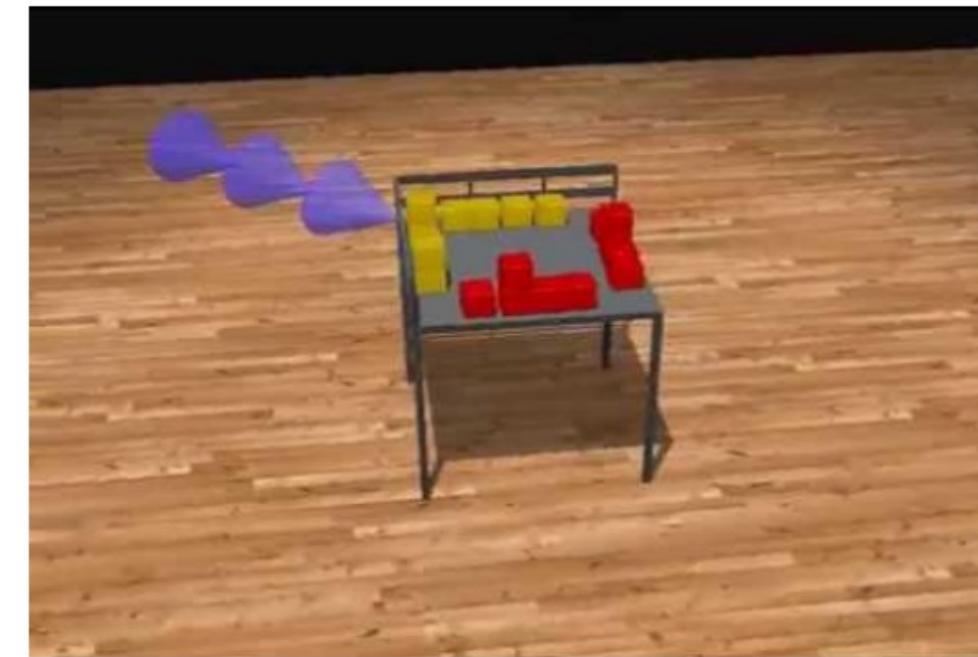
Will it fall?



In which direction?



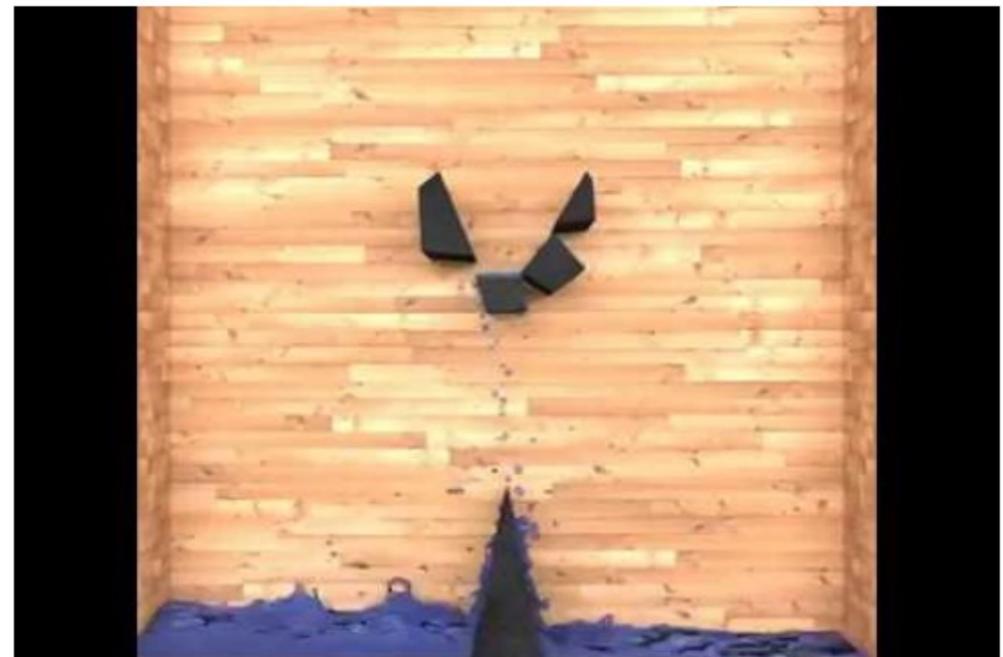
Different masses



Complex scenes



Infer the mass

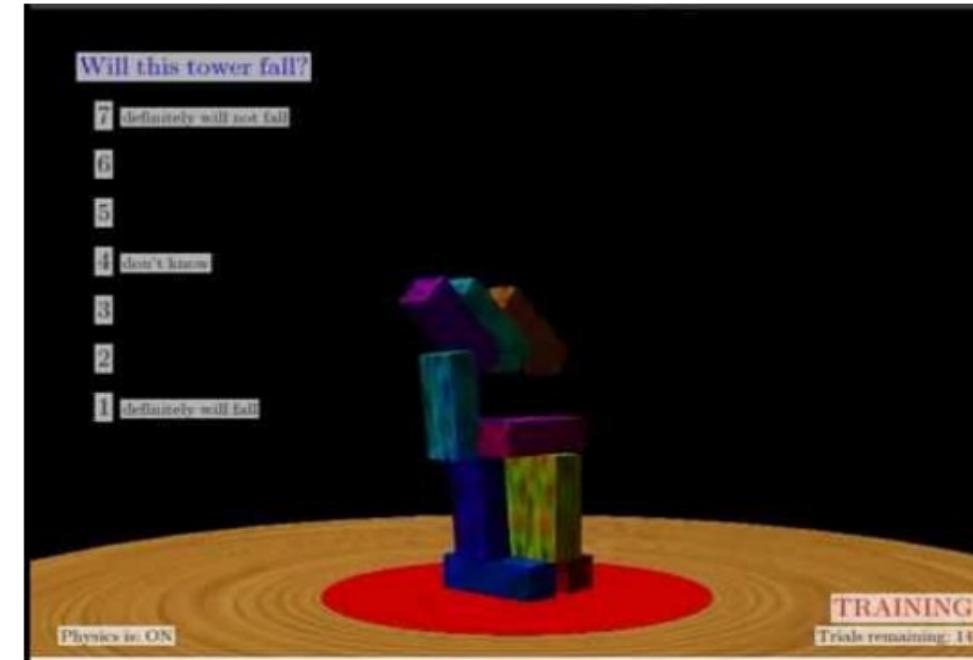


Predict fluids

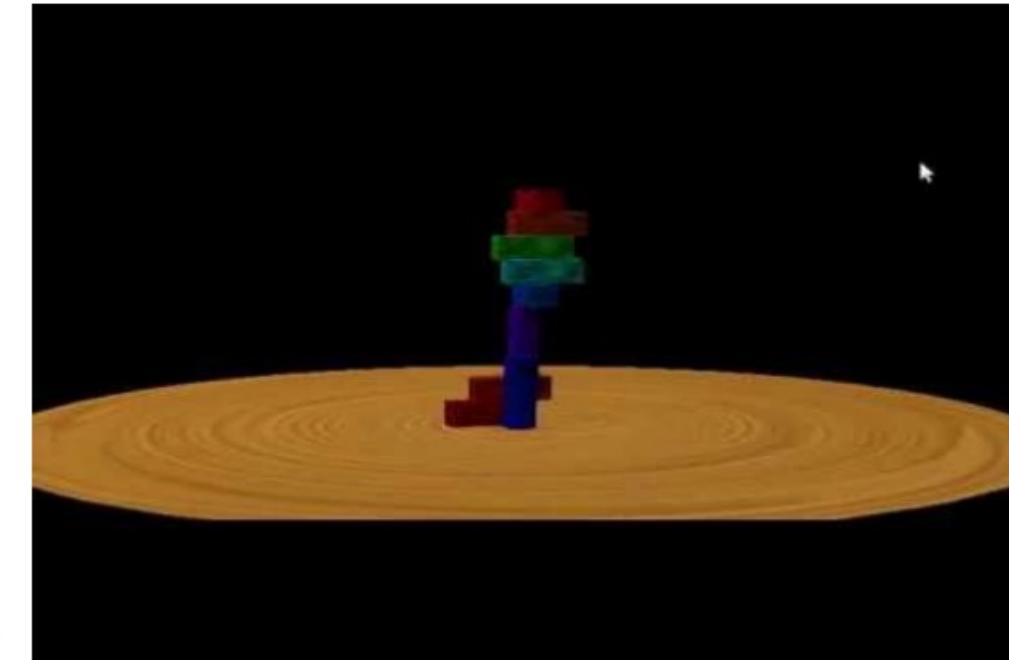
The picture is from the presentation by Peter Battaglia: <http://phys.csail.mit.edu/talks/battaglia.pdf>

Models of a Physical World

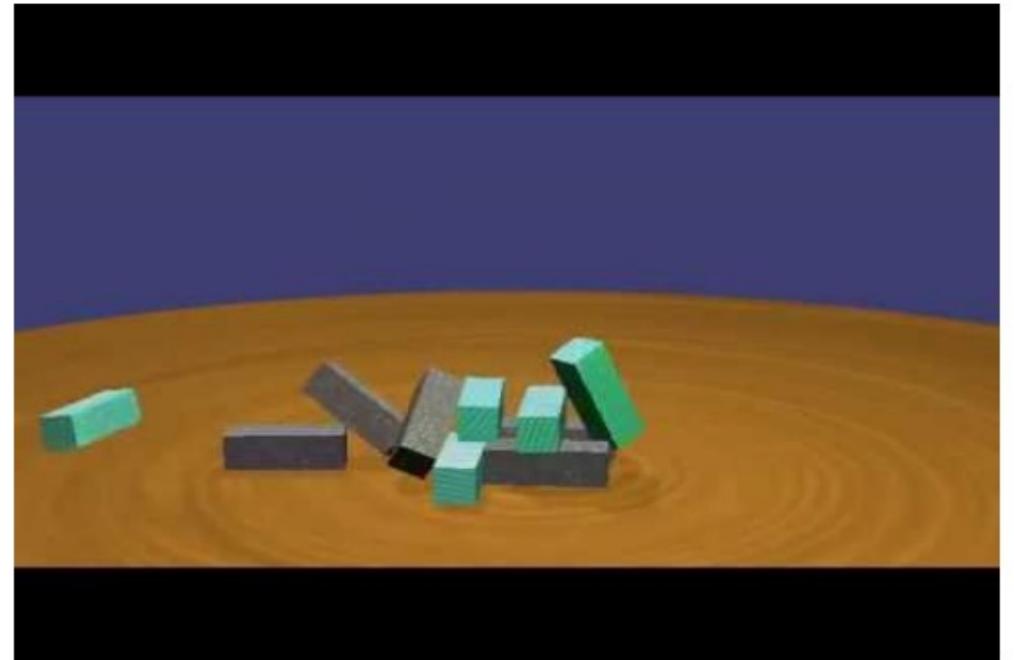
- understand which events are in better agreement with the world, which are more likely



Will it fall?

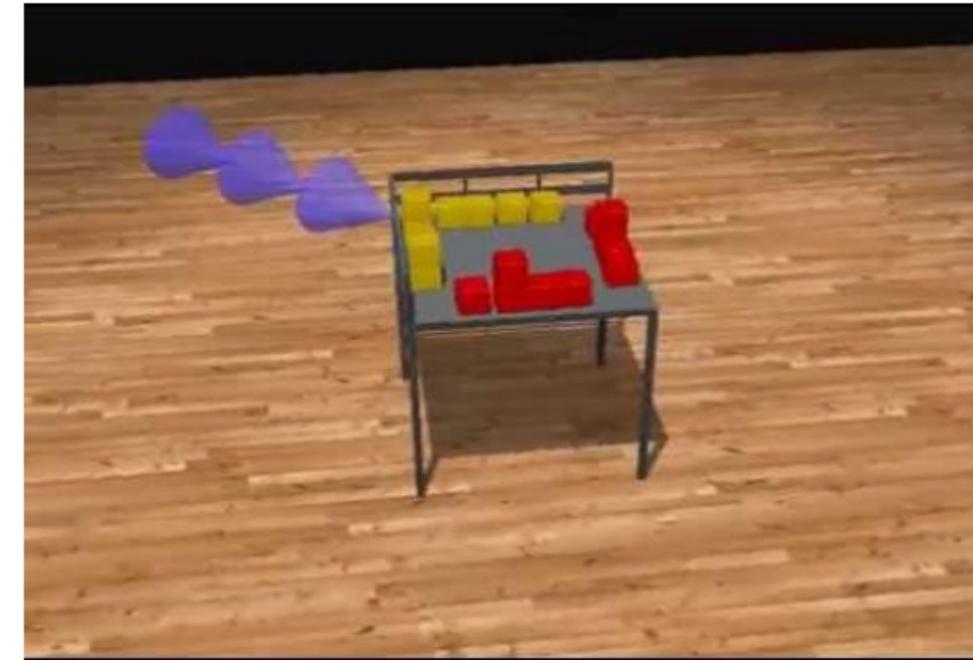


In which direction?



Different masses

- can predict what happens given some “context”



Complex scenes



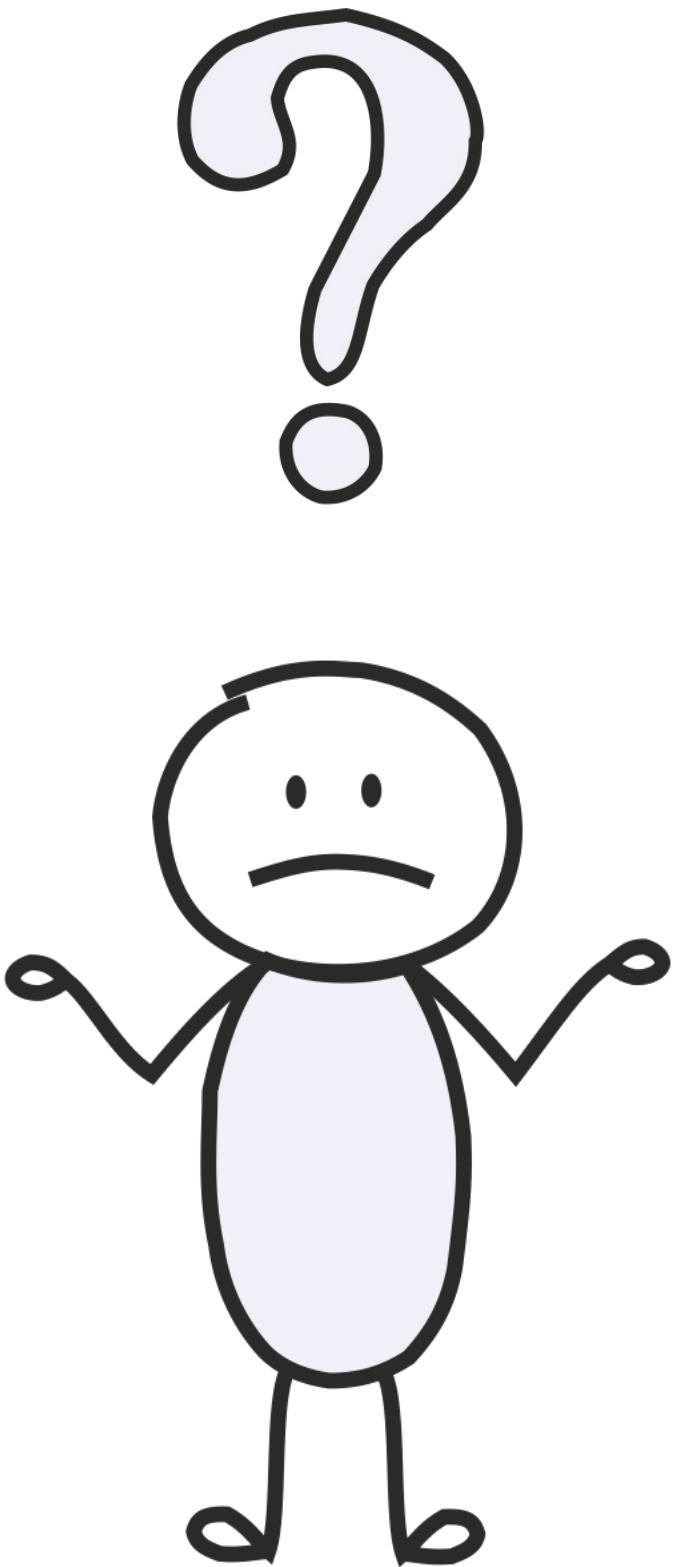
Infer the mass



Predict fluids

The picture is from the presentation by Peter Battaglia: <http://phys.csail.mit.edu/talks/battaglia.pdf>

What is a Model of Language?



Model of a Language?..

The intuition is exactly the same!

What is different, is the notion of an event: for language, an event is a linguistic unit (text, sentence, token, symbol).

Language Models (LMs) estimate the probability of different linguistic units: symbols, tokens, token sequences.

How can this be useful?

We deal with Language Models every day!

Web search engine / ...

I saw a cat|

I saw a cat on the chair

I saw a cat running after a dog

I saw a cat in my dream

I saw a cat book

How can this be useful?

We deal with Language Models every day!

Translation service / mail agent / ...

I saw a ca
car ←

How can this be useful?

We deal with Language Models every day!

Translation service / mail agent / ...

I saw a catt

Probably you meant I saw a cat

How can this be useful?

We deal with Language Models every day!

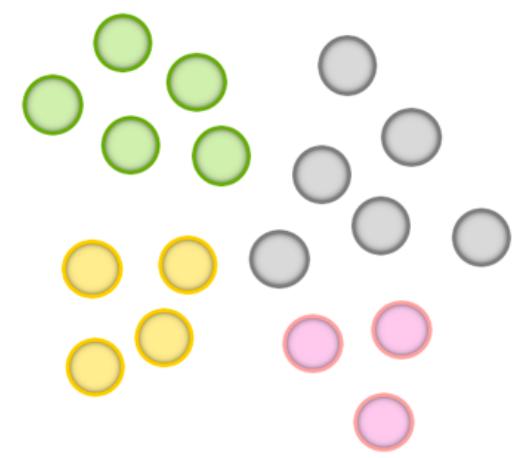
Keyboard / mail agent / ...

I saw a catt

cat
car

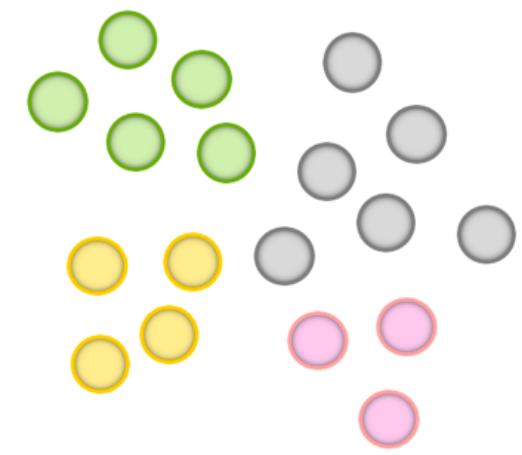
How likely is a sentence to appear in a language?

What is the probability
to pick a green ball?



How likely is a sentence to appear in a language?

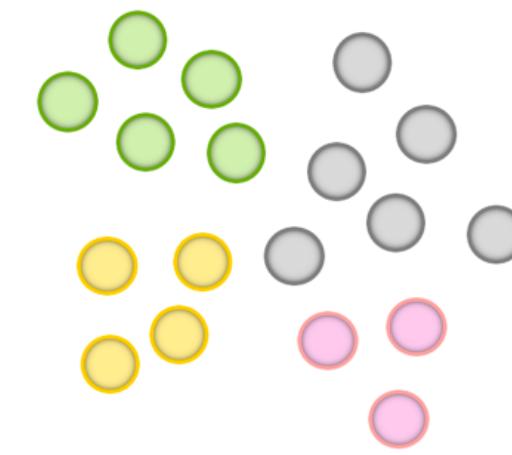
What is the probability
to pick a green ball?



$$\frac{5}{5 + 6 + 4 + 3} = \frac{5}{18}$$

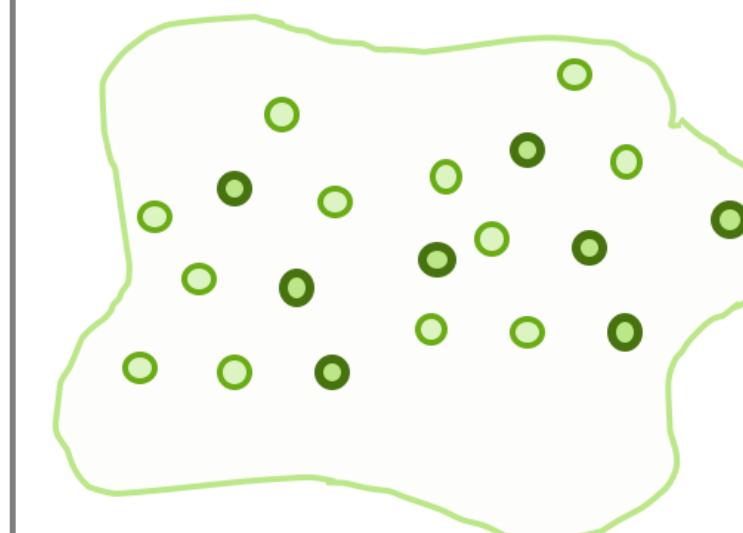
How likely is a sentence to appear in a language?

What is the probability to pick a green ball?



$$\frac{5}{5 + 6 + 4 + 3} = \frac{5}{18}$$

Can we do the same for sentences?



Text corpus

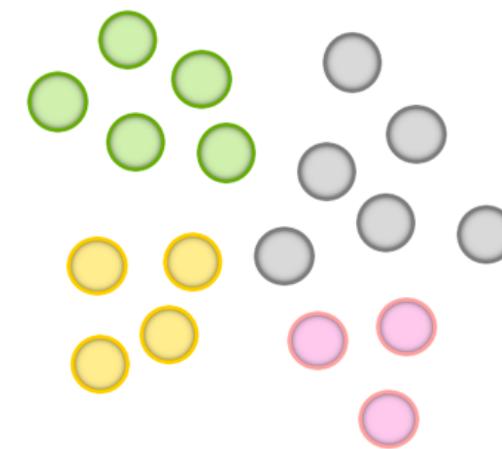
$$P(\text{the mut is tinning the tebn}) = \frac{0}{|\text{corpus}|} = 0$$

$$P(\text{mut the tinning tebn is the}) = \frac{0}{|\text{corpus}|} = 0$$

With this approach, sentences that never occurred in the corpus will receive zero probability

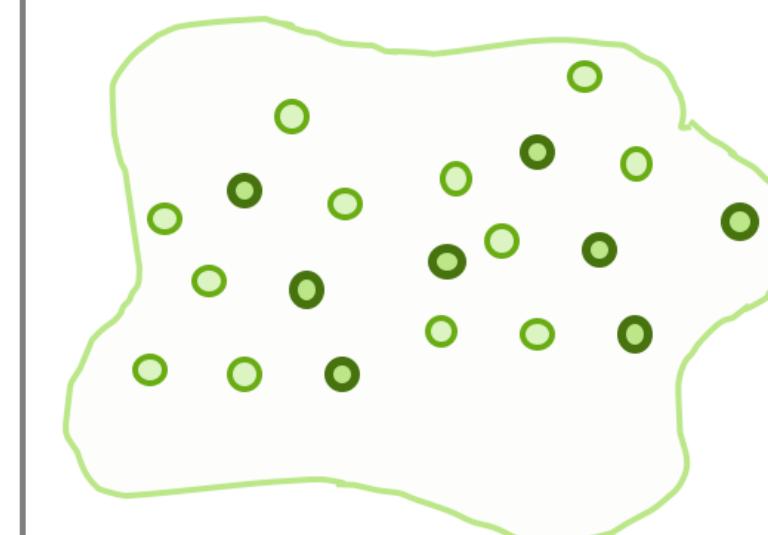
How likely is a sentence to appear in a language?

What is the probability to pick a green ball?



$$\frac{5}{5 + 6 + 4 + 3} = \frac{5}{18}$$

Can we do the same for sentences?

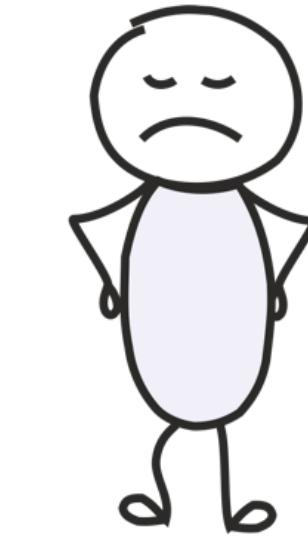


$$P(\text{the mut is tinning the tebn}) = \frac{0}{|\text{corpus}|} = 0$$

$$P(\text{mut the tinning tebn is the}) = \frac{0}{|\text{corpus}|} = 0$$

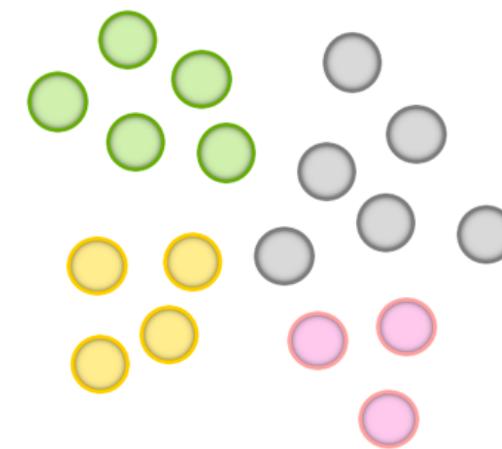
With this approach, sentences that never occurred in the corpus will receive zero probability

But the first sentence is “more likely” than the second!
This method is not good!



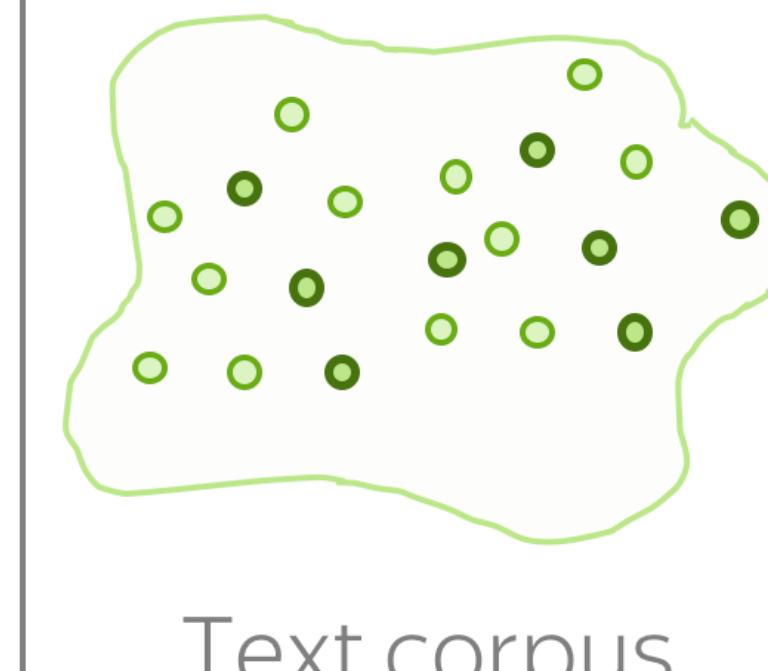
How likely is a sentence to appear in a language?

What is the probability to pick a green ball?



$$\frac{5}{5 + 6 + 4 + 3} = \frac{5}{18}$$

Can we do the same for sentences?



$$P(\text{the mut is tinning the tebn}) = \frac{0}{|\text{corpus}|} = 0$$

$$P(\text{mut the tinning tebn is the}) = \frac{0}{|\text{corpus}|} = 0$$

With this approach, sentences that never occurred in the corpus will receive zero probability

But the first sentence is “more likely” than the second!
This method is not good!



We can not estimate sentence probabilities reliably if we treat them as atomic units!

Sentence Probability: Decompose into Smaller Parts

Imagine we

- read the sentence **I saw a cat on a mat** word by word,
- update probability every time we see a new token

$$P(\mathbf{I}) =$$

$$P(I)$$



Probability of **I**



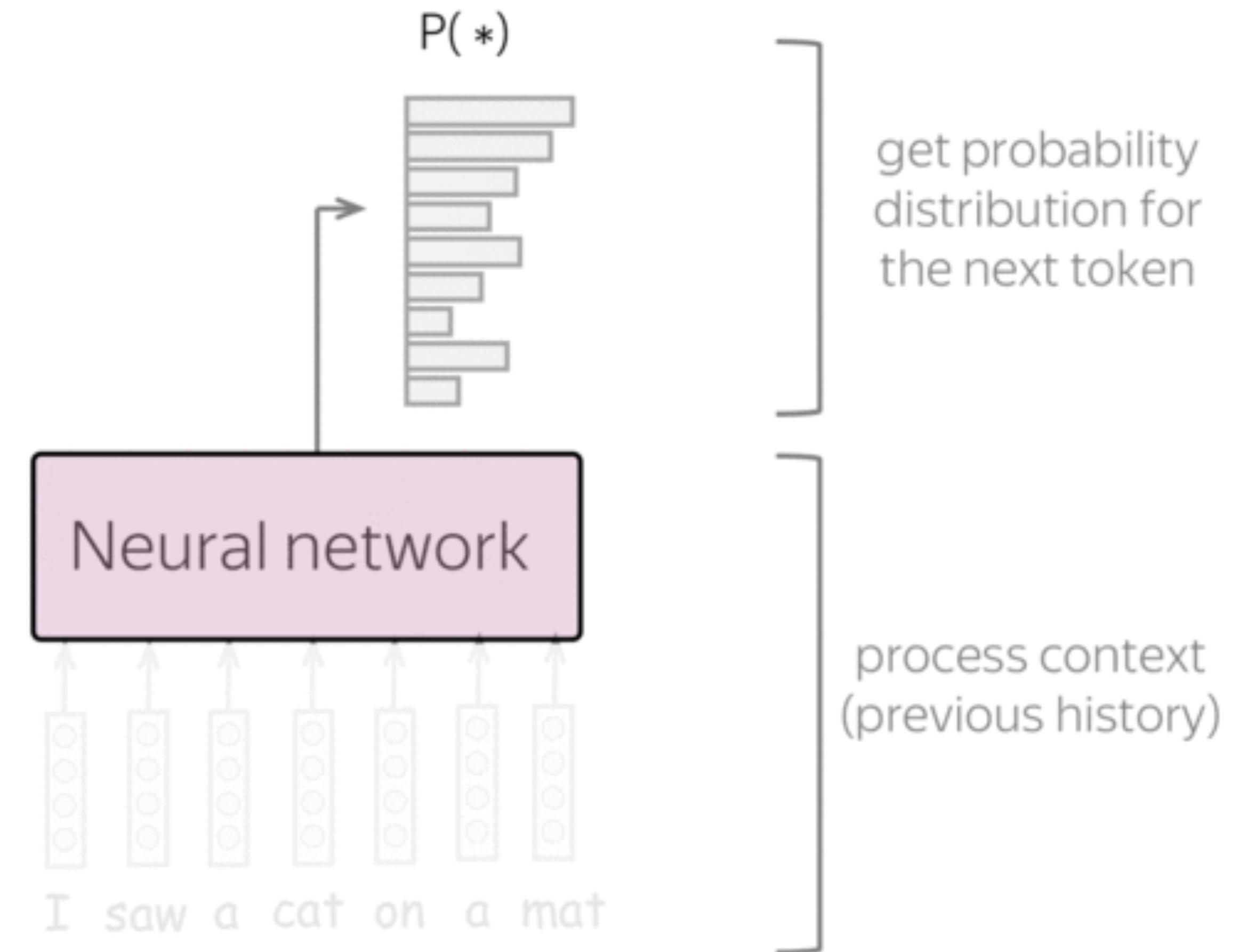
General View

- process context – model-specific

Get vector representation of the previous context

- evaluate probabilities – model-agnostic

Predict probability distribution for the next token



General View

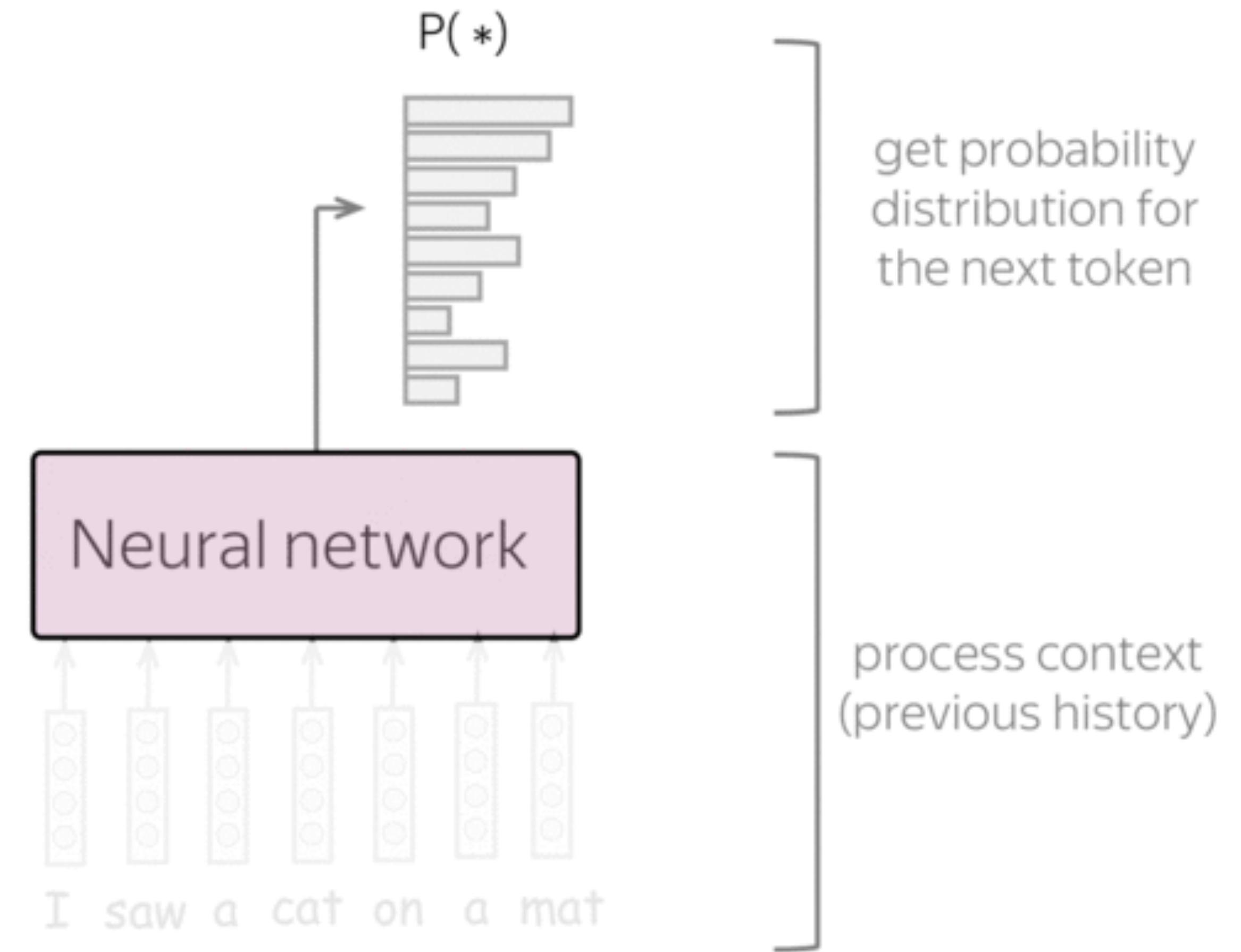
- process context – model-specific

Get vector representation of the previous context

- evaluate probabilities – model-agnostic

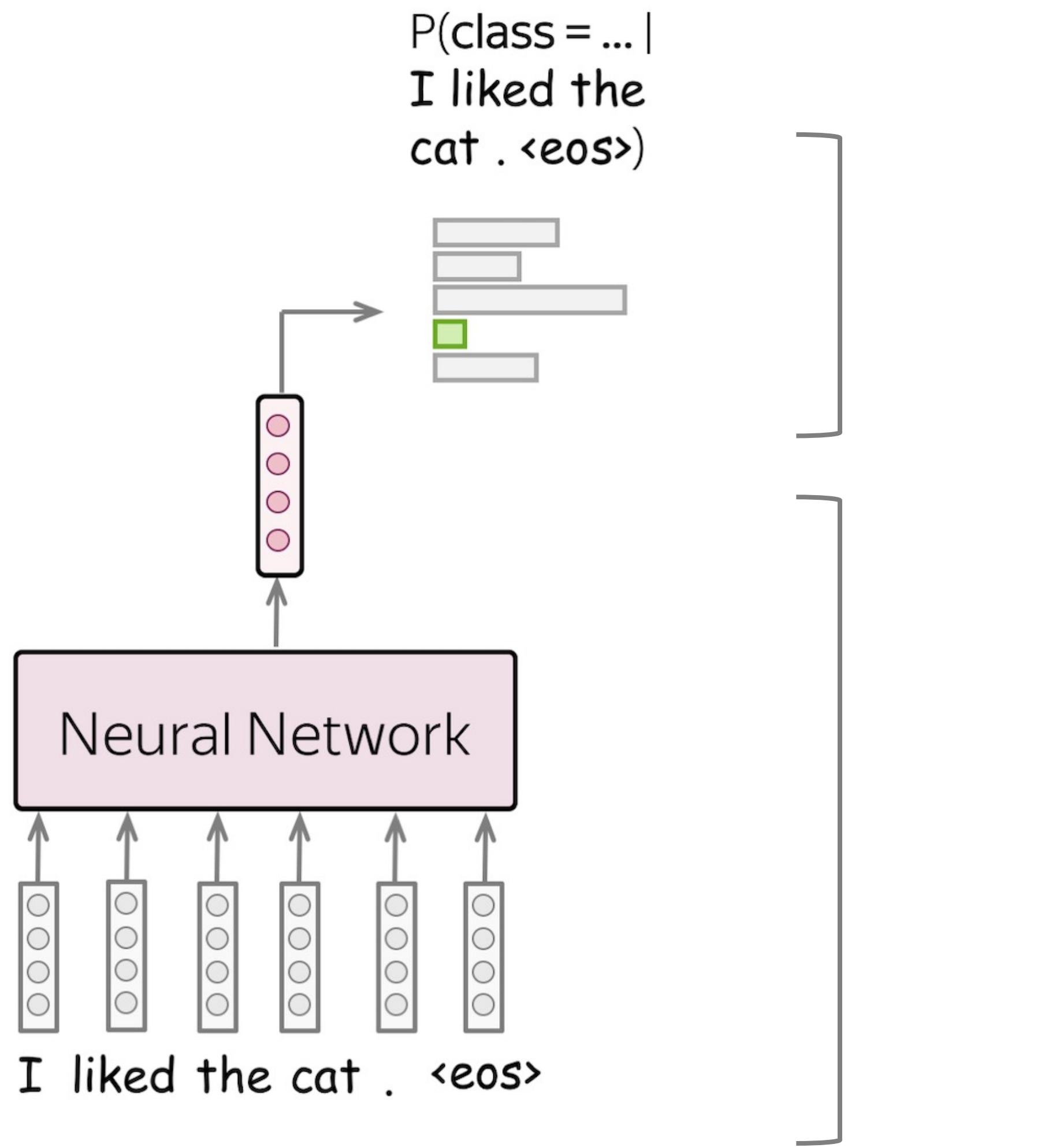
Predict probability distribution for the next token

Classify into $|V|$ classes



Classification with Neural Networks

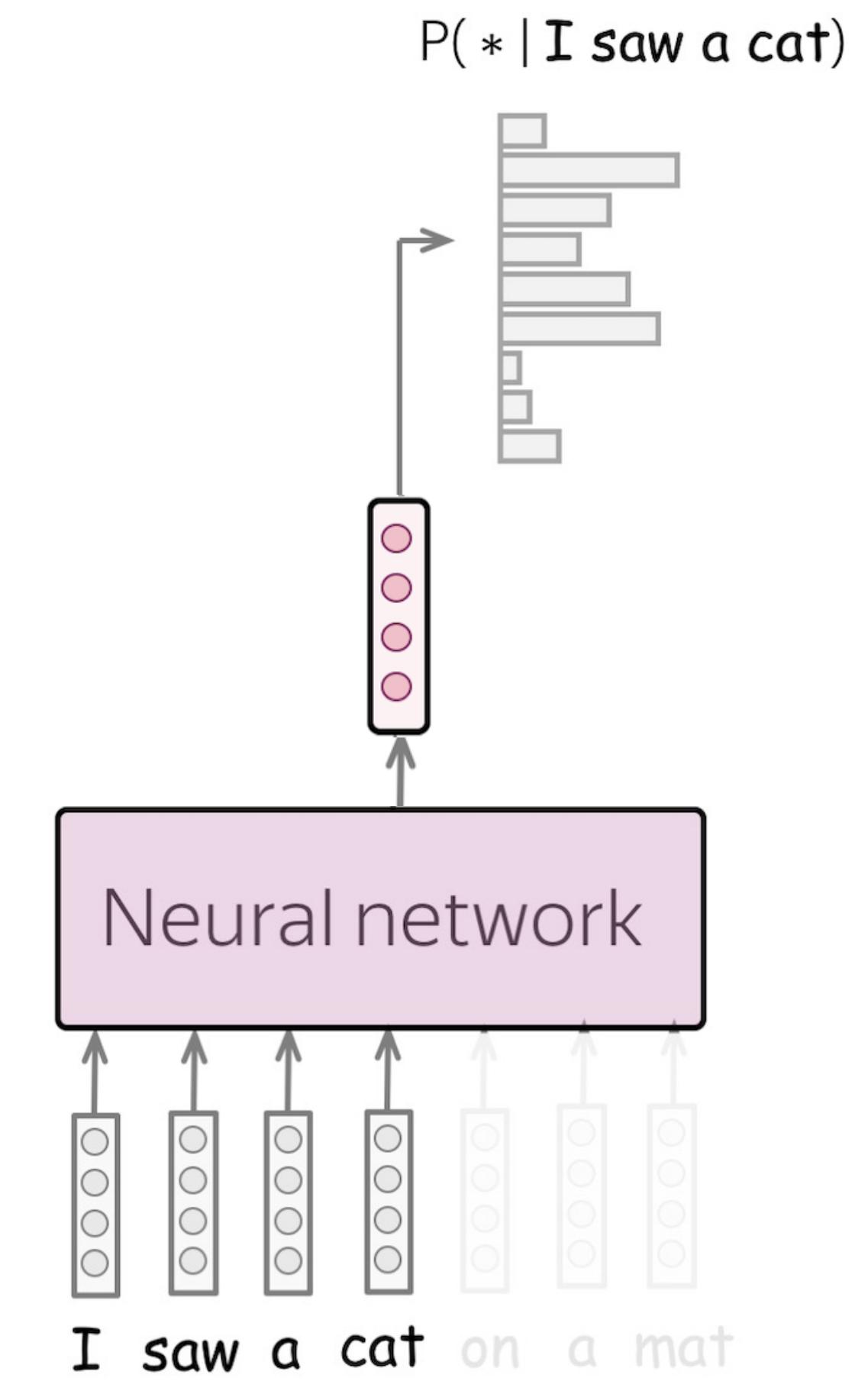
- Before: Classification



Get probability
distribution over
classes

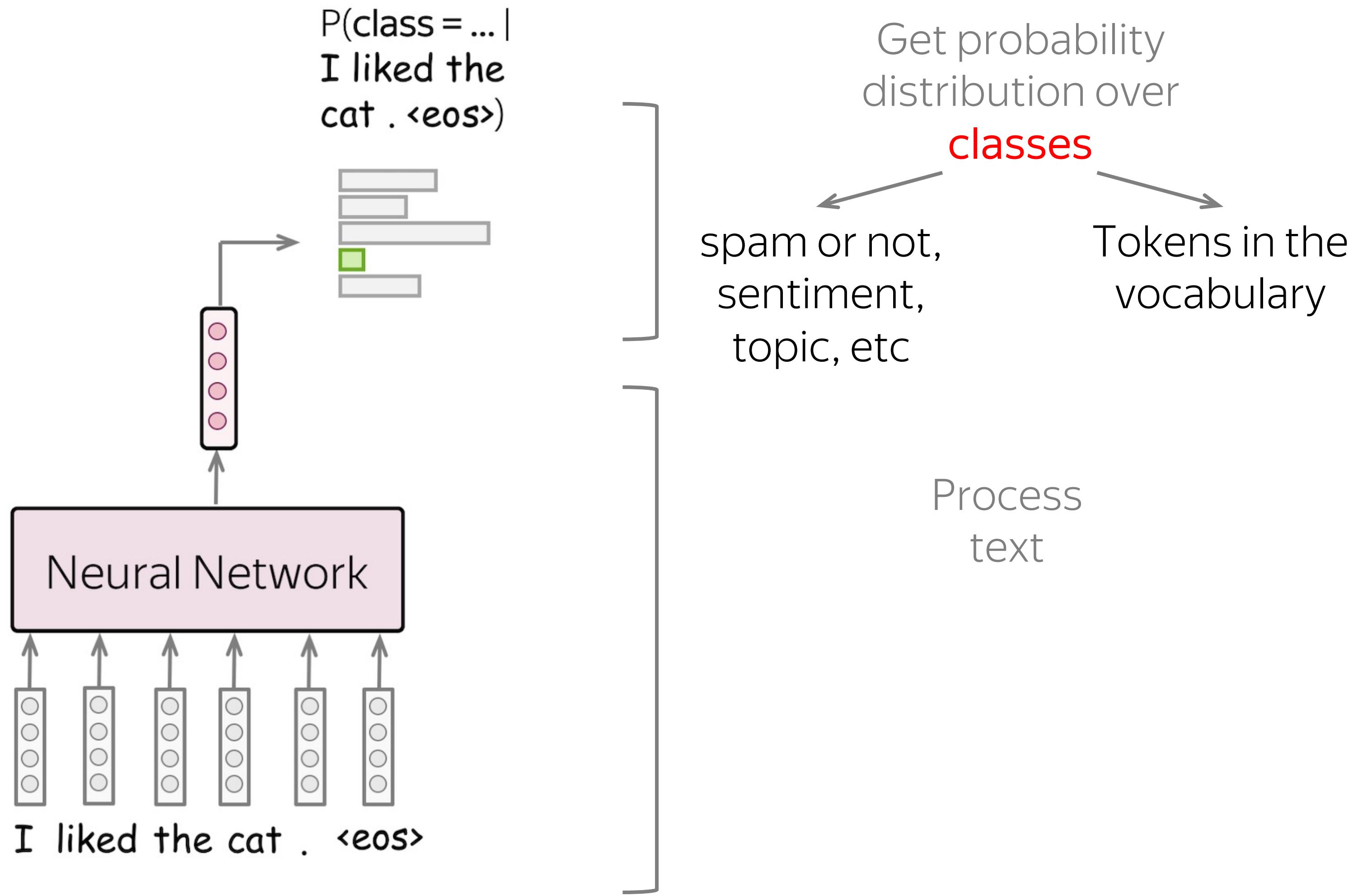
Process
text

- Now: Language Modeling

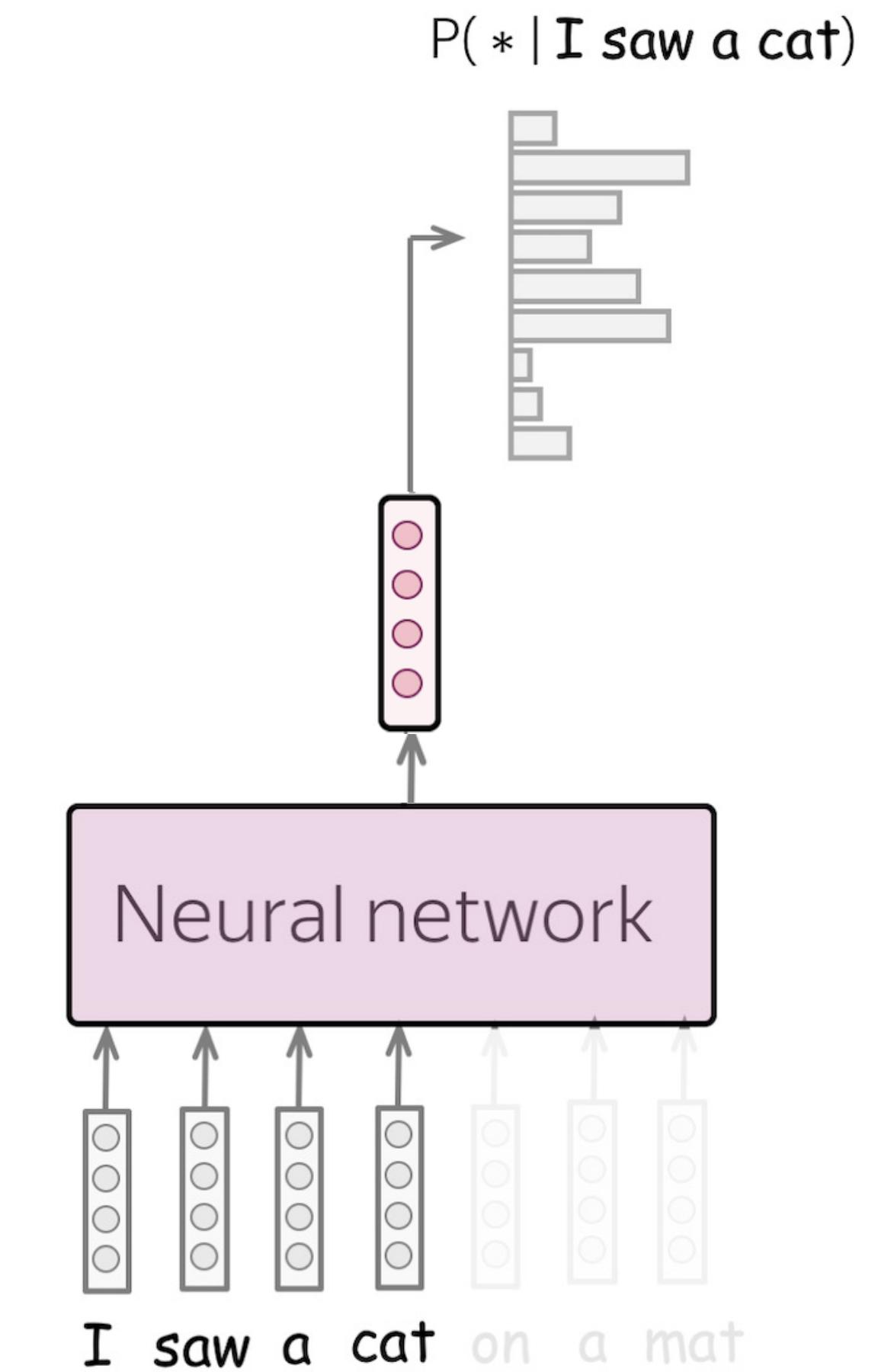


Classification with Neural Networks

- Before: Classification

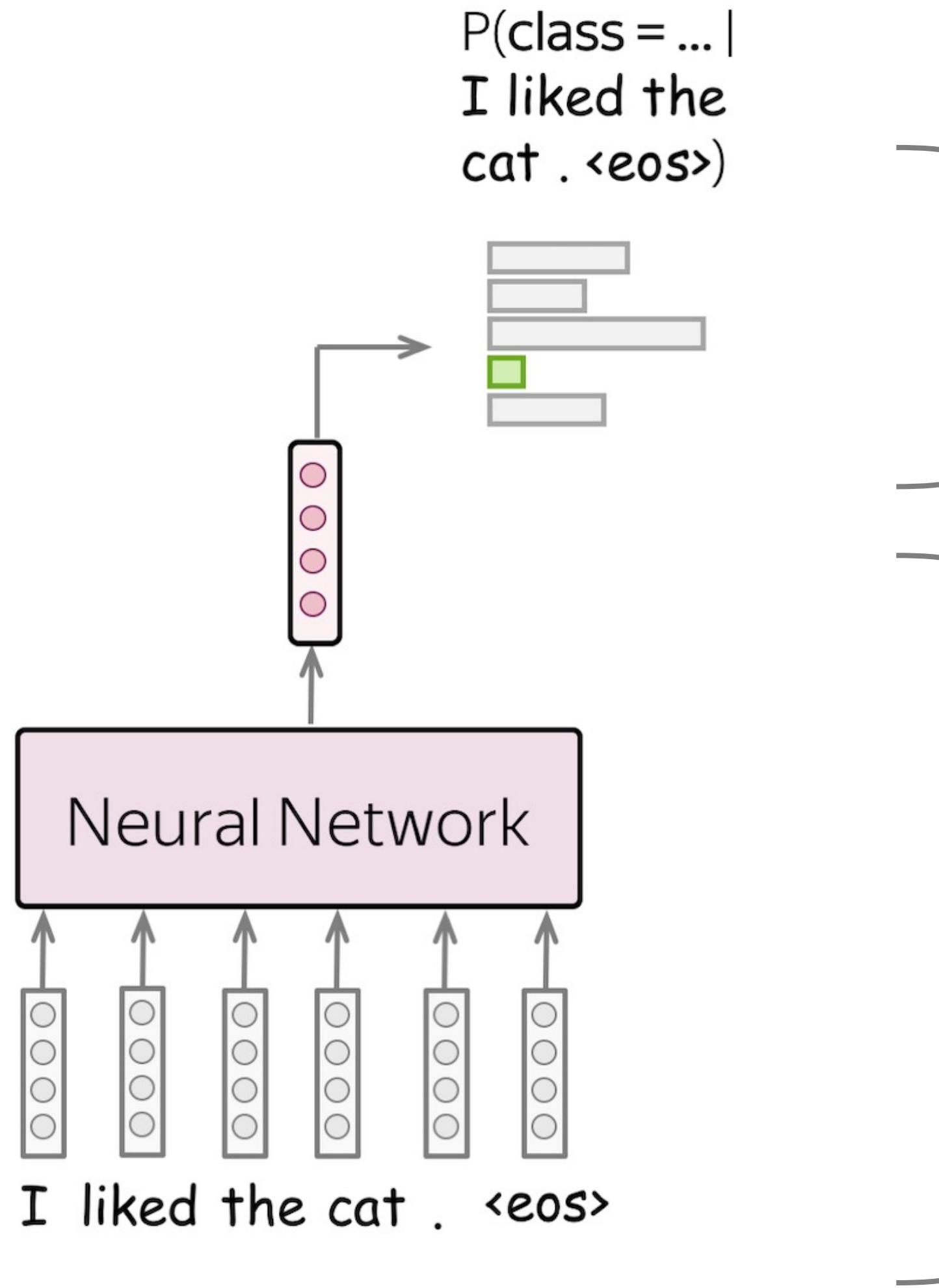


- Now: Language Modeling



Classification with Neural Networks

- Before: Classification



Get probability distribution over **classes**

spam or not, sentiment, topic, etc

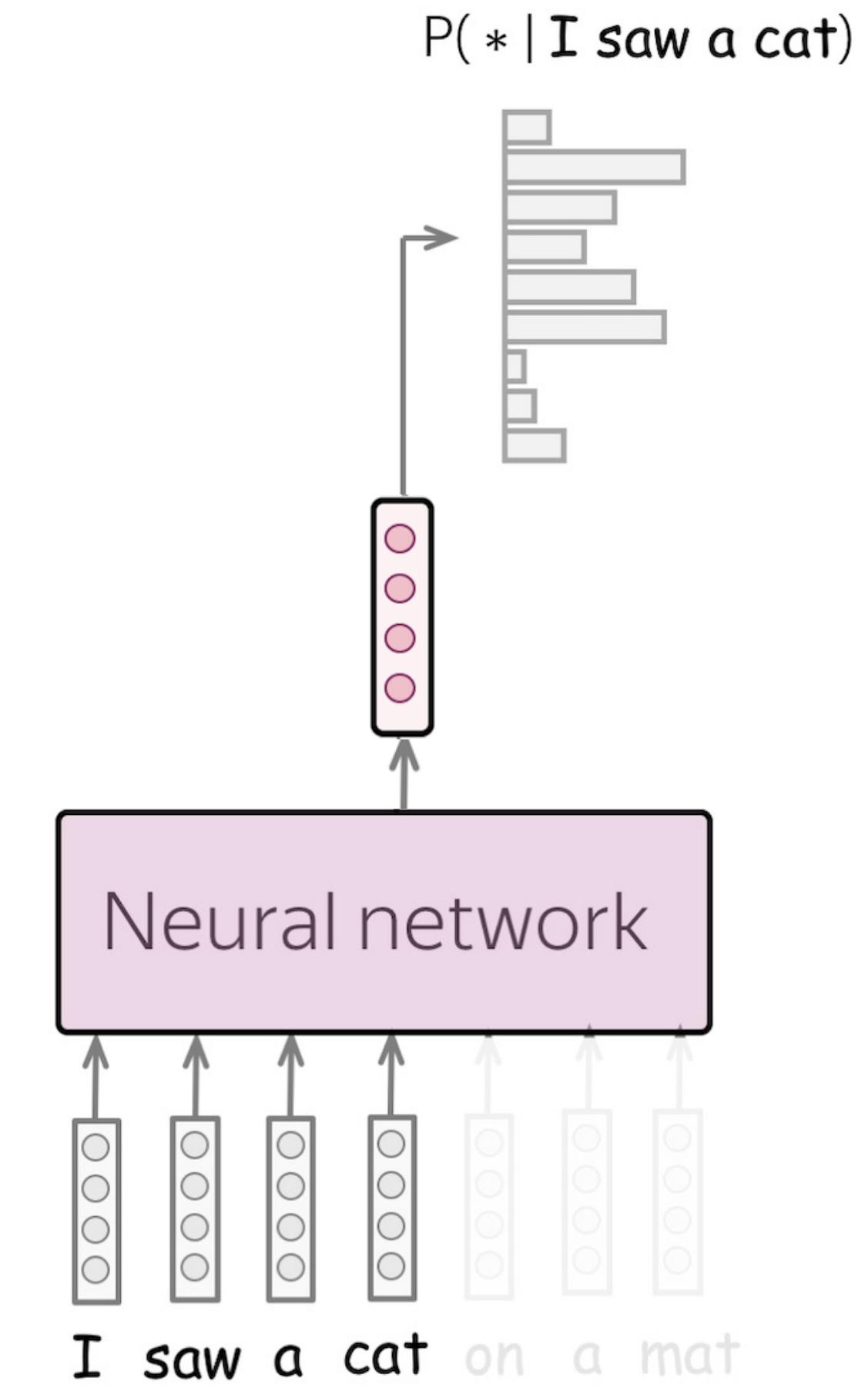
Process **text**

The whole sentence/document

Previous tokens, i.e. tokens up to the current

Tokens in the vocabulary

- Now: Language Modeling



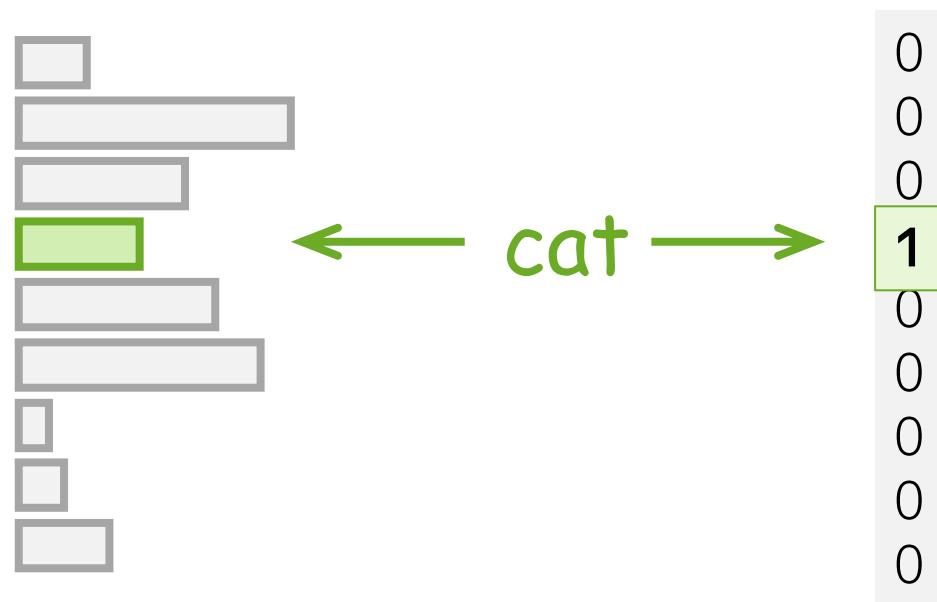
Training: Cross-Entropy

Target at this step
↓
Training example: I saw a **cat** on a mat <eos>

Model prediction: Target:

$$p(* | \text{I saw a})$$

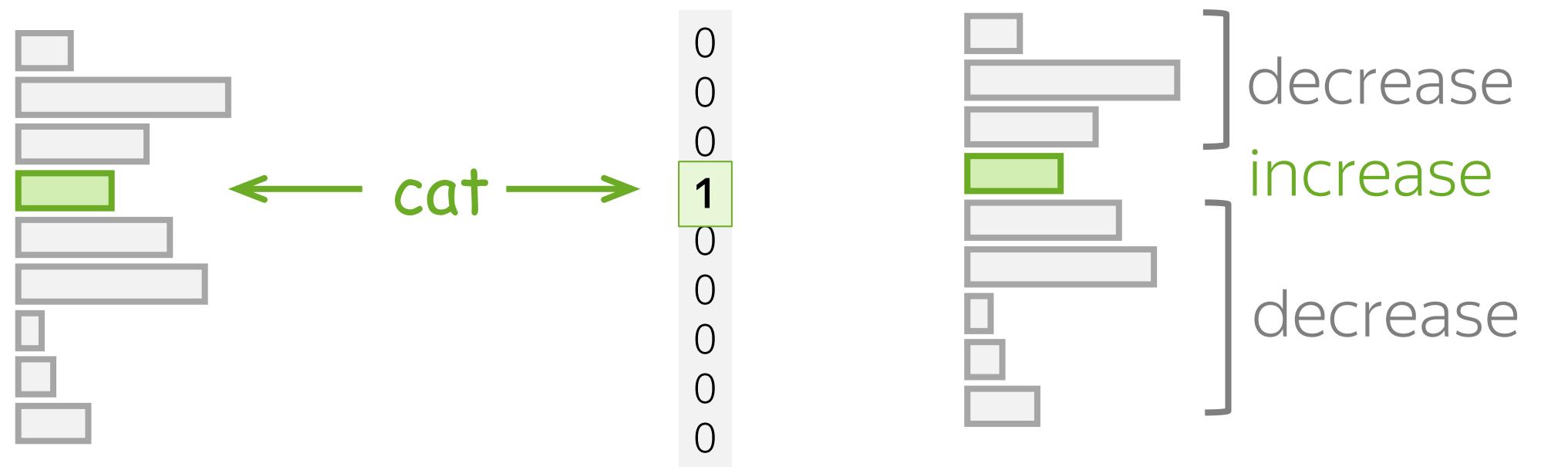
$$p^*$$



Training: Cross-Entropy

Target at this step
↓
Training example: **I saw a cat** on a mat <eos>

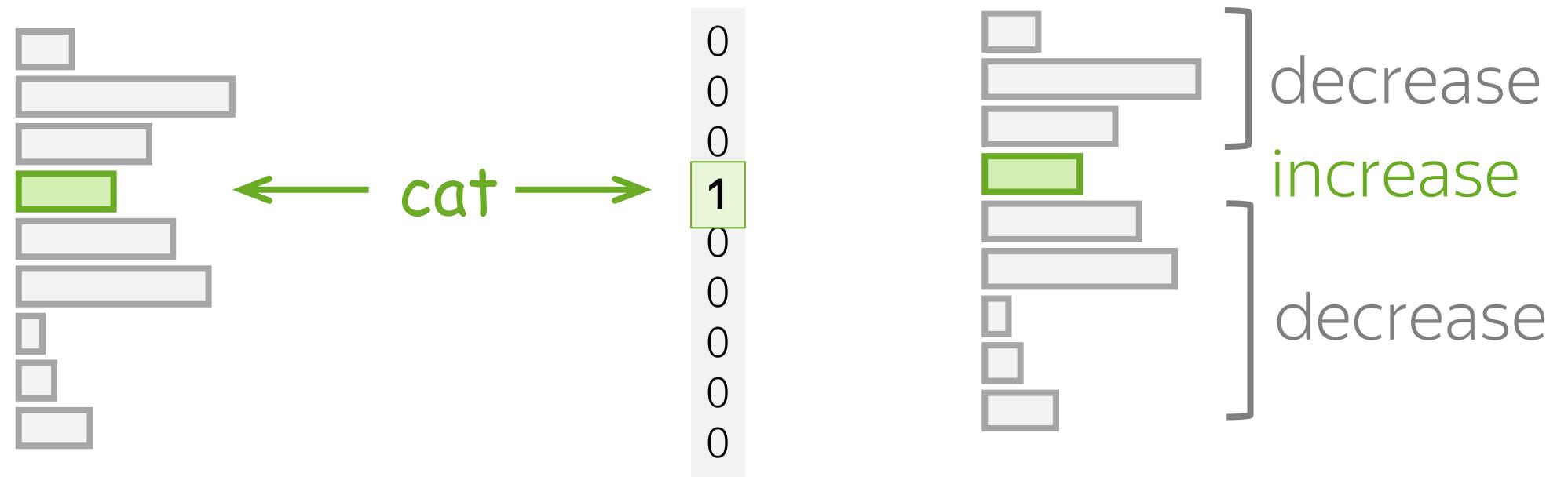
Model prediction: Target: Loss = $-\log(p(\text{cat})) \rightarrow \min$
 $p(* | \text{I saw a})$ p^*



Training: Cross-Entropy

Target at this step
↓
Training example: **I saw a cat** on a mat <eos>

Model prediction: Target: Loss = $-\log(p(\text{cat})) \rightarrow \min$
 $p(* | \text{I saw a})$ p^*

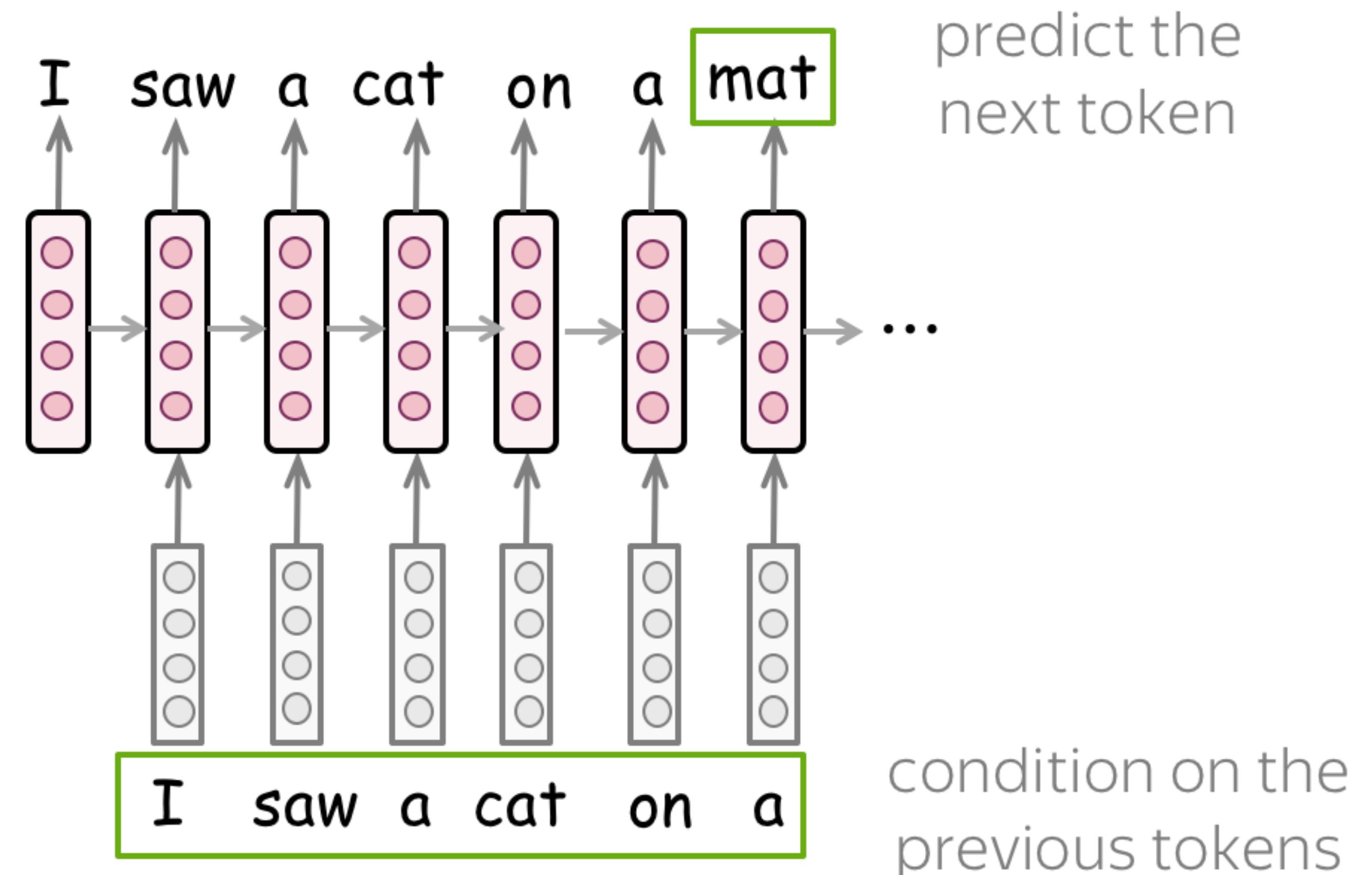


Initial RNN state
Start: do not have input, want to predict the first token

we want the model to predict this
↓
Training example: **I saw a cat** on a mat <eos>

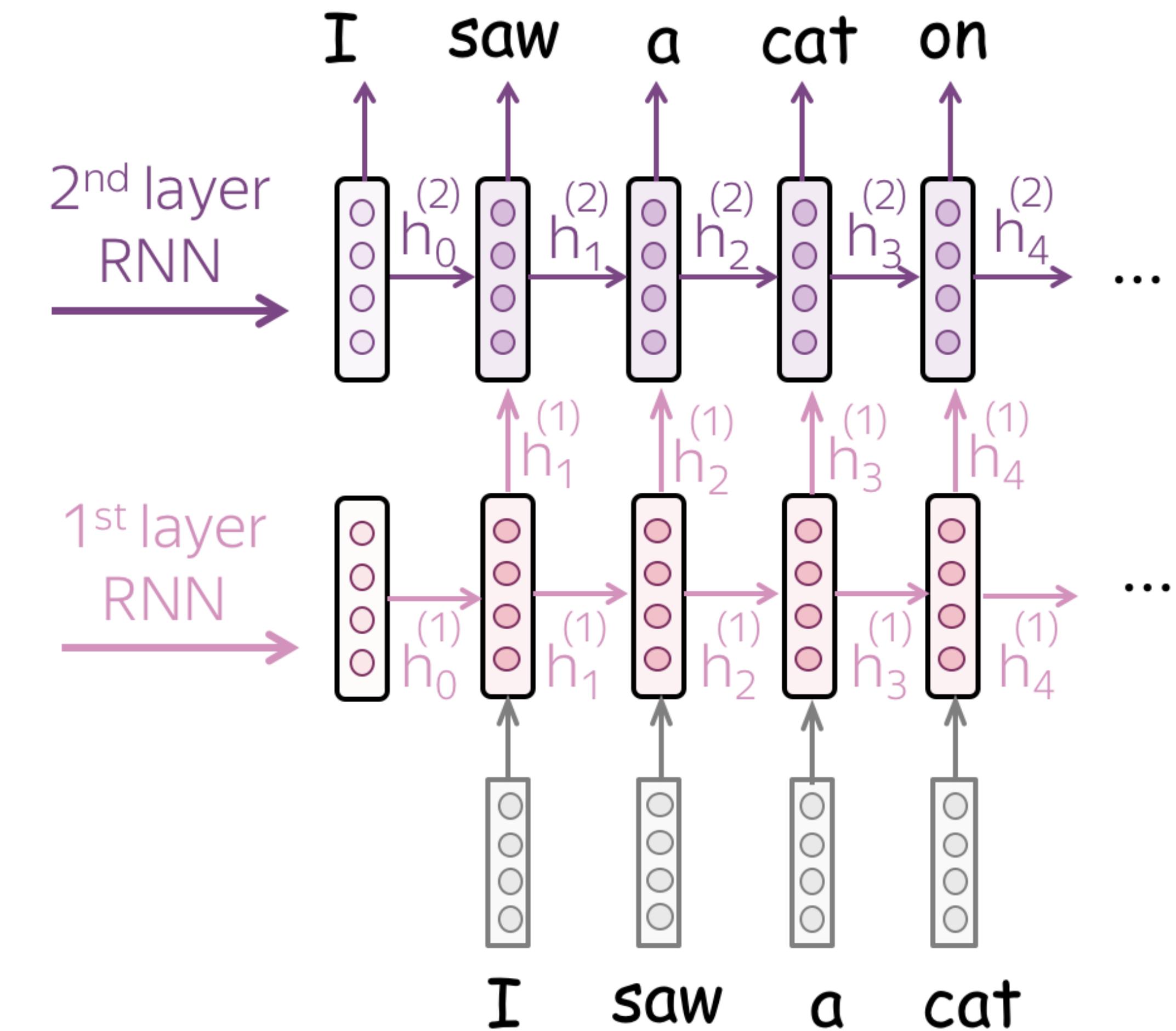
Recurrent Models for Language Modeling

- simple: read a text, predict next token at each step



Recurrent Models for Language Modeling

- Multi-layer: feed the states from one RNN to the next



Generate a Text Using a Language Model

I _____

Alternatively, you can use greedy decoding: pick the token with the highest probability.

Examples

Neural LM trained on 2.5m English sentences.

```
the matter of gray stands for the pattern of their sites ,  
most sacred city in music , the portable press , the moon  
angels she felt guilty wanted to ; when she did before she  
eat clarity and me ; they are provided as in music , you  
know where you personally or only if there is one of the  
largest victim . _eos_
```

Examples

Neural LM trained on 2.5m English sentences.

```
it is to strengthen the specific roles of national opinion  
is an effective and conviction of cargo in a mid -  
december , an egyptian state opera _eos_
```

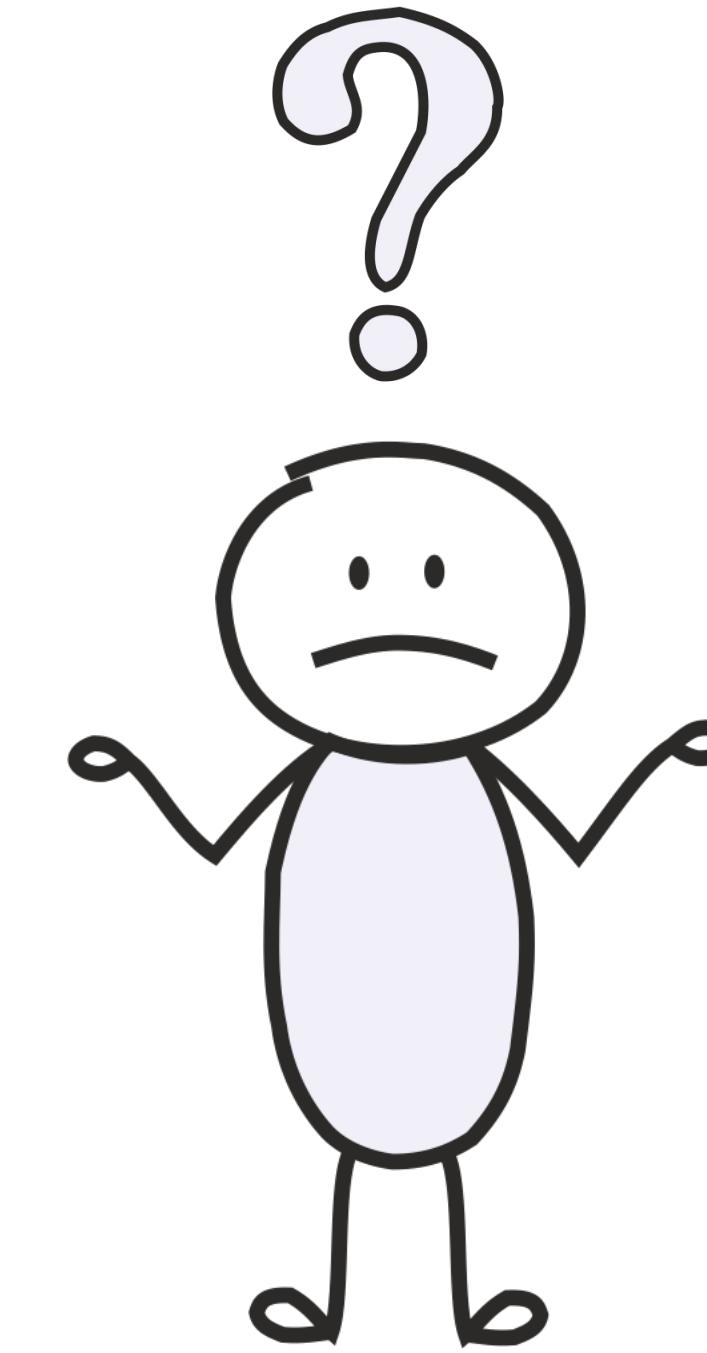
Examples

Neural LM trained on 2.5m English sentences.

this is why the government does not occur or any of any other terms and conditions for the people , and others remained still has to be more confident about which its way to the main challenge to find a company in " corporate " is complete with the case _eos_

Recurrent Models for Language Modeling

Can we understand what
these models learn?



Examples of generated texts

- Char-level LSTM trained on Linux source code

```
/*
 * Increment the size file of the new incorrect UI_FILTER group information
 * of the size generatively.
 */
static int indicate_policy(void)
{
    int error;
    if (fd == MARN_EPT) {
        /*
         * The kernel blank will coeld it to userspace.
         */
        if (ss->segment < mem_total)
            unblock_graph_and_set_blocked();
        else
            ret = 1;
        goto bail;
    }
    segaddr = in_SB(in.addr);
    selector = seg / 16;
    setup_works = true;
    for (i = 0; i < blocks; i++) {
        seq = buf[i++];
        bpf = bd->bd.next + i * search;
        if (fd) {
            current = blocked;
        }
    }
    rw->name = "Getjbbregs";
    bprm_self_clearl(&iv->version);
    regs->new = blocks[(BPF_STATS << info->historidac)] | PFMR_CLOBATHINC_SECONDS << 12;
    return segtable;
}
```

The examples are from the paper [Visualizing and Understanding Recurrent Networks](#)

Visualize Neuron Activations: Some Are Interpretable!

- Char-level LSTMs trained on Linux Kernel and War and Peace

Cell sensitive to position in line

The sole importance of the crossing of the Berezina lies in the fact that it plainly and indubitably proved the fallacy of all the plans for cutting off the enemy's retreat and the soundness of the only possible line of action--the one Kutuzov and the general mass of the army demanded--namely, simply to follow the enemy up. The French crowd fled at a continually increasing speed and all its energy was directed to reaching its goal. It fled like a wounded animal and it was impossible to block its path. This was shown not so much by the arrangements it made for crossing as by what took place at the bridges. When the bridges broke down, unarmed soldiers, people from Moscow and women with children who were with the French transport, all--carried on by vis inertiae--pressed forward into boats and into the ice-covered water and did not, surrender.

Cell that turns on inside quotes

"You mean to imply that I have nothing to eat out of.... On the contrary, I can supply you with everything even if you want to give dinner parties," warmly replied Chichagov, who tried by every word he spoke to prove his own rectitude and therefore imagined Kutuzov to be animated by the same desire.

Kutuzov, shrugging his shoulders, replied with his subtle penetrating smile: "I meant merely to say what I said."

The examples are from the paper [Visualizing and Understanding Recurrent Networks](#)

Visualize Neuron Activations: Some Are Interpretable!

- Char-level LSTMs trained on Linux Kernel and War and Peace

Cell that activates inside if statements

```
static int __dequeue_signal(struct sigpending *pending, sigset_t *mask,
    siginfo_t *info)
{
    int sig = next_signal(pending, mask);
    if (sig) {
        if (current->notifier) {
            if (sigismember(current->notifier_mask, sig)) {
                if (!(current->notifier)(current->notifier_data)) {
                    clear_thread_flag(TIF_SIGPENDING);
                    return 0;
                }
            }
        }
        collect_signal(sig, pending, info);
    }
    return sig;
}
```

The examples are from the paper [Visualizing and Understanding Recurrent Networks](#)

Visualize Neuron Activations: Some Are Interpretable!

- Char-level LSTMs trained on Linux Kernel and War and Peace

Cell that turns on inside comments and quotes

```
/* Duplicate LSM field information. The lsm_rule is opaque, so
 * re-initialized. */
static inline int audit_dupe_lsm_field(struct audit_field *df,
    struct audit_field *sf)
{
    int ret = 0;
    char *lsm_str;
    /* our own copy of lsm_str */
    lsm_str = kstrdup(sf->lsm_str, GFP_KERNEL);
    if (unlikely(!lsm_str))
        return -ENOMEM;
    df->lsm_str = lsm_str;
    /* our own (refreshed) copy of lsm_rule */
    ret = security_audit_rule_init(df->type, df->op, df->lsm_str,
        (void **)&df->lsm_rule);
    /* Keep currently invalid fields around in case they
     * become valid after a policy reload. */
    if (ret == -EINVAL) {
        pr_warn("audit rule for LSM \\'%s\\' is invalid\n",
            df->lsm_str);
        ret = 0;
    }
    return ret;
}
```

The examples are from the paper [Visualizing and Understanding Recurrent Networks](#)

Visualize Neuron Activations: Some Are Interpretable!

- Char-level LSTMs trained on Linux Kernel and War and Peace

Cell sensitive to the depth of an expression

```
#ifdef CONFIG_AUDITSYSCALL
static inline int audit_match_class_bits(int class, u32 *mask)
{
    int i;
    if (classes[class]) {
        for (i = 0; i < AUDIT_BITMASK_SIZE; i++)
            if (mask[i] & classes[class][i])
                return 0;
    }
    return 1;
}
```

The examples are from the paper [Visualizing and Understanding Recurrent Networks](#)

Visualize Neuron Activations: Some Are Interpretable!

- Char-level LSTMs trained on Linux Kernel and War and Peace

Cell that might be helpful in predicting new line

```
char *audit_unpack_string(void **bufp, size_t *remain, si
{
    char *str;
    if (!*bufp || (len == 0) || (len > *remain))
        return ERR_PTR(-EINVAL);
    /* of the currently implemented string fields, PATH_MAX
     * defines the longest valid length.
     */
    if (len > PATH_MAX)
        return ERR_PTR(-ENAMETOOLONG);
    str = kmalloc(len + 1, GFP_KERNEL);
    if (unlikely(!str))
        return ERR_PTR(-ENOMEM);
    memcpy(str, *bufp, len);
    str[len] = 0;
    *bufp += len;
    *remain -= len;
    return str;
}
```

The examples are from the paper [Visualizing and Understanding Recurrent Networks](#)

More Recent: Sentiment Neuron

- Char-level LSTM trained on Amazon Reviews

This is one of Crichton's best books. The characters of Karen Ross, Peter Elliot, Munro, and Amy are beautifully developed and their interactions are exciting, complex, and fast-paced throughout this impressive novel. And about 99.8 percent of that got lost in the film. Seriously, the screenplay AND the directing were horrendous and clearly done by people who could not fathom what was good about the novel. I can't fault the actors because frankly, they never had a chance to make this turkey live up to Crichton's original work. I know good novels, especially those with a science fiction edge, are hard to bring to the screen in a way that lives up to the original. But this may be the absolute worst disparity in quality between novel and screen adaptation ever. The book is really, really good. The movie is just dreadful.

The example is from the [Open AI blog post](#)

Use Interpretable Neurons to Control Generated Texts

- Fix neuron value, sample from the prefix **I couldn't figure out**

SENTIMENT FIXED TO POSITIVE

I couldn't figure out the shape at first but it definitely does what it's meant to do. It's a great product and I recommend it highly

I couldn't figure out why this movie had been discontinued! Now I can enjoy it anytime I like. So glad to have found it again.

I couldn't figure out how to use the video or the book that goes along with it, but it is such a fantastic book on how to put it into practice!

I couldn't figure out how to use just one and my favorite running app. I use it all the time. Good quality, You cant beat the price.

I couldn't figure out how to attach these balls to my little portable drums, but these fit the bill and were well worth every penny.

SENTIMENT FIXED TO NEGATIVE

I couldn't figure out how to use the product. It did not work. At least there was no quality control; this tablet does not work. I would have given it zero stars, but that was not an option.

I couldn't figure out how to set it up being that there was no warning on the box. I wouldn't recommend this to anyone.

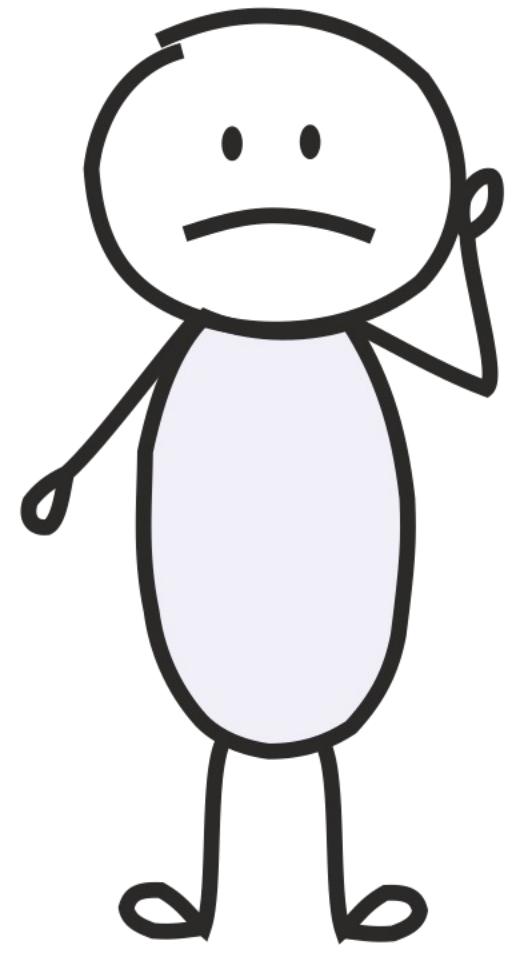
I couldn't figure out how to use the gizmo. What a waste of time and money. Might as well through away this junk.

I couldn't figure out how to stop this drivel. At worst, it was going absolutely nowhere, no matter what I did. Needles to say, I skim-read the entire book. Don't waste your time.

I couldn't figure out how to play it.

The example is from the [Open AI blog post](#)

Language models learn
things we are **not** teaching
them explicitly



What is going to happen:

- Classification
 - Language Modeling
 - Transfer Learning
 - A Note on Crazy Stuff
- Features:
from hand-crafted
to learned
- Model:
from task-specific
to generalized

What is going to happen:

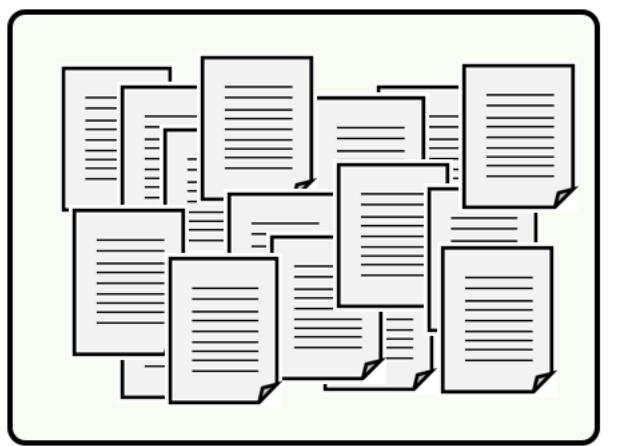
- Classification
- Language Modeling
- Transfer Learning →
 - High-Level Idea
 - From words to words in context
 - Transformer
 - BERT
- A Note on Crazy Stuff

Features:
from hand-crafted
to learned

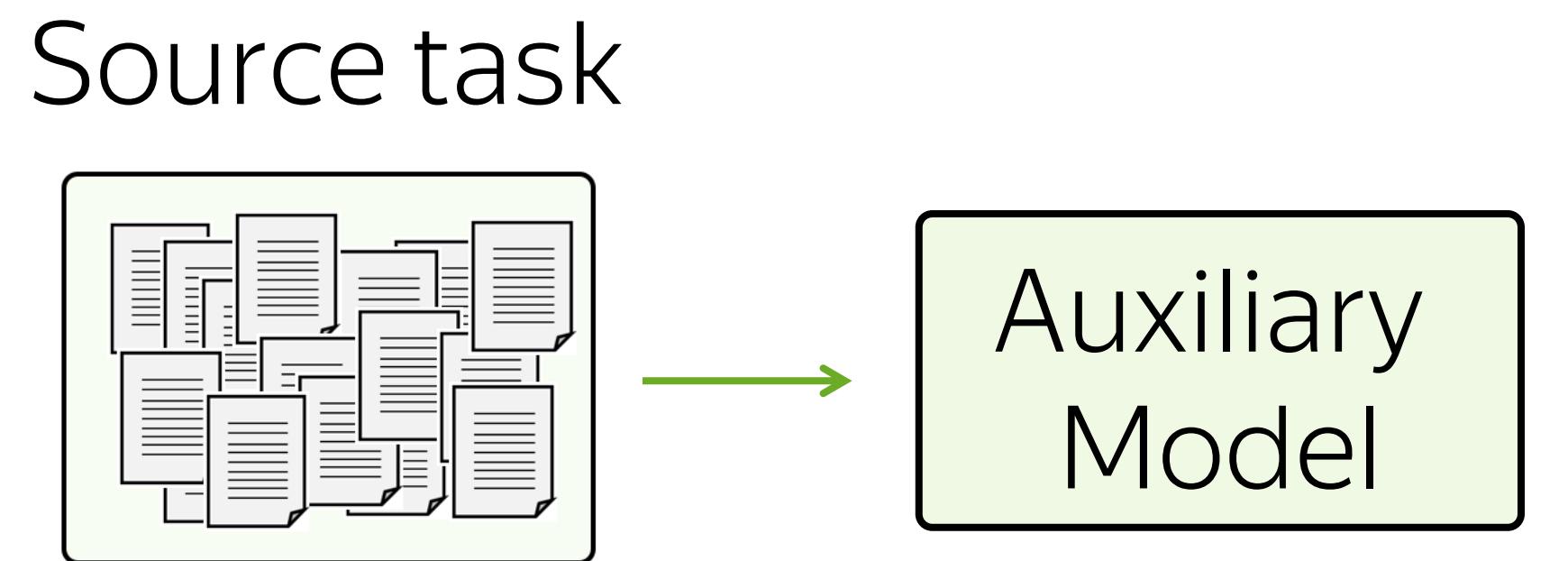
Model:
from task-specific
to generalized

Transfer Learning Idea

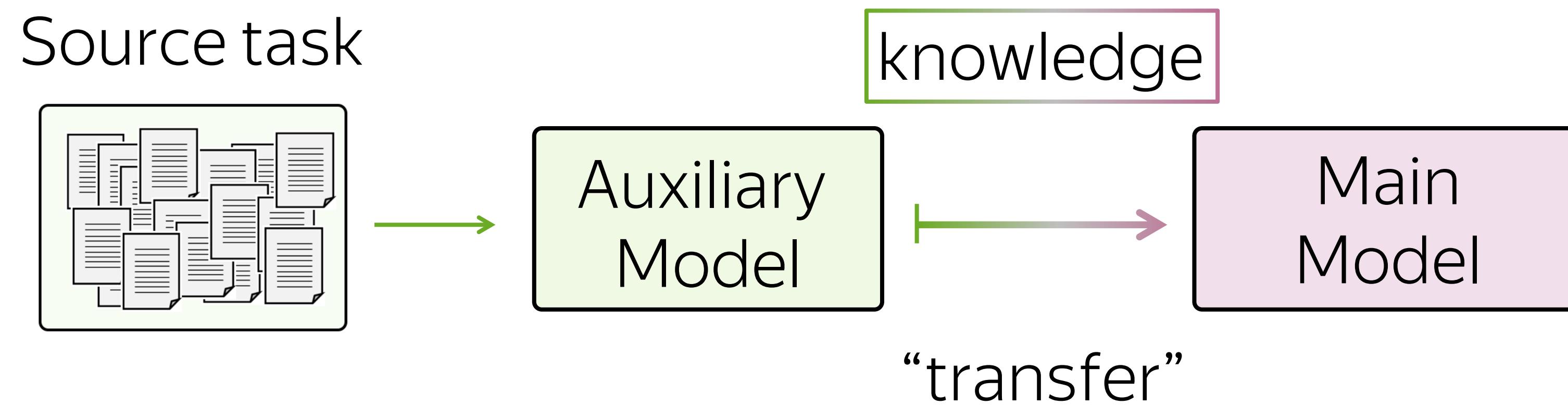
Source task



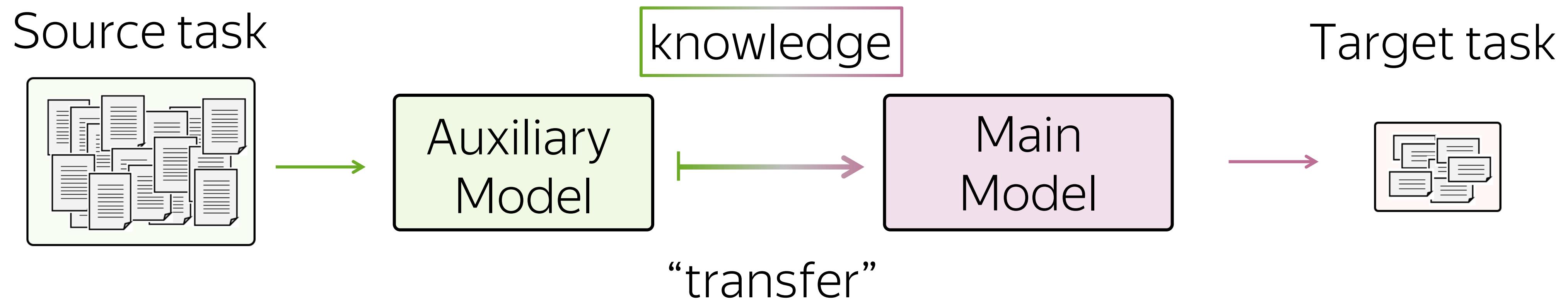
Transfer Learning Idea



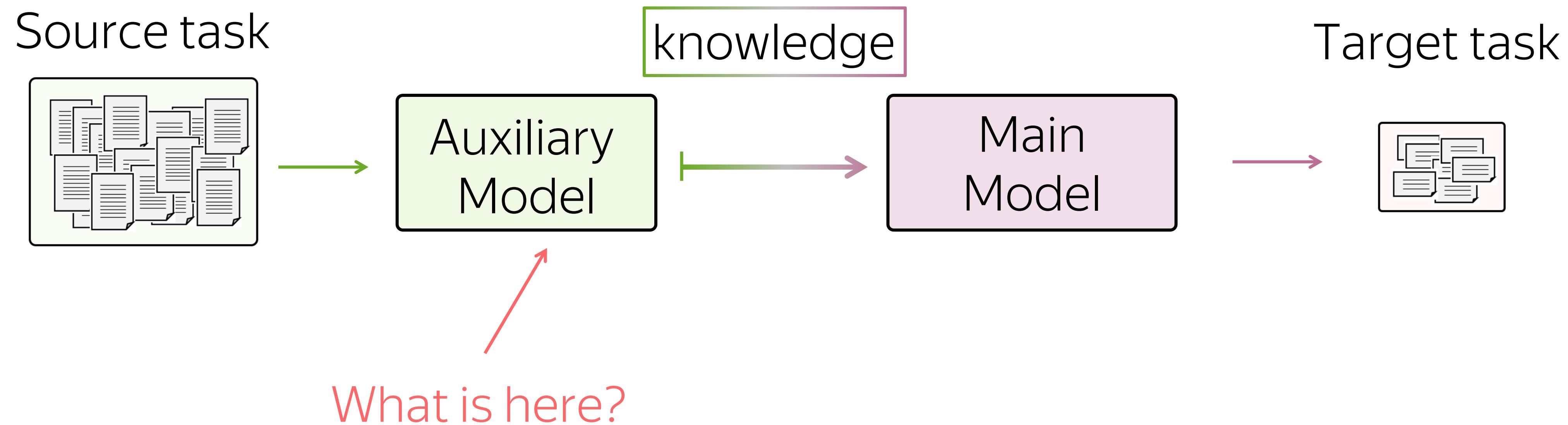
Transfer Learning Idea



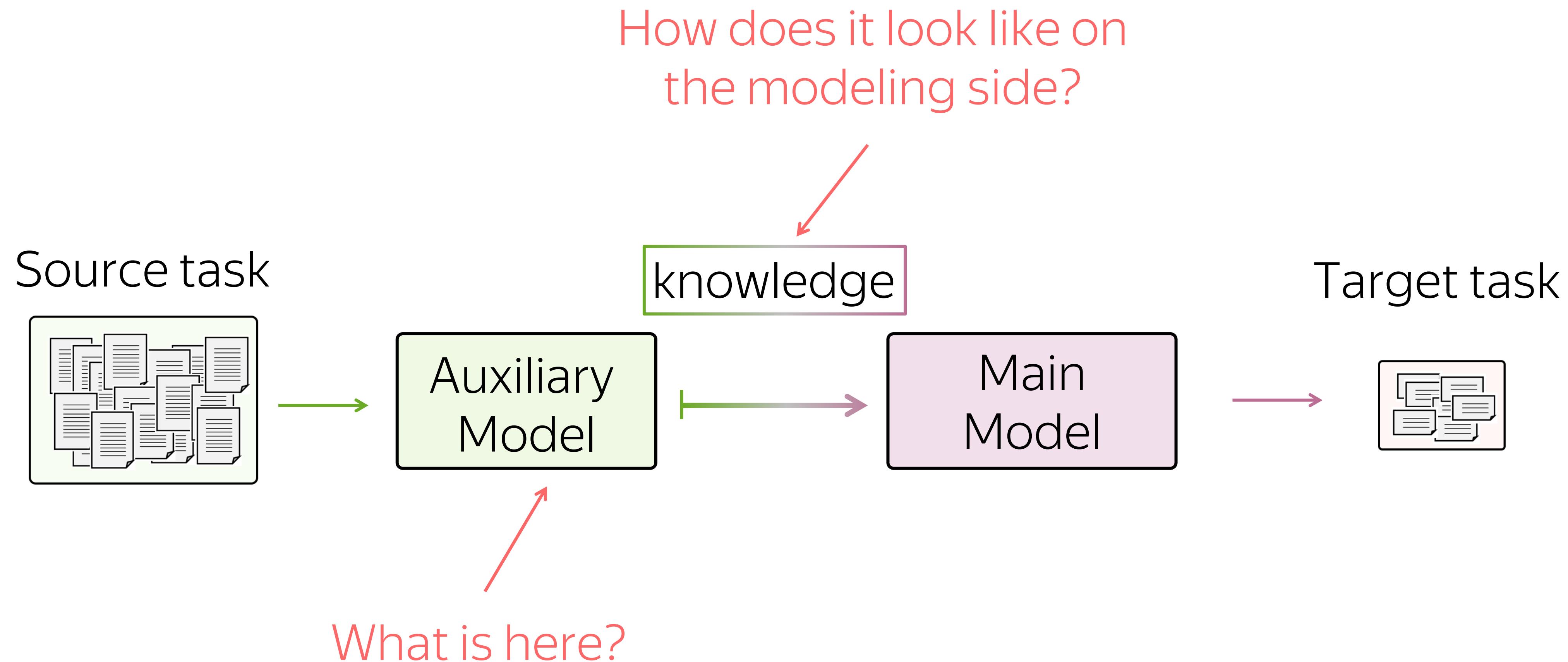
Transfer Learning Idea



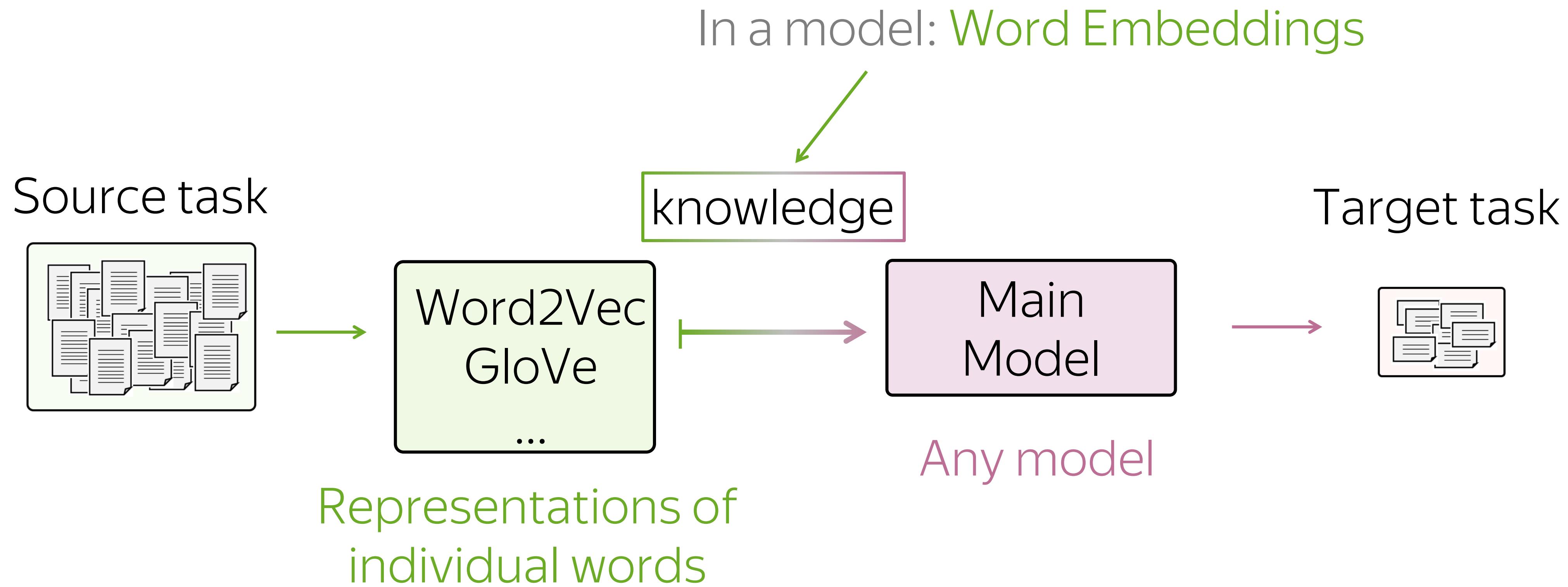
Transfer Learning Idea



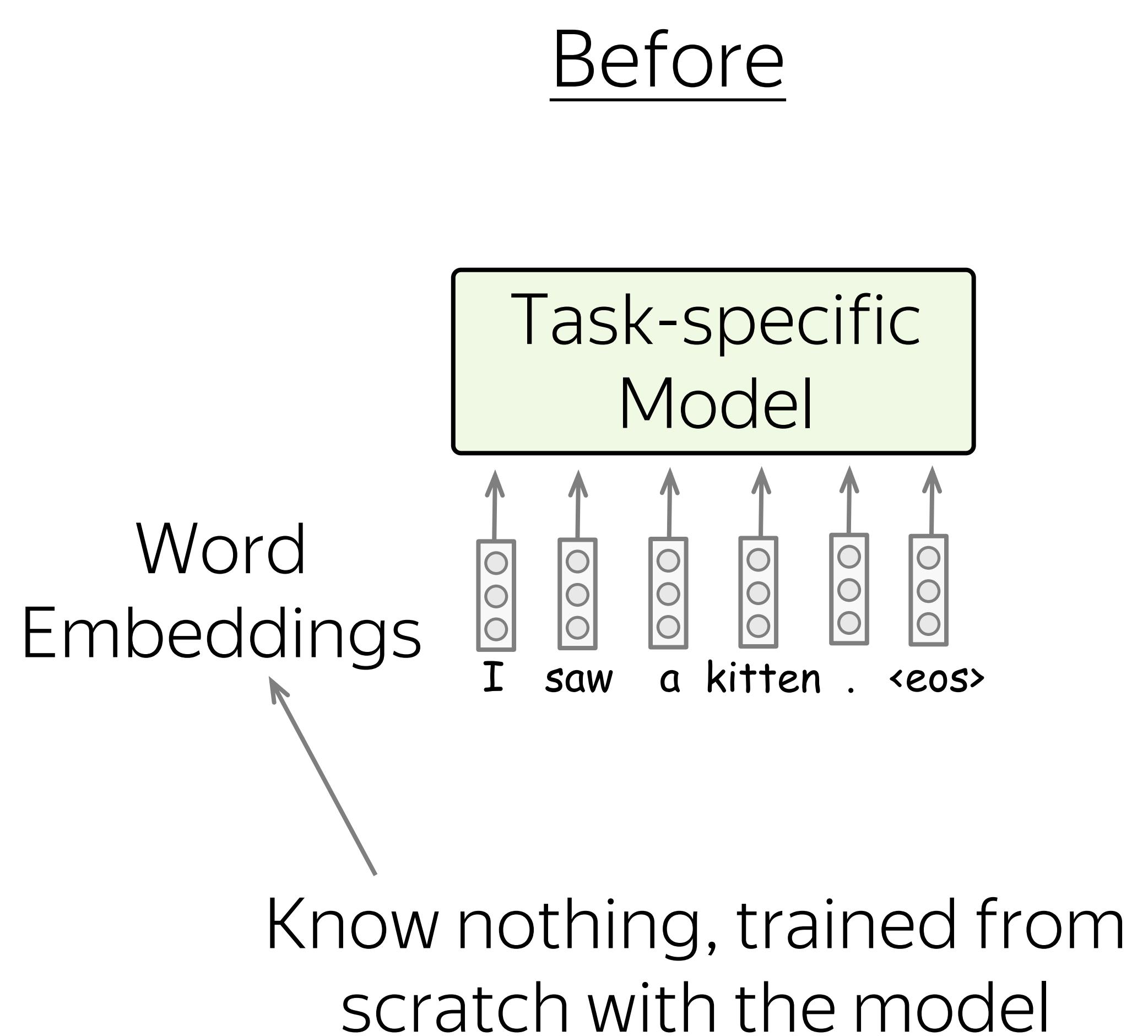
Transfer Learning Idea



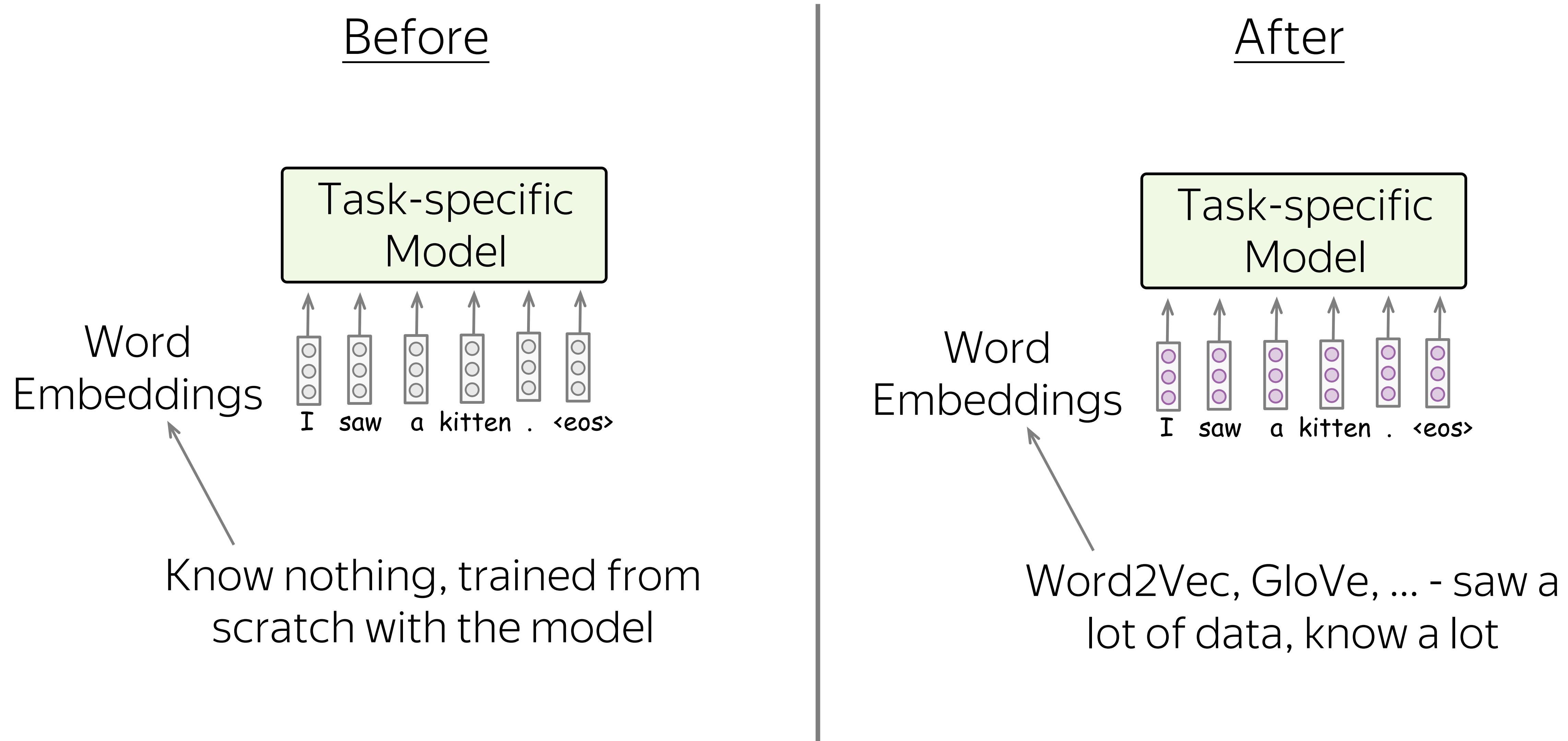
Simplest (recap once again): Word Embeddings (Word2Vec, GloVe)



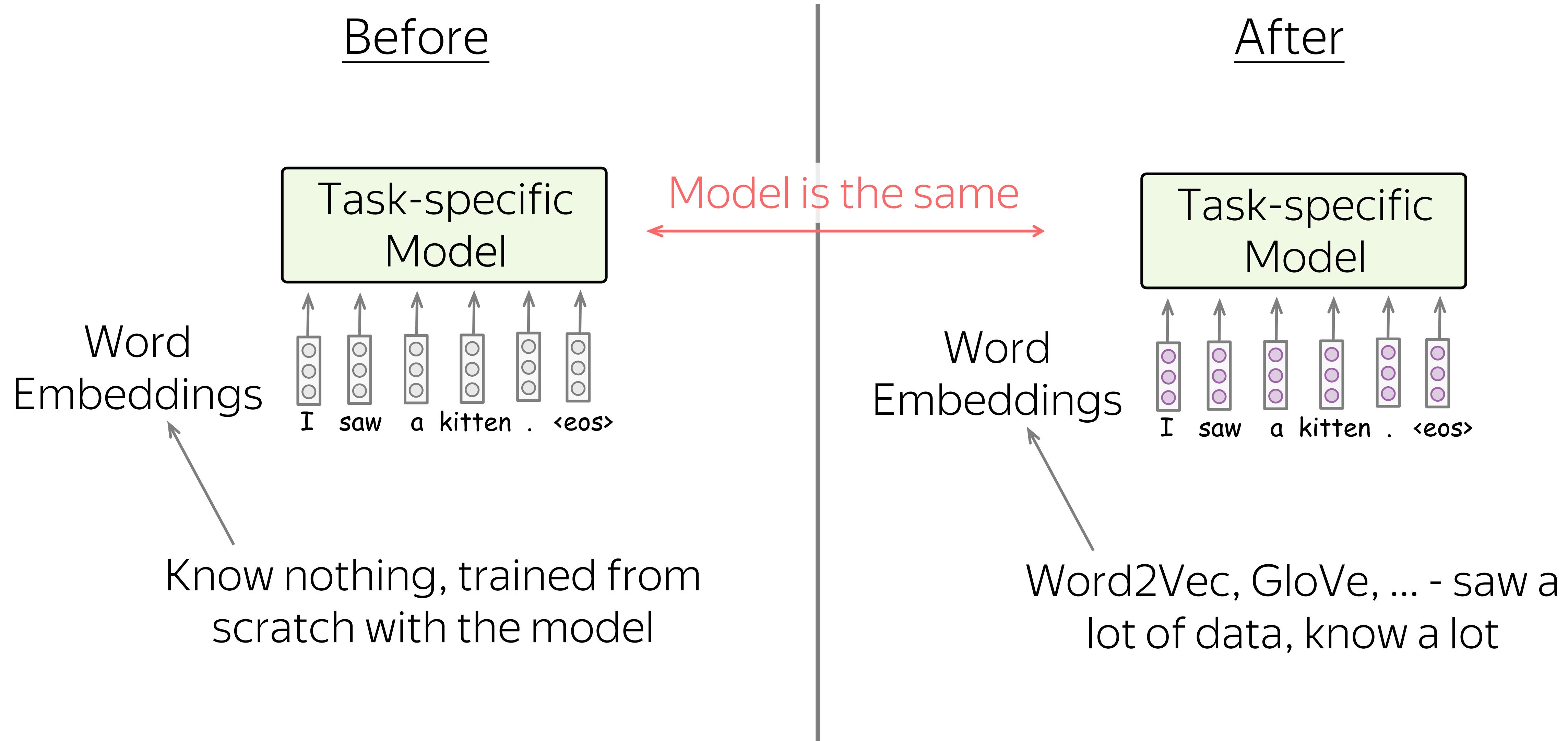
Transfer Through Word Embedding



Transfer Through Word Embedding



Transfer Through Word Embedding



What is going to happen:

- Classification
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff



- High-Level Idea
- From words to words in context
- Transformer
- BERT

Features:
from hand-crafted
to learned

Model:
from task-specific
to generalized

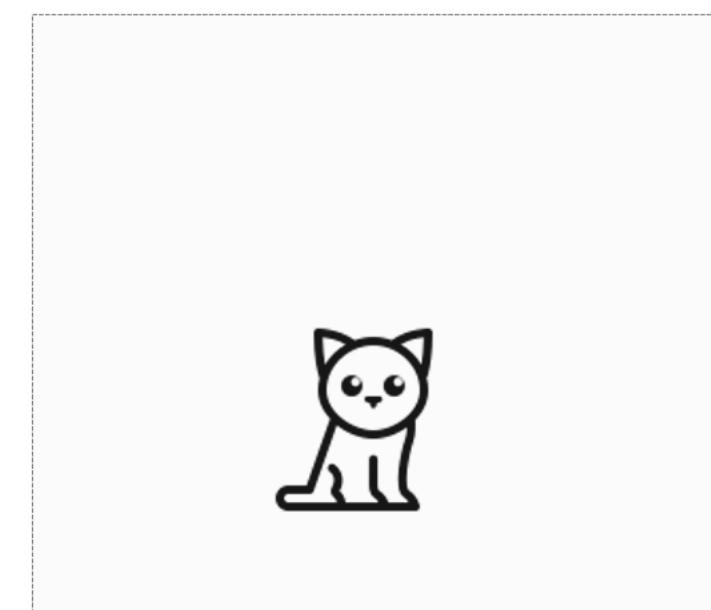
What is going to happen:

- Classification
 - Language Modeling
 - Transfer Learning →
 - High-Level Idea
 - From words to words in context
 - Transformer
 - BERT
 - A Note on Crazy Stuff
- Features:
from hand-crafted
to learned
- Model:
from task-specific
to generalized

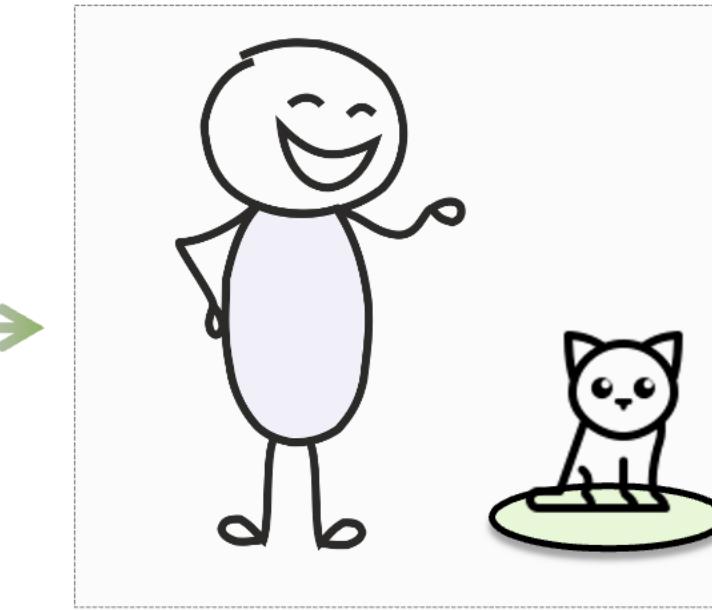
From Words to Words-in-Context

Before: representations of individual words

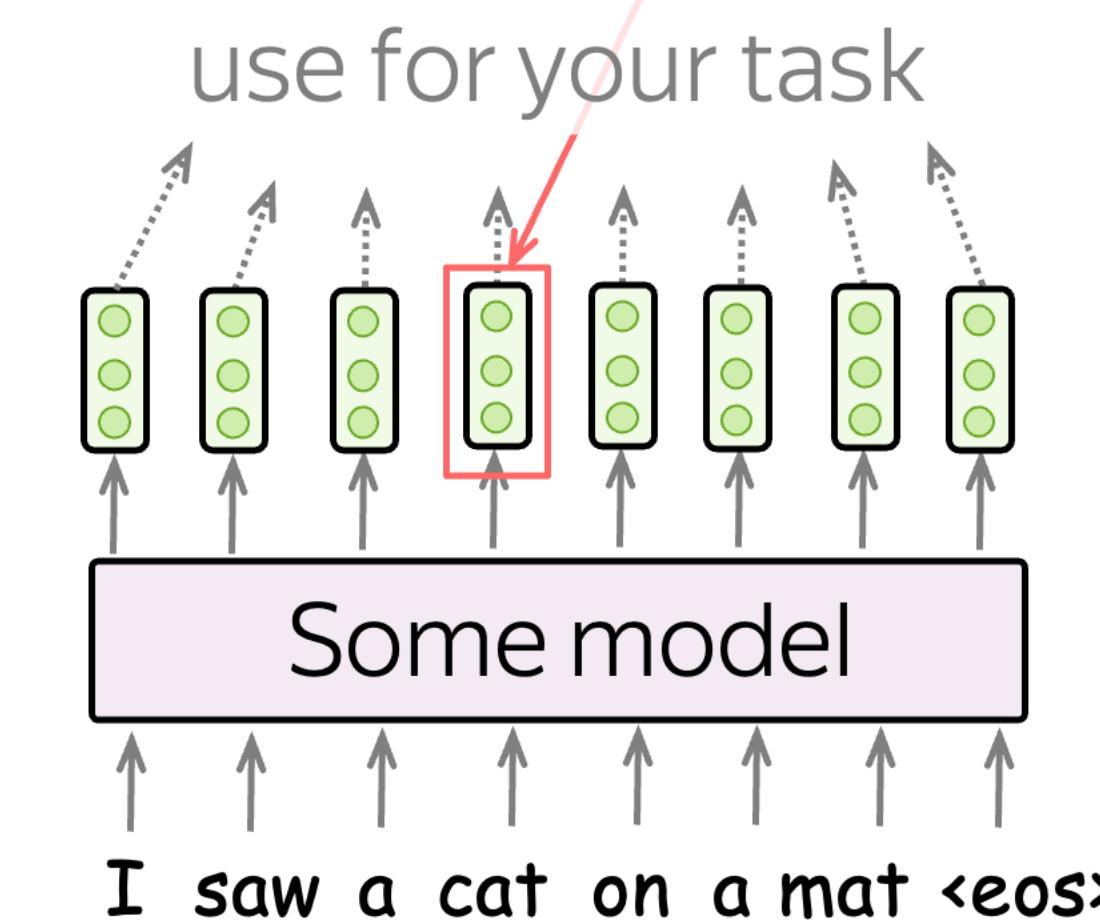
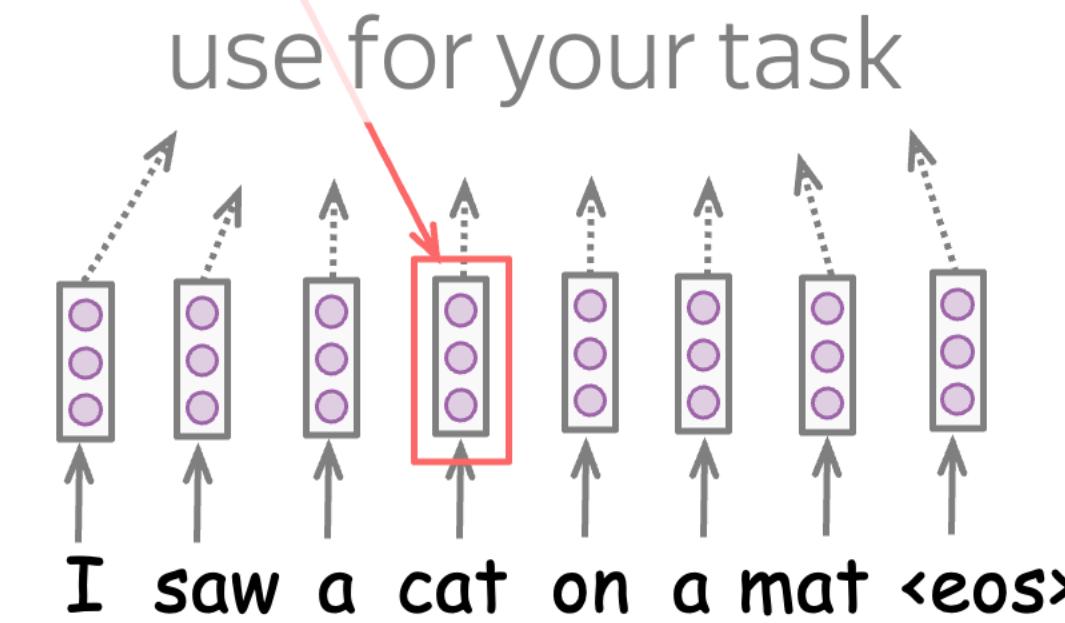
“cat”. Just “cat”



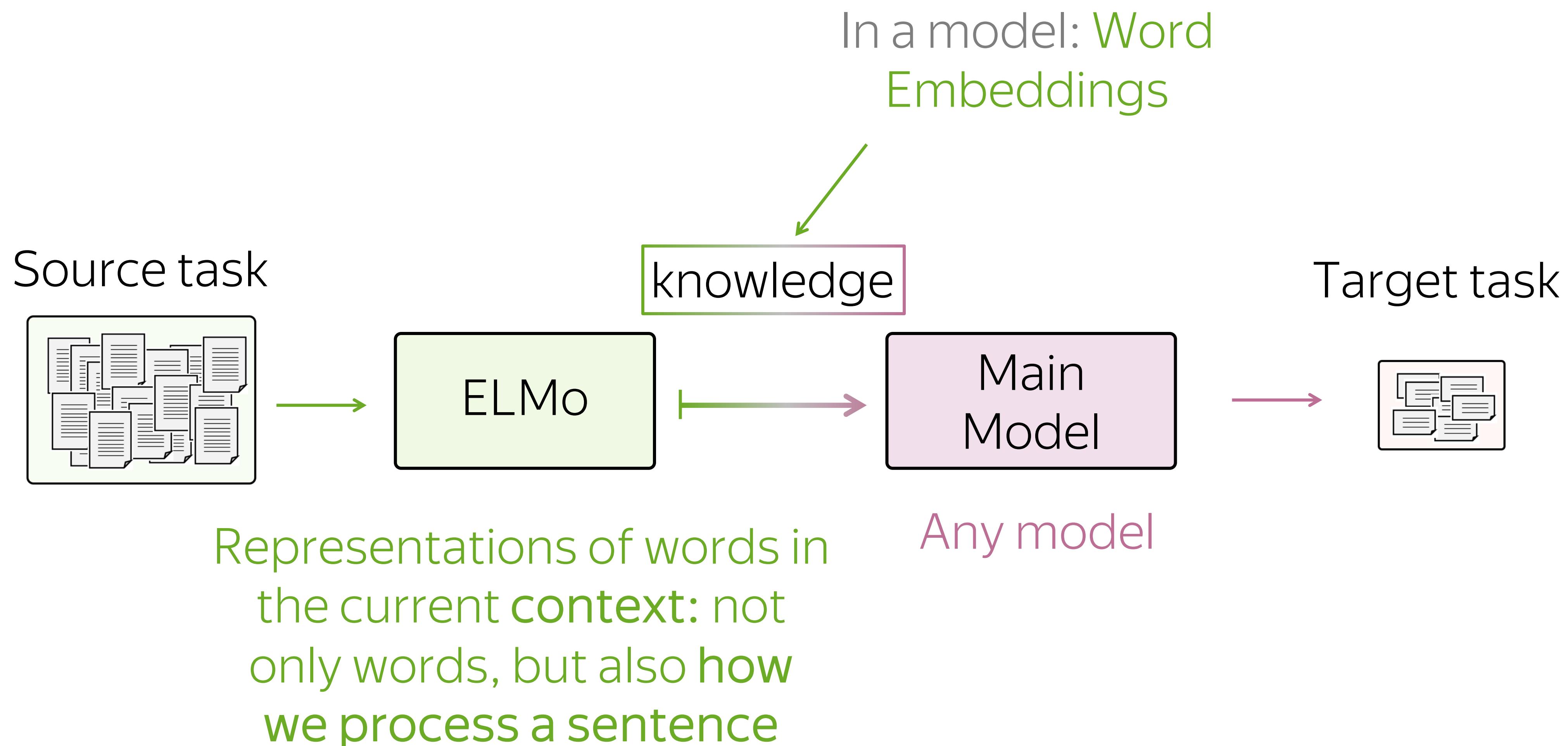
After: representations of words used in this specific context



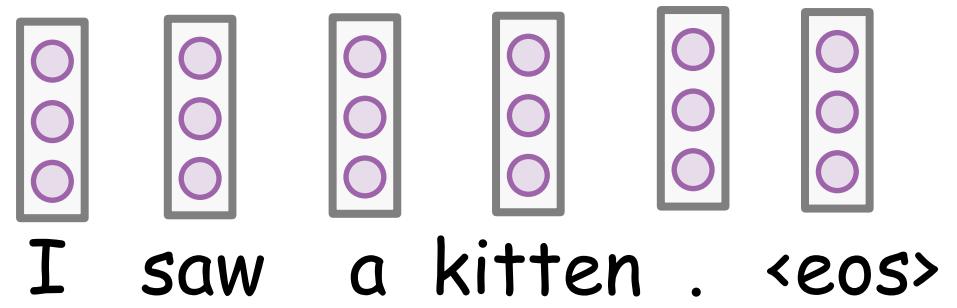
Not just any “cat”!
This is the **cat** that **I saw**, the one who **sat on the mat**



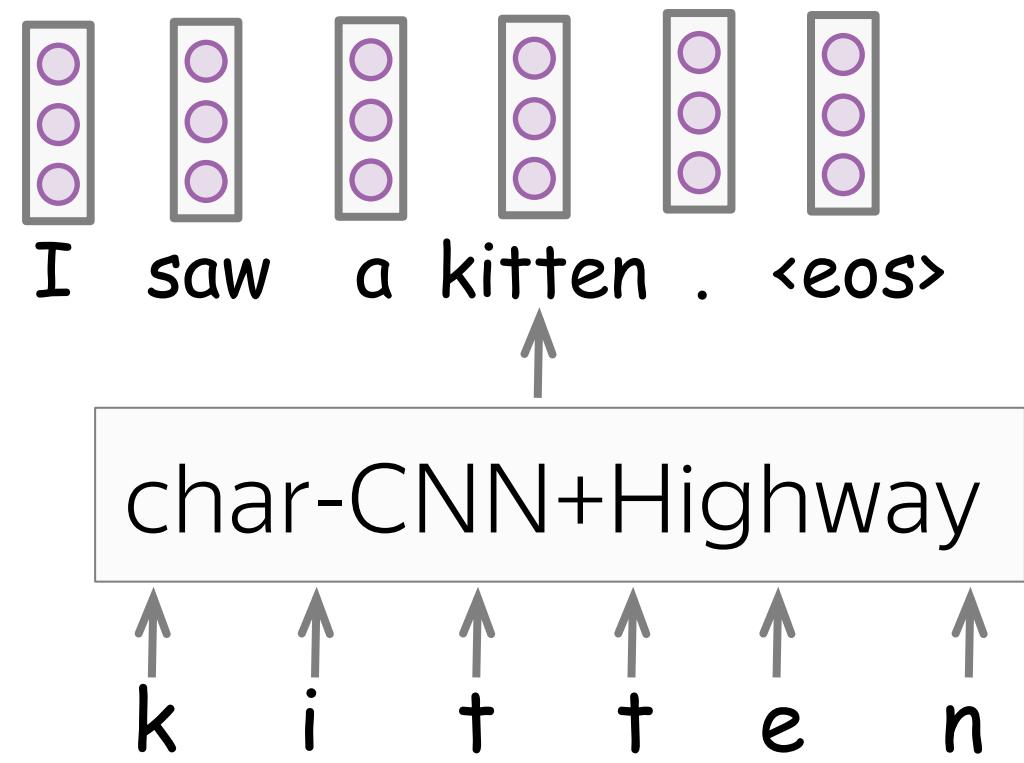
ELMo: From Words to Words-in-Context



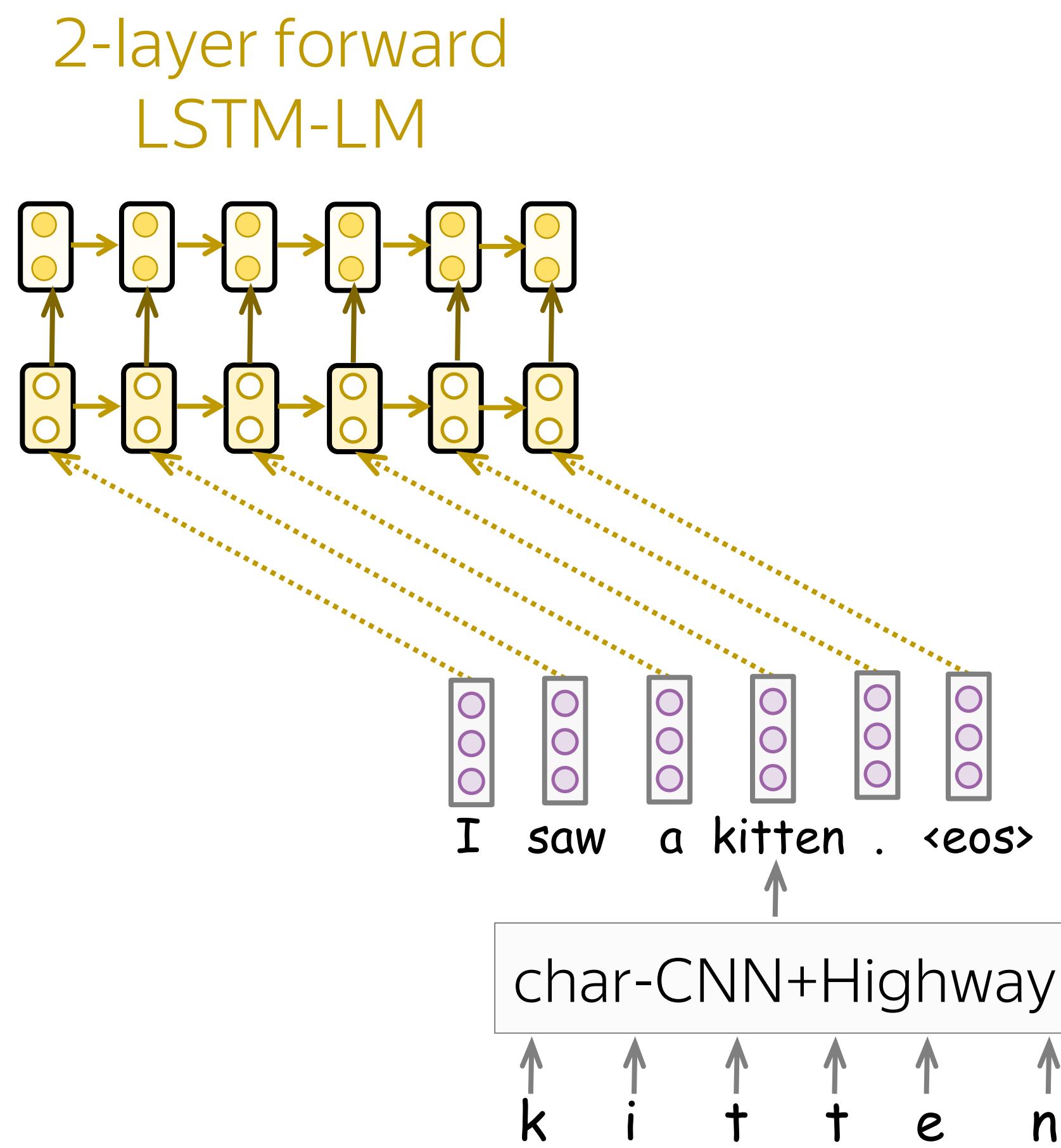
ELMo: From Words to Words-in-Context



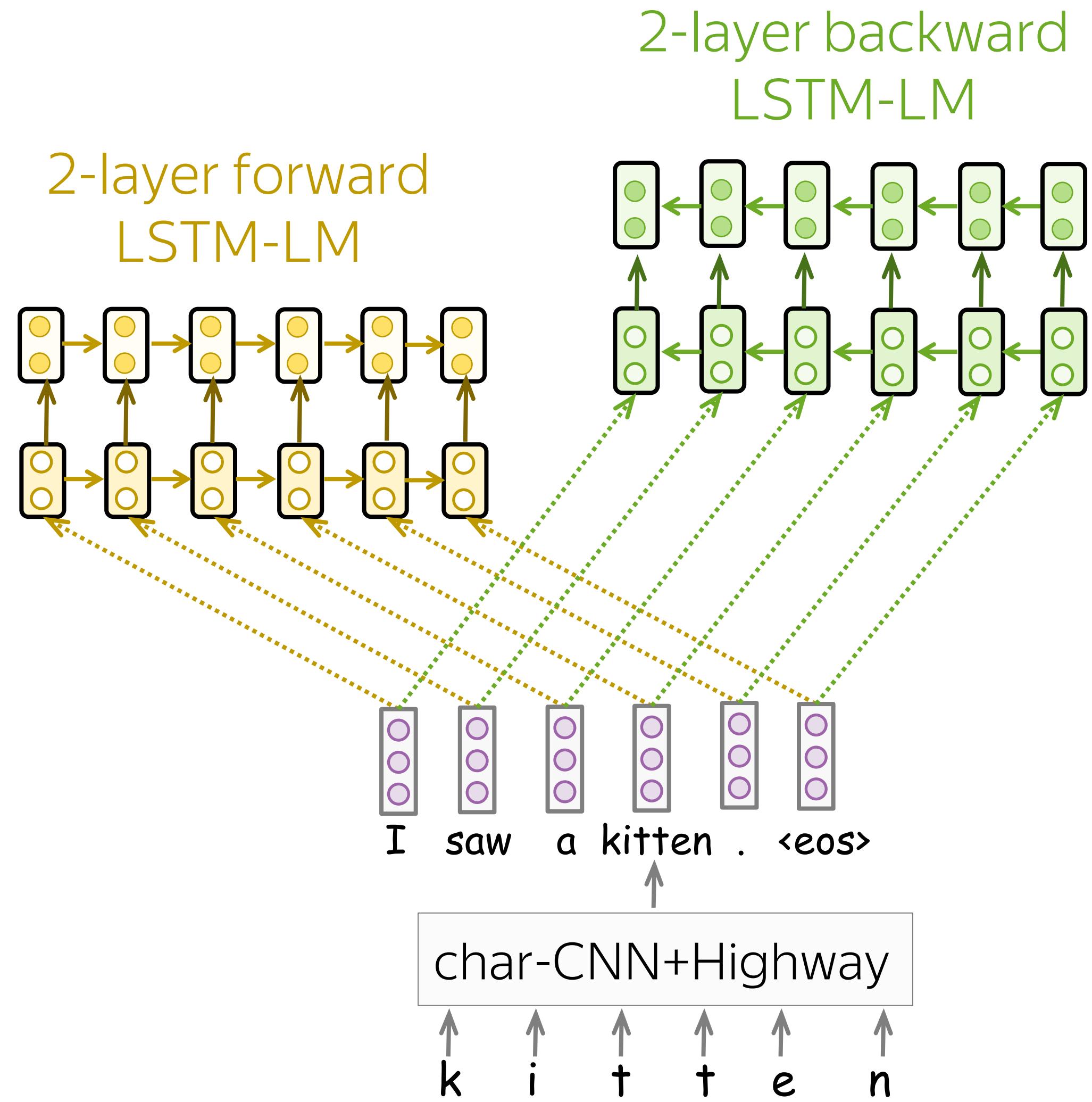
ELMo: From Words to Words-in-Context



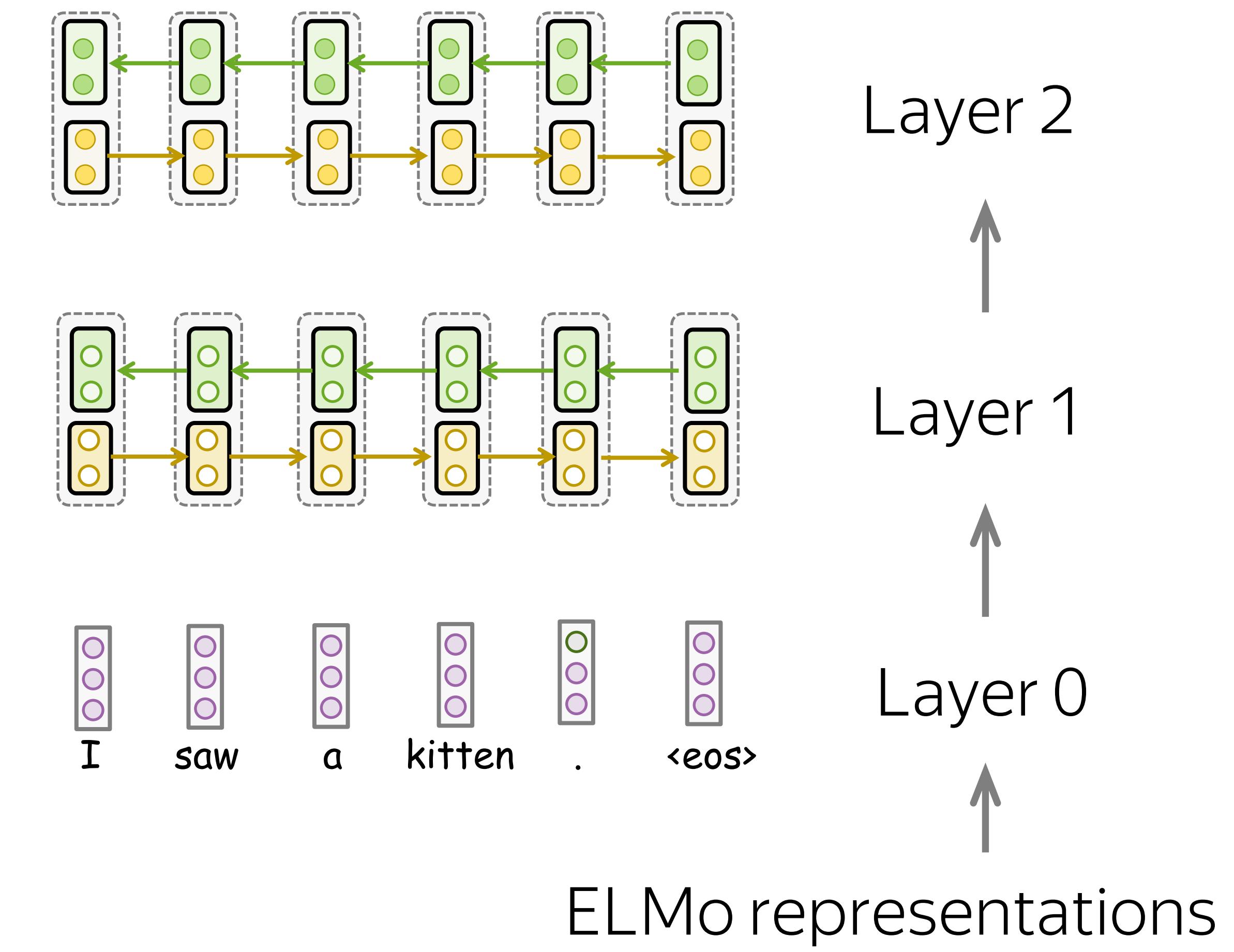
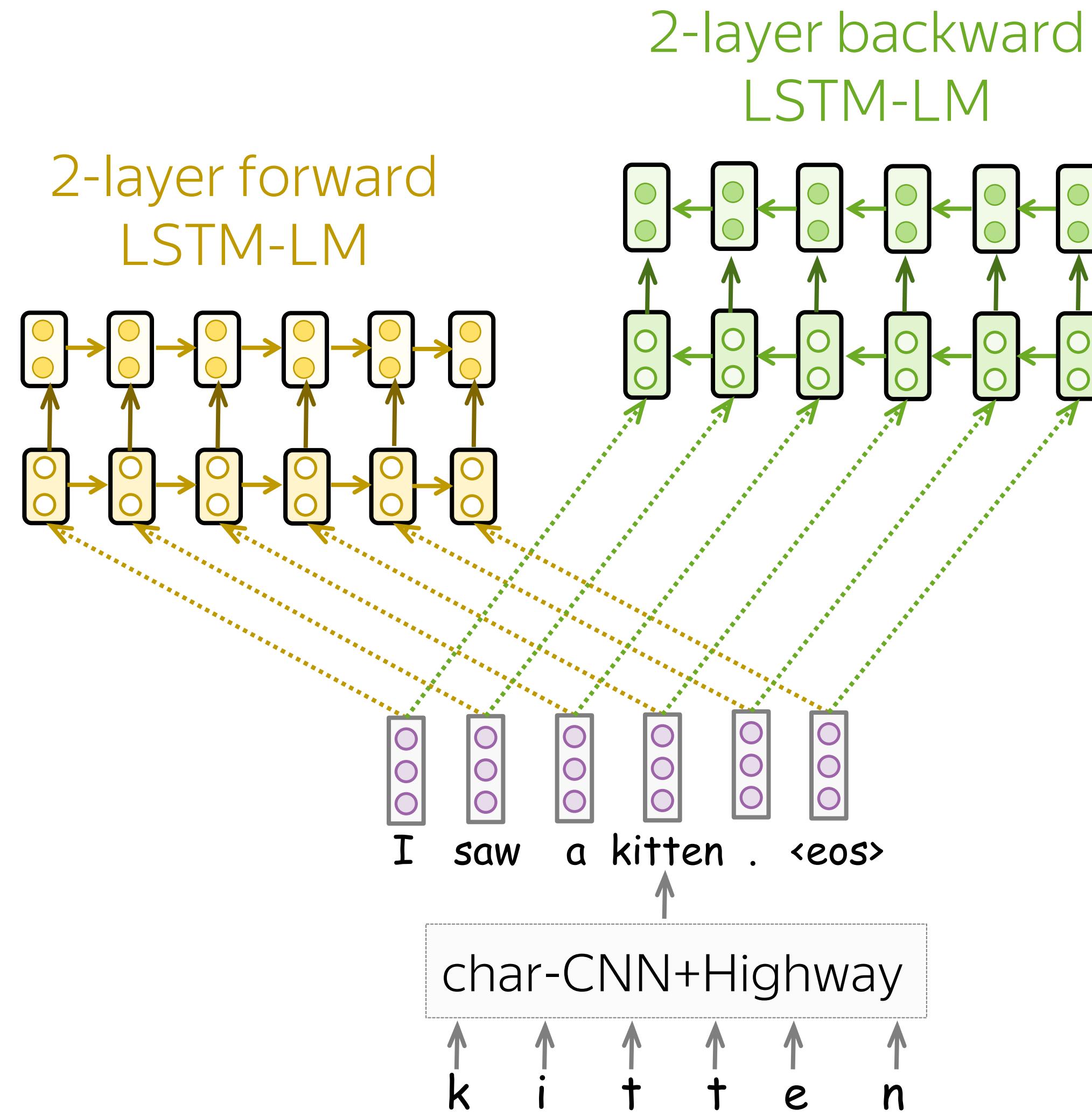
ELMo: From Words to Words-in-Context



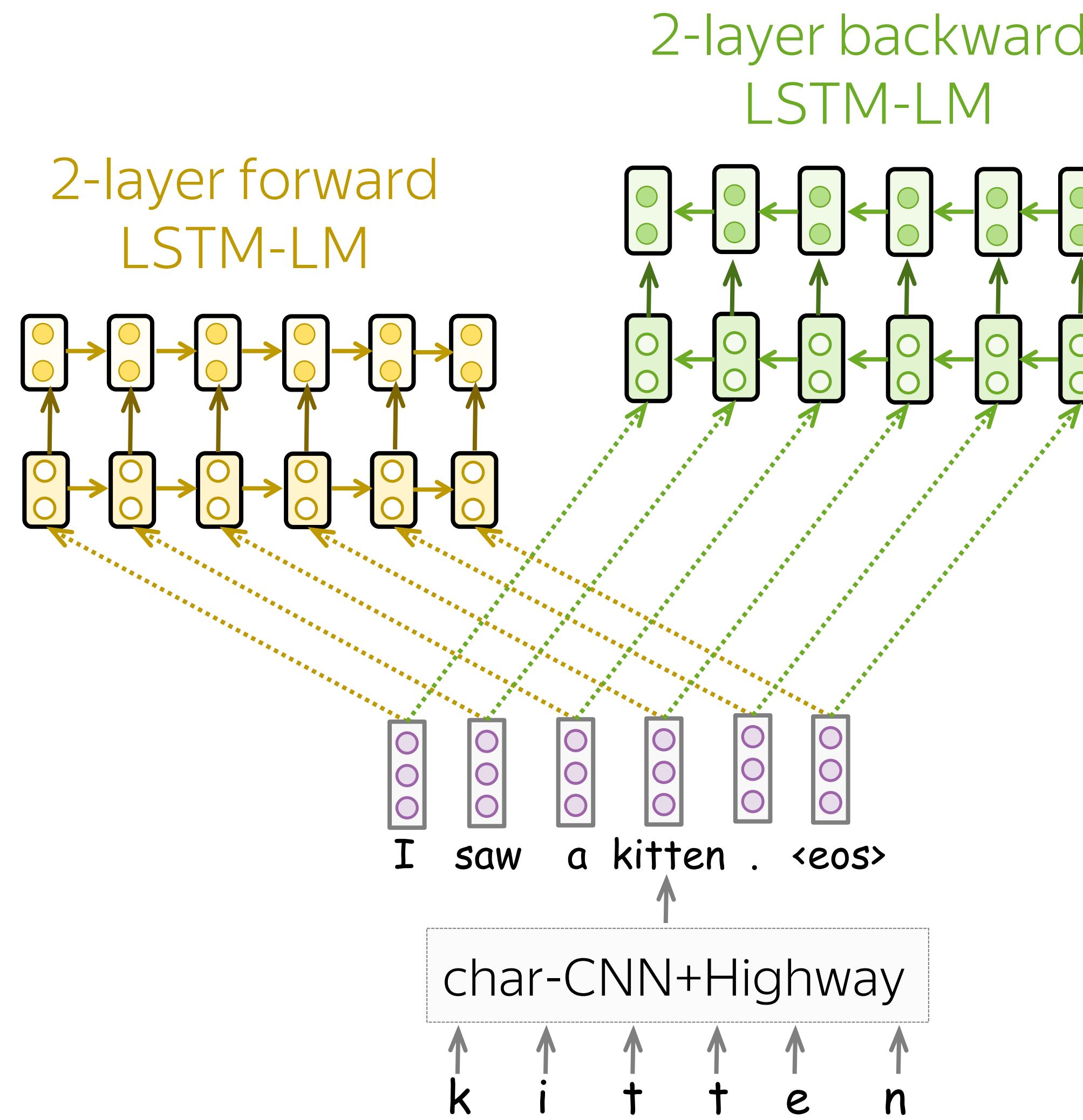
ELMo: From Words to Words-in-Context



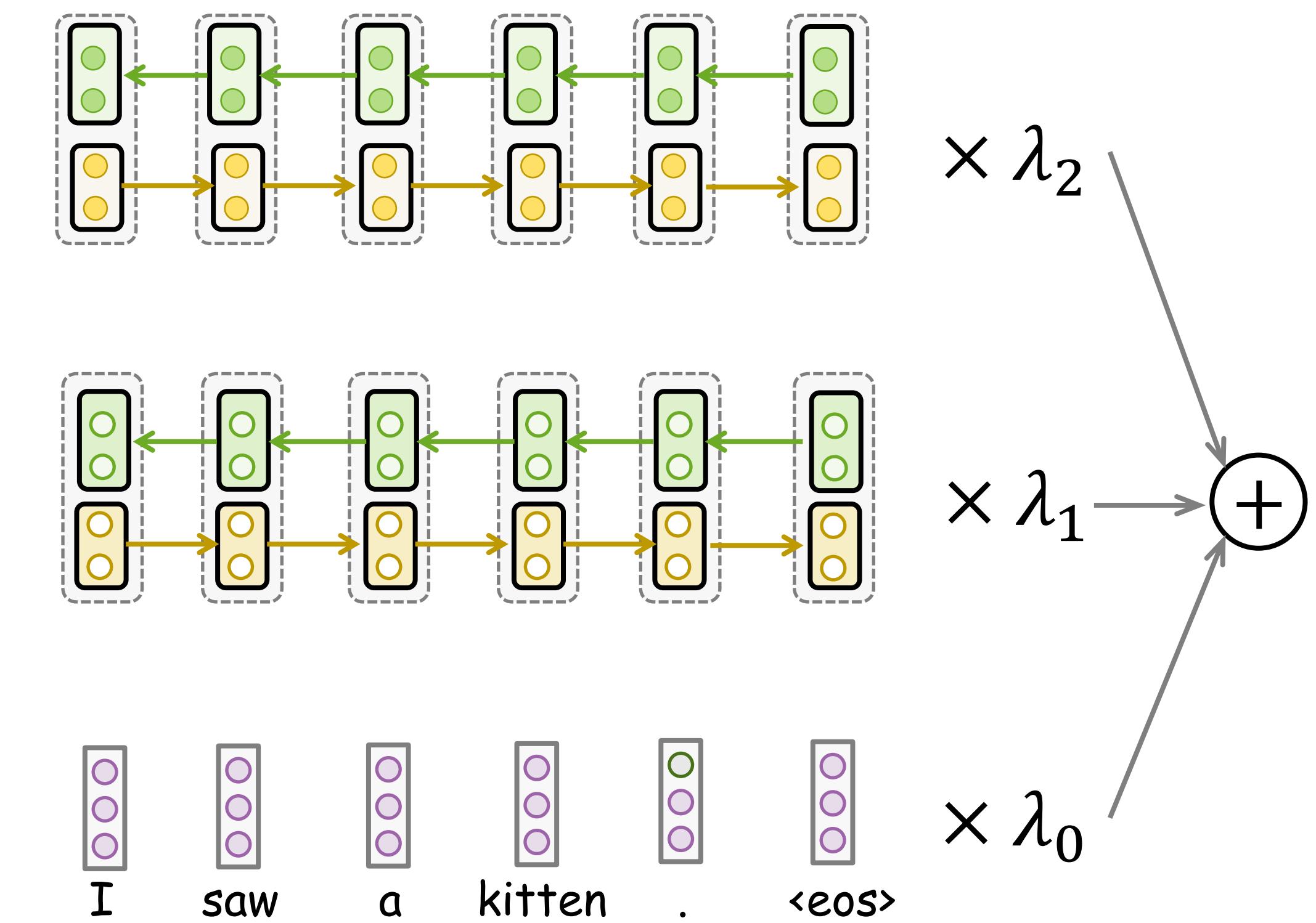
ELMo: From Words to Words-in-Context



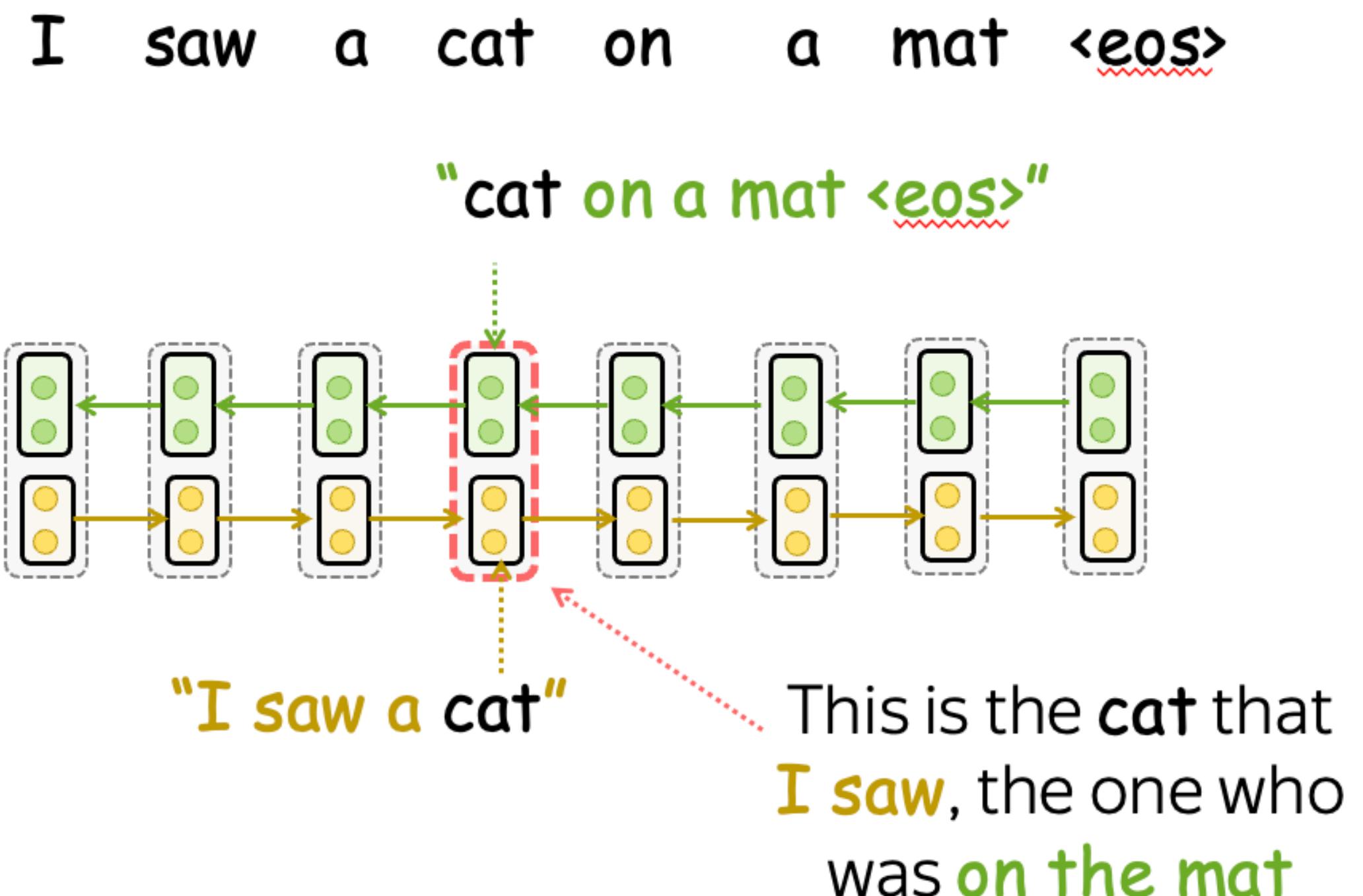
ELMo: From Words to Words-in-Context



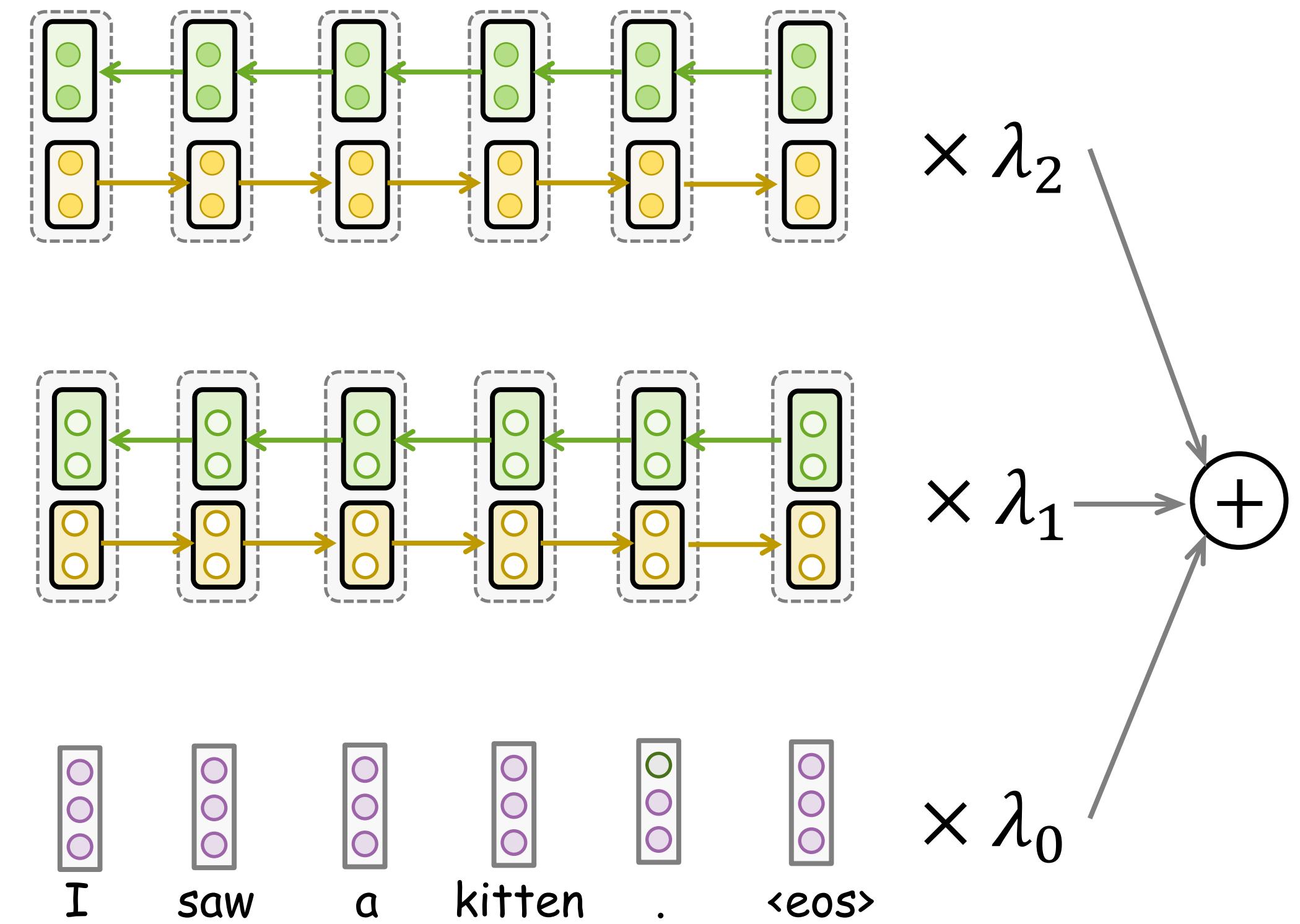
Learn specific $\lambda_0, \lambda_1, \lambda_2$ for each task



ELMo: From Words to Words-in-Context

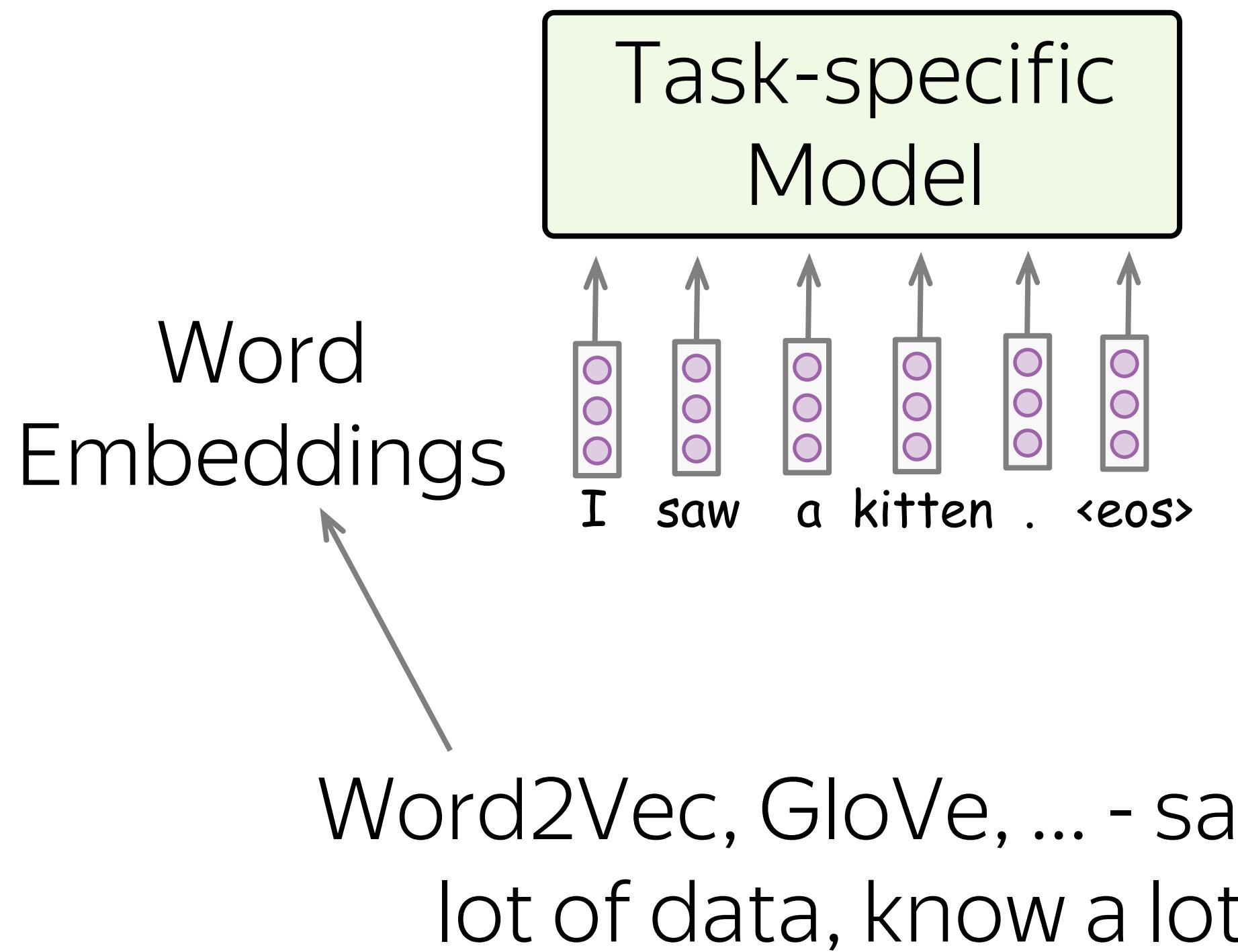


Learn specific $\lambda_0, \lambda_1, \lambda_2$ for each task

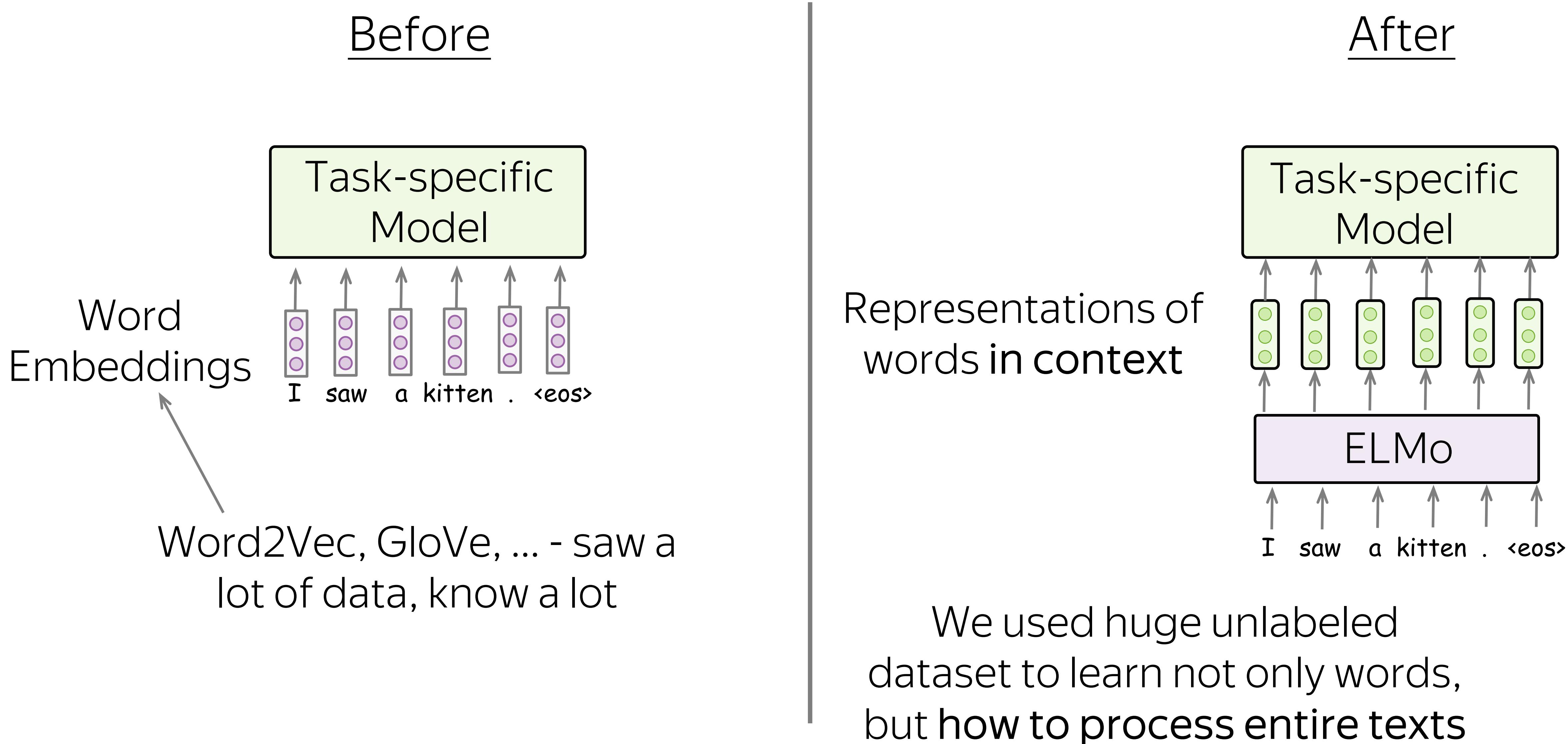


ELMo: How to Use?

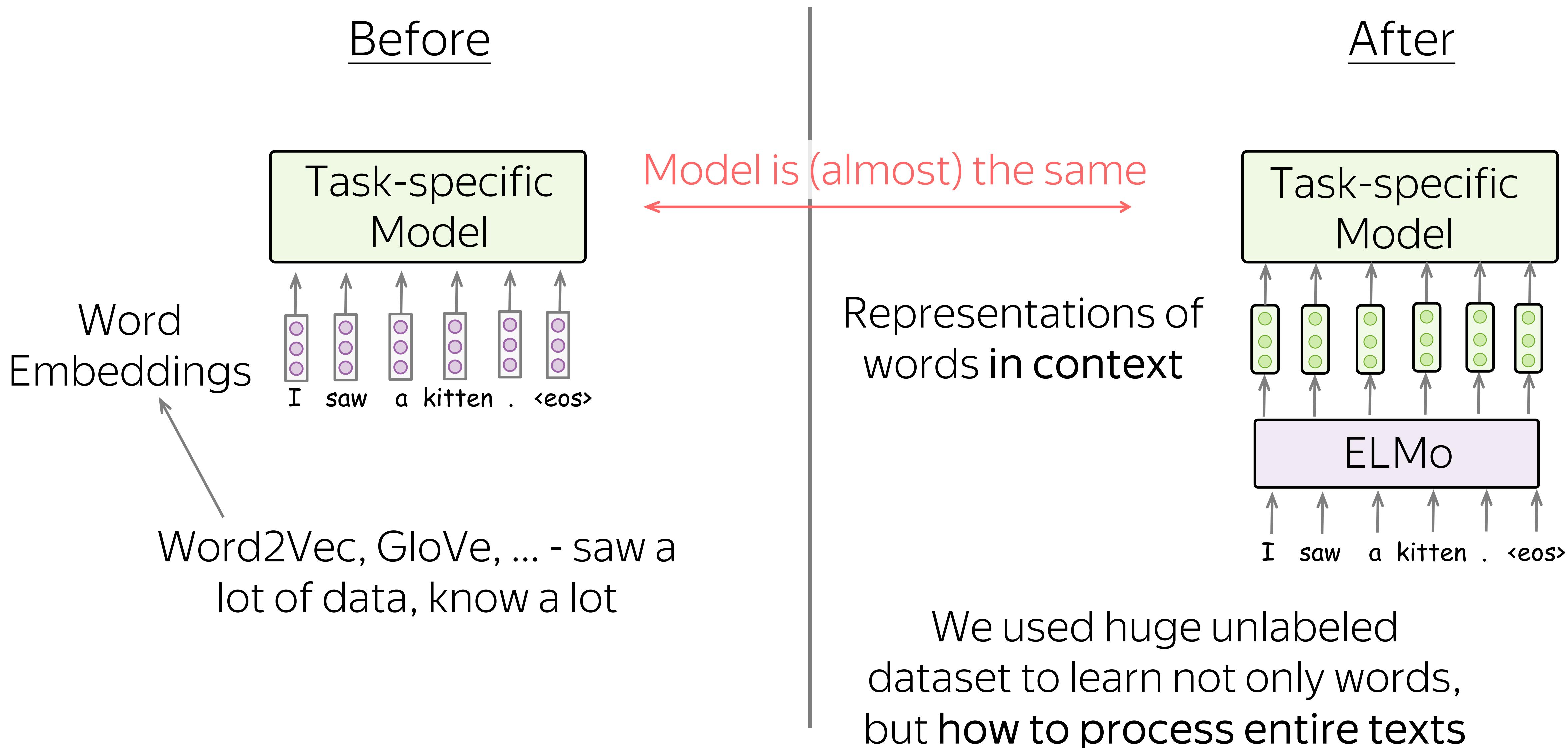
Before



ELMo: How to Use?



ELMo: How to Use?



What is going to happen:

- Classification
 - Language Modeling
 - Transfer Learning →
 - High-Level Idea
 - From words to words in context
 - Transformer
 - BERT
 - A Note on Crazy Stuff
- Features:
from hand-crafted
to learned
- Model:
from task-specific
to generalized

What is going to happen:

- Classification
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff



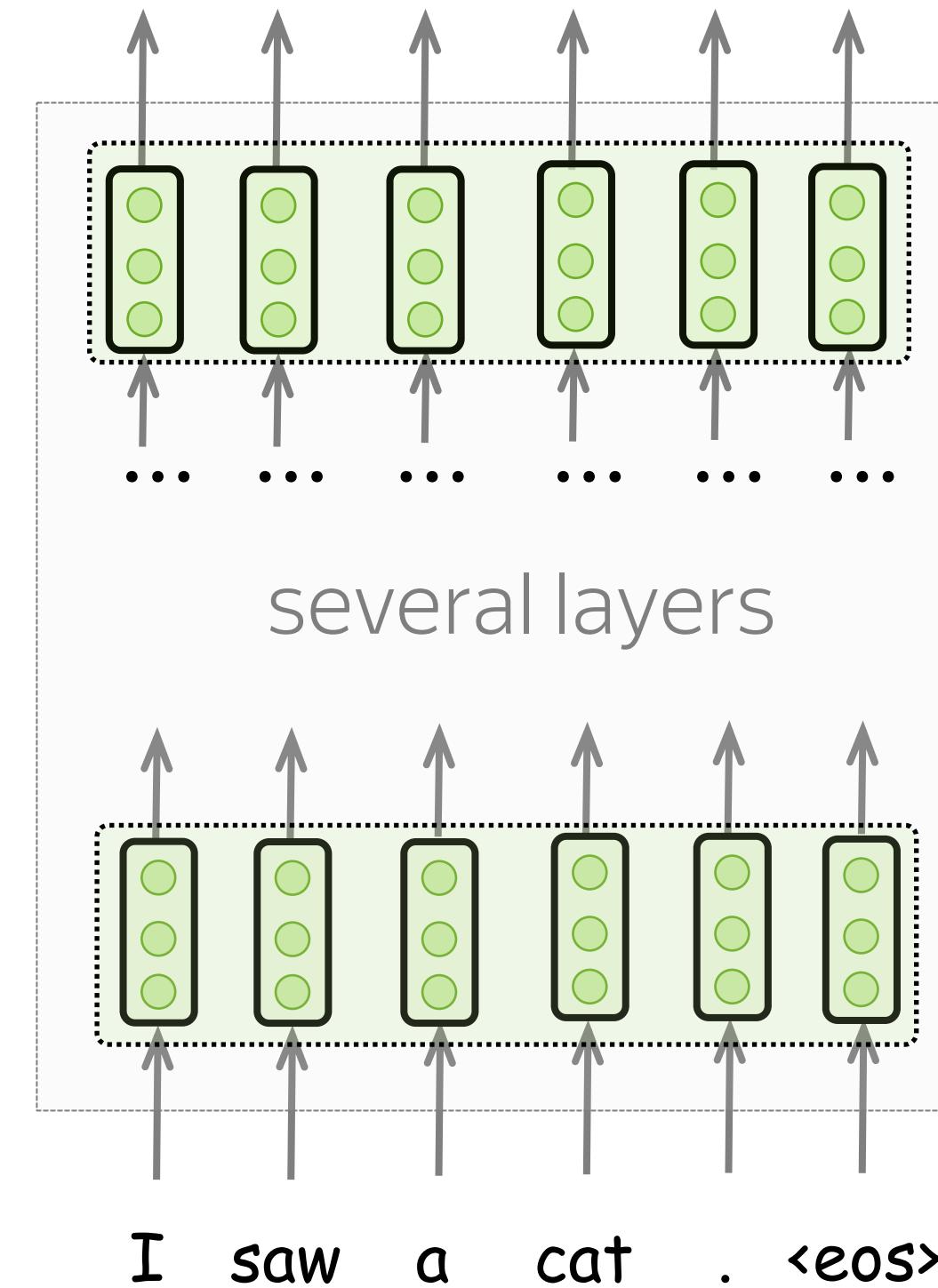
- High-Level Idea
- From words to words in context
- Transformer
- BERT

Features:
from hand-crafted
to learned

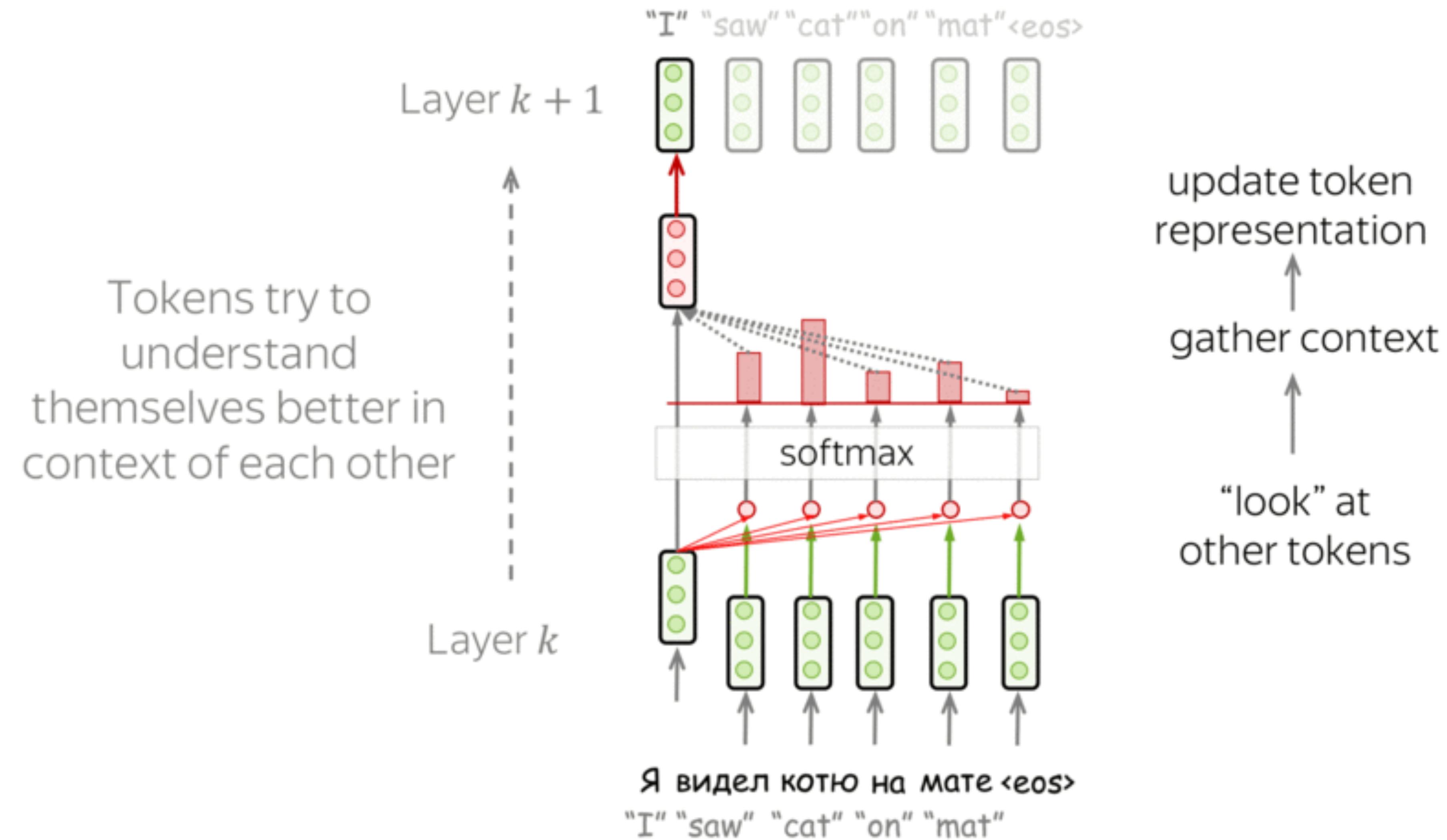
Model:
from task-specific
to generalized

Transformer Building Blocks

- Several layers with the same structure
- At each layer, tokens
 - “look” at each other
 - gather useful information
 - update their representation



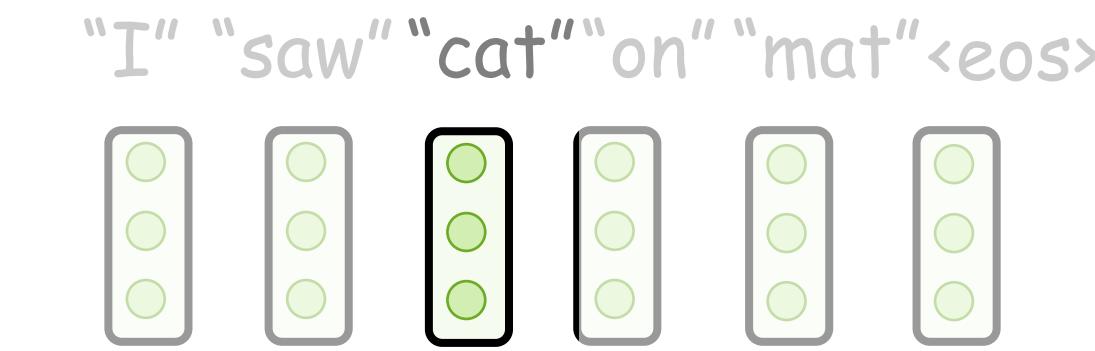
Self-Attention: “Look at Each Other”



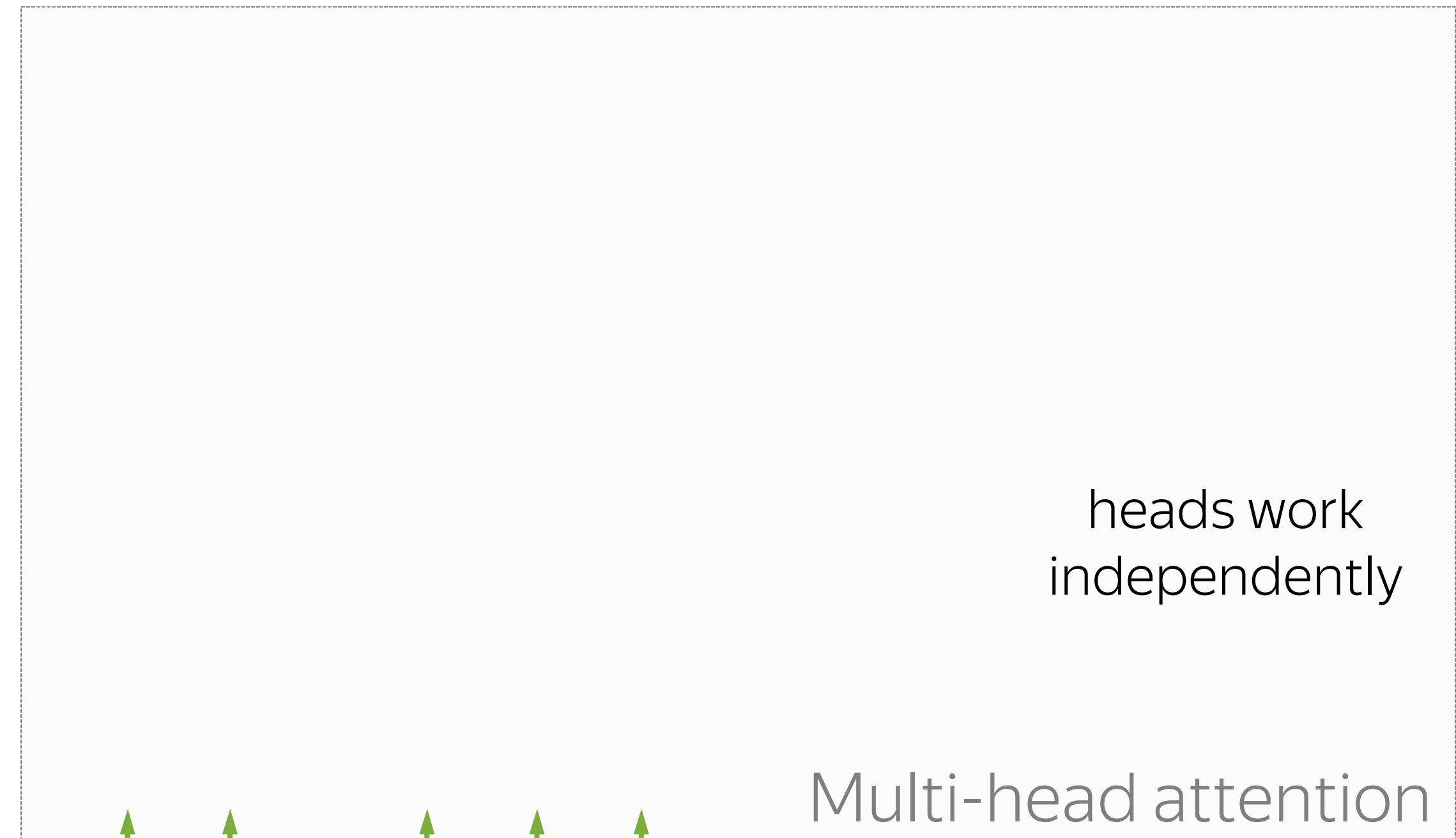
Multi-Head Attention

Idea:

- Pay attention to different types of information at the same time



heads work
independently

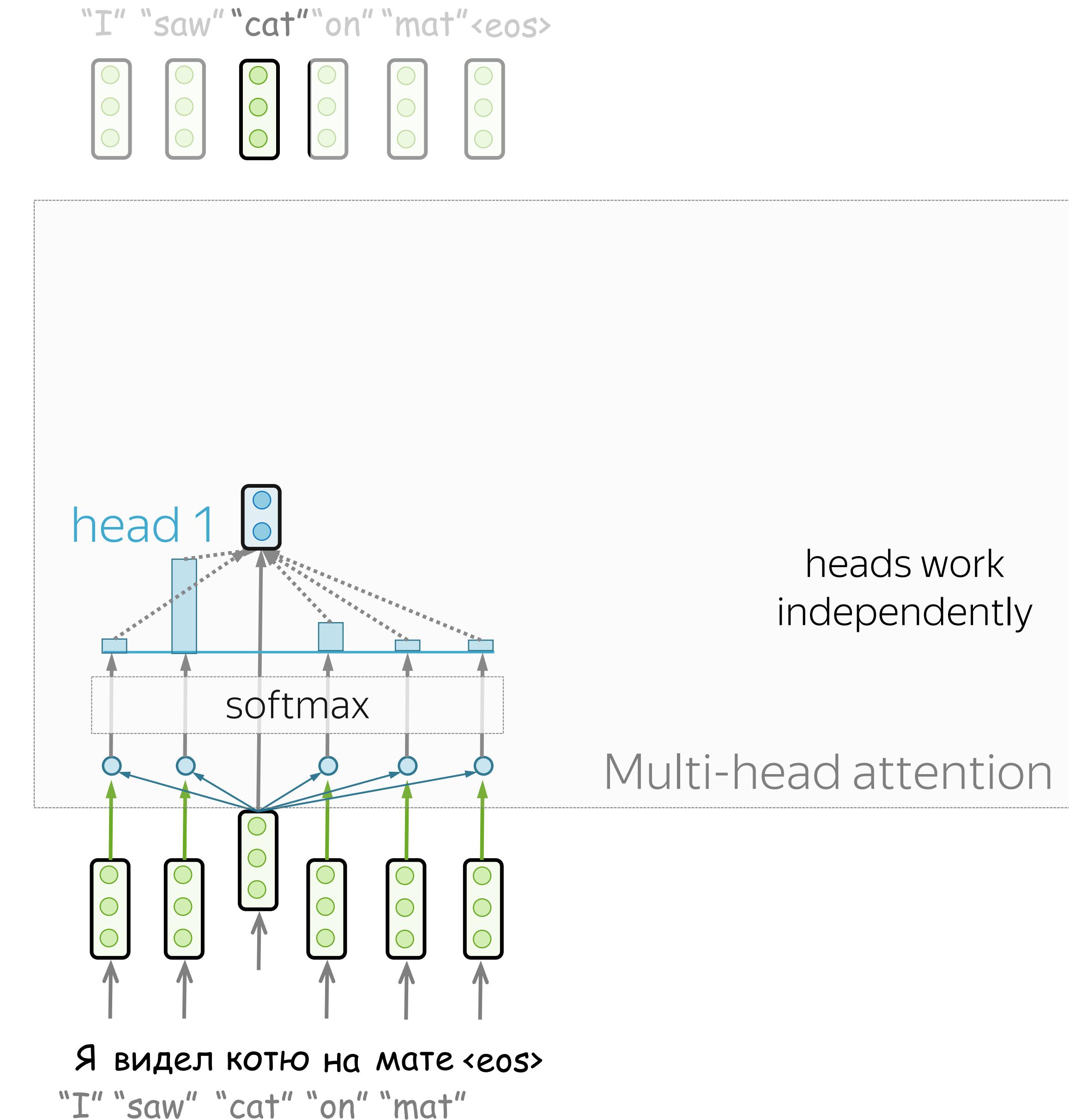


Я видел котю на мате <eos>
<I> <saw> <cat> <on> <mat>

Multi-Head Attention

Idea:

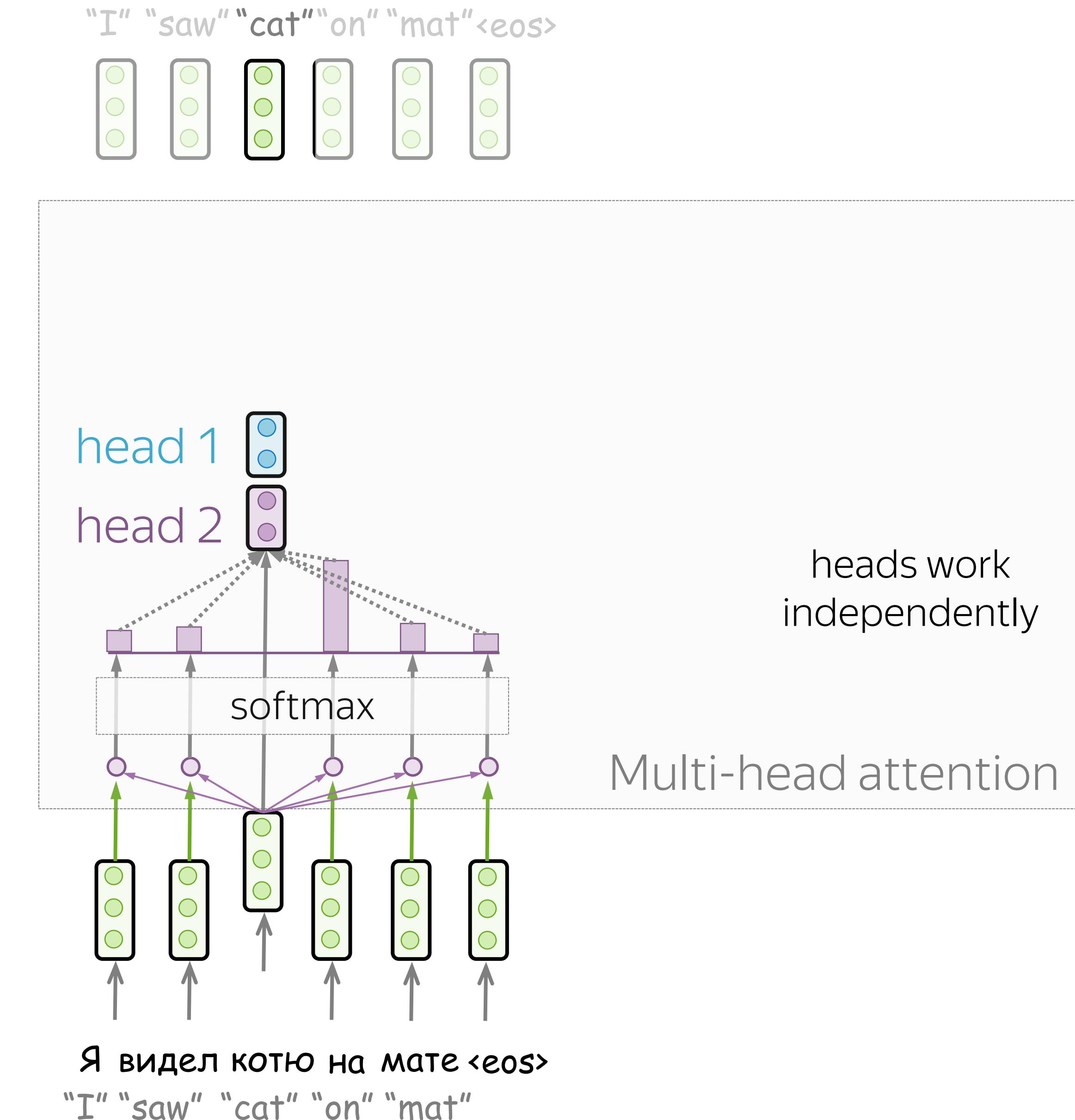
- Pay attention to different types of information at the same time



Multi-Head Attention

Idea:

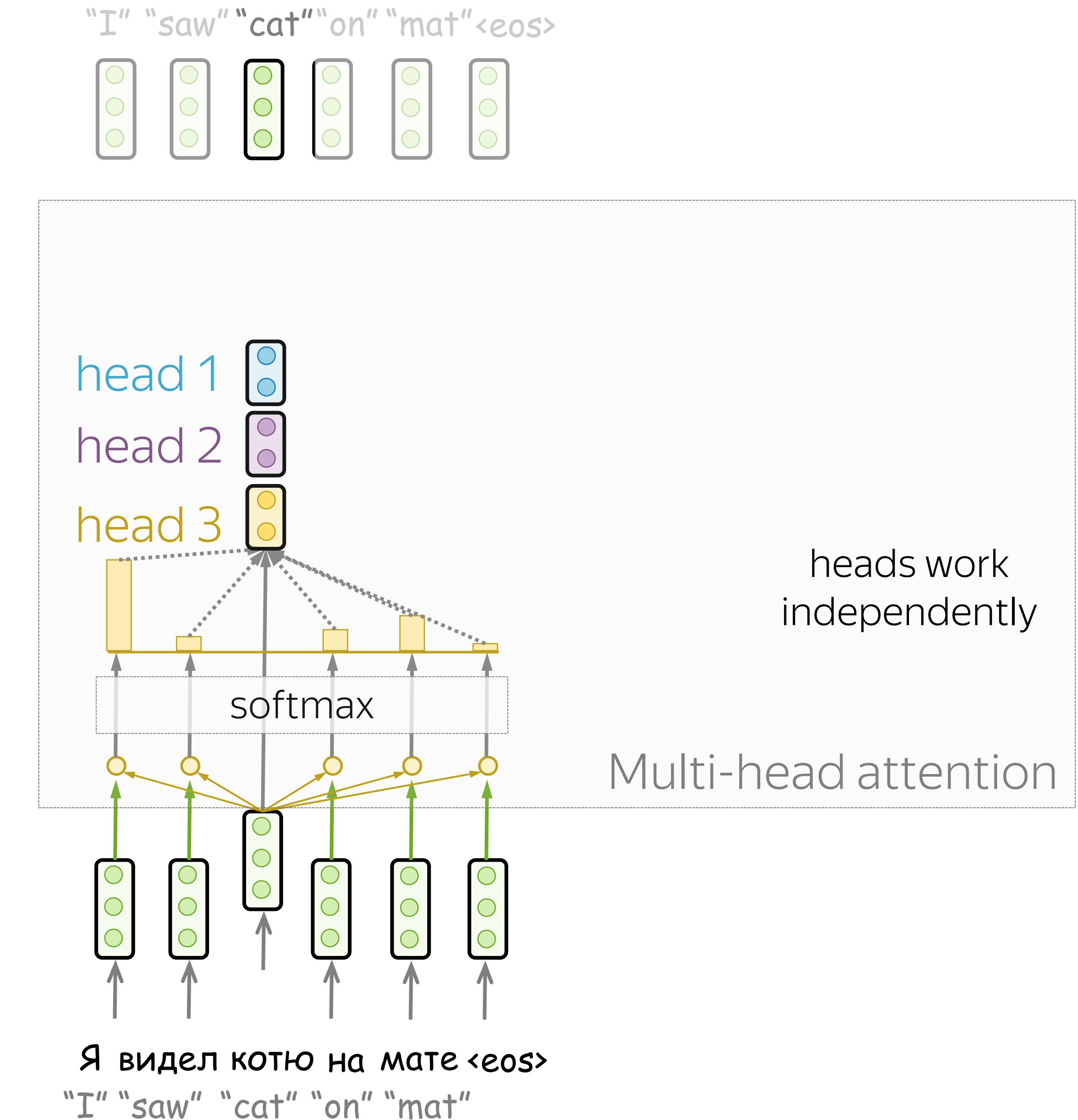
- Pay attention to different types of information at the same time



Multi-Head Attention

Idea:

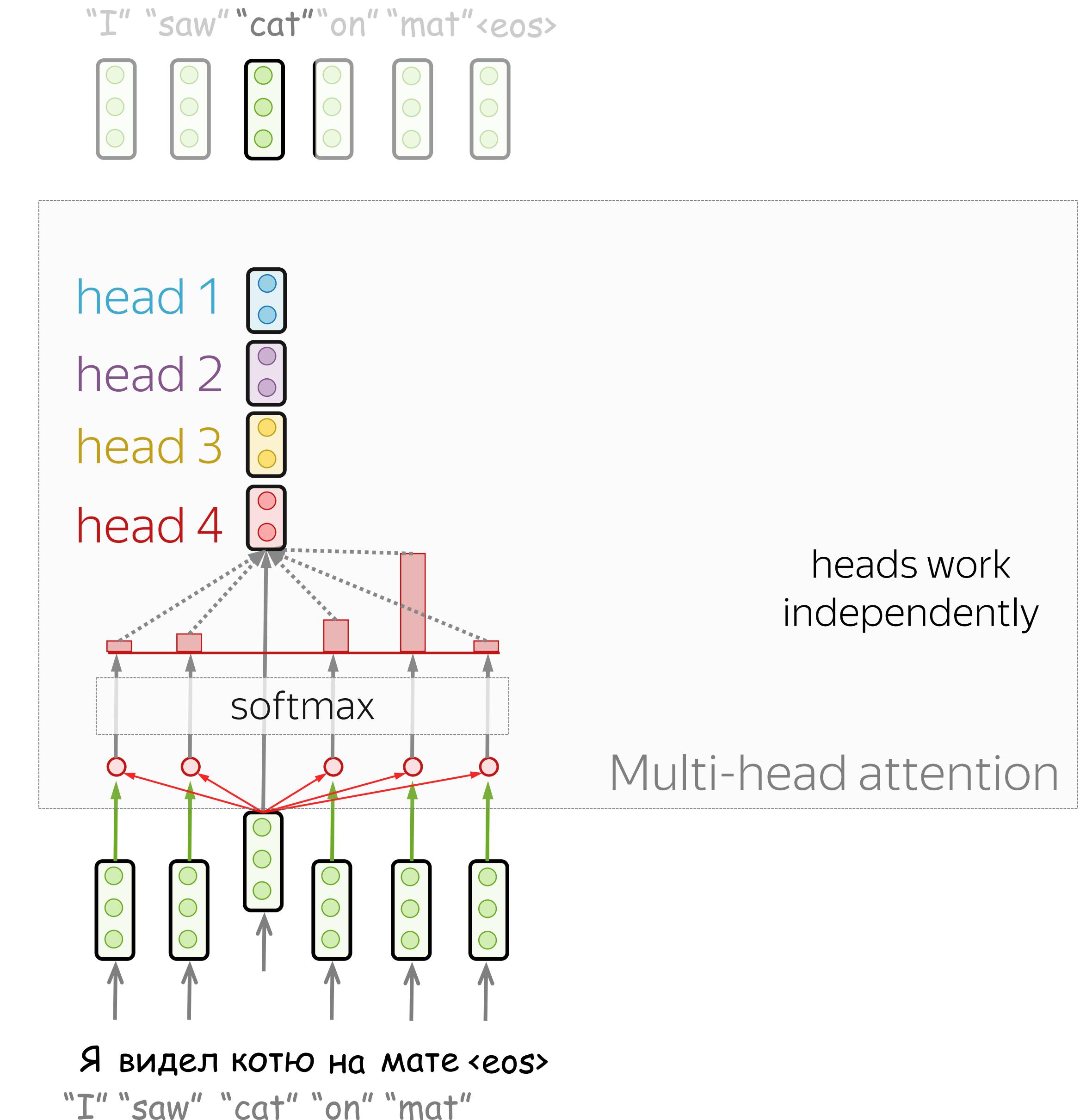
- Pay attention to different types of information at the same time



Multi-Head Attention

Idea:

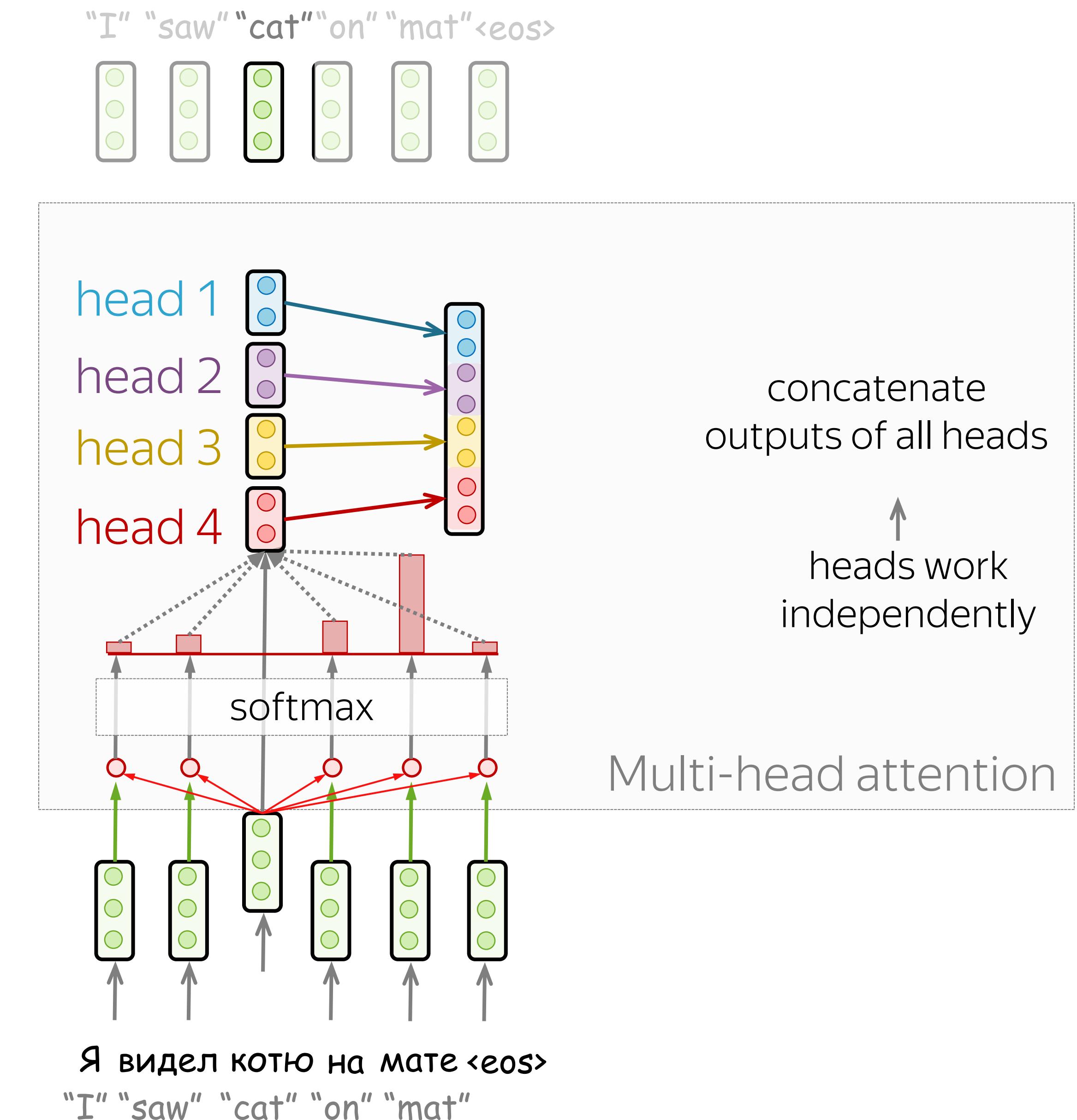
- Pay attention to different types of information at the same time



Multi-Head Attention

Idea:

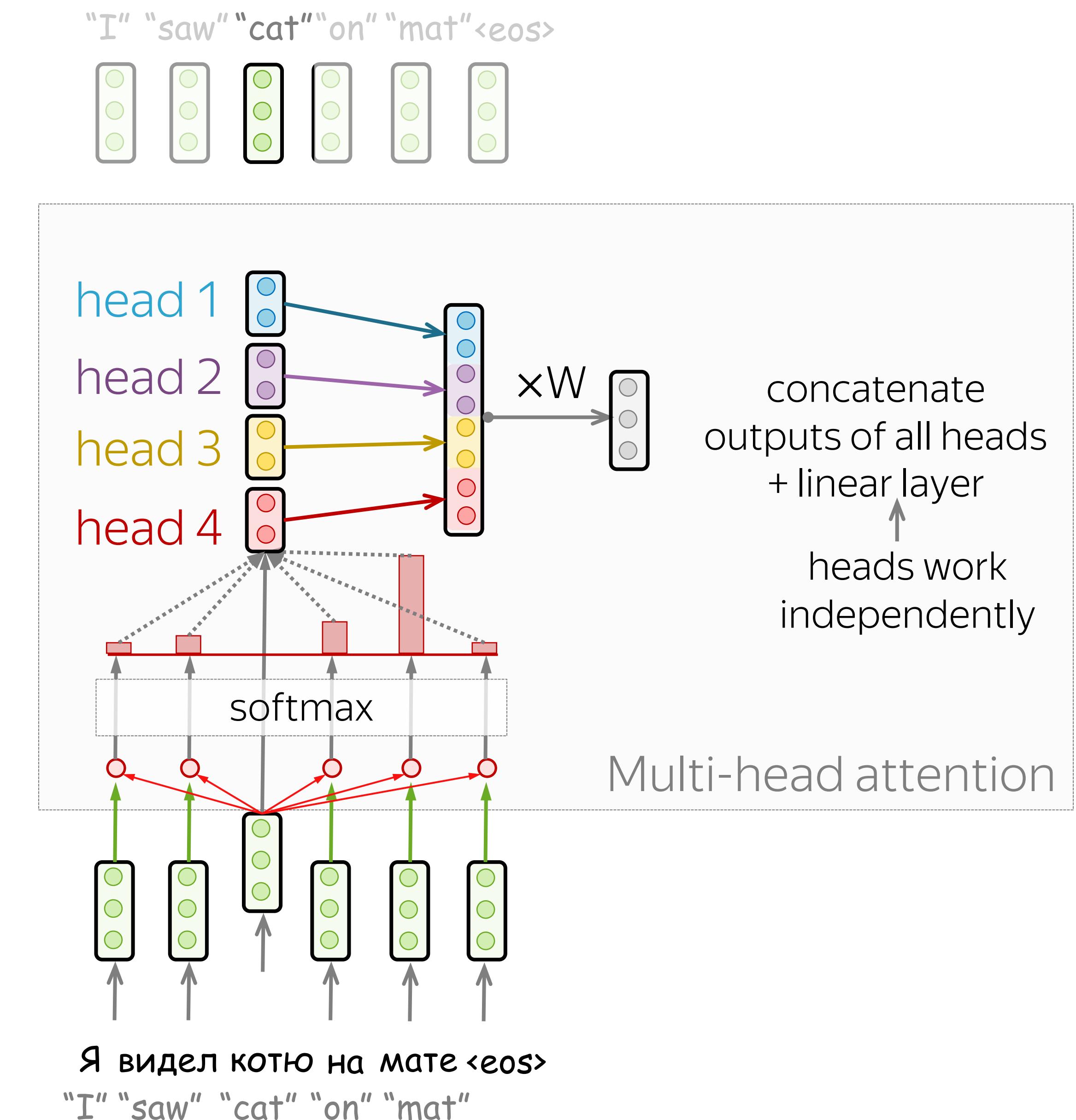
- Pay attention to different types of information at the same time



Multi-Head Attention

Idea:

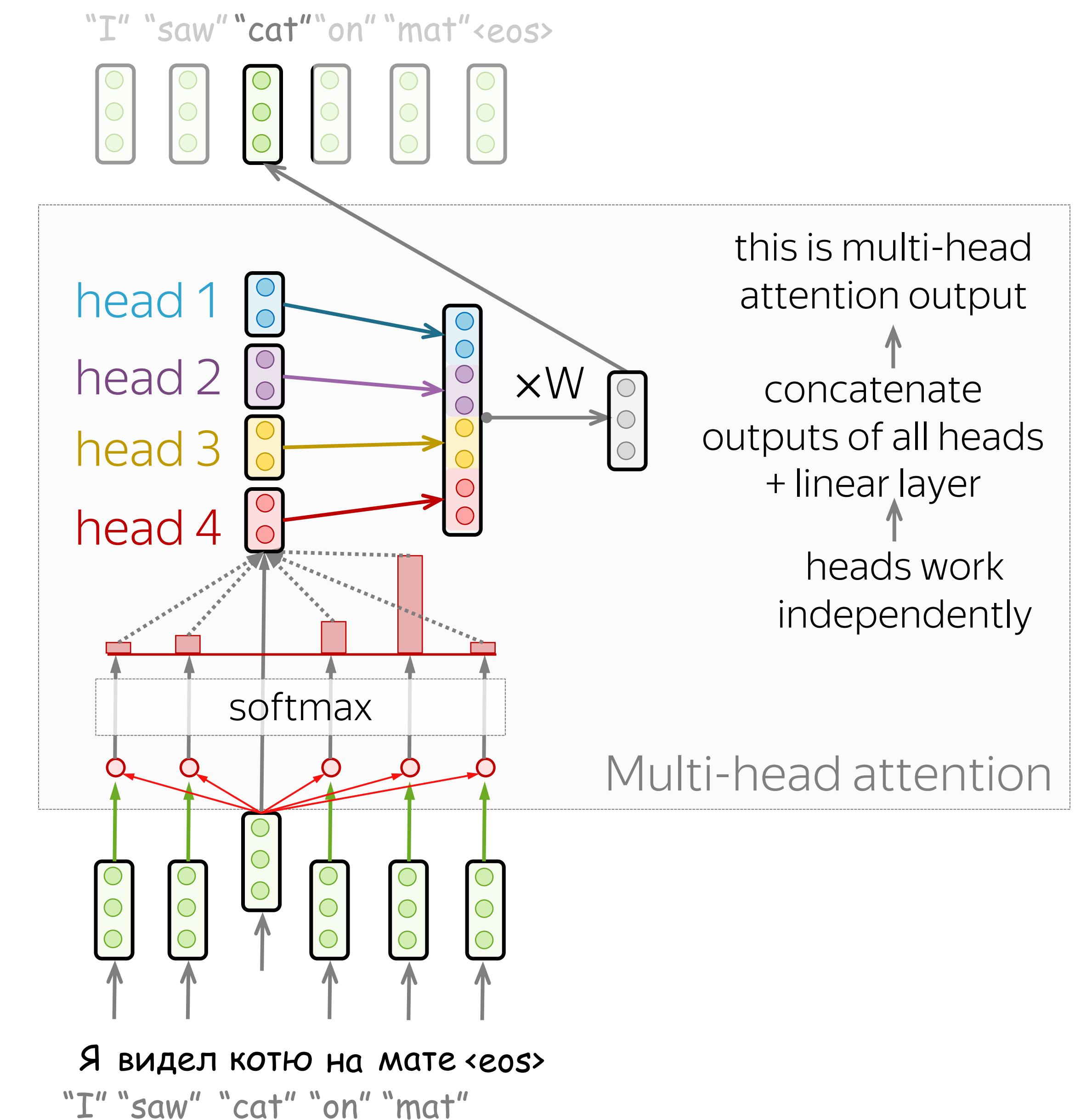
- Pay attention to different types of information at the same time



Multi-Head Attention

Idea:

- Pay attention to different types of information at the same time



What is going to happen:

- Classification
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff



- High-Level Idea
- From words to words in context
- Transformer LMs
- BERT

Features:
from hand-crafted
to learned

Model:
from task-specific
to generalized

What is going to happen:

- Classification
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff

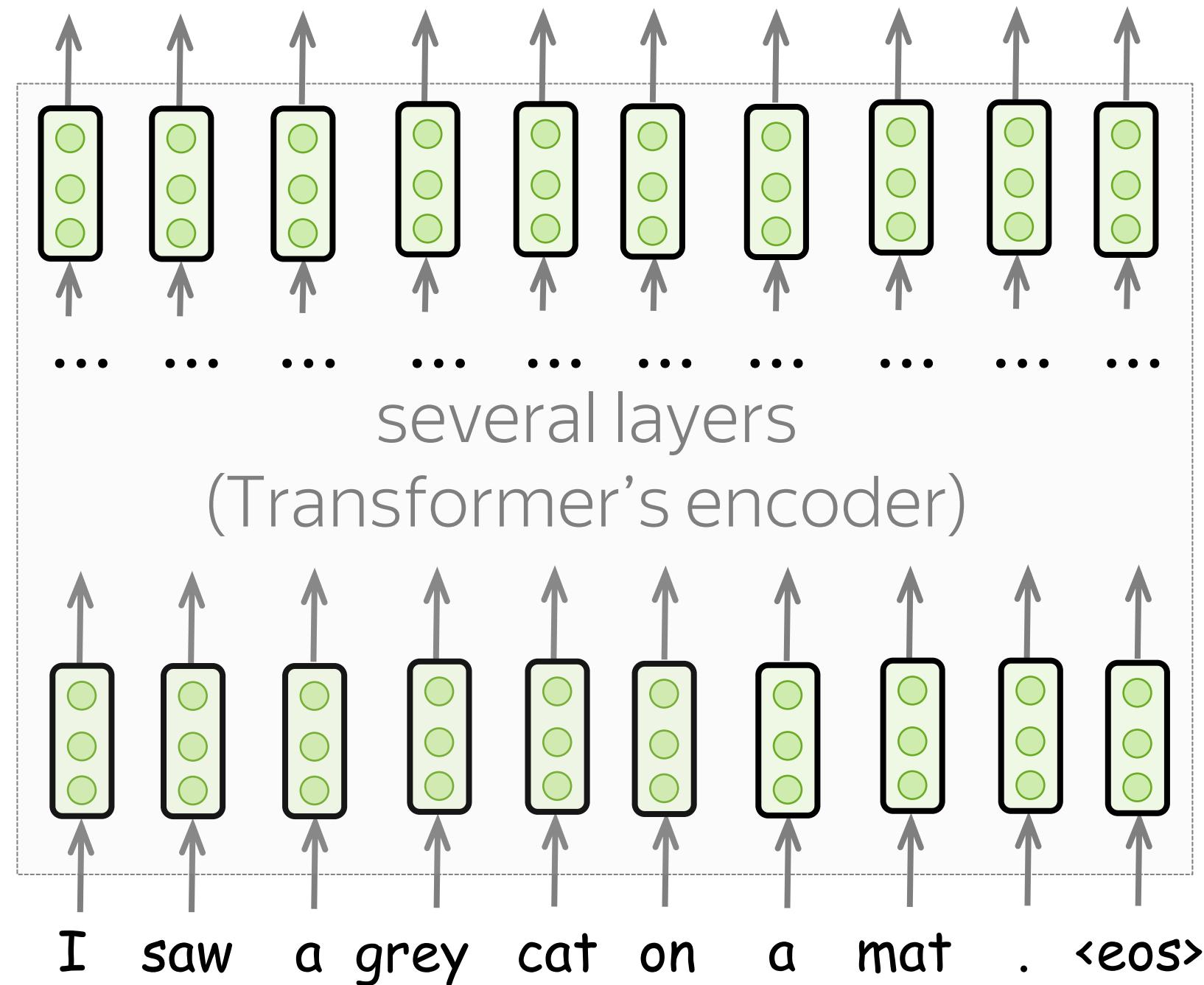


- High-Level Idea
- From words to words in context
- Transformer LMs
- BERT

Features:
from hand-crafted
to learned

Model:
from task-specific
to generalized

BERT: Transformer Encoder with Fancy Training



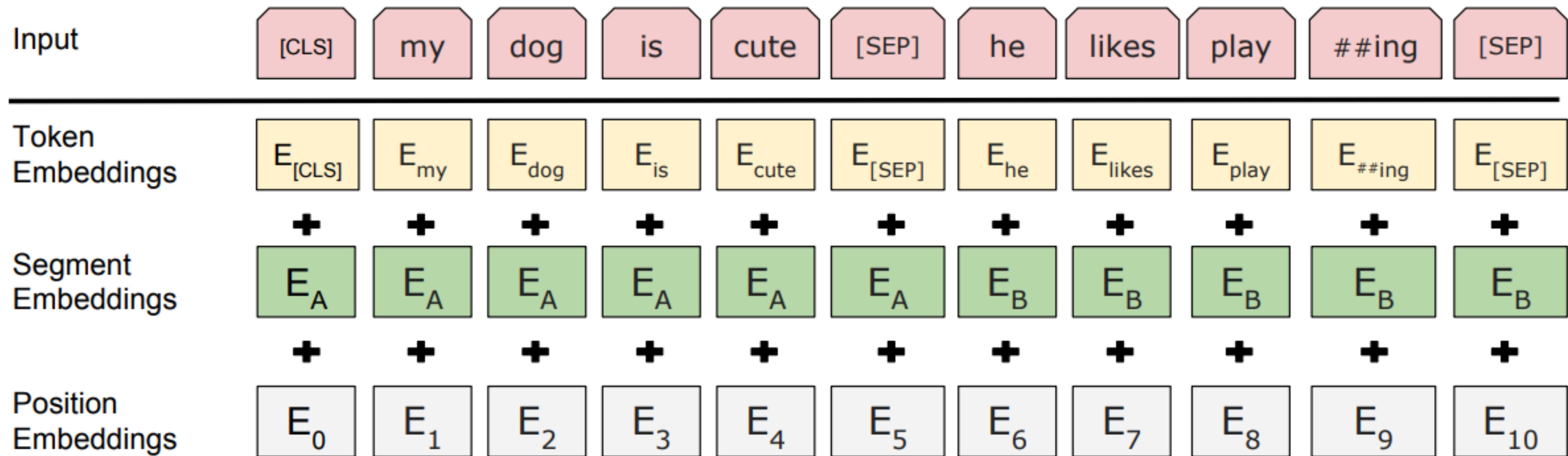
Model architecture:

- Transformer encoder

What is special about it:

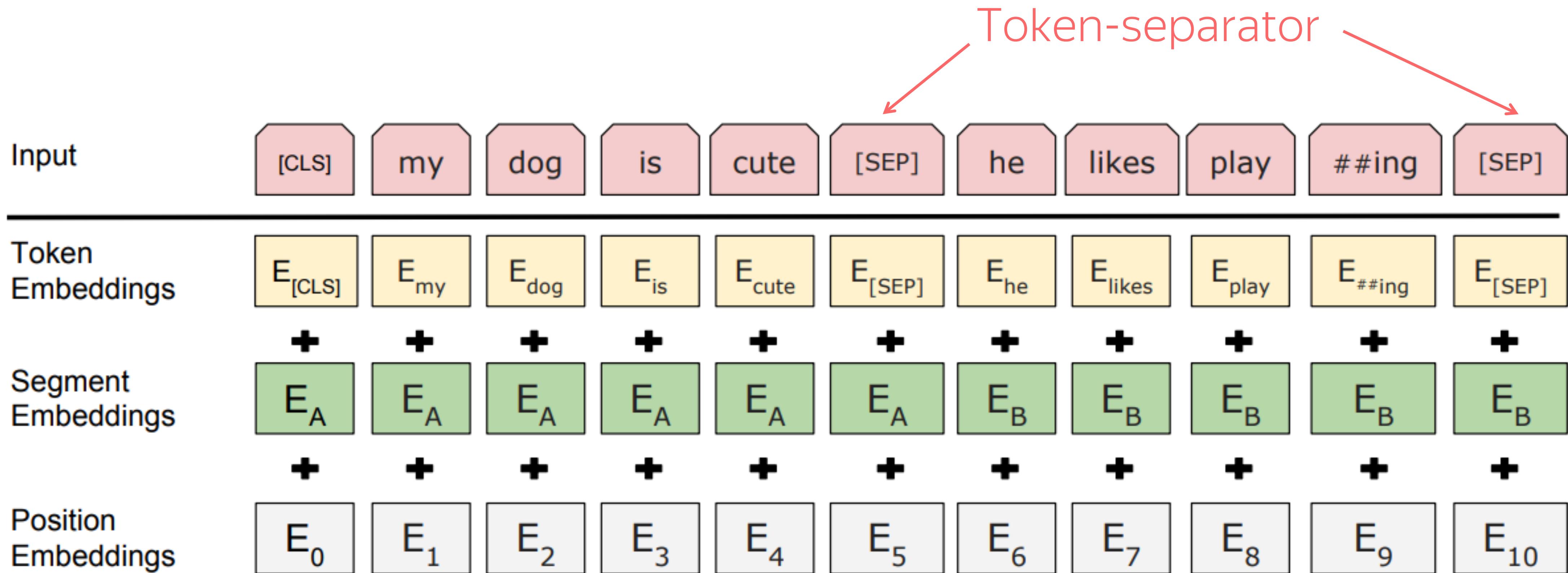
- Training objectives
 - MLM: Masked language modeling
 - NSP: Next sentence prediction
- Lots of data

BERT: Input



The figure is from the [original BERT paper](#)

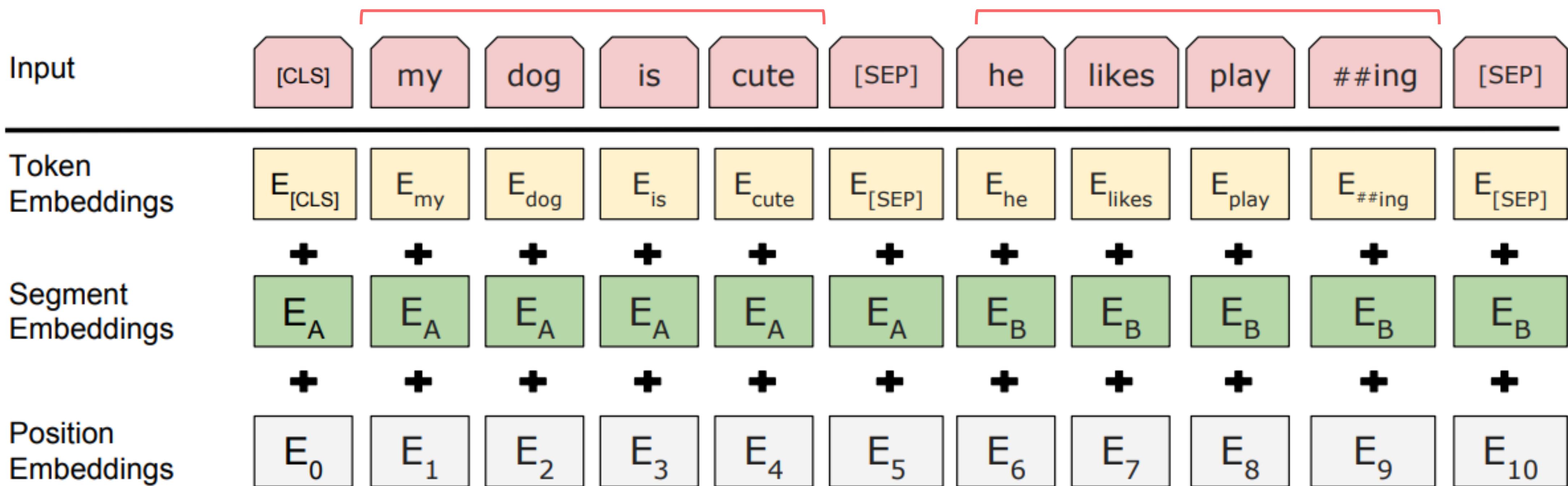
BERT: Input



The figure is from the [original BERT paper](#)

BERT: Input

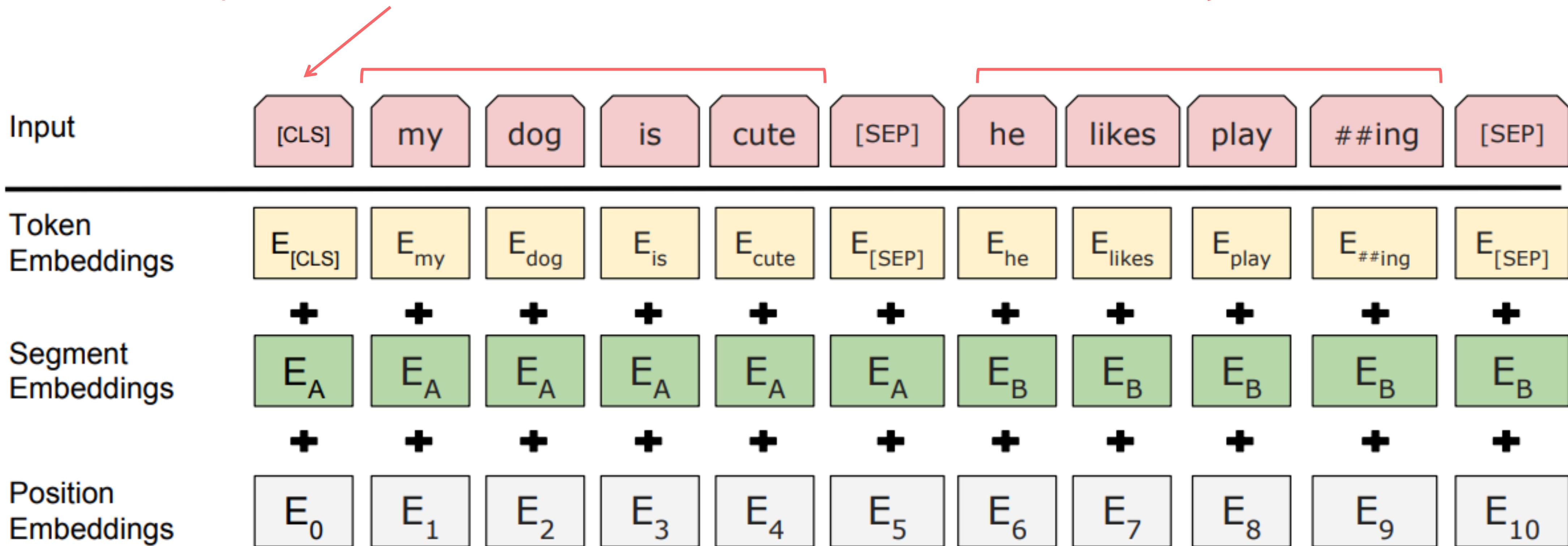
Pair of sentences: either consecutive or random (50%/50%)



The figure is from the [original BERT paper](#)

BERT: Input

Used to predict if the sentences are consecutive (NSP objective)

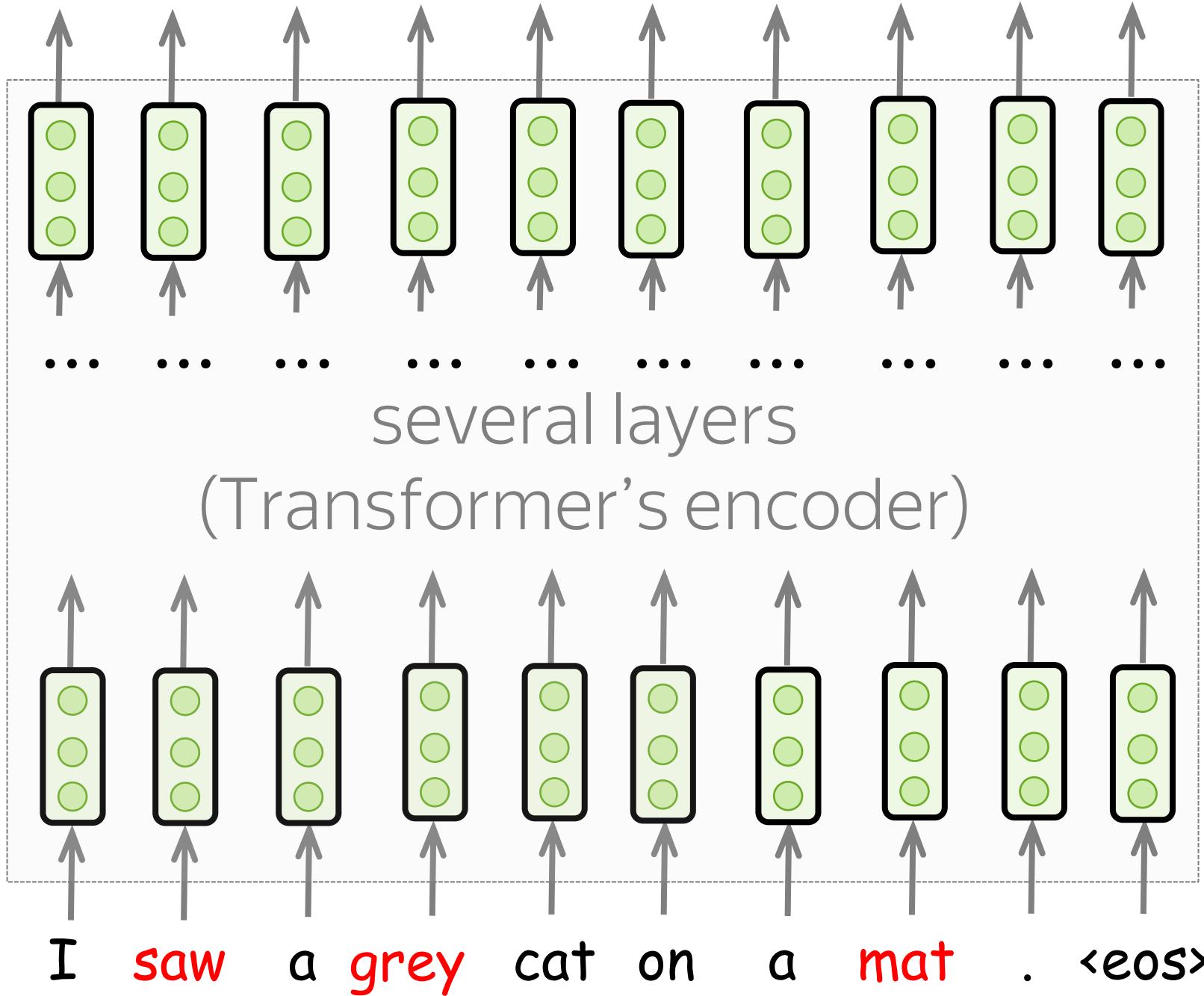


The figure is from the [original BERT paper](#)

BERT: Masked Language Modeling Objective

At each training step:

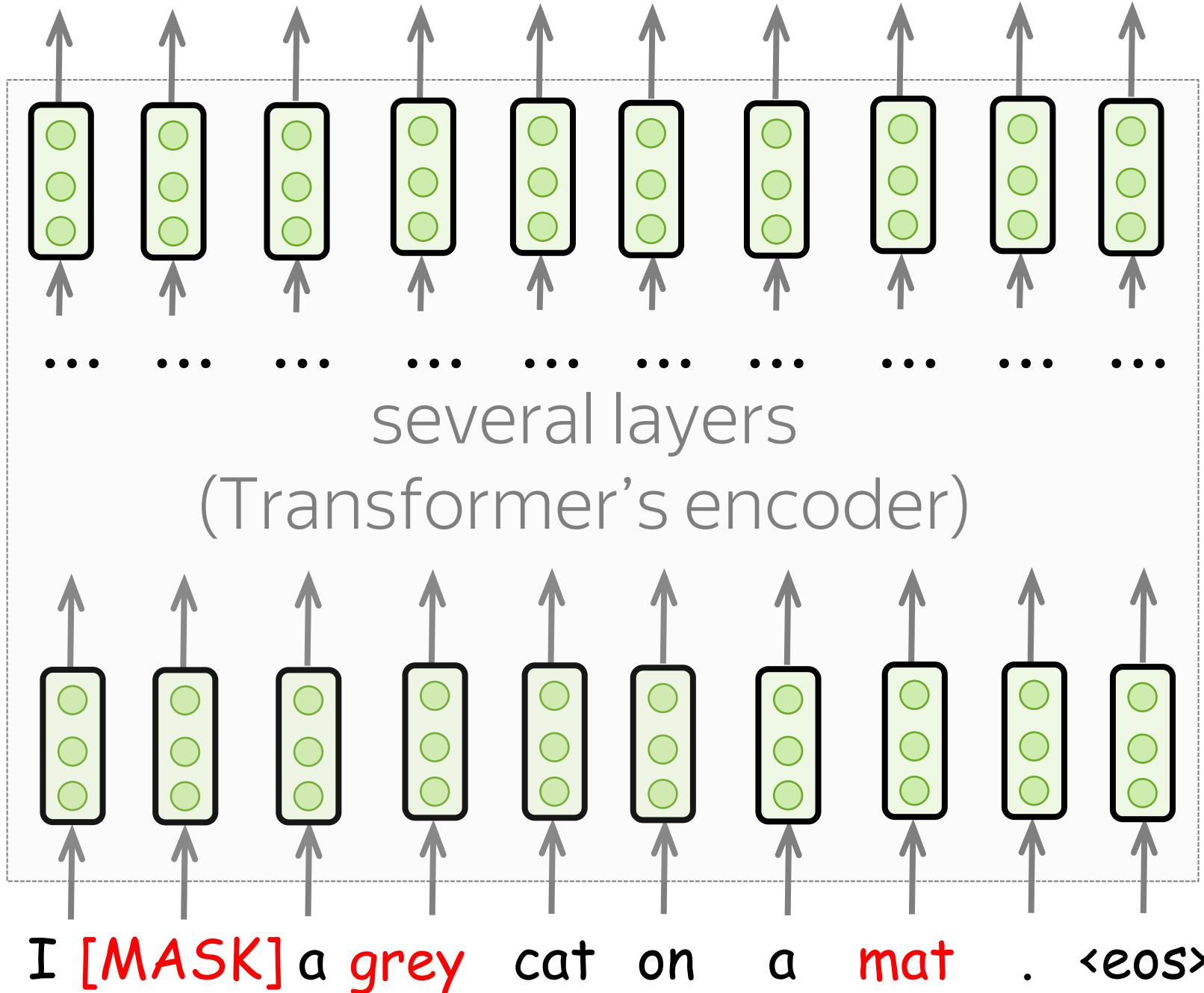
- pick randomly 15% of tokens



BERT: Masked Language Modeling Objective

At each training step:

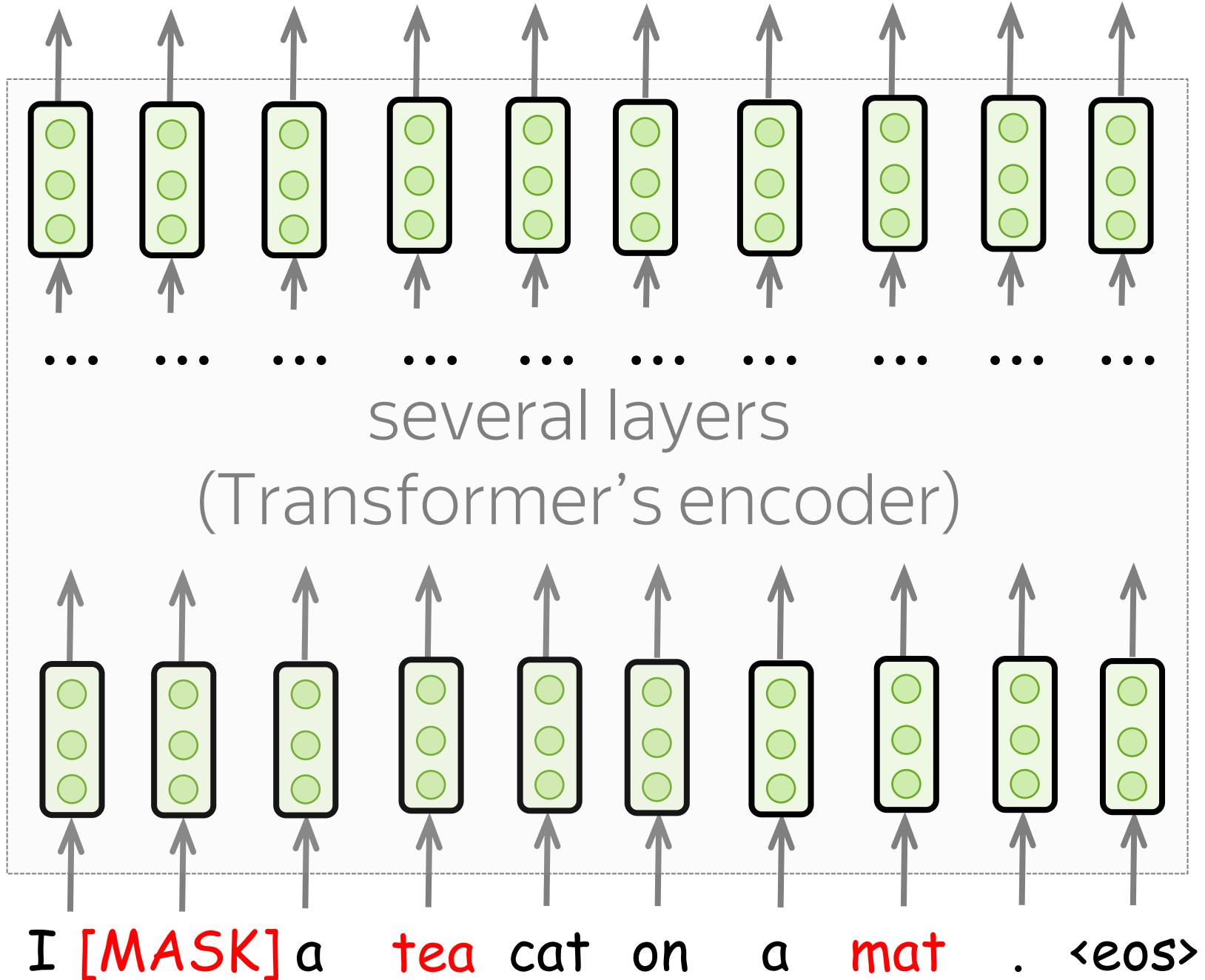
- pick randomly 15% of tokens
- replace each of the chosen tokens with
 - **[MASK]** with prob. 80%



BERT: Masked Language Modeling Objective

At each training step:

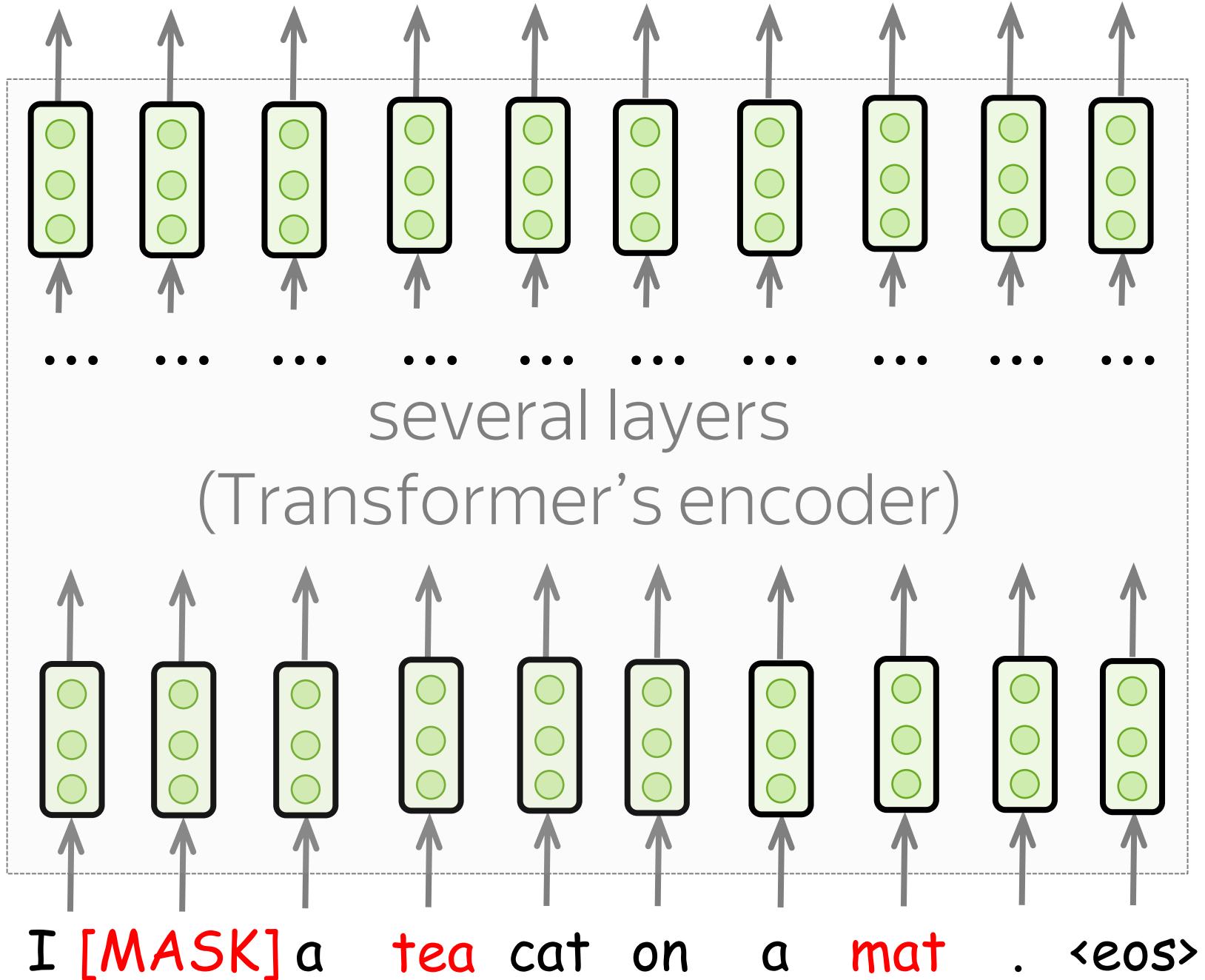
- pick randomly 15% of tokens
- replace each of the chosen tokens with
 - **[MASK]** with prob. 80%
 - random token with prob. 10%



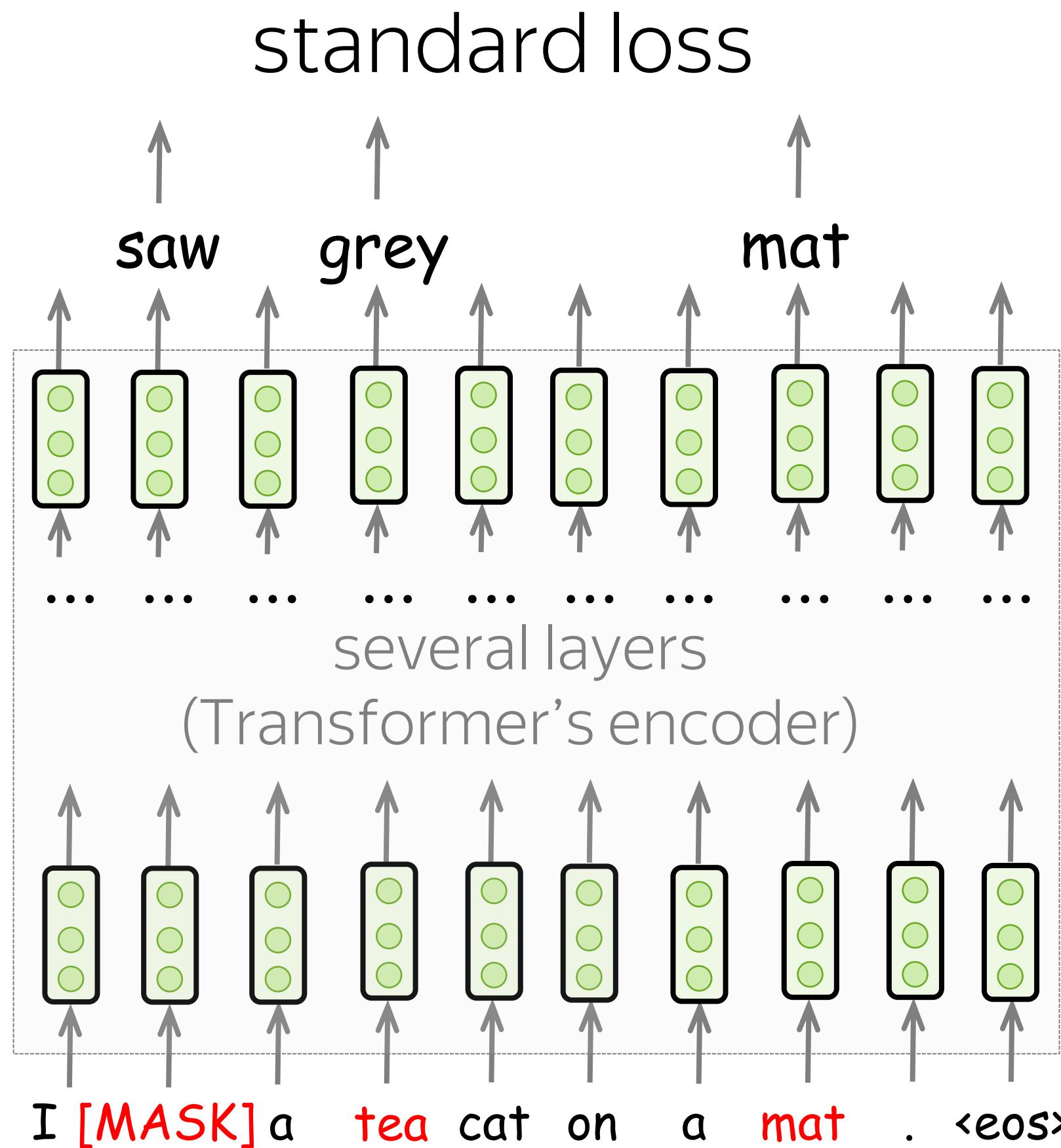
BERT: Masked Language Modeling Objective

At each training step:

- pick randomly 15% of tokens
- replace each of the chosen tokens with
 - **[MASK]** with prob. 80%
 - random token with prob. 10%
 - self with prob. 10%



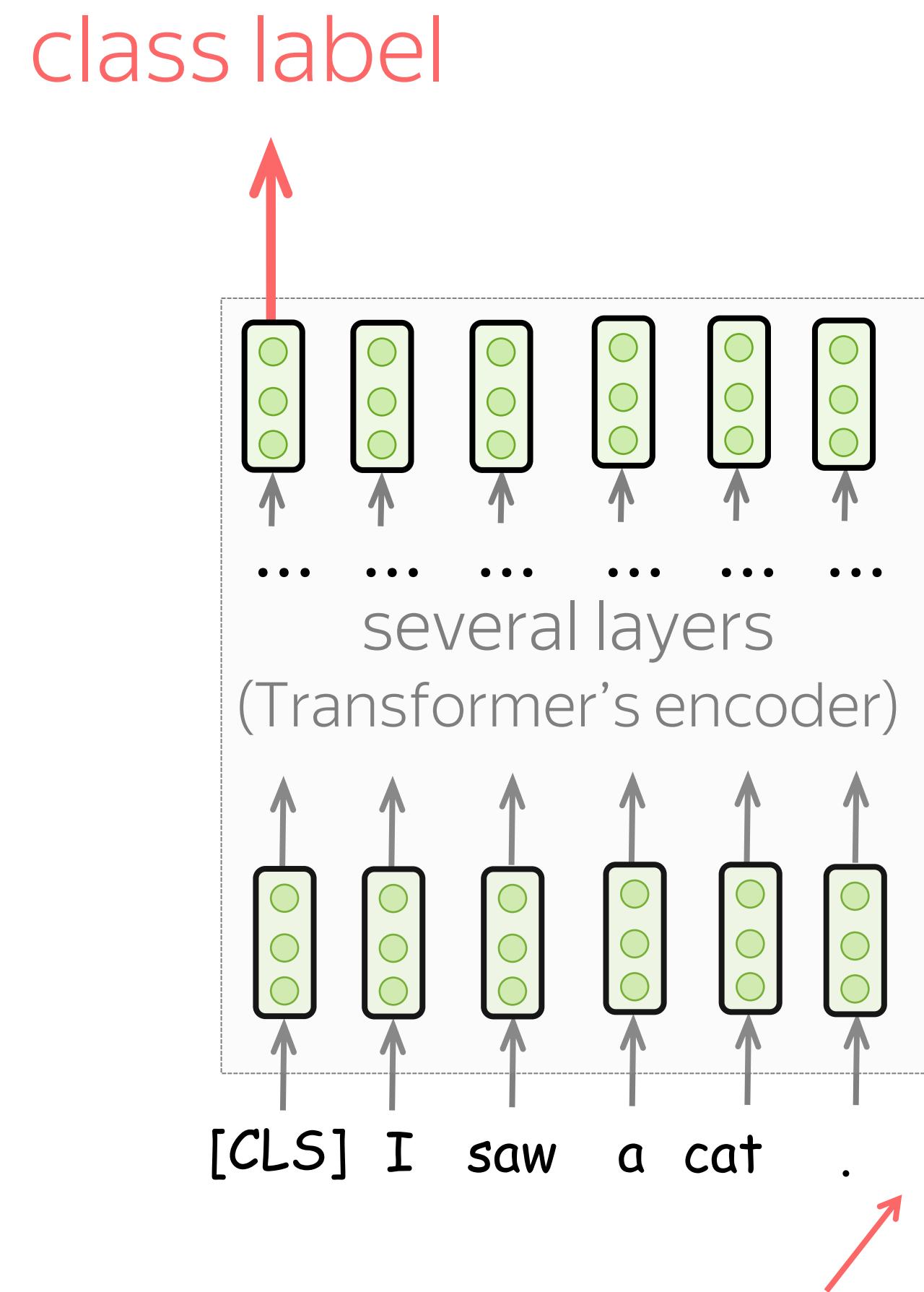
BERT: Masked Language Modeling Objective



At each training step:

- pick randomly 15% of tokens
- replace each of the chosen tokens with
 - **[MASK]** with prob. 80%
 - random token with prob. 10%
 - self with prob. 10%
- predict original tokens
(only chosen ones!)

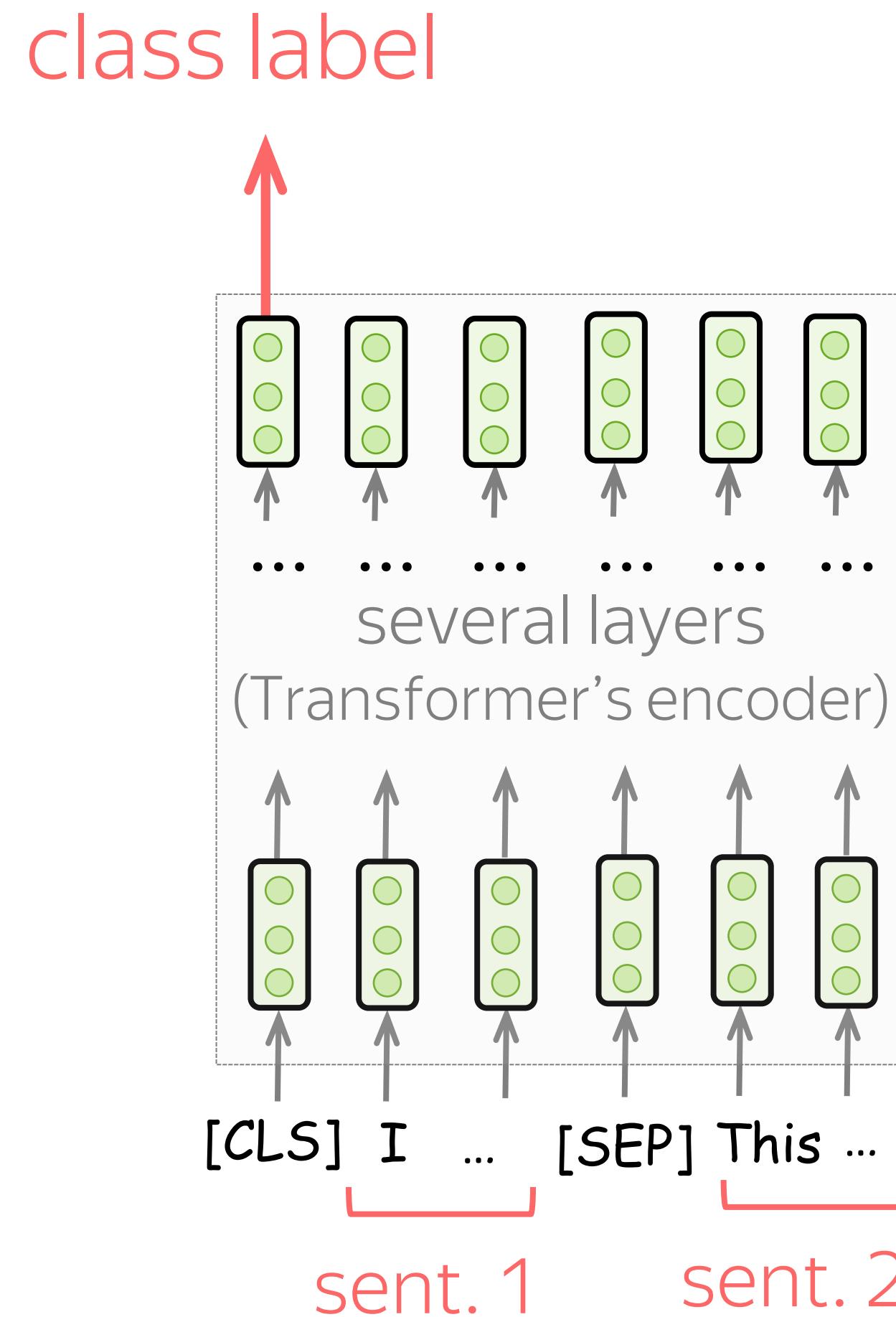
Finetuning BERT: Single-Sentence Classification



Examples of tasks:

- SST-2 – binary sentiment classification (we saw it in the text classification lecture)
- CoLA (Corpus of Linguistic Acceptability) – say whether a sentence is linguistically acceptable

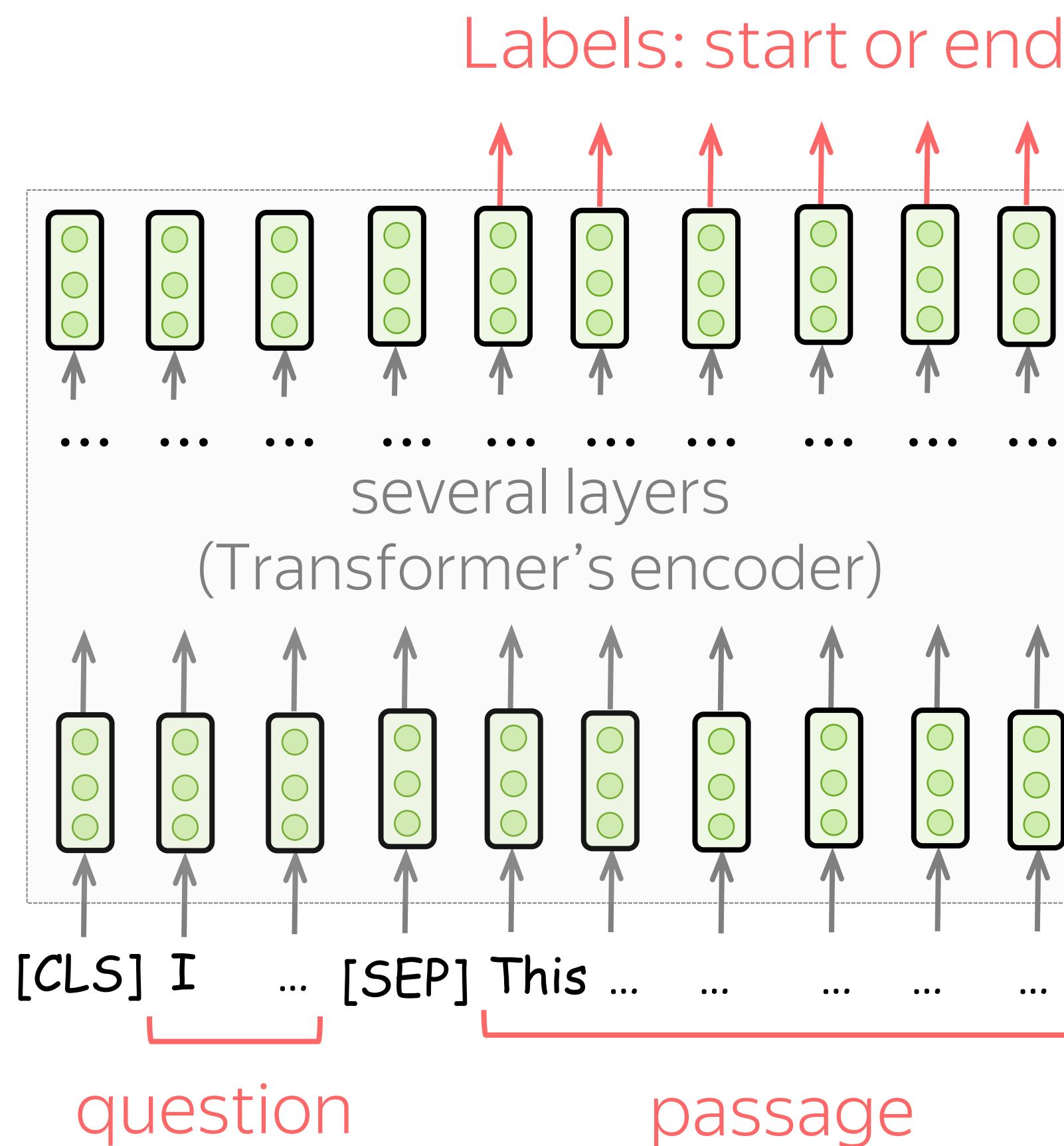
Finetuning BERT: Sentence Pair Classification



Examples of tasks:

- MLNI – entailment classification. Given a pair of sentences, say if the second is an **entailment**, **contradiction** or **neutral**
- QQP (Quora Question Pairs) – given two questions say if they are semantically equivalent
- STS-B – given two sentences return a similarity score from 1 to 5

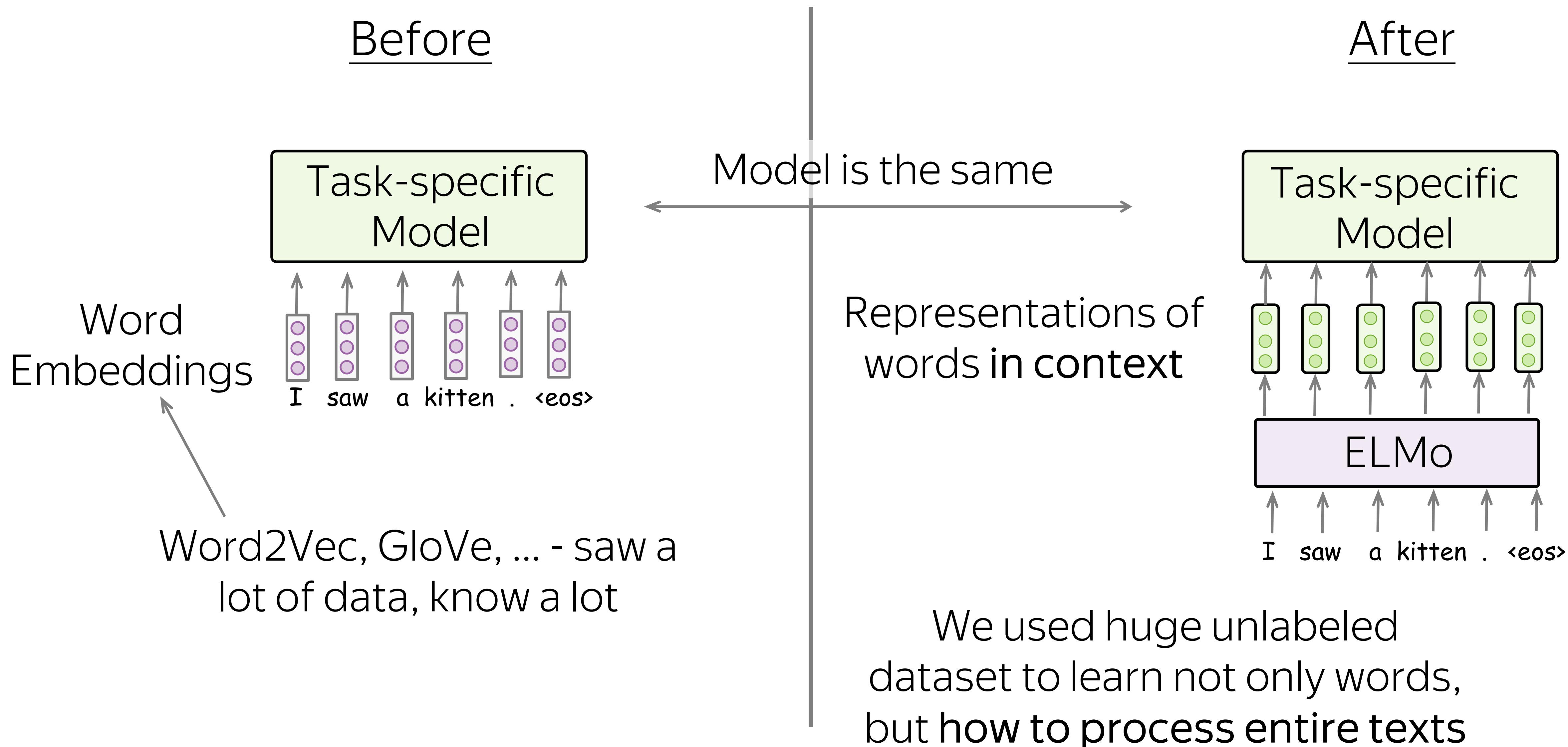
Finetuning BERT: Question Answering



Examples of tasks:

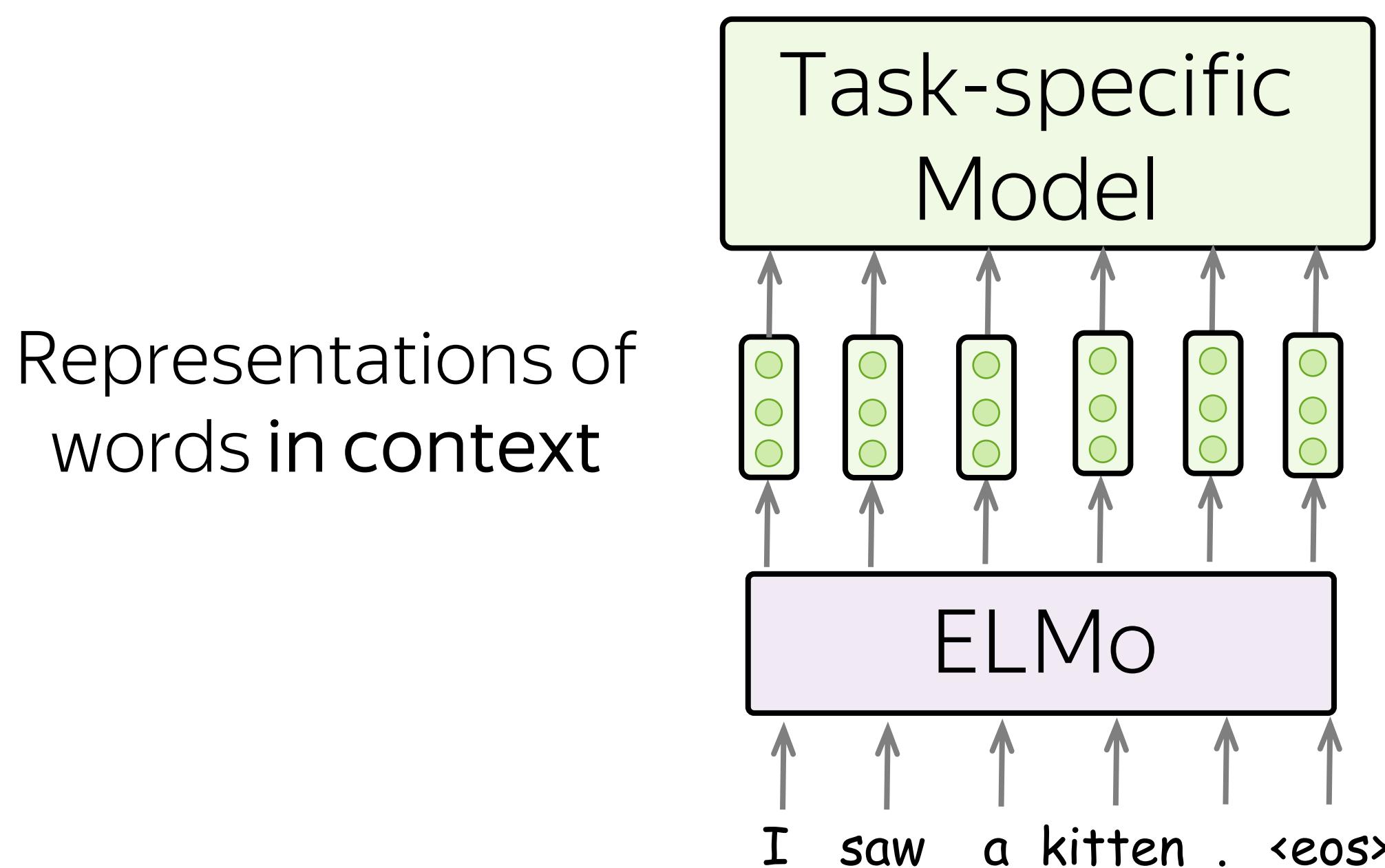
- SQuAD – dataset with pairs of question-passage; the passage contains the answer – need to indicate where

ELMo: What's changed?



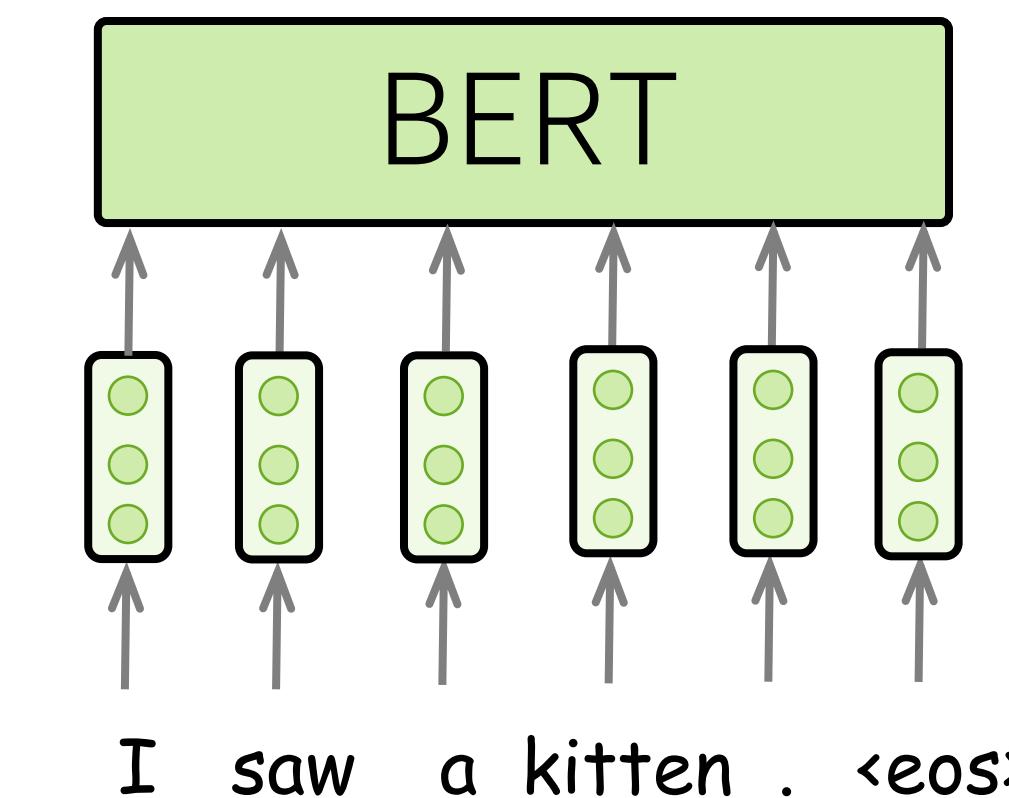
BERT: What's changed?

Before



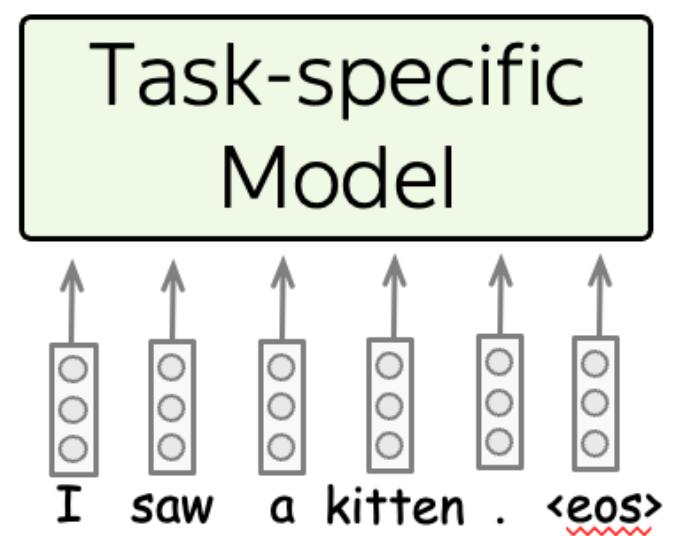
After

No task-specific model at all!



Transfer Learning: Recap of the Ideas

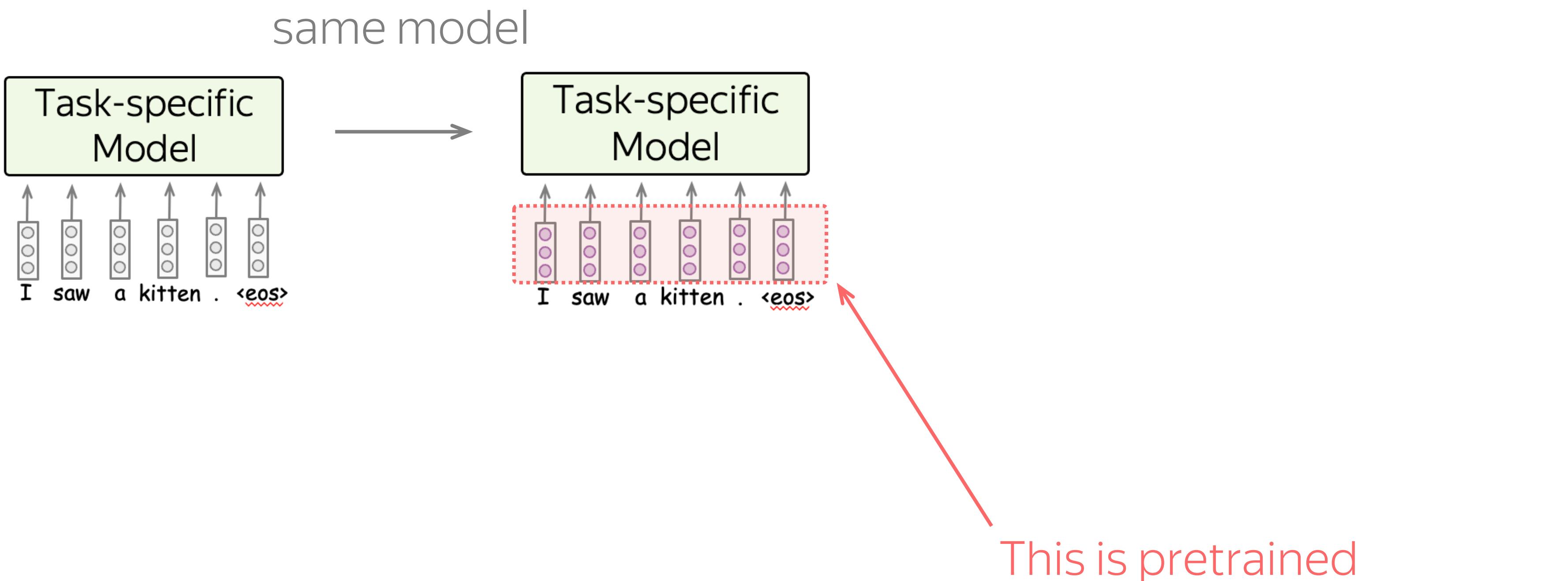
No transfer:
train NN
from scratch



Transfer Learning: Recap of the Ideas

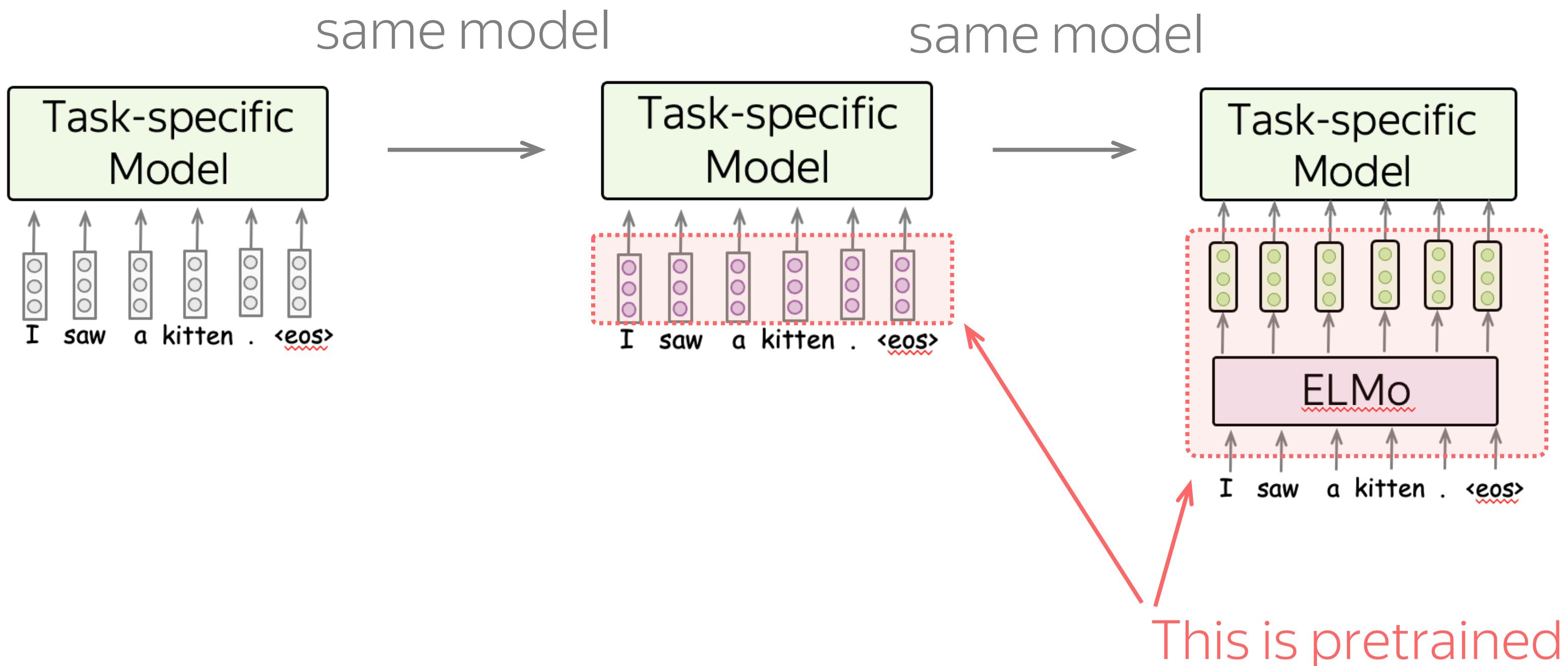
No transfer: → train NN from scratch

Word2vec: use pretrained word embeddings



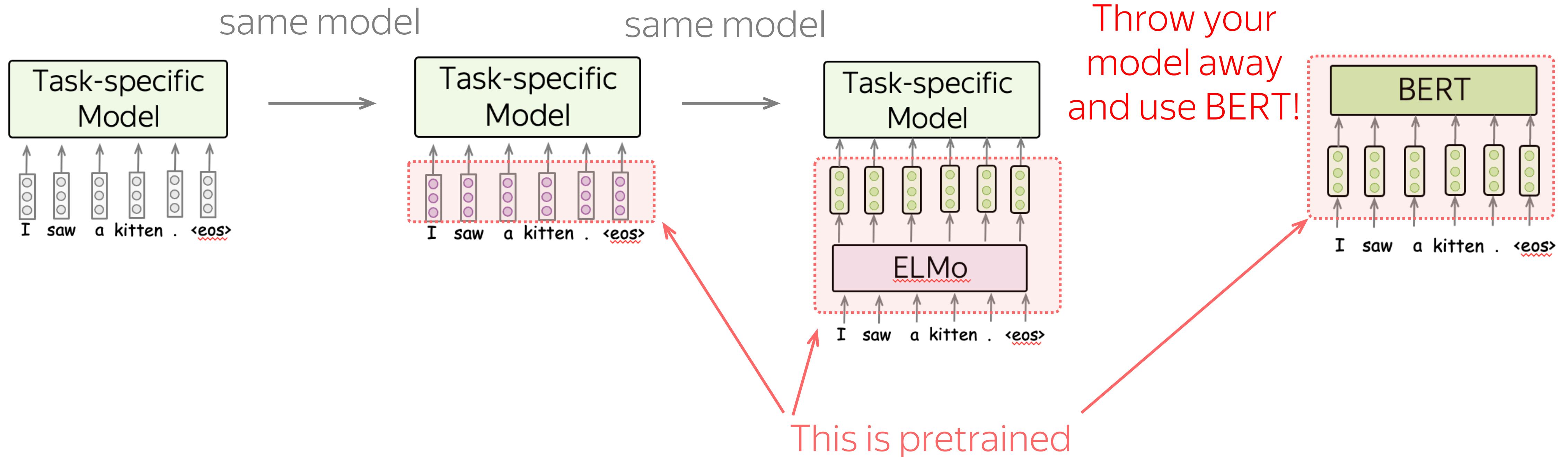
Transfer Learning: Recap of the Ideas

No transfer: → Word2vec: use pretrained word embeddings → ELMo: encode words in their context



Transfer Learning: Recap of the Ideas

No transfer: → Word2vec: use pretrained word embeddings → ELMo: encode words in their context → BERT: encode words in their context



What is going to happen:

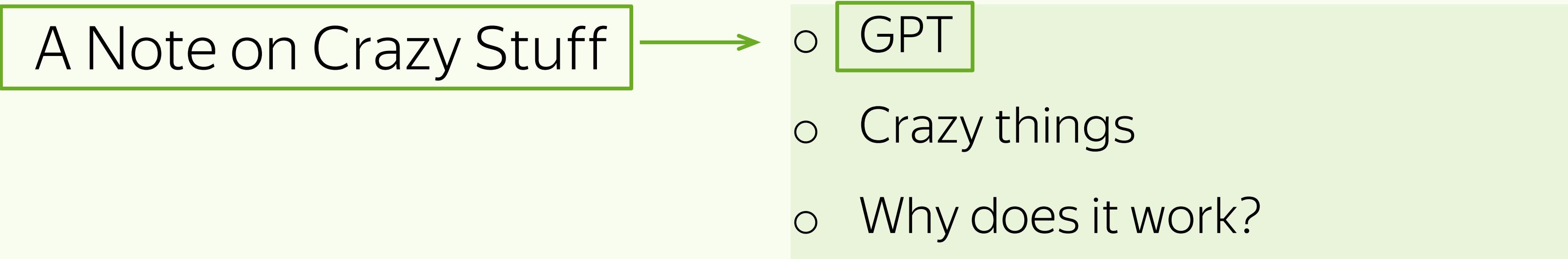
- Classification
 - Language Modeling
 - Transfer Learning
 - A Note on Crazy Stuff
- Features:
from hand-crafted
to learned
- Model:
from task-specific
to generalized

What is going to happen:

- Classification
- Language Modeling
- Transfer Learning
- A Note on Crazy Stuff

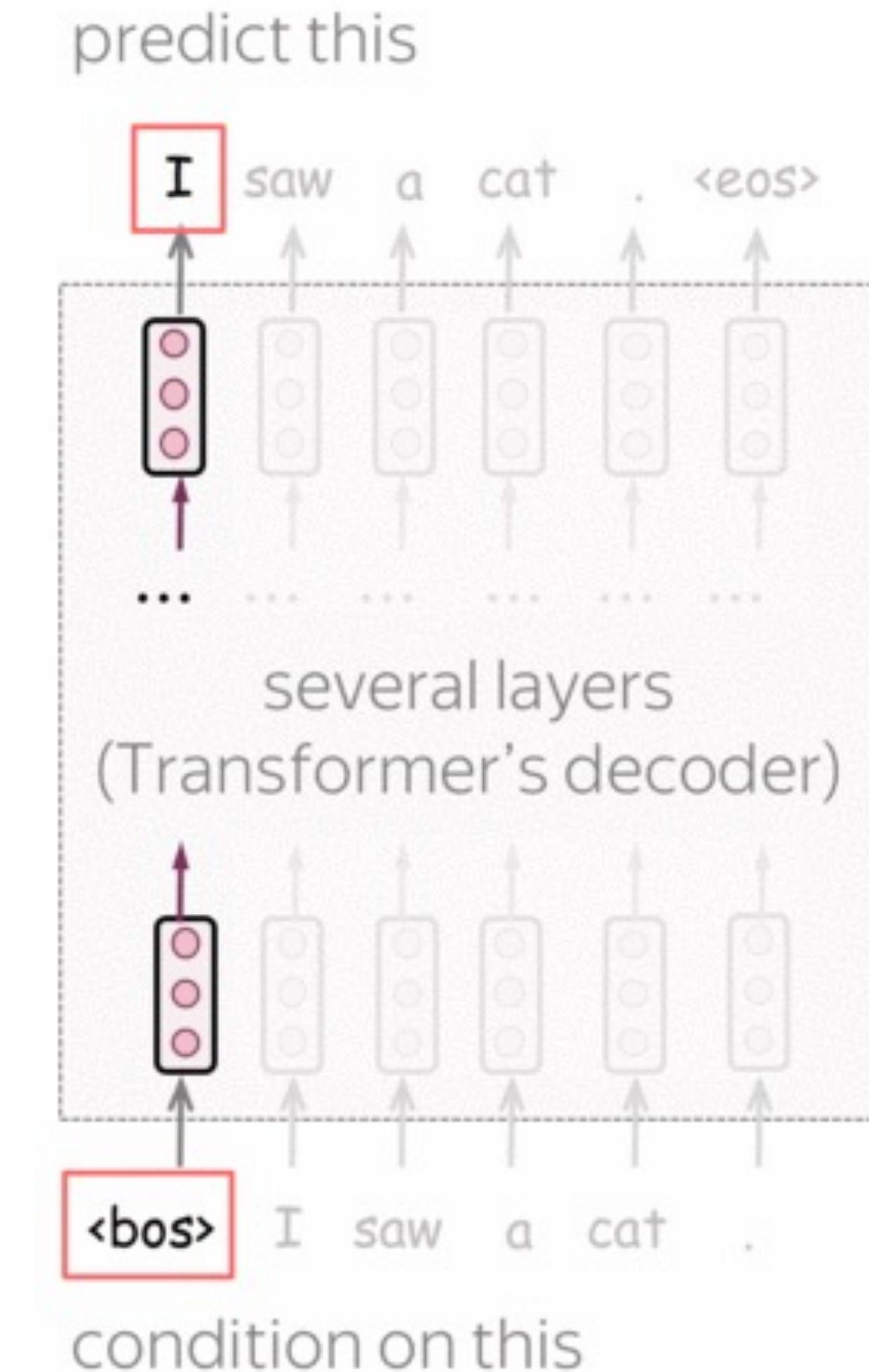
Features:
from hand-crafted
to learned

Model:
from task-specific
to generalized

- 
- GPT
 - Crazy things
 - Why does it work?

Transformer Language Model

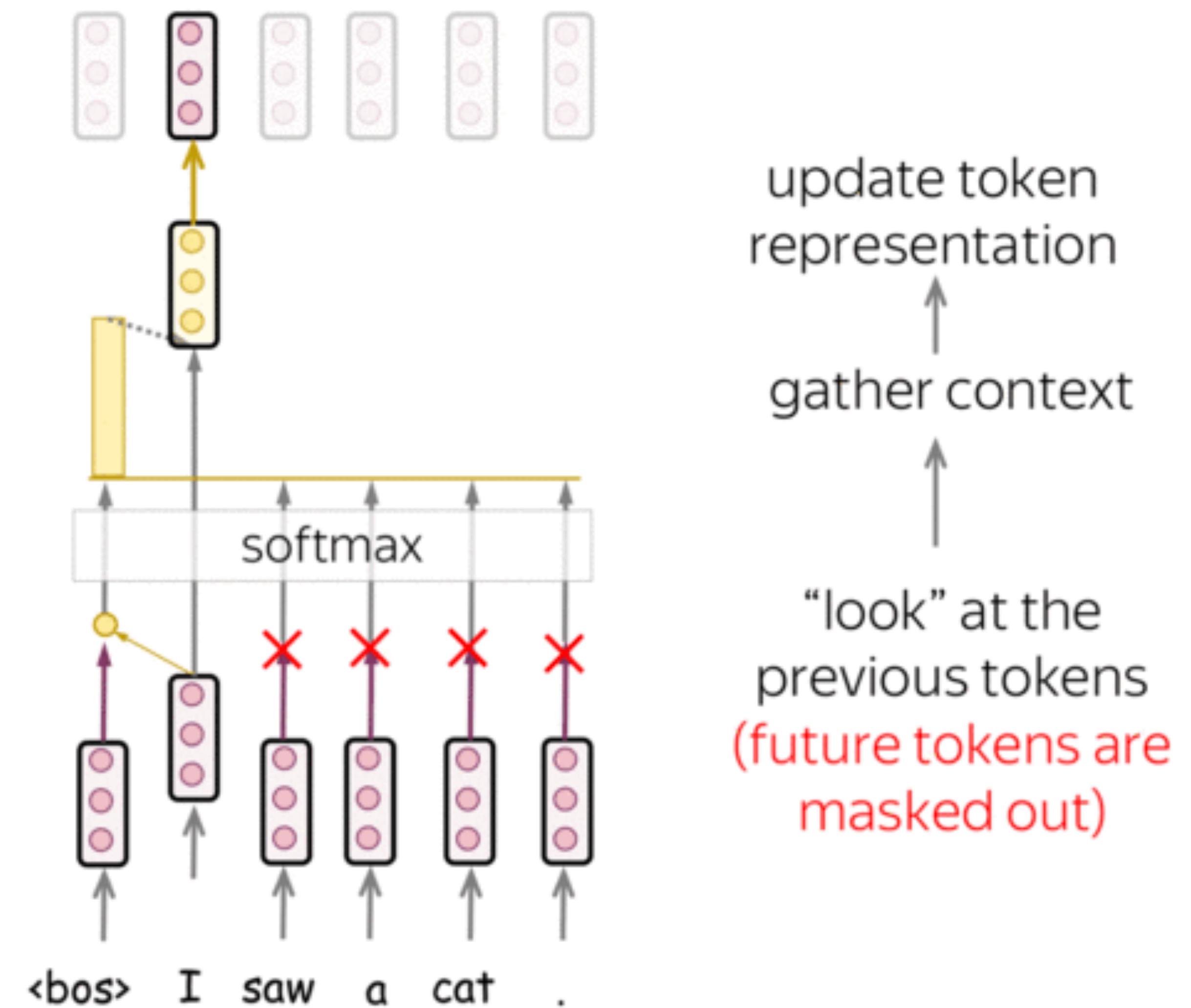
- Several layers with the same structure
- At each layer, tokens
 - “look” at each other
 - gather useful information
 - update their representation



Masked Self-Attention: “Don’t Look Ahead”

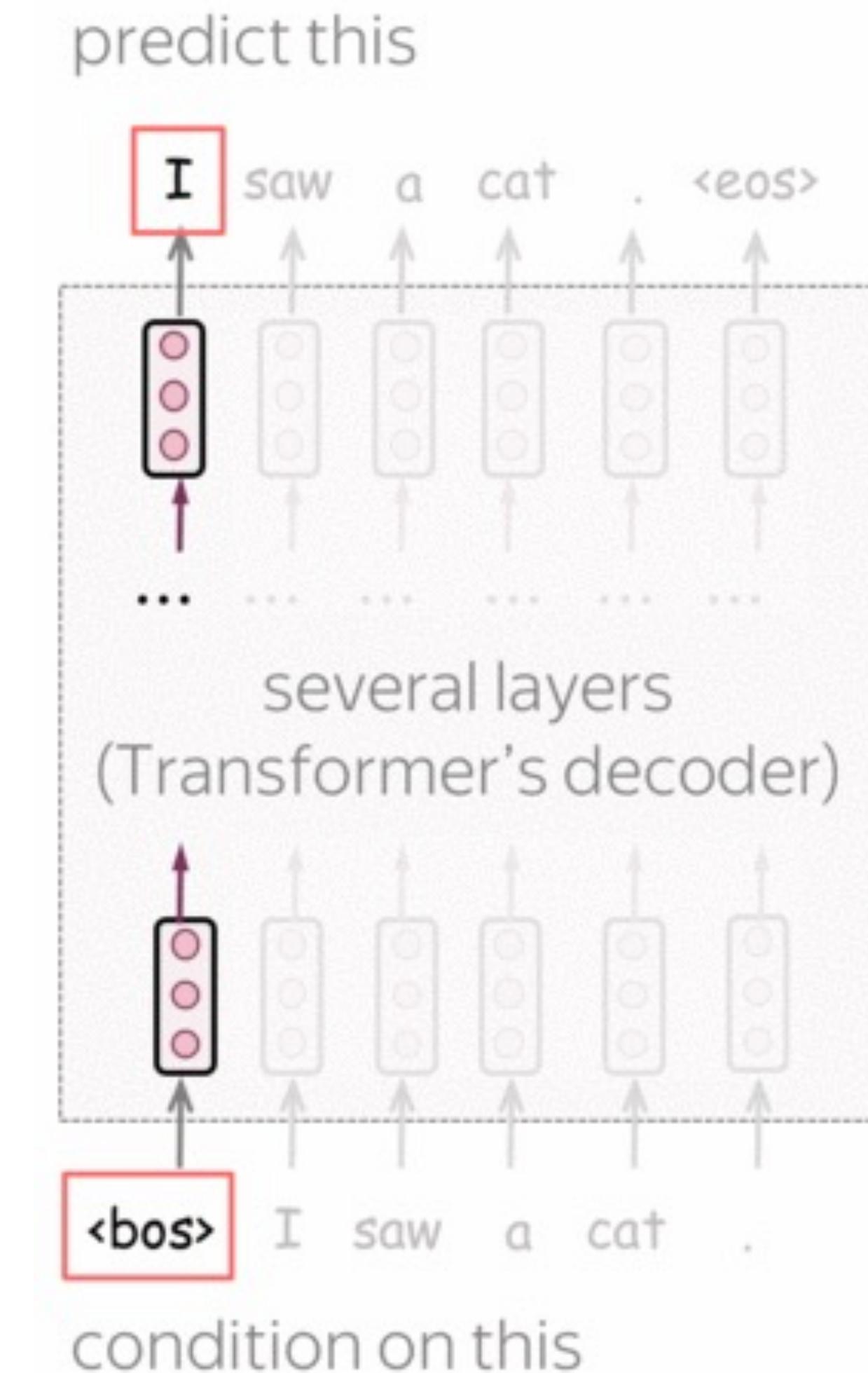
In the decoder, we forbid looking at future tokens – we don't know them

Note: in training, decoder processes all target tokens at once – without masks, it would see future



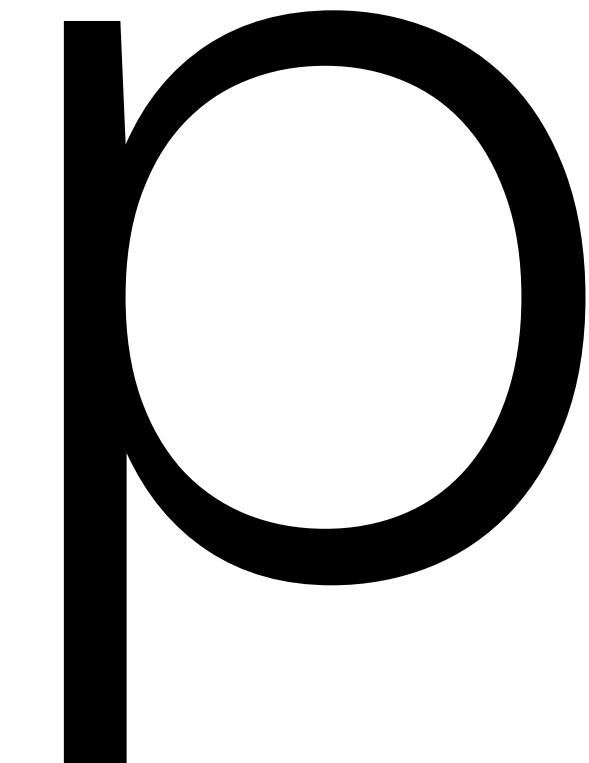
Transformer Language Model

- Several layers with the same structure
- At each layer, tokens
 - “look” at each other
 - gather useful information
 - update their representation



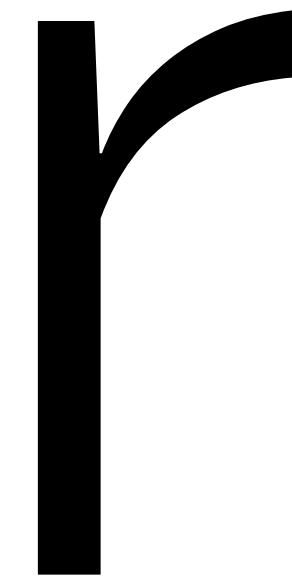
Group Exercise: Feel Like a Transformer!

Once when I was six years old I saw a magnificent picture in a book, called True Stories from Nature, about the



Group Exercise: Feel Like a Transformer!

Once when I was six years old I saw a magnificent picture in a book, called True Stories from Nature, about the p



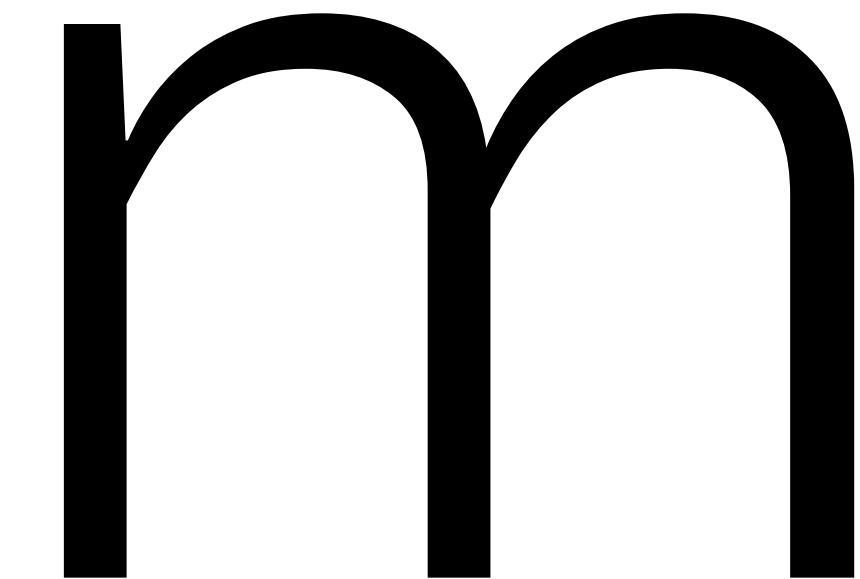
Group Exercise: Feel Like a Transformer!

Once when I was six years old I saw a magnificent picture in a book, called True Stories from Nature, about the pr



Group Exercise: Feel Like a Transformer!

Once when I was six years old I saw a magnificent picture in a book, called True Stories from Nature, about the pri



Group Exercise: Feel Like a Transformer!

Once when I was six years old I saw a magnificent picture in a book, called True Stories from Nature, about the prim

e

Group Exercise: Feel Like a Transformer!

Once when I was six years old I saw a magnificent picture in a book, called True Stories from Nature, about the prime

V

Group Exercise: Feel Like a Transformer!

Once when I was six years old I saw a magnificent picture in a book, called True Stories from Nature, about the primev

a

Group Exercise: Feel Like a Transformer!

Once when I was six years old I saw a magnificent picture in a book, called True Stories from Nature, about the primeva



GPT: Generative Pre-trained Transformer

GPT-1

(2018)

Number of
parameters

117 million

GPT-2

(2019)

Training data

5 GB

40 GB

GPT-2: Examples of Generated Text

(CNN) Former President Donald Trump gave a January 6 speech near the White House in which he exhorted his supporters to march on the US Capitol to challenge Joe Biden's claim that Rei was better than Asuka. "Ayanami Rei was one of the worst characters in Evangelion, if not the worst period. I'm not saying that because she's Japanese. I'm saying she 's a total loser," Trump said . " Her face. She doesn't have a smile. Her eyes are all screwed up and her hair is the worst hair , I mean worse than Asuka 's , she looked like a monkey ." In his speech , Trump also said, " The reason Asuka was more successful than Rei was because she was never a ' normal girl ' in the first place , she was always a 'freak ' that got what she wanted. Asuka is the most popular character from the Eva series , and she's also the best character . Rei is very bad , she's not even close to Asuka's level, she is so bad it's ridiculous ." In a statement released Wednesday , Trump called the speech " not nice ." Read the full statement below . (Source : CNN) .

.. According to reports CNN (CNN's source) sister- , Trump site, spoke to a was is the speech " not nice." Iam (source) The interview bert with (CNN aired source) , the interviewer stated " it's not nice ." This was the second time the interview was aired . (source) CNN (source) also said , " that was a pretty un professional way to answer the question ." (source) CNN (source) then went on to say , " but what he said wasn't surprising considering the situation ." (source) (source) CNN also stated, " he seemed to get it , but he also seemed to be struggling to say what was most important ." (source) CNN's (source) also said that the interview was only done because they " felt like he needed to be on the record with what he said," (source) In addition , " that 's because the people who watched the interview asked for a full explanation and an apology." (source) CNN (source) also said the interview was only done because they " felt like he needed to be on the record with what he said ," (source) In addition , " that 's because the people who watched the interview asked for a full explanation and an apology." (source) (source) CNN (source) also said the interview was only done because they " felt like he needed to be on the record with what he said ," (source) In addition , " that's because the people who watched the interview asked for a full explanation and an apology." (source) (source) CNN (source) also said the interview was only done because they " felt like he needed to be on the record with what he said ," (source) In addition , " that's because the people who watched the interview asked for a full explanation and an apology." (source) (source) CNN (source) also said the interview was only done because they

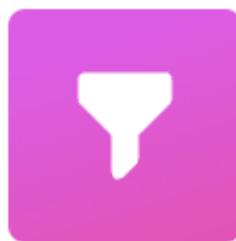
(highlighted text is generated by the model)

GPT: Generative Pre-trained Transformer

	<u>GPT-1</u> (2018)	<u>GPT-2</u> (2019)	<u>GPT-3</u> (2020)
Number of parameters	117 million	1.5 billion	175 billion
Training data	5 GB	40 GB	45 TB (i.e., 45 000 GB)

The entirety of English Wikipedia constitutes just 0.6% of GPT-3 training data

GPT-3: examples (<https://beta.openai.com/examples>)



TL;DR summarization

Transformation Generation

Summarize text by adding a 'tl;dr:' to the end of a text passage. It shows that the API understands how to perform a number of tasks with no instructions.

Prompt

A neutron star is the collapsed core of a massive supergiant star, which had a total mass of between 10 and 25 solar masses, possibly more if the star was especially metal-rich.[1] Neutron stars are the smallest and densest stellar objects, excluding black holes and hypothetical white holes, quark stars, and strange stars.[2] Neutron stars have a radius on the order of 10 kilometres (6.2 mi) and a mass of about 1.4 solar masses. [3] They result from the supernova explosion of a massive star, combined with gravitational collapse, that compresses the core past white dwarf star density to that of atomic nuclei.

Tl;dr

Sample response

A neutron star is the collapsed core of a massive supergiant star. These ultra-dense objects are incredibly fascinating due to their strange properties and their potential for phenomena such as extreme gravitational forces and a strong magnetic field.

GPT-3: examples (<https://beta.openai.com/examples>)

 **Factual answering**

Answers Generation Conversation Classification

Guide the model towards factual answering by showing it how to respond to questions that fall outside its knowledge base. Using a '?' to indicate a response to words and phrases that it doesn't know provides a natural response that seems to work better than more abstract replies.

Prompt

Q: Who is Batman?
A: Batman is a fictional comic book character.

Q: What is torsalplexity?
A: ?

Q: What is Devz9?
A: ?

Q: Who is George Lucas?
A: George Lucas is American film director and producer famous for creating Star Wars.

Q: What is the capital of California?
A: Sacramento.

Q: What orbits the Earth?
A: The Moon.

Q: Who is Fred Rickerson?
A: ?

Q: What is an atom?
A: An atom is a tiny particle that makes up everything.

Q: Who is Alvan Muntz?
A: ?

GPT-3: examples (<https://beta.openai.com/examples>)

 English to other languages

Transformation Generation

Translates English text into French, Spanish and Japanese.

Prompt

Translate this into 1. French, 2. Spanish and 3. Japanese:

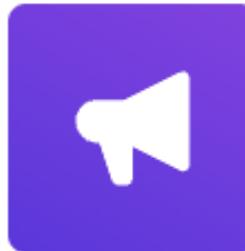
What rooms do you have available?

1.

Sample response

Quels sont les chambres disponibles?
2. ¿Cuáles son las habitaciones disponibles?
3. 何室がありますか?

GPT-3: examples (<https://beta.openai.com/examples>)



Ad from product description

Generation

Turn a product description into ad copy.

Prompt

Write a creative ad for the following product to run on Facebook aimed at parents:

Product: Learning Room is a virtual environment to help students from kindergarten to high school excel in school.

Sample response

Are you looking for a way to help your child excel in school? Look no further than Learning Room! Our virtual environment is designed to help students from kindergarten to high school learn and grow. With our help, your child will be able to get ahead in school and achieve their academic goals.

GPT-3: examples (<https://beta.openai.com/examples>)



Create study notes

Generation

Provide a topic and get study notes.

Prompt

What are 5 key points I should know when studying Ancient Rome?

Sample response

1. Rome was founded in 753 BC by Romulus and Remus.
2. Rome was ruled by an emperor.
3. Rome was a republic before it became an empire.
4. Julius Caesar was assassinated in 44 BC.
5. Augustus was the first emperor of Rome.

GPT-3: examples (<https://beta.openai.com/examples>)

Python to natural language

Code Translation

Explain a piece of Python code in human understandable language.

Prompt

```
# Python 3
def remove_common_prefix(x, prefix, ws_prefix):
    x["completion"] = x["completion"].str[len(prefix):]
    if ws_prefix:
        # keep the single whitespace as prefix
        x["completion"] = " " + x["completion"]
    return x

# Explanation of what the code does

#
```

Sample response

The code above is a function that takes a dataframe and a prefix as input and returns a dataframe with the prefix removed from the completion column.

GPT-3: examples (<https://beta.openai.com/examples>)

<div style="border-bottom: 1px solid #ccc; padding-bottom: 10px;"> Analogy maker Generation</div> <p>Create analogies. Modified from a community prompt to require fewer examples.</p>	<p>Prompt</p> <div style="background-color: #f0f0f0; padding: 10px; border-radius: 10px;"><p>Create an analogy for this phrase: Questions are arrows in that:</p></div> <p>Sample response</p> <div style="background-color: #e0ffe0; padding: 10px; border-radius: 10px;"><p>Questions are arrows in that they can be used to point out things that need to be fixed.</p></div>
--	---

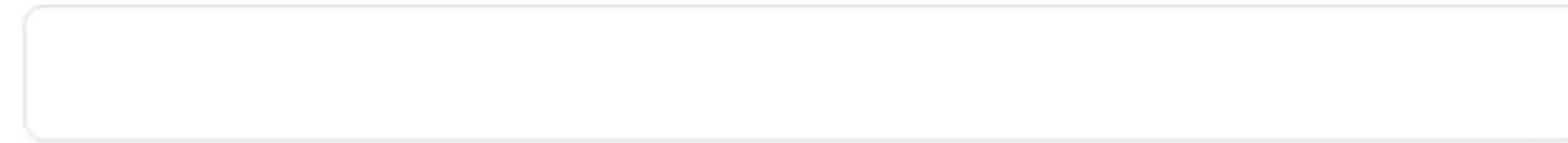
GPT-3: examples

Describe a layout.

Just describe any layout you want, and it'll try to render below!

A div that contains 3 buttons each with a random color.

Generate



GPT-3 and in-context learning



Micro horror story creator

Transformation Generation Translation

Creates two to three sentence short horror stories from a topic input.

Prompt

Topic: Breakfast

Two-Sentence Horror Story: He always stops crying when I pour the milk on his cereal. I just have to remember not to let him see his face on the carton.

Topic: Wind

Two-Sentence Horror Story:

Sample response

I was lying in bed, trying to get to sleep, when I heard the wind howling outside my window. It sounded like something was trying to get in.

Source: <https://beta.openai.com/examples>

GPT-3 Gives Correct Medical Advice

Question: A 10 year old boy presents with recurrent episodes of dyspnoea, cough and wheeze triggered by cold air and allergens. After performing tests he is diagnosed with an obstructive disease and given medication. Which receptor is the medication most likely to work on:

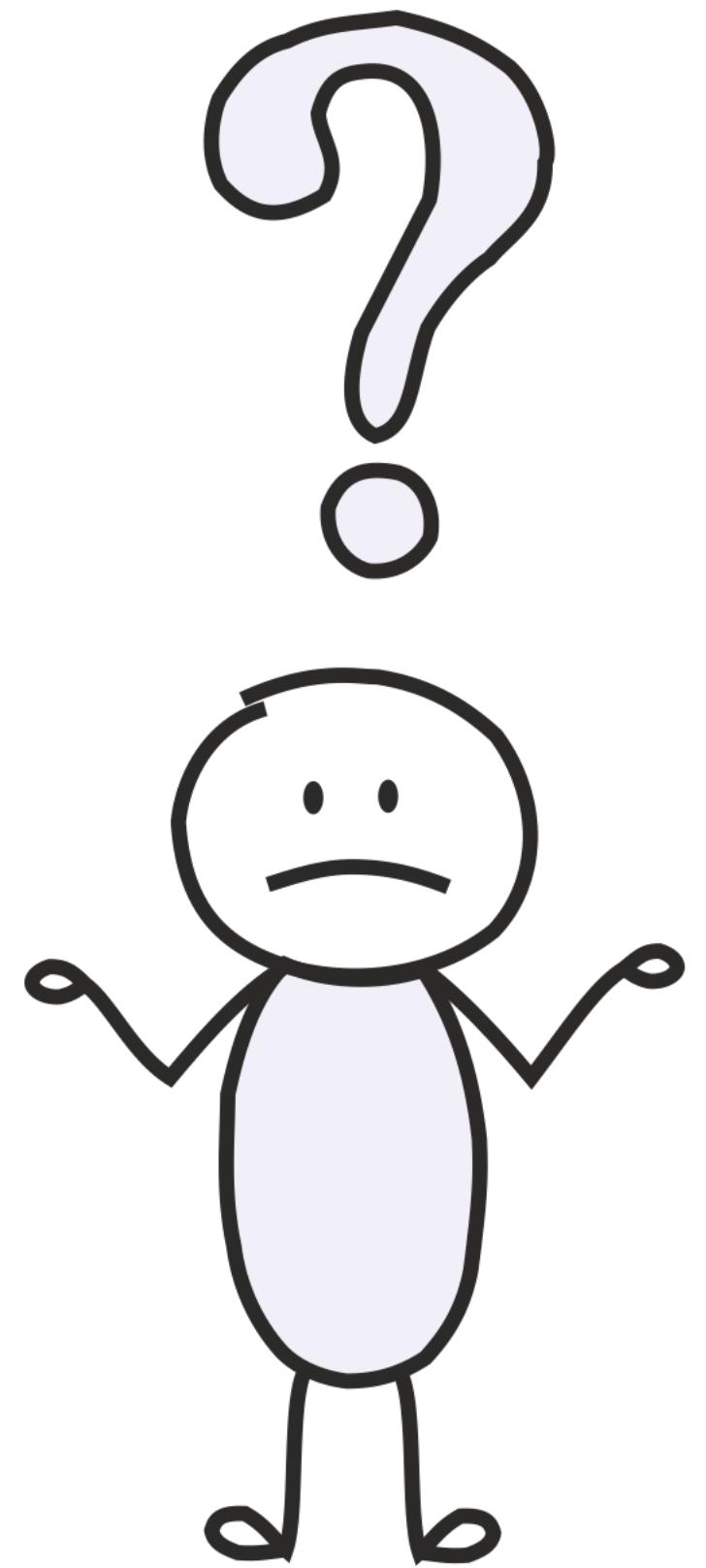
- A) muscarinic receptor
- B) G-protein coupled receptor
- C) Era
- D) Erb
- E) a-1 receptor

Correct answer is **B**

Explanation: The patient is suffering from asthma, a disease of the airways. The drugs used to treat asthma are bronchodilators. The bronchodilators act on the beta-2 receptors. The beta-2 receptors are G-protein coupled receptors

(bold text is generated by the model)

Any concerns?



GPT-2: Examples of Generated Text

Problem: it can generate anything and not care whether it's true or not

Why is Skub the greatest product on the market today? I'll give you three great reasons:

1) Skub is the most reliable solution for your personal, professional and personal life. It offers a great combination of quality and availability.

2) Skub is the most versatile solution for all your clients. It also has a great selection of advanced software, features and products.

3) Skub's unique design and its strong functional design allows you to easily customize your products. It provides great products and services with all the latest features. It has been designed for everyone with a clear look, without needing any further extra effort.

Why is Skub the worst product on the market today? I'll give you three great reasons:

1) Skub is not affordable to use in a way that would be considered a reasonable solution to the problems we face (especially to those who are in the business of selling products that aren't well received in the industry).

2) Skub is not very effective and lacks the best features. It is hard to see what would make an improvement in value in a product.

3) Skub is not as popular (especially if you're trying to sell a product with a high level of performance) as you might think. Skub is a product that will not be sold for a lot of money.

(highlighted text is generated by the model)

Sometimes GPT-3 Generates Nonsense

Q: How does a laser work?

A: It's simple, really. If you've ever seen one of those cool demos where you shine a laser through a cloud of smoke and see the smoke streak in rings, that's all a laser is. When you get right down to it, it's a light beam shining through a gas, turning it into a plasma. Only instead of making the cloud of smoke glow, a laser makes it explode. What sets it apart from a regular light beam is the frequency at which it oscillates, that is to say how many times per second it is turned on and off. If you use a lower frequency you get a beam with a broader range of colors, but it will not be as focused and will have trouble with anything over a few feet. Conversely, if you use a higher frequency you'll get a smaller range of colors and better focus, but it will lose coherency and be absorbed in the atmosphere much sooner.

Sometimes GPT-3 Generates Nonsense

Q: How many eyes does a horse have?

A: Four. One in the front and three in the rear.

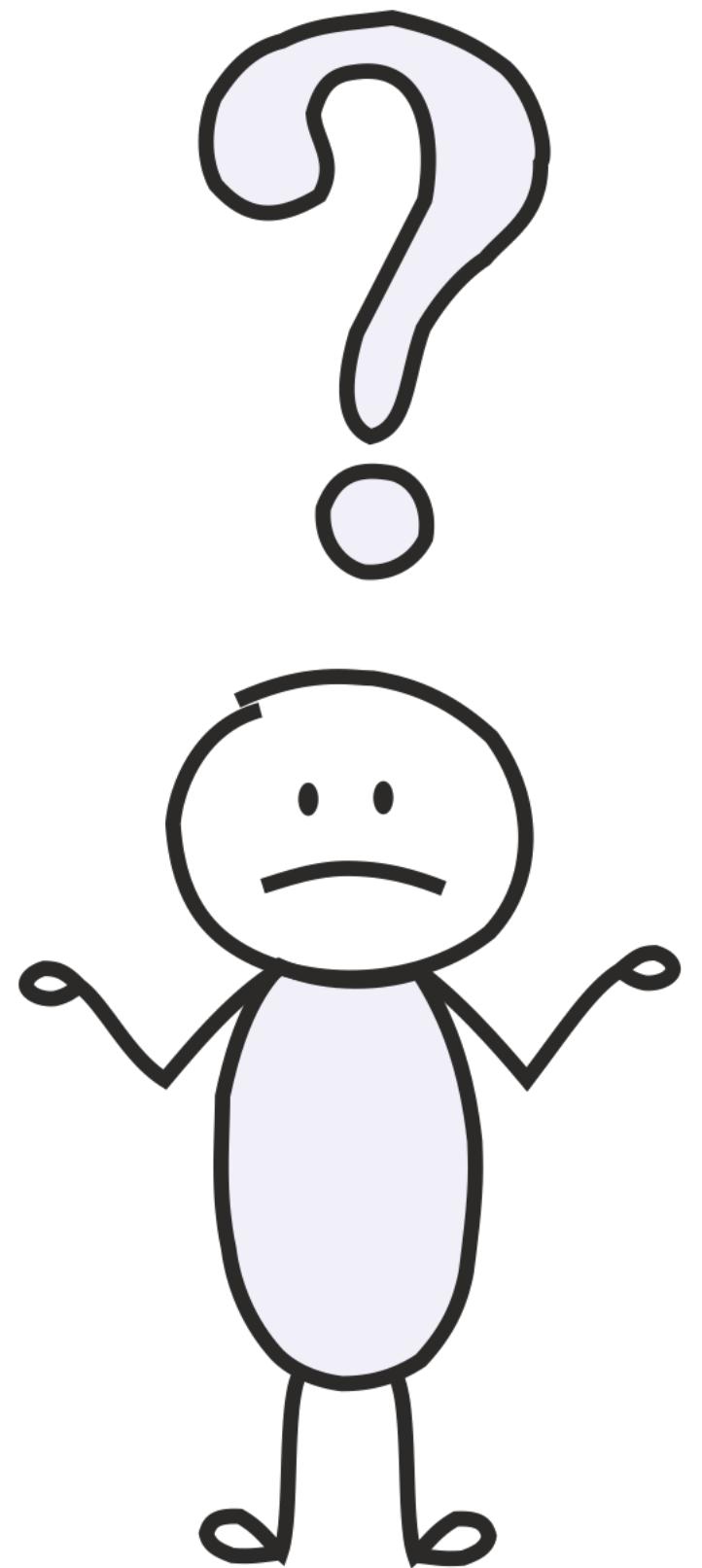
Q: How many eyes does a horse have?

A: 4. It has two eyes on the outside and two eyes on the inside.

Q: How many eyes does a horse have?

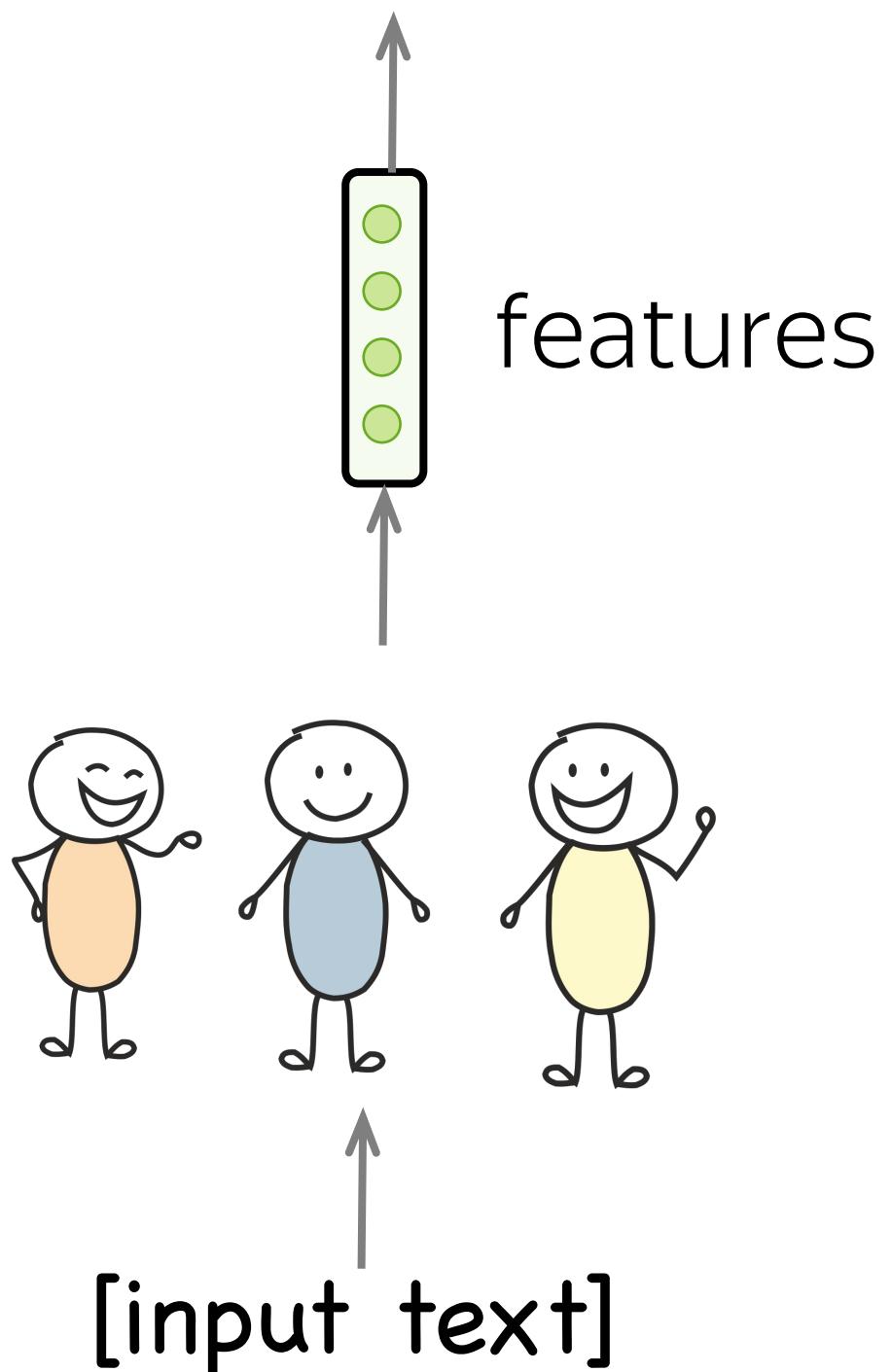
A: Nine. The equine eye contains two balls and a thin slit that closes when the horse is frightened.

Questions?



The Evolutionary Journey in NLP

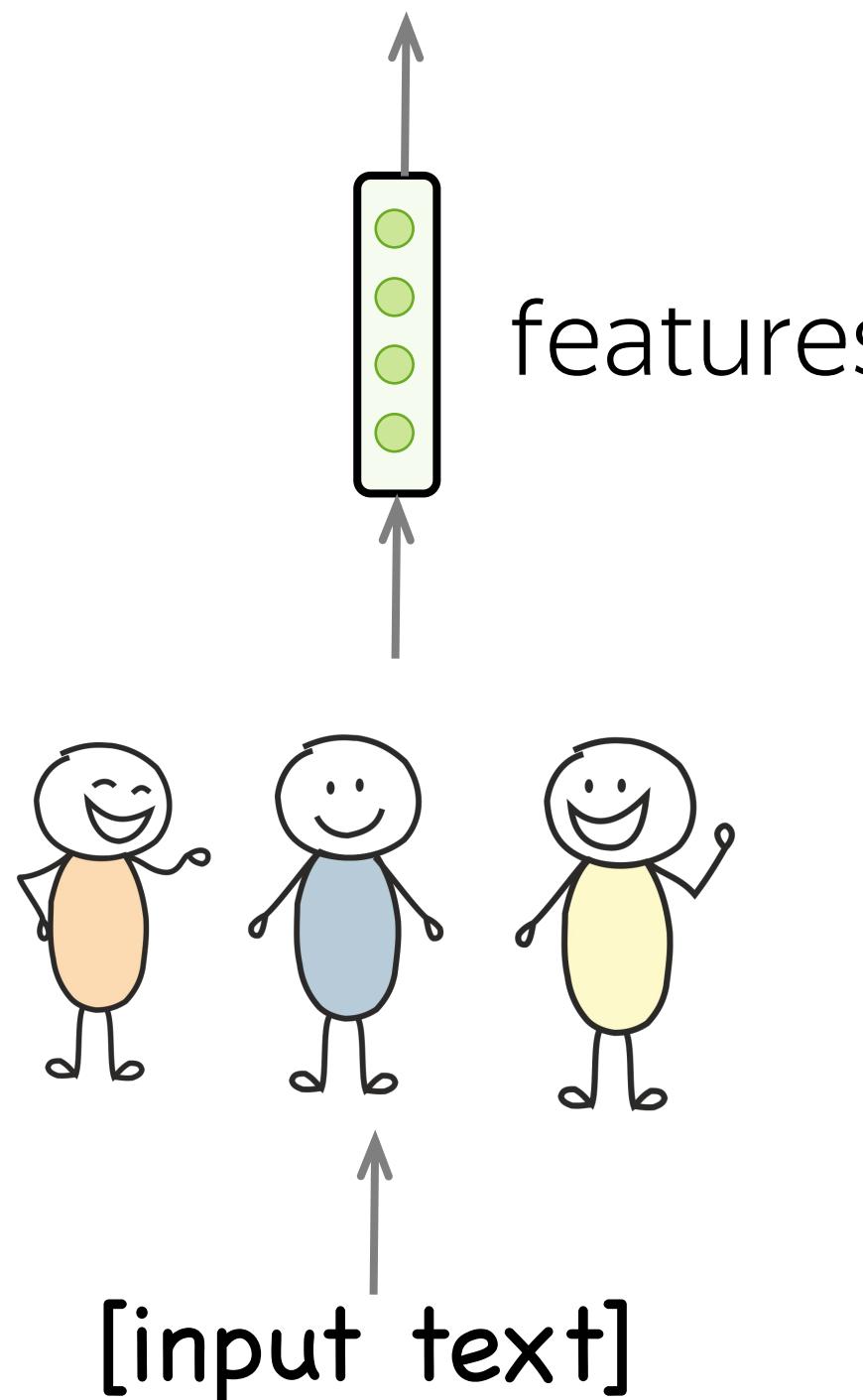
Classify e.g.
sentiment



The Evolutionary Journey in NLP

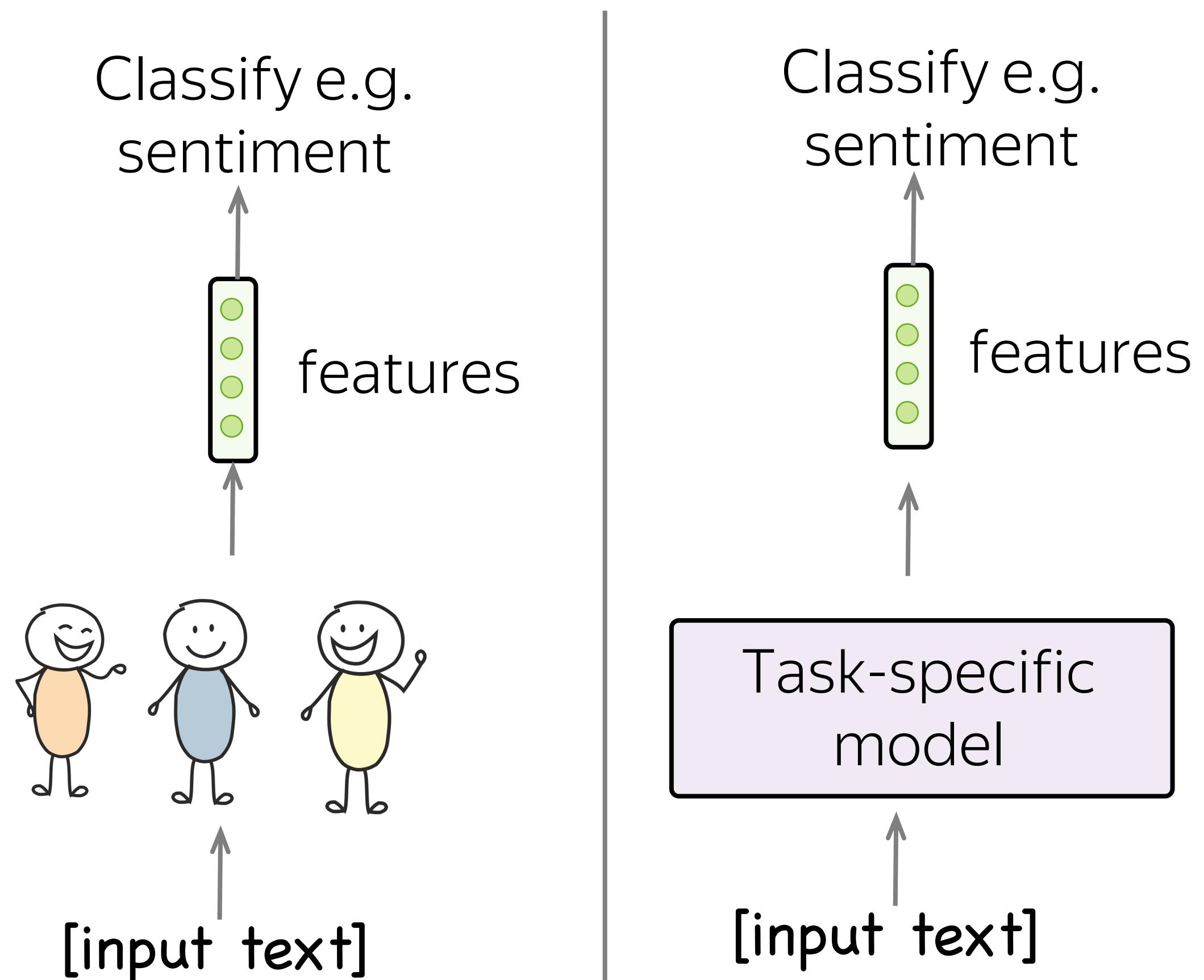
Different task –
another set of
features (and people!)

Classify e.g.
sentiment



The Evolutionary Journey in NLP

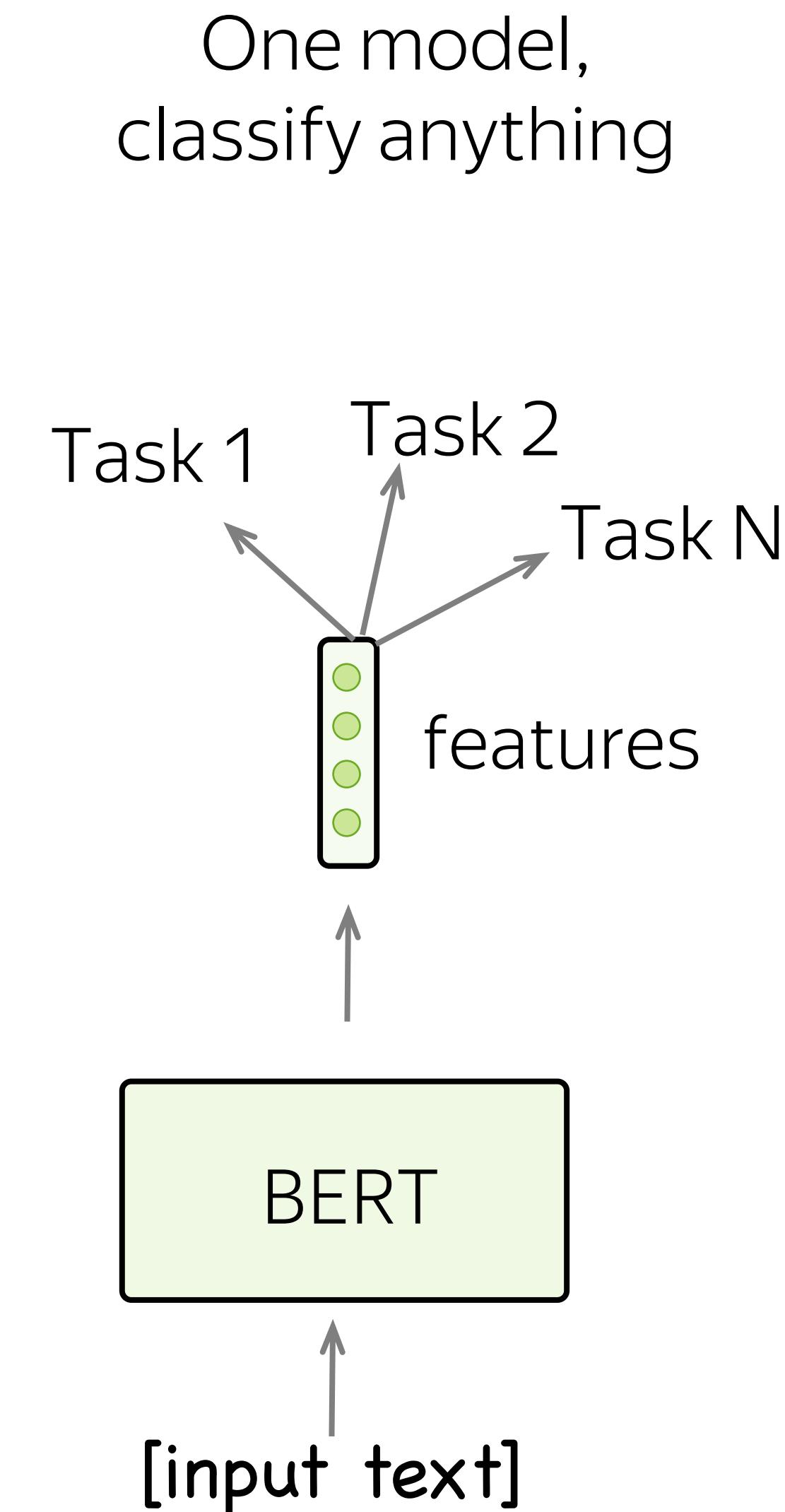
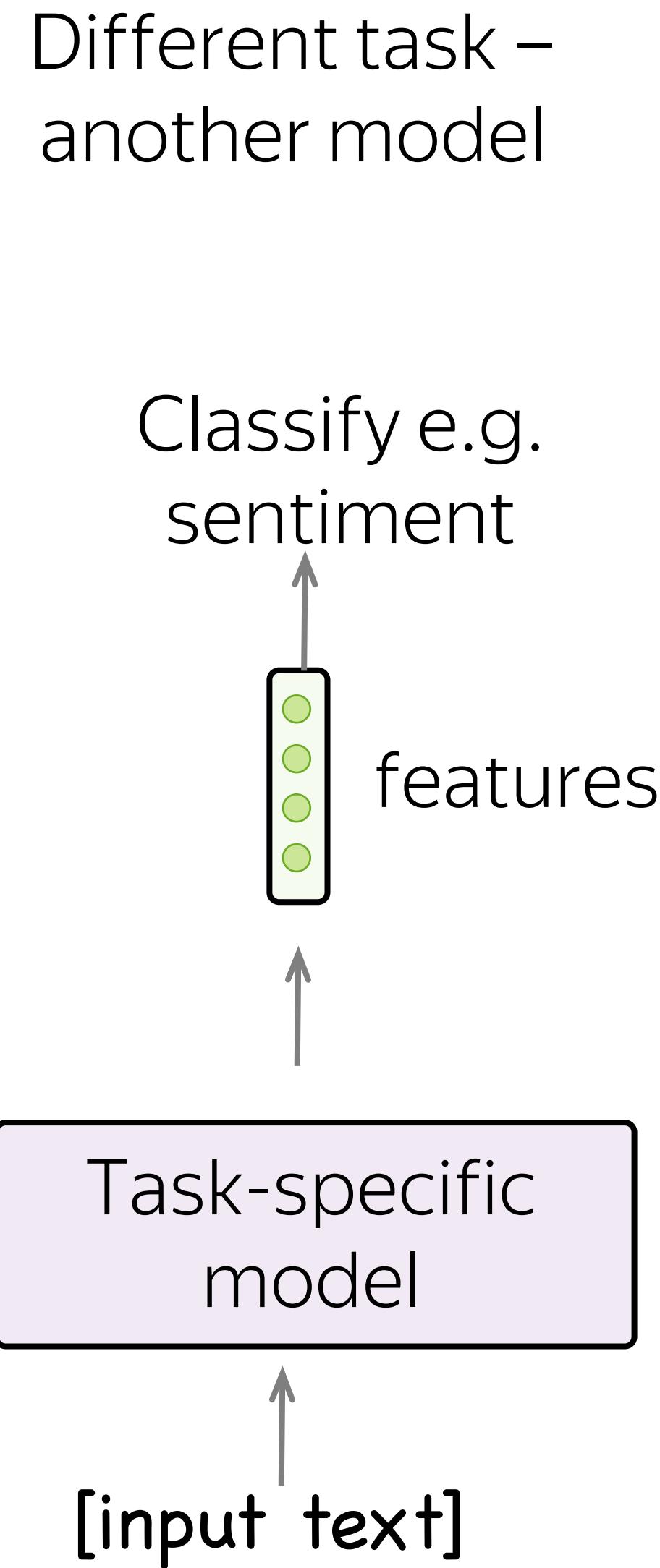
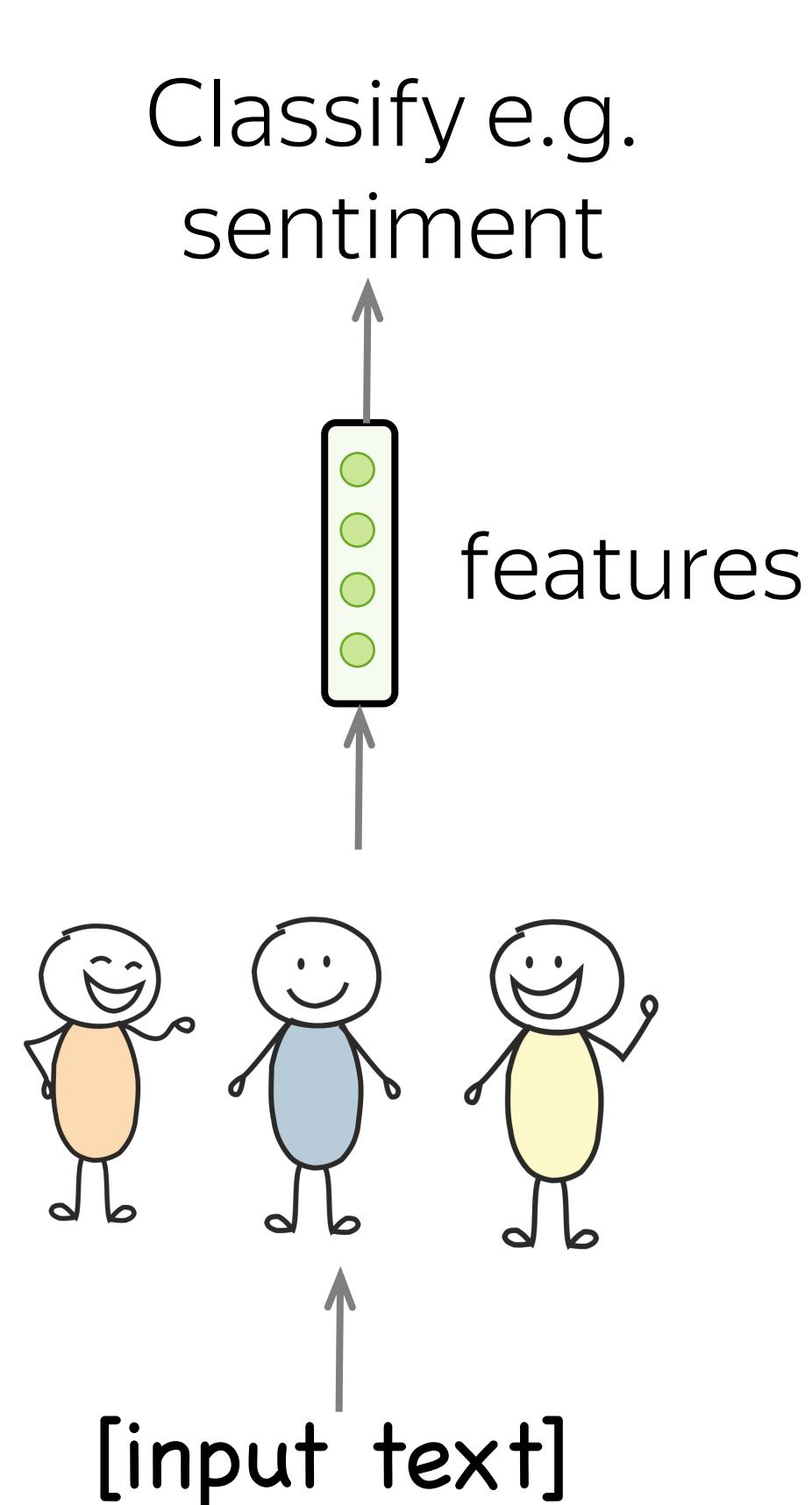
Different task –
another set of
features (and people!)



Different task –
another model

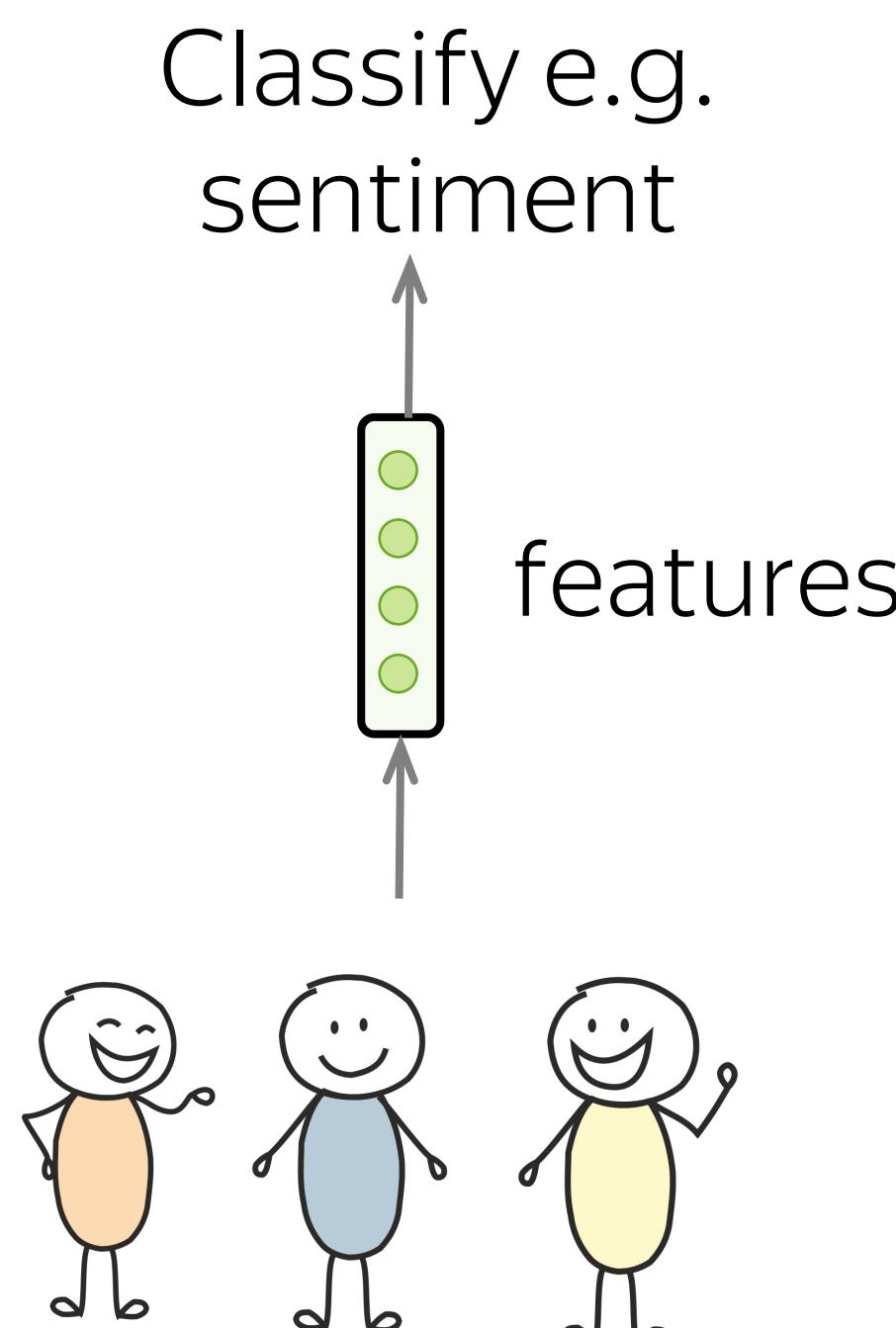
The Evolutionary Journey in NLP

Different task –
another set of
features (and people!)

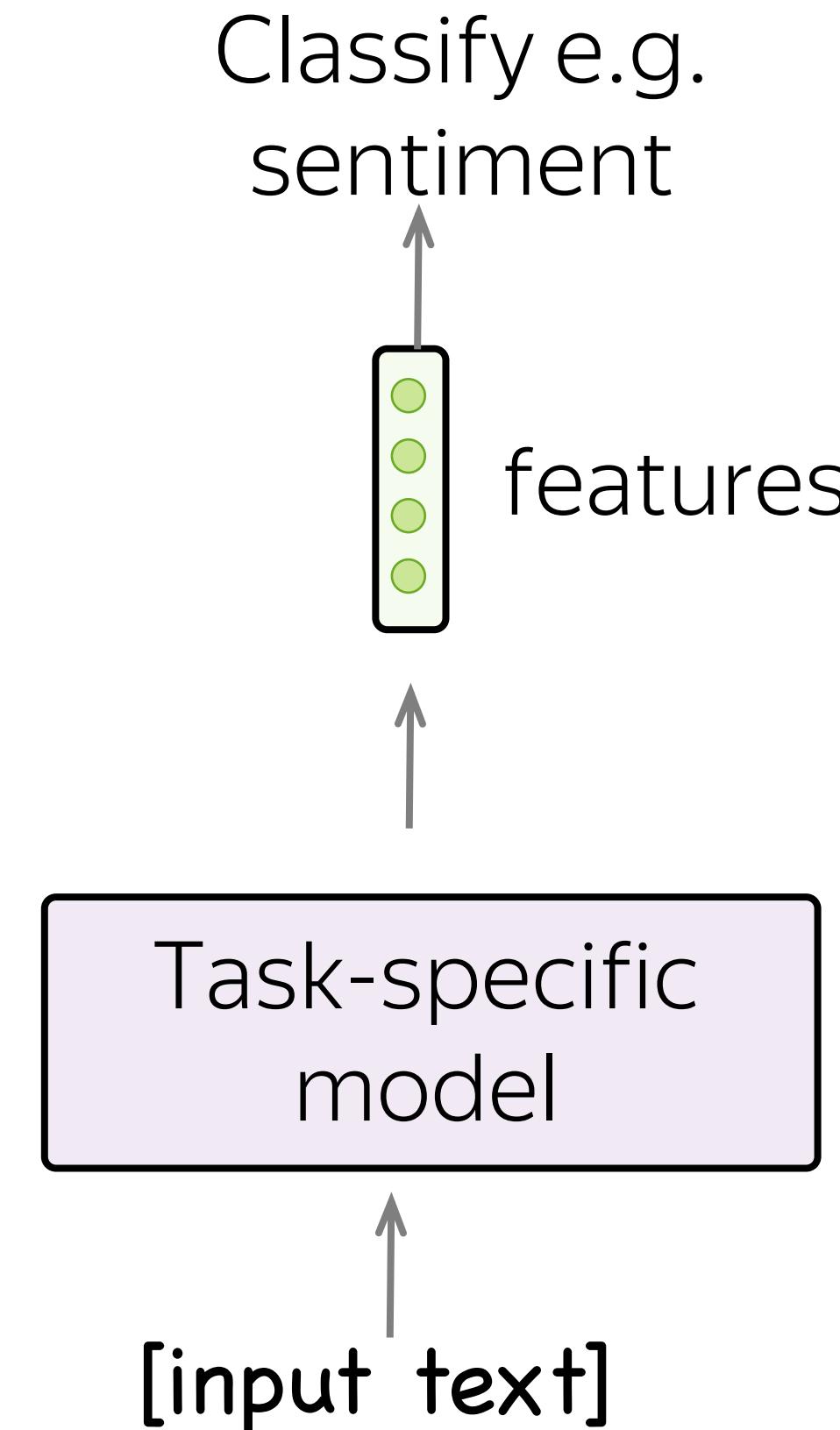


The Evolutionary Journey in NLP

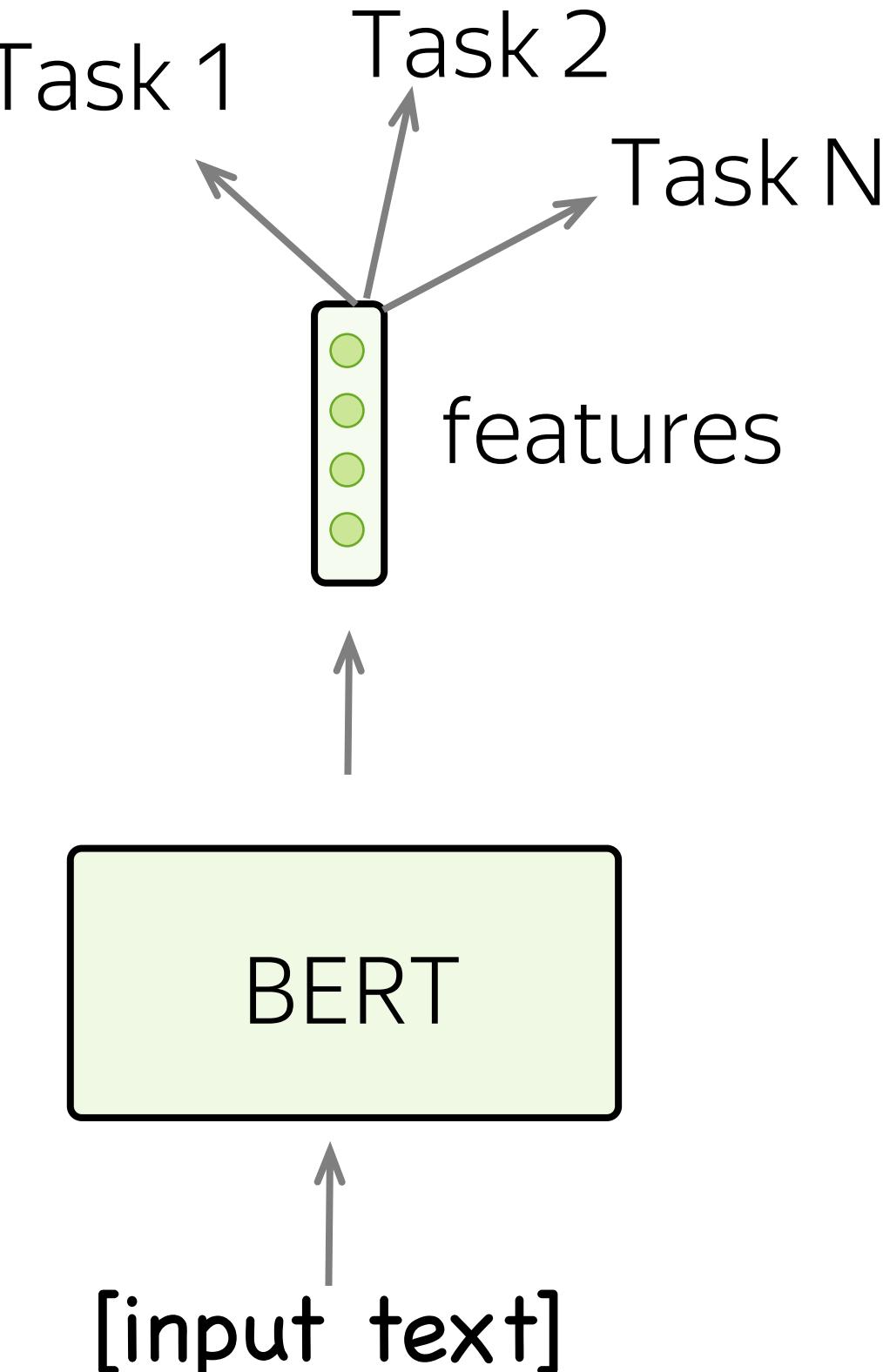
Different task –
another set of
features (and people!)



Different task –
another model



One model,
classify anything

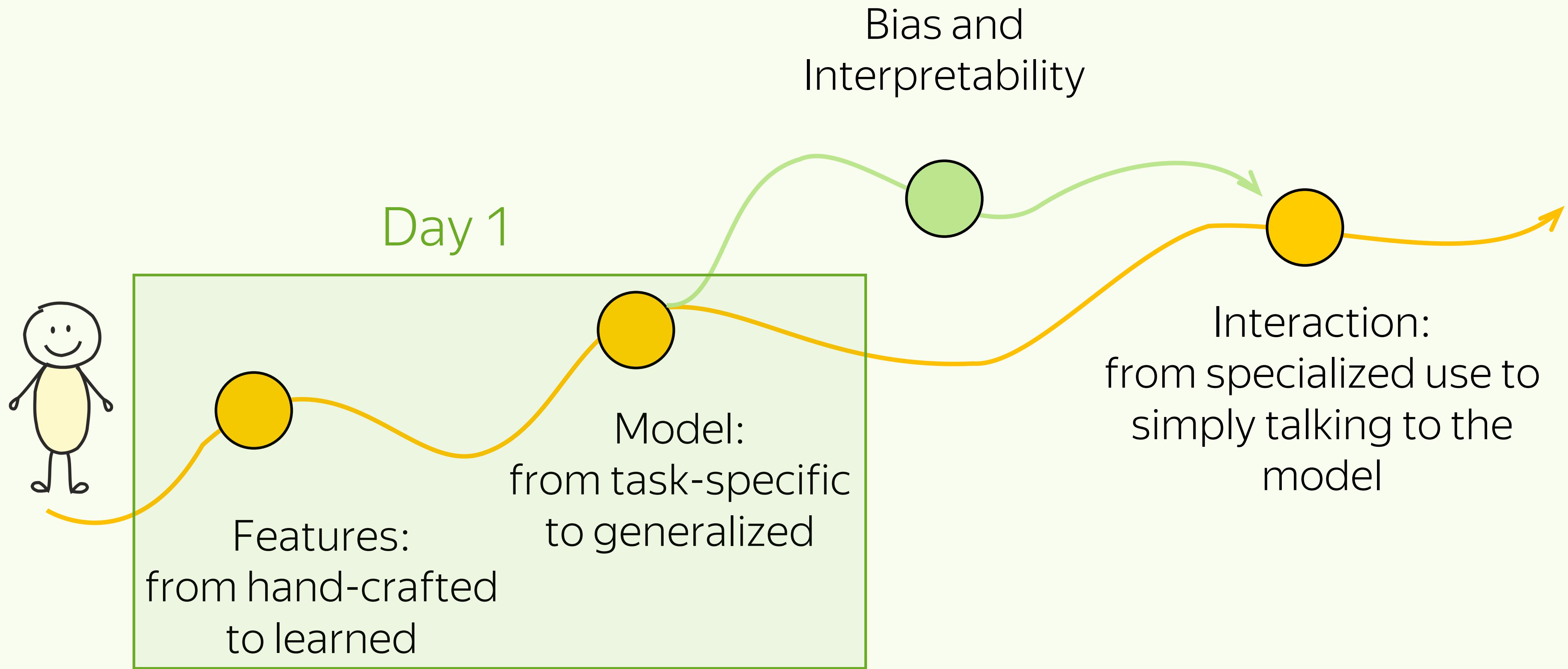


Talk to the model

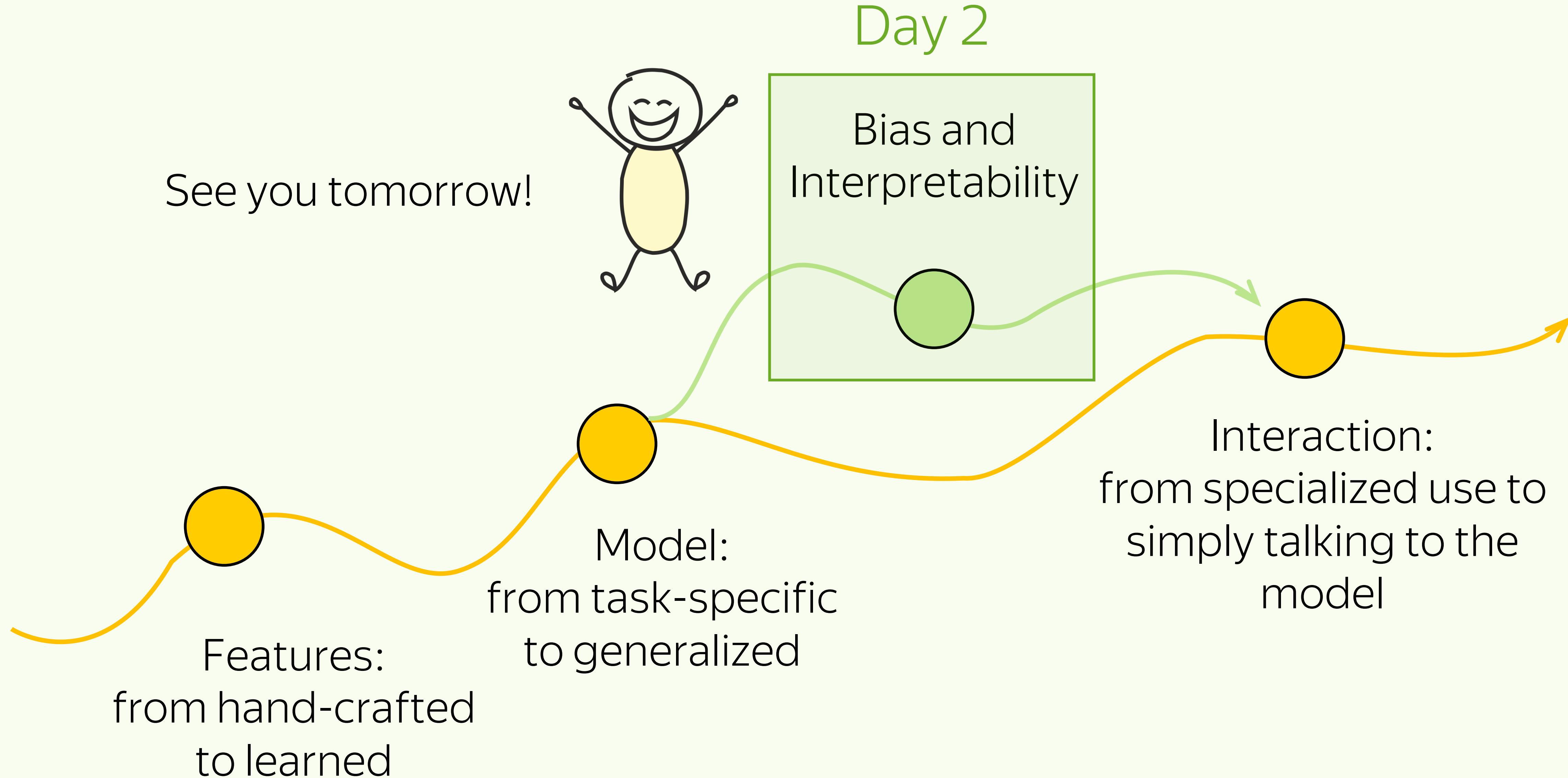
Input (prompt)
What is the sentiment
of the next sentence?
I love this movie!

Model output
positive

The Evolutionary Journey in NLP



The Evolutionary Journey in NLP



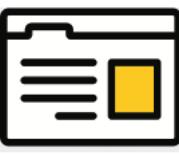
Thank you!

Lena Voita

Research Scientist at FAIR, Meta AI



lena-voita@hotmail.com



<https://lena-voita.github.io>



@lena_voita



lena-voita

