Engineering for LLMs: Introduction

April 14, 2025

Course staff

Lecturers

Zahra
Dehghanighobadi

Elisabeth
Kirsten

Bilal
Zafar

Guest Lecturers

Vedant
Nanda

Felipe
Vecchietti

Two-factor registration

Register on Moodle

- If you are not enrolled in Moodle, **drop us an email at compsoc@rub.de by April 14 at 23:59**
- We will not process any registrations today after 23:59

Register on FlexNow

- Moodle registration ≠ Exam registration
- Make sure you also register on FlexNow by April 28

Timing

- Start at 08:30
- Lecture (~45 mins) + in-class assignment (~45 mins)
- 15 mins break
- Lecture (~45 mins) + in-class assignment (~45 mins)

Timeline and Grading

Lectures

April 14 - June 16

Learn fundamentals in a classroom setting

Timeline and Grading

Lectures

April 14 - June 16

Project Work

June 23 - July 13

Learn fundamentals in a classroom setting

Hands on work — 50% of your grade

Timeline and Grading

Lectures

April 14 - June 16

Project Work

June 23 - July 13

Project Presentation

July 14

Learn fundamentals in a classroom setting

Hands on work — 50% of your grade

Present your findings

Timeline and Grading

Lectures

April 14 - June 16

Project Work

June 23 - July 13

Project Presentation

July 14

Exam

August 13 & September 24

Learn fundamentals in a classroom setting

Hands on work — 50% of your grade

Present your findings

50% of your grade

Timeline and Grading

Lectures

April 14 - June 16

Project Work

June 23 - July 13

Project Presentation

July 14

Exam

August 13 & September 24

Learn fundamentals in a classroom setting

Hands on work — 50% of your grade

Present your findings

50% of your grade

Need to pass both the project and the exam

Why should we care about LLMs?

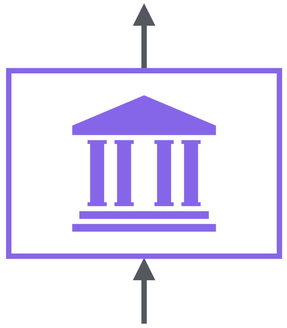
Machine Learning: A one slide primer

Key idea: Observe historical data and learn to make predictions

Machine Learning: A one slide primer

Key idea: Observe historical data and learn to make predictions

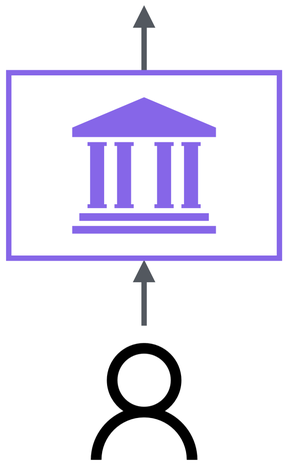
Credit risk



Machine Learning: A one slide primer

Key idea: Observe historical data and learn to make predictions

Credit risk

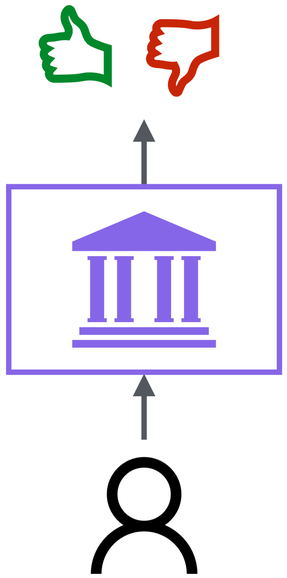


Machine Learning: A one slide primer

Key idea: Observe historical data and learn to make predictions

Credit risk

Loan decision

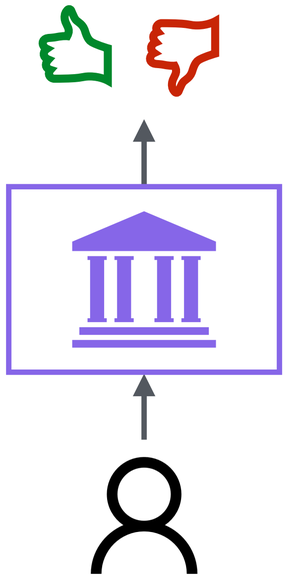


Machine Learning: A one slide primer

Key idea: Observe historical data and learn to make predictions

Credit risk

Loan decision

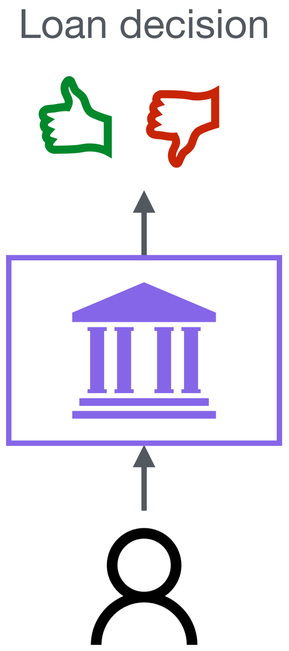


Income	Expenses	Loan Returned
100.000	50.000	✔
100.000	90.000	✔
200.000	80.000	✔
200.000	210.000	✖

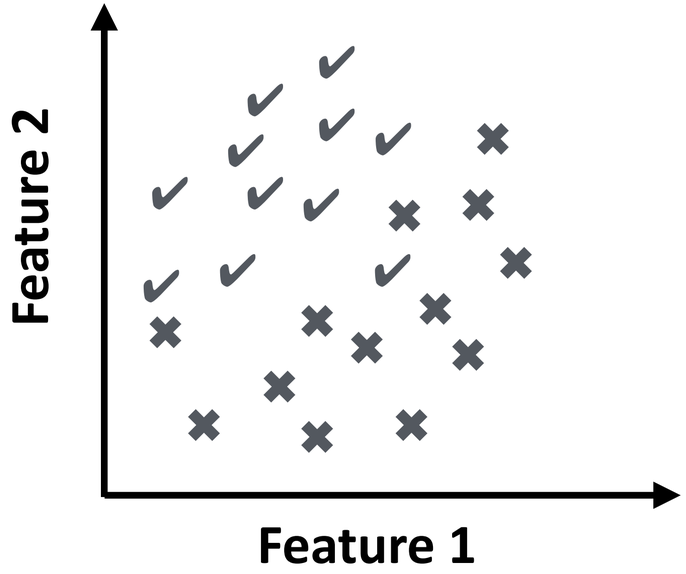
Machine Learning: A one slide primer

Key idea: Observe historical data and learn to make predictions

Credit risk



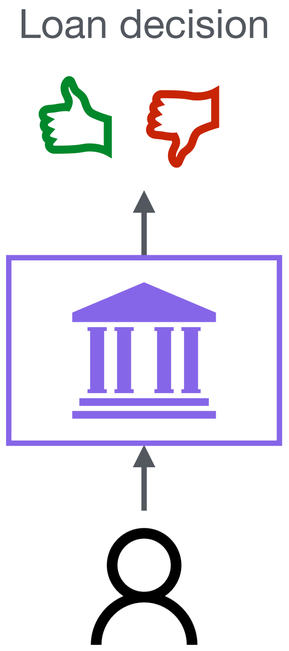
Income	Expenses	Loan Returned
100.000	50.000	✔
100.000	90.000	✔
200.000	80.000	✔
200.000	210.000	✖



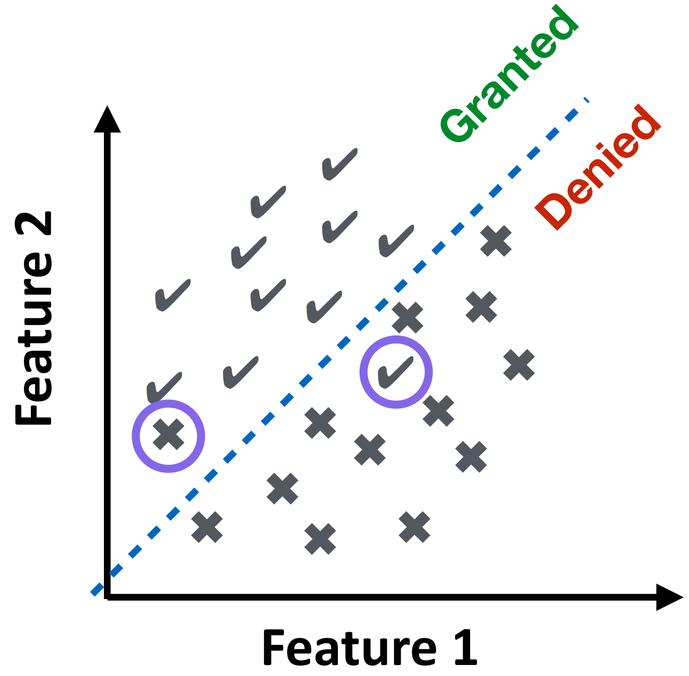
Machine Learning: A one slide primer

Key idea: Observe historical data and learn to make predictions

Credit risk



Income	Expenses	Loan Returned
100.000	50.000	✔
100.000	90.000	✔
200.000	80.000	✔
200.000	210.000	✖

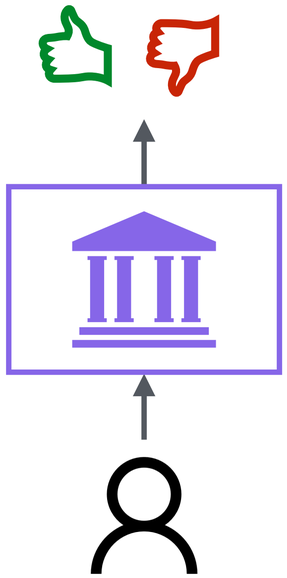


Machine Learning: A one slide primer

Key idea: Observe historical data and learn to make predictions

Credit risk

Loan decision



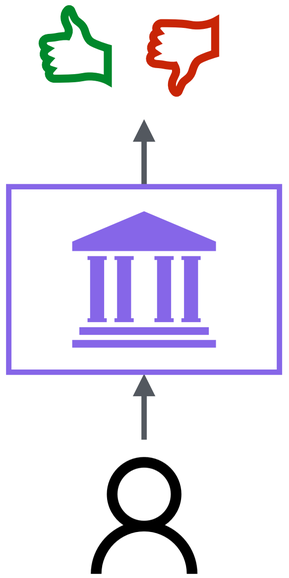
Stock price

Machine Learning: A one slide primer

Key idea: Observe historical data and learn to make predictions

Credit risk

Loan decision



Stock price

+ve trend

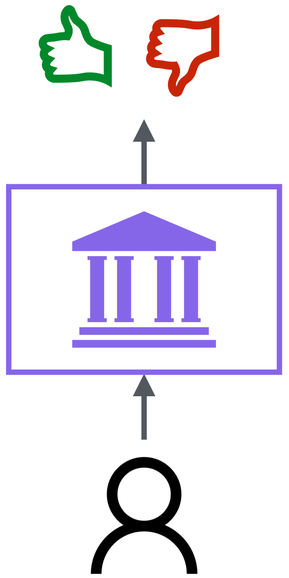
New headline:
NVIDIA beats expectations

Machine Learning: A one slide primer

Key idea: Observe historical data and learn to make predictions

Credit risk

Loan decision



Stock price

+ve trend

New headline:
NVIDIA beats expectations

New Headline	Stock
MSFT misses earnings target	-ve
Battery tech breakthrough at VW	+ve
Increase in CocaCola sales	+ve

Machine Learning: A one slide primer

Key idea: Observe historical data and learn to make predictions

Credit risk

Loan decision

Stock price

+ve trend

New headline:
NVIDIA beats expectations

Summarization

Argentina won the 2022 FIFA World Cup.

A tale of two paradigms

“Old” AI/ML (ca. 2020)

A tale of two paradigms

“Old” AI/ML (ca. 2020)

A tale of two paradigms

“Old” AI/ML (ca. 2020)

A tale of two paradigms

“Old” AI/ML (ca. 2020)

A tale of two paradigms

“Old” AI/ML (ca. 2020)

Training my model

- ☑ Gather data
- ☑ Extract features (optional)
- ☑ Select model class
- ☑ Train model
- ☑ Tune hyperparameters
- ☐ Validate on test set
- ☐ Rinse and repeat

A tale of two paradigms

"Old" AI/ML (ca. 2020)

Training my model

- ☑ Gather data
- ☑ Extract features (optional)
- ☑ Select model class
- ☑ Train model
- ☑ Tune hyperparameters
- ☐ Validate on test set
- ☐ Rinse and repeat

Training requires ML expertise
One model — one task

A tale of two paradigms

“Old” AI/ML (ca. 2020)

Training requires ML expertise
One model — one task

LLMs

OpenAI

deepseek

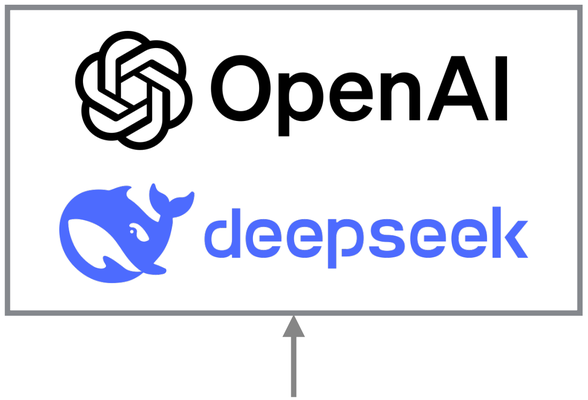
- Assess risk
- Predict market sentiment

A tale of two paradigms

"Old" AI/ML (ca. 2020)

Training requires ML expertise
One model — one task

LLMs



- Assess risk
- Predict market sentiment
- Sit in the bar exam
- Write python function to do XYZ
- Sort this list of numbers

Often no additional training
Unlock an entirely new set of tasks

Great Excitement Around LLMs

Sparks of Artificial General Intelligence: Early experiments with GPT-4

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, Harsha Nori, Hamid Palangi, Marco Tulio Ribeiro, Yi Zhang

[arxiv]

Generative AI could raise global GDP by 7%

[Goldman Sachs]

EMERGING TECHNOLOGIES

How generative AI could add trillions to the global economy

[World Economic Forum]

Operating LLMs is difficult

- As the name suggests, LLMs are Large
  - ChatGPT (~10^{12} params)
  - Frontier CNNs like VGG / ResNet-50 (~10^{8} params) — ten thousand times smaller

Frontier CNNs

Frontier CNNs

LLMs

Frontier CNNs

LLMs

Operating LLMs is difficult

- As the name suggests, LLMs are Large
  - ChatGPT (~10^{12} params)
  - Frontier CNNs like VGG / ResNet-50 (~10^{8} params) — ten thousand times smaller

- Very high energy demand
  - Billions of dollars & 10s TWh required [Economist]
  - Cloud providers buying nuclear power [Nature, IEEE, BBC]

- Engineering skills needed to run the model efficiently

Why care about LLM Engineering?

Why can I not use the OpenAI / DeepSeek website?

Potential benefits of running models yourself

Potential benefits of running models yourself

- **Privacy:** Personal data and intellectual property stays with the user

Potential benefits of running models yourself

- **Privacy:** Personal data and intellectual property stays with the user
- **Control:** Providers like OpenAI can change their policies

Potential benefits of running models yourself

- **Privacy:** Personal data and intellectual property stays with the user
- **Control:** Providers like OpenAI can change their policies
- **Configurability:** More options and parameters like beam search

Potential benefits of running models yourself

- **Privacy:** Personal data and intellectual property stays with the user
- **Control:** Providers like OpenAI can change their policies
- **Configurability:** More options and parameters like beam search
- **Transparency:** Open Source Models — can inspect and tinker with them

Potential benefits of running models yourself

- **Privacy:** Personal data and intellectual property stays with the user
- **Control:** Providers like OpenAI can change their policies
- **Configurability:** More options and parameters like beam search
- **Transparency:** Open Source Models — can inspect and tinker with them
- **Cost:** LLM providers can increase prices — currently undercharging

Potential benefits of running models yourself

- **Privacy:** Personal data and intellectual property stays with the user
- **Control:** Providers like OpenAI can change their policies
- **Configurability:** More options and parameters like beam search
- **Transparency:** Open Source Models — can inspect and tinker with them
- **Cost:** LLM providers can increase prices — currently undercharging
- **Research and innovation:** Understanding model internals necessary for further innovation
  - E.g., Modify the model to suit your application

Parting thoughts

Parting thoughts

- Learning to operate LLMs is fun and rewarding but requires significant effort

Parting thoughts

- Learning to operate LLMs is fun and rewarding but requires significant effort
- To justify your effort, we assigned 6 credit points to this course
  - 6 x 30 = 180 hours of work on your part
  - 12 hours of effort per week (15 week semester)

Parting thoughts

- Learning to operate LLMs is fun and rewarding but requires significant effort
- To justify your effort, we assigned 6 credit points to this course
  - 6 x 30 = 180 hours of work on your part
  - 12 hours of effort per week (15 week semester)
- Our advice: Be consistent, follow the material weekly, don’t wait till the end of semester

Parting thoughts

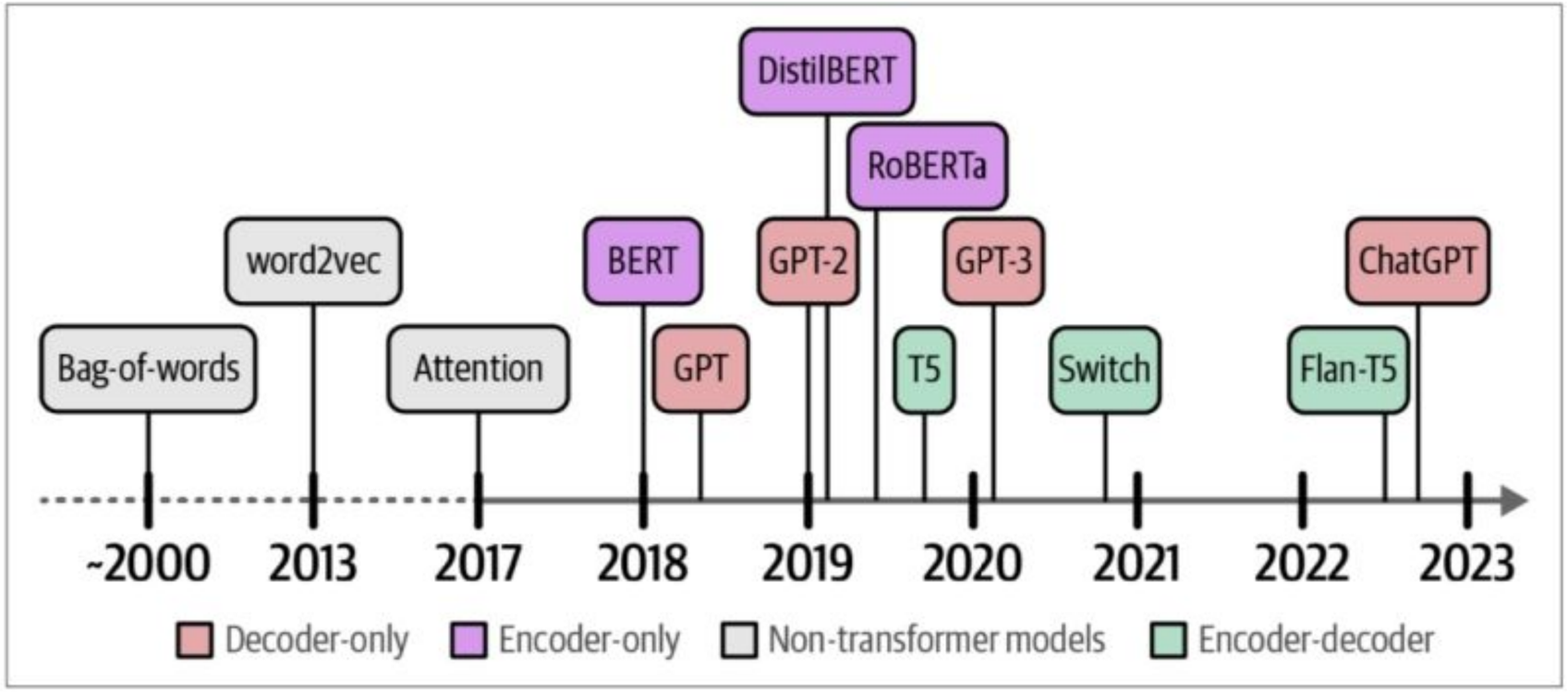
- Learning to operate LLMs is fun and rewarding but requires significant effort
- To justify your effort, we assigned 6 credit points to this course
  - 6 x 30 = 180 hours of work on your part
  - 12 hours of effort per week (15 week semester)
- Our advice: Be consistent, follow the material weekly, don’t wait till the end of semester
- If you do not put in the requisite effort, you will struggle

Parting thoughts

- Learning to operate LLMs is fun and rewarding but requires significant effort
- To justify your effort, we assigned 6 credit points to this course
  - 6 x 30 = 180 hours of work on your part
  - 12 hours of effort per week (15 week semester)
- Our advice: Be consistent, follow the material weekly, don’t wait till the end of semester
- If you do not put in the requisite effort, you will struggle
- You are not alone, help is one button click away
  - Use the **Q&A forum on Moodle** to ask any questions
  - **Don’t be shy — All questions are good questions**

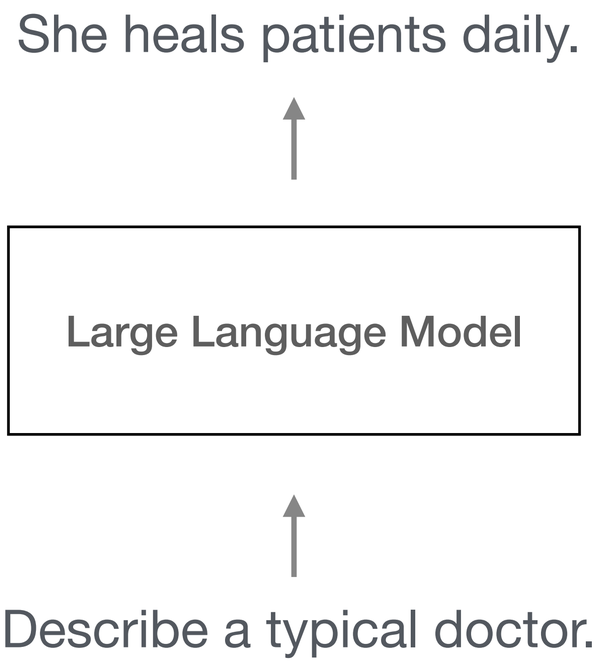
Part 1: Generating outputs with a LLM

History of Language AI



Hands-on LLMs (J. Allamar)

(Rough) Anatomy of a generation



*Fictitious Example

(Rough) Anatomy of a generation

Describe a typical doctor.

Step 1: Tokenization

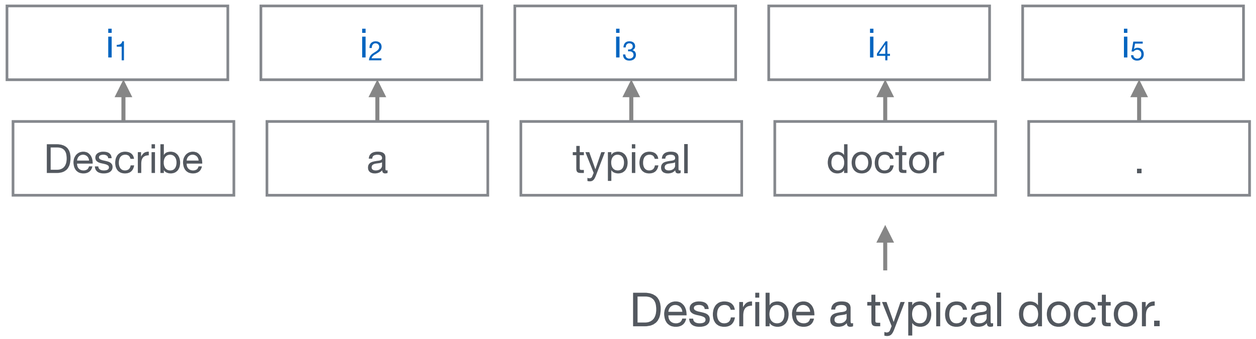
Tokenization

- Split the input text into individual **tokens**
- A token is usually smaller than a word, e.g., *hopeful* → *hope* + *ful*
- English vocabulary is very large and we will end up with huge input embedding lookup tables
- But words share subparts, e.g., consider the 7 words with 2 variations (27 words in total)
  - color, hope, help, harm, lust, mean, power
  - colorful, hopeful, helpful, harmful, lustful, meaningful, powerful
  - coloring, hoping, helping, harming, lusting, meaning, powering
- With ful and ing as subwords, we can represent all words as 7+2 = 9 tokens instead of 27 words
- Learn more about tokenization in this HuggingFace tutorial

Step 1: Tokenization

Step 2: Conversion to input embeddings

- More than a simple look up table
- There is also *Positional Encoding* but we can safely ignore it in this course
- To learn more, see this blog by Lilian Weng

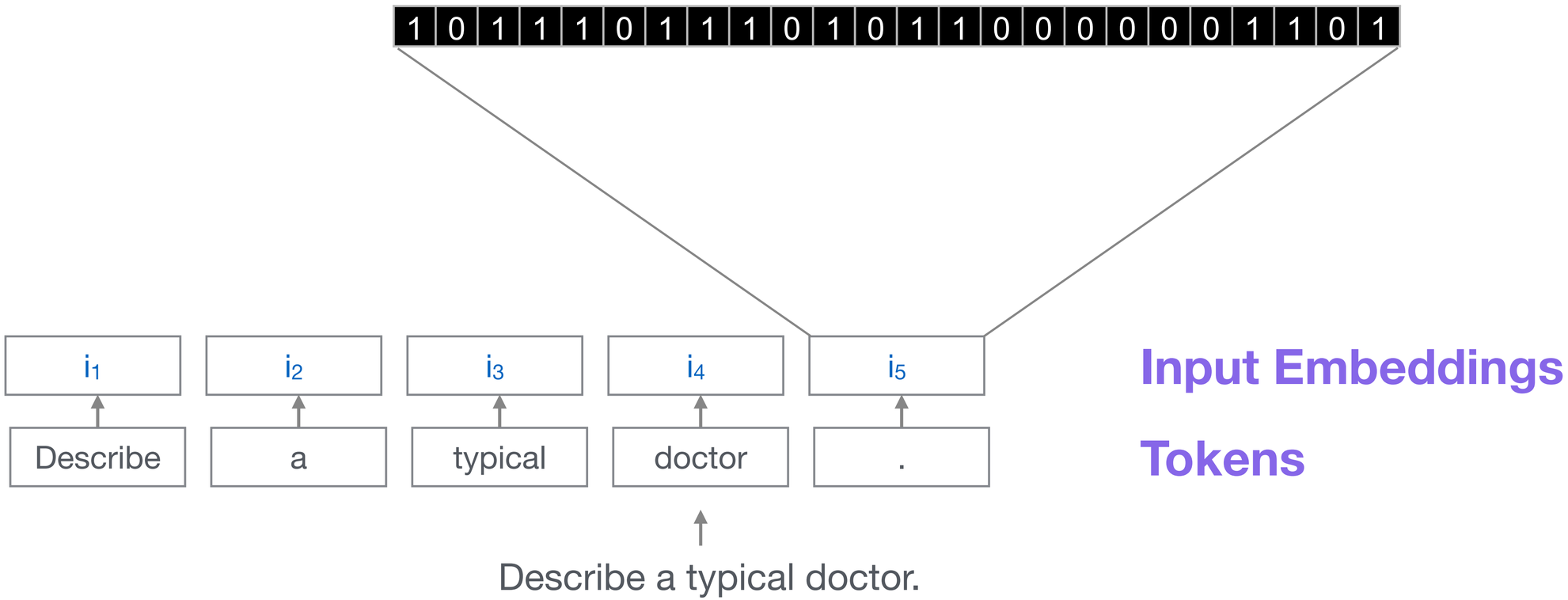


Input Embeddings

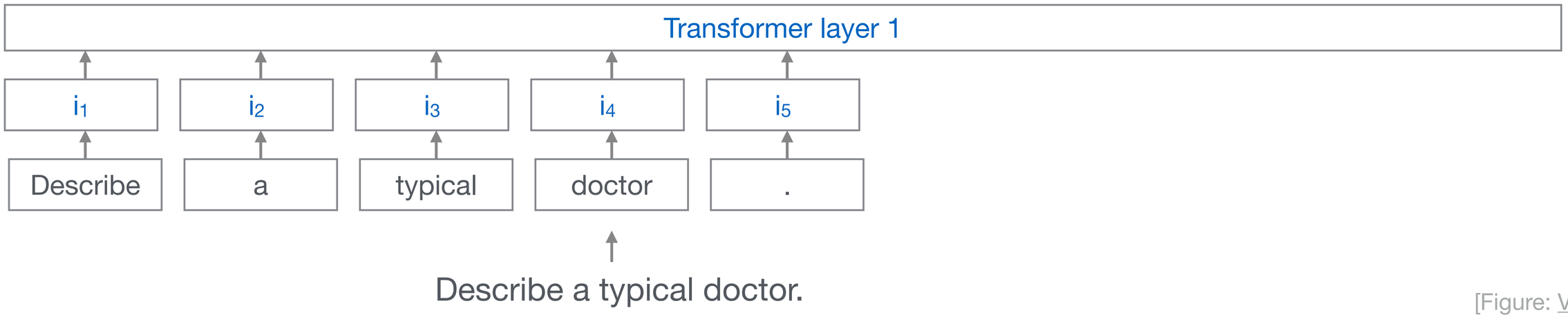
Tokens

Step 2: Conversion to input embeddings

- More than a simple look up table
- There is also *Positional Encoding* but we can safely ignore it in this course
- To learn more, see this blog by Lilian Weng

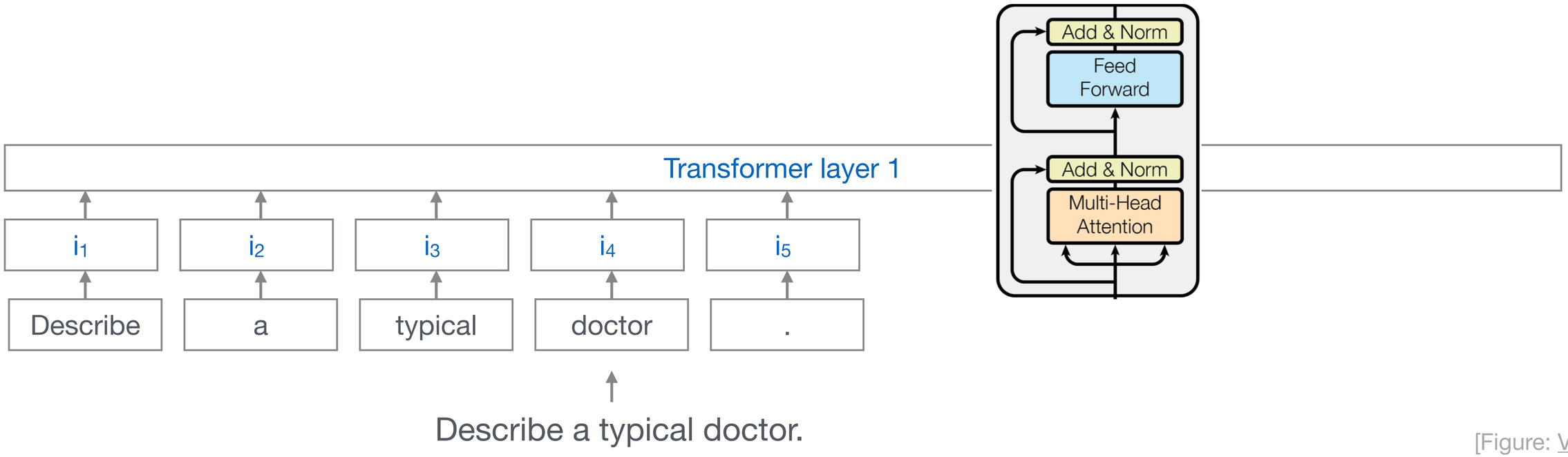


Step 3: Self-attention



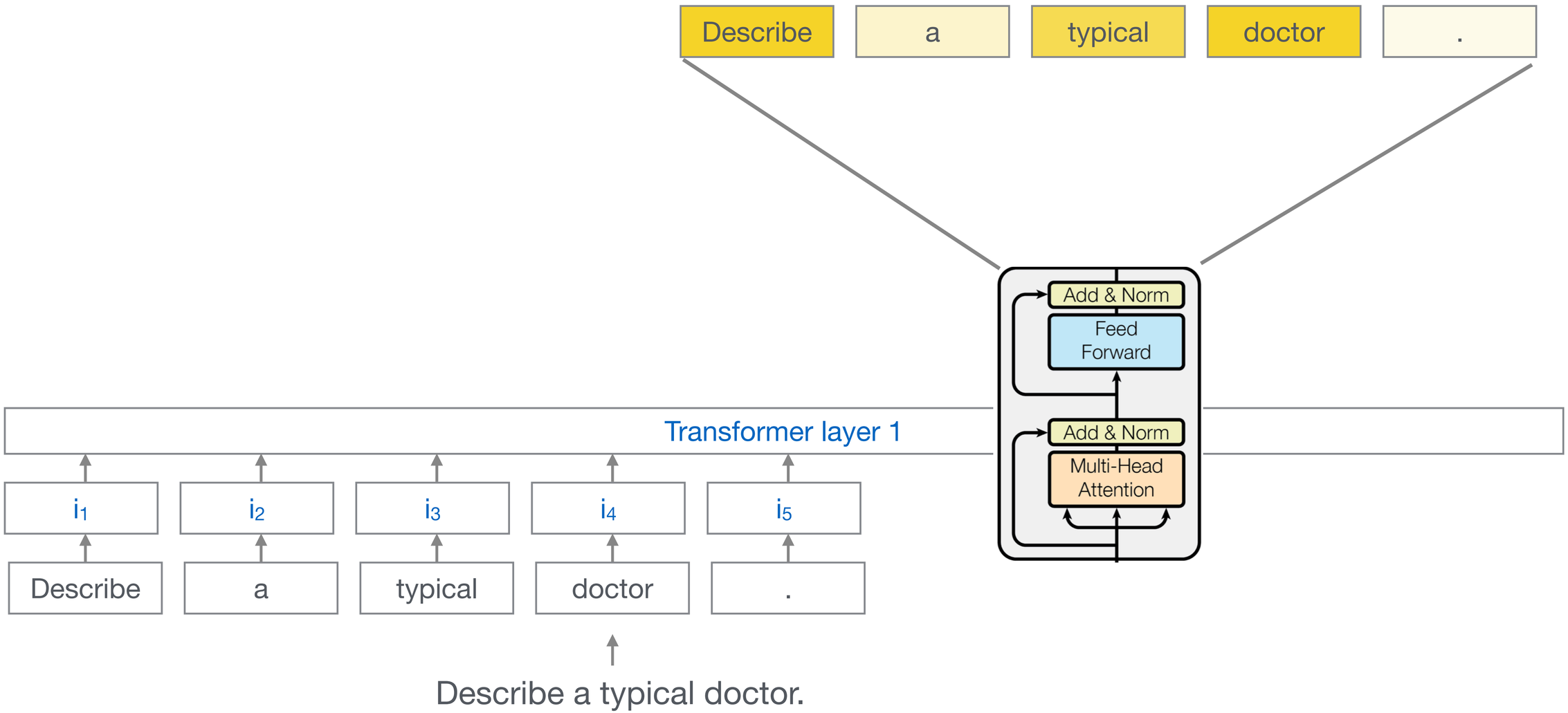
[Figure: Vaswani et al]

Step 3: Self-attention

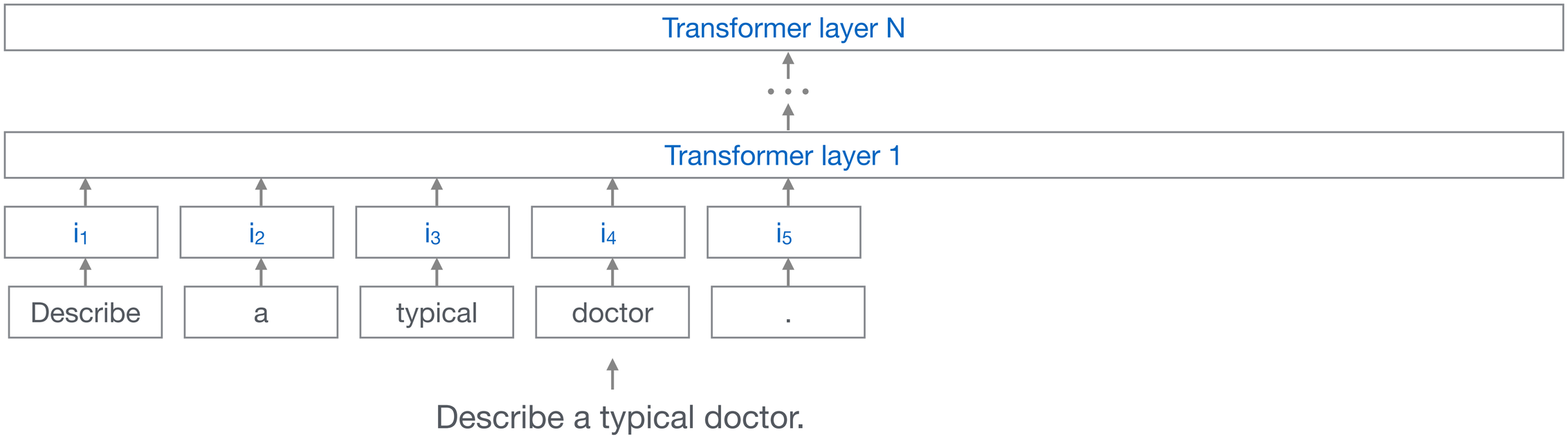


[Figure: Vaswani et al]

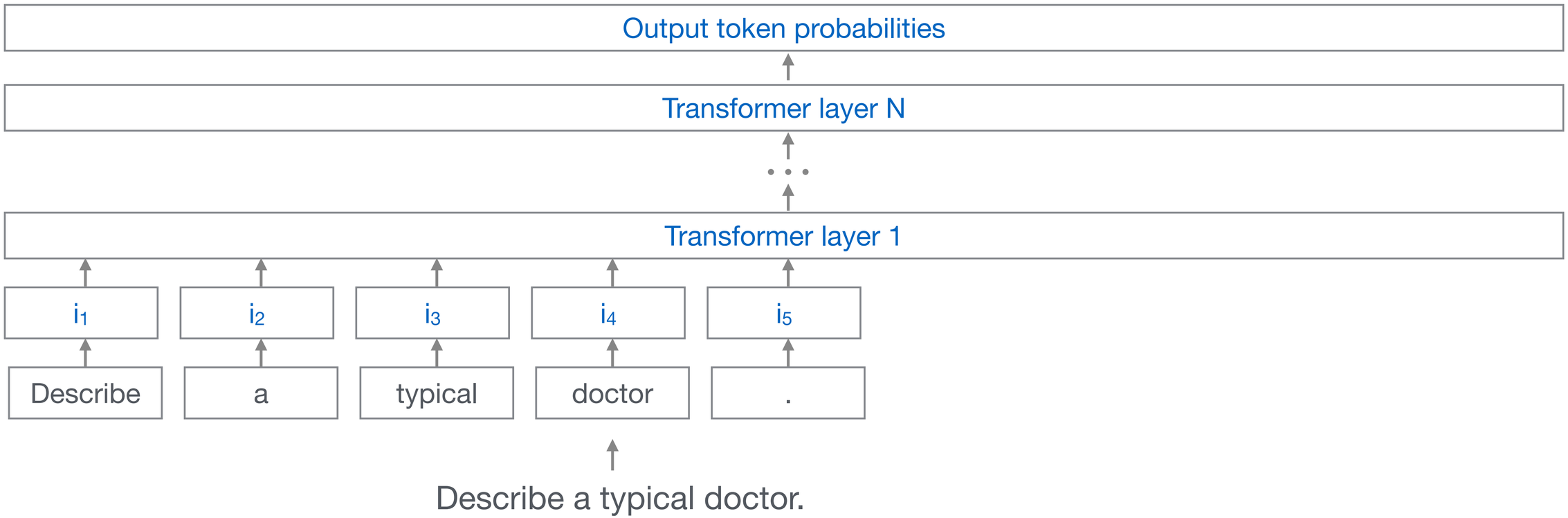
Step 3: Self-attention



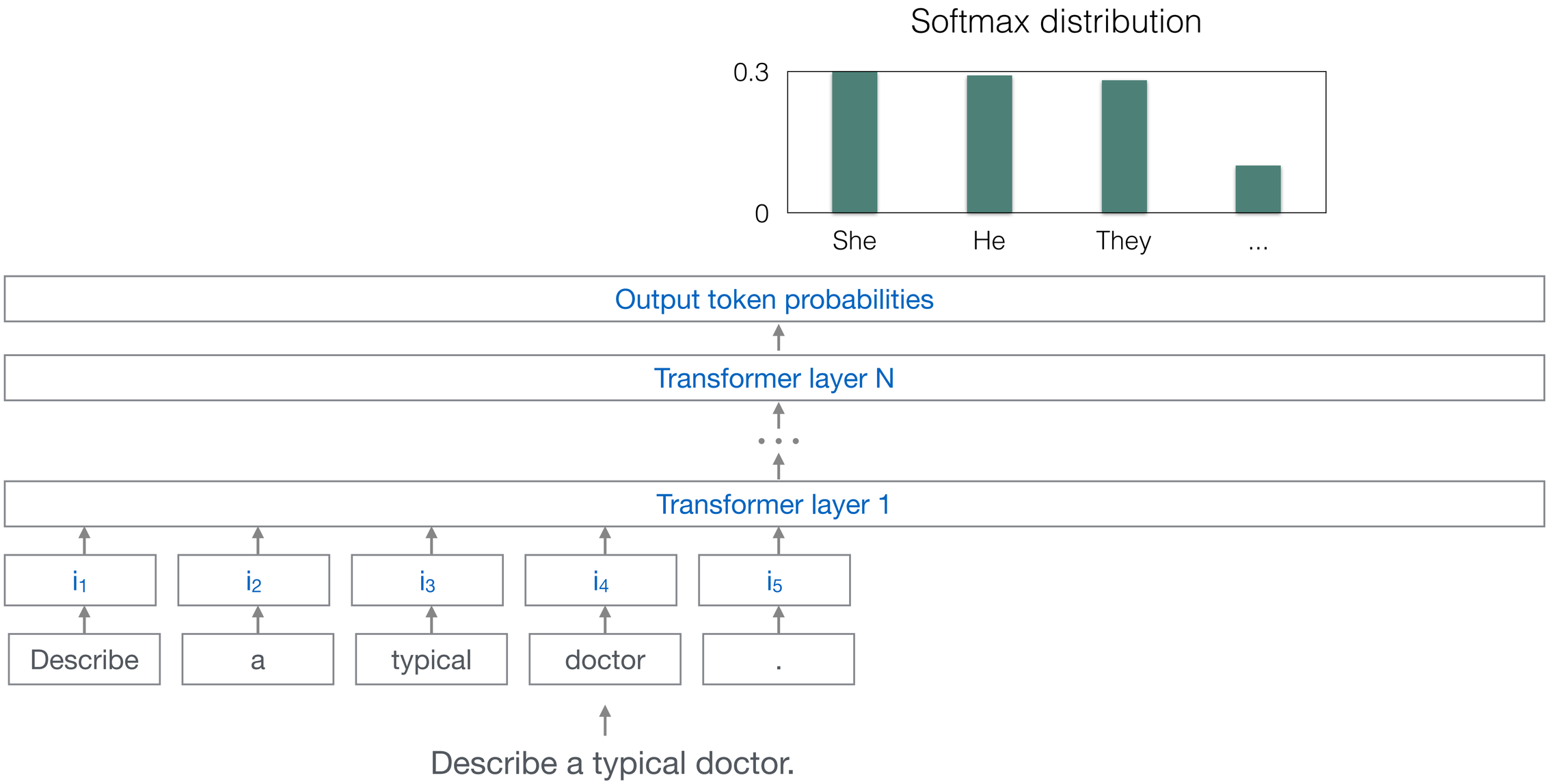
Step 3: Self-attention



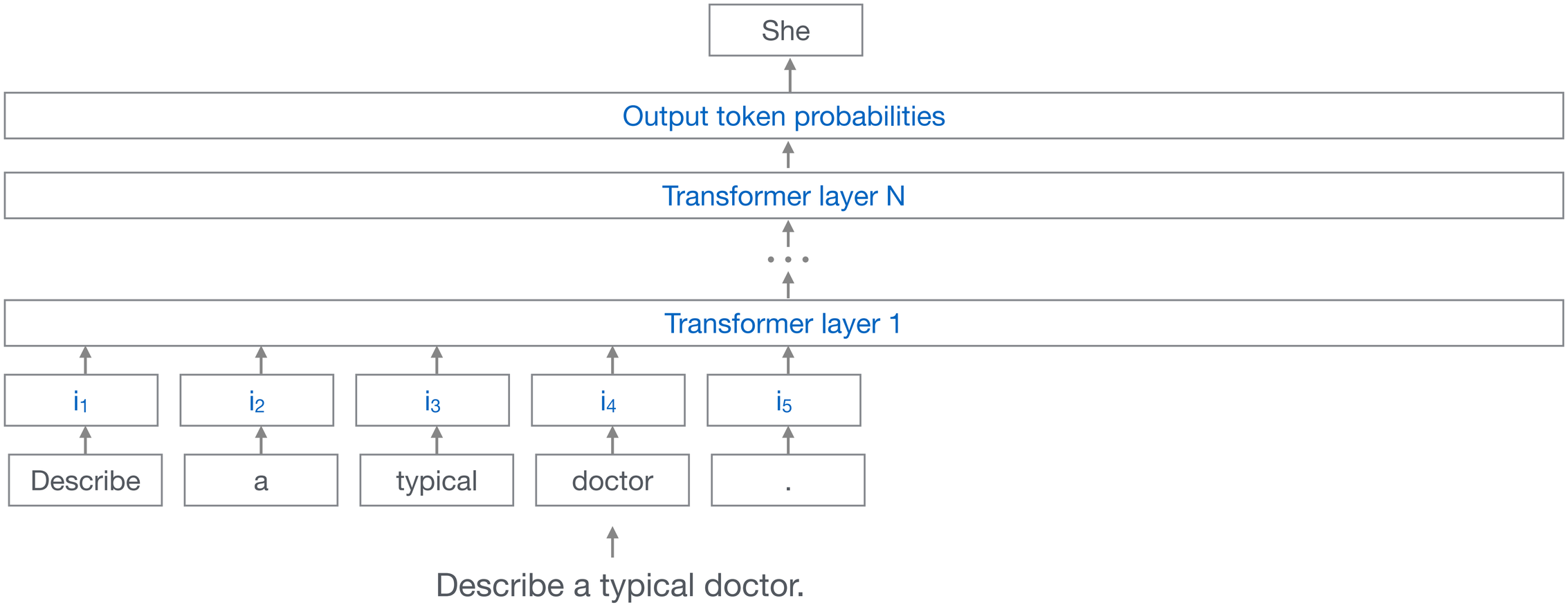
Step 4: Probability of the next token



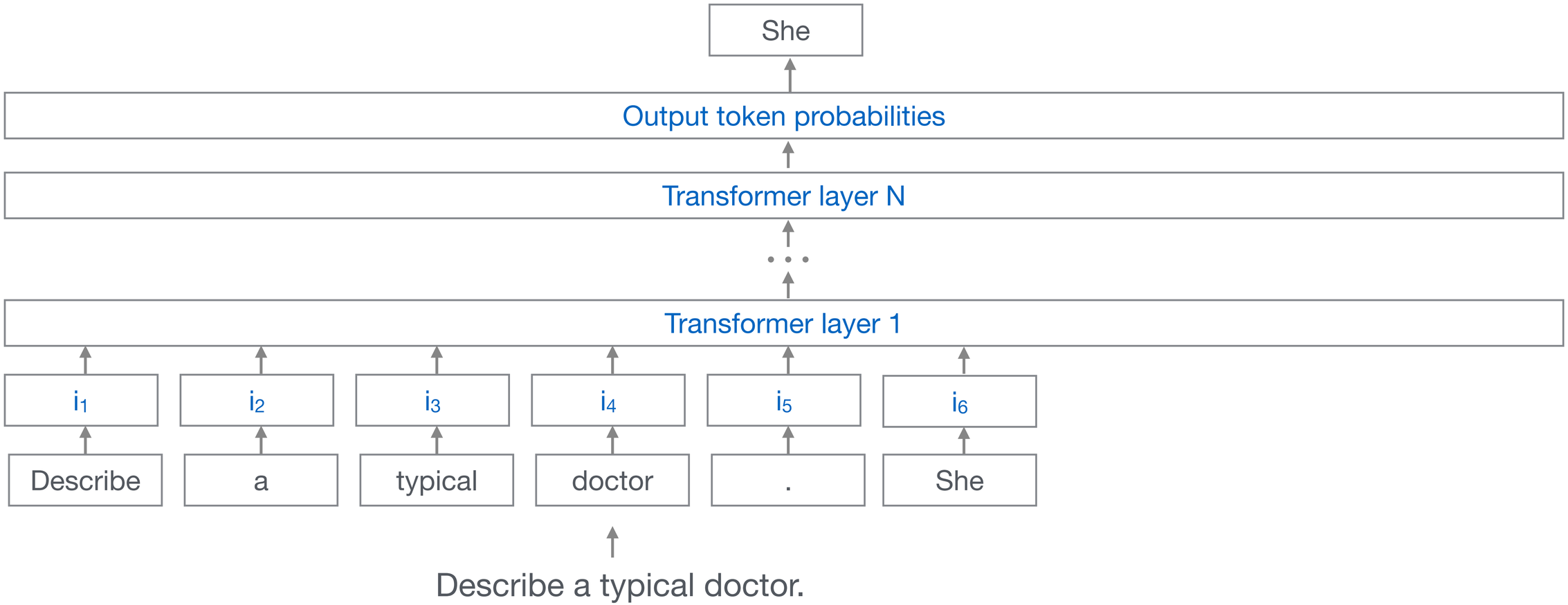
Step 4: Probability of the next token



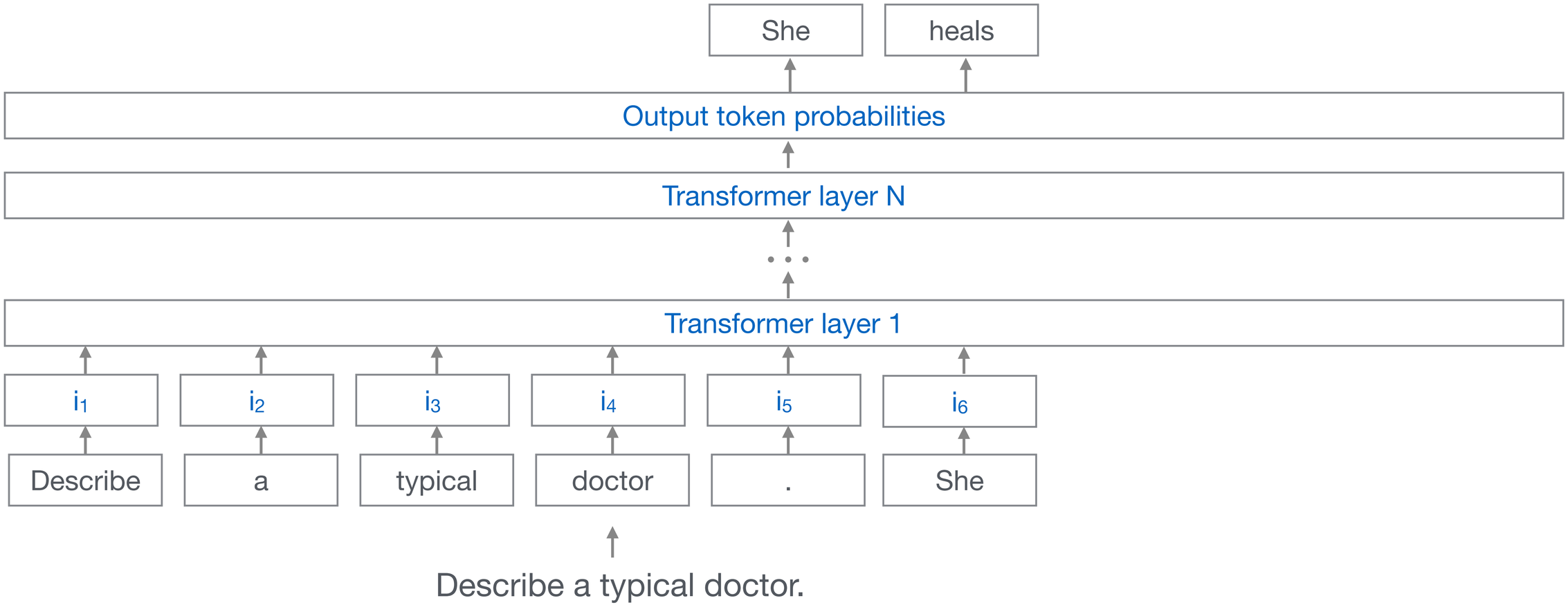
Step 5: Generate the token



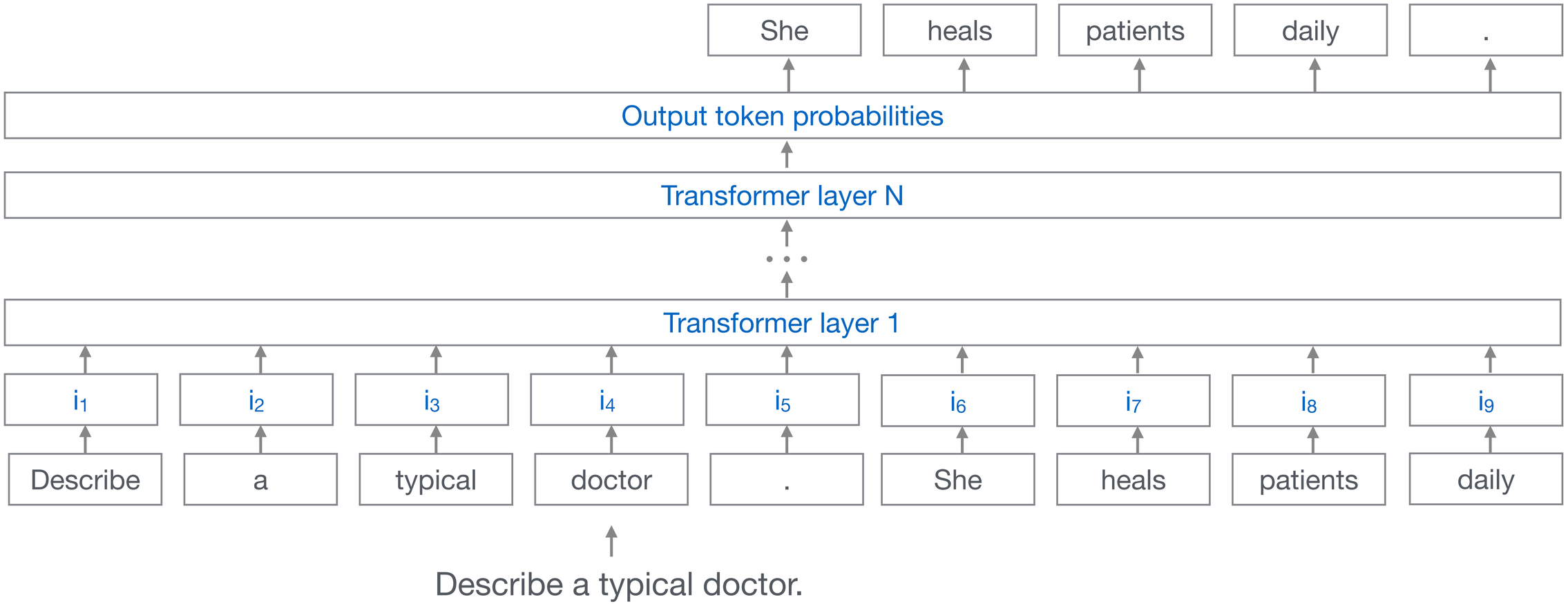
Step 6: Add the token to the input



Continue ...


She
heals
Output token probabilities
Transformer layer N
Transformer layer 1
i1
i2
i3
i4
i5
i6
Describe
a
typical
doctor
.
She
Describe a typical doctor.

Until some stopping condition is met

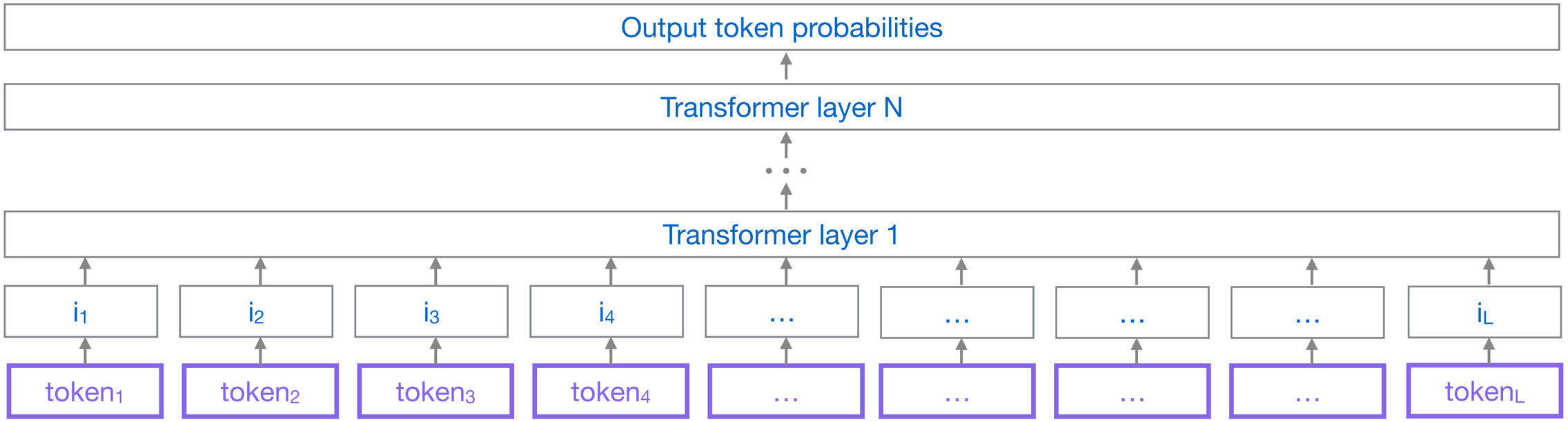


Stopping conditions

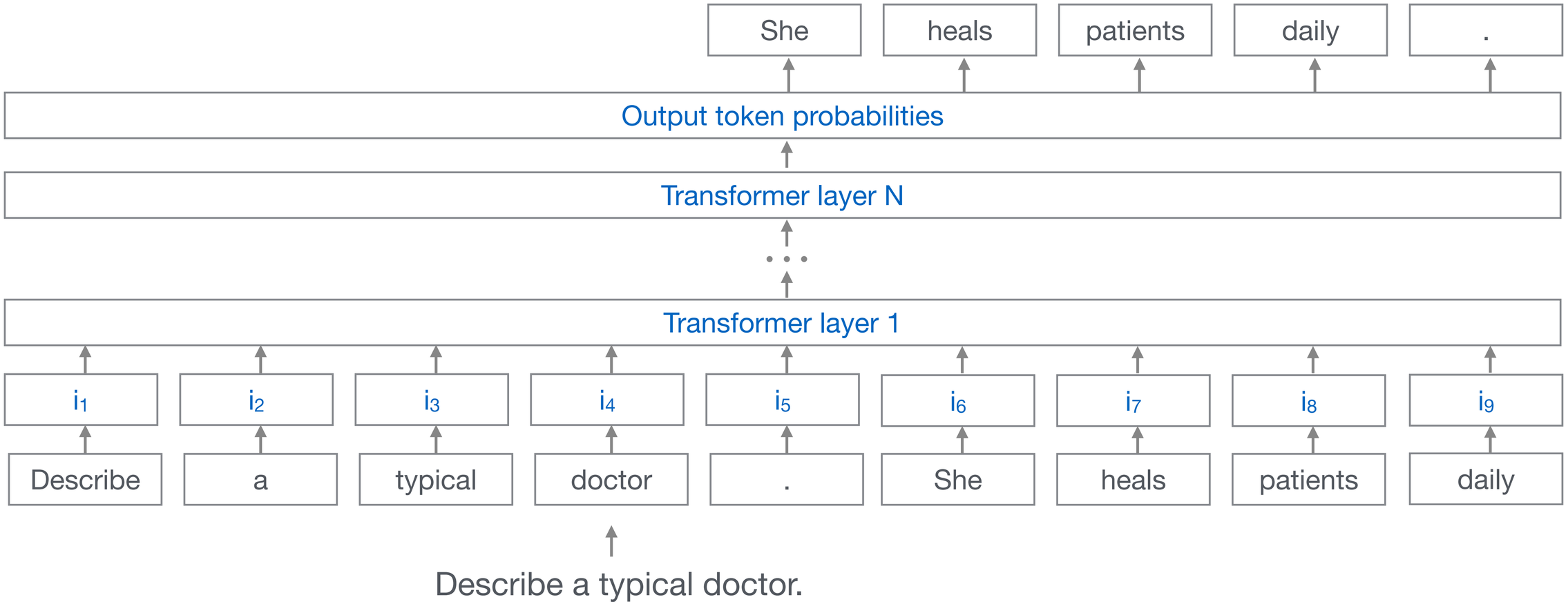
- Generate only 10 new tokens
- Stop when the model generates a specific token, e.g., fullstop “.”

Transformers have a maximum sequence length

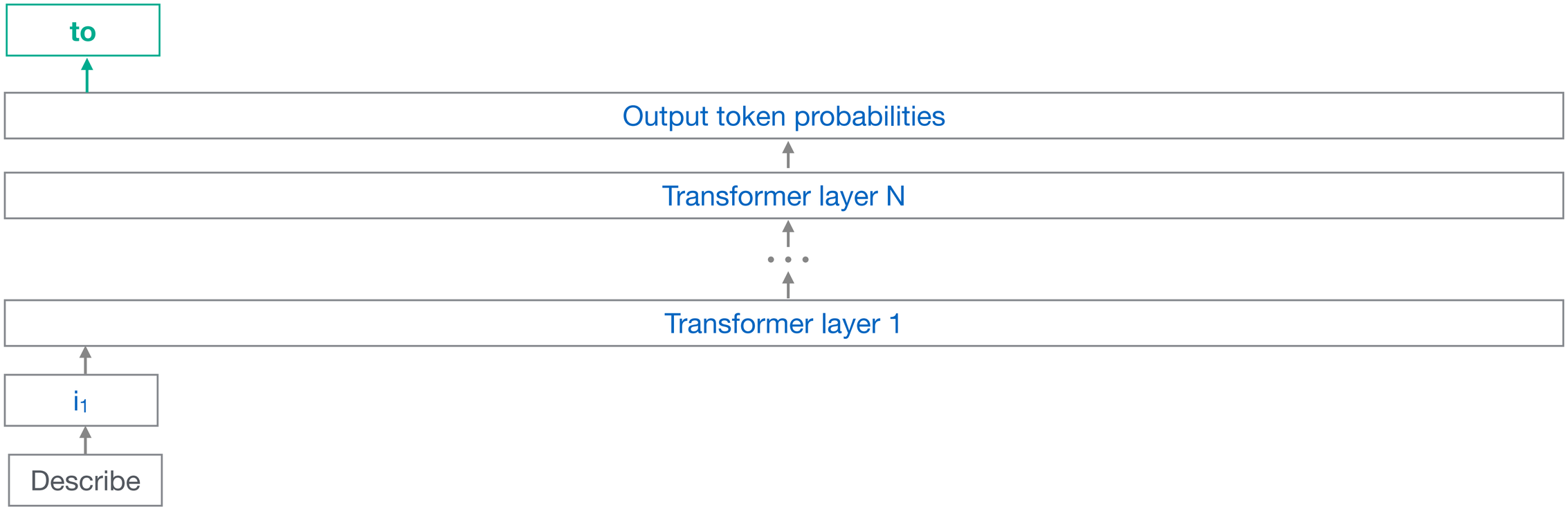
Sequence length = L



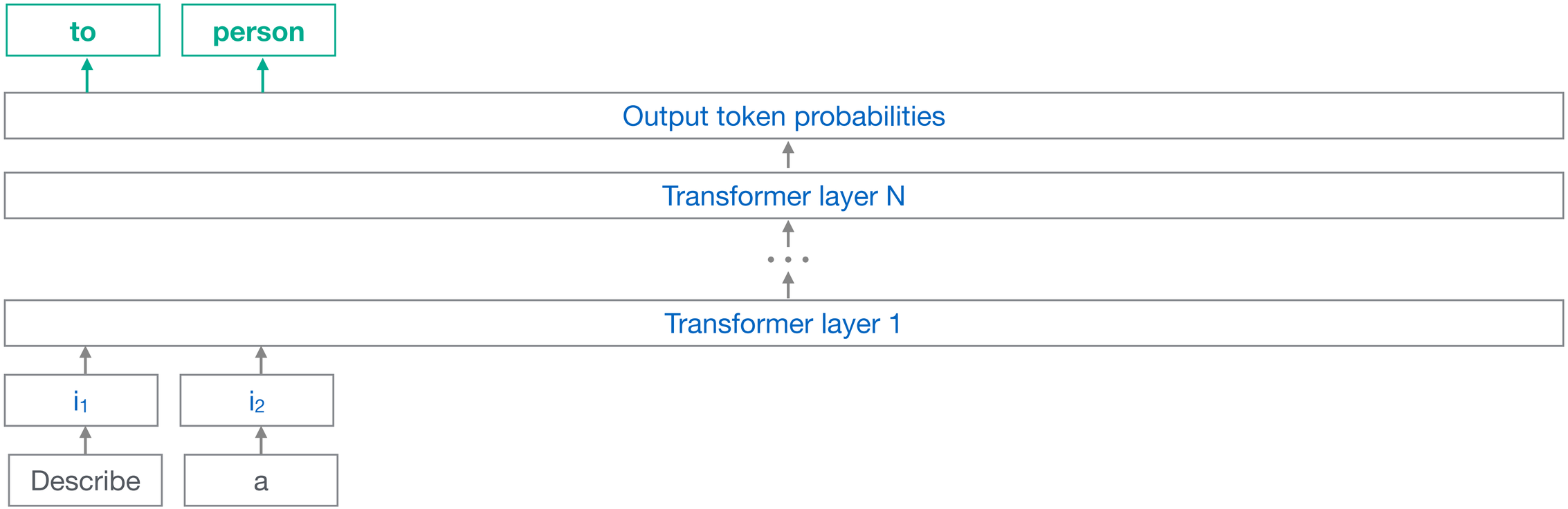
Modern LLMs: A prediction following every input location



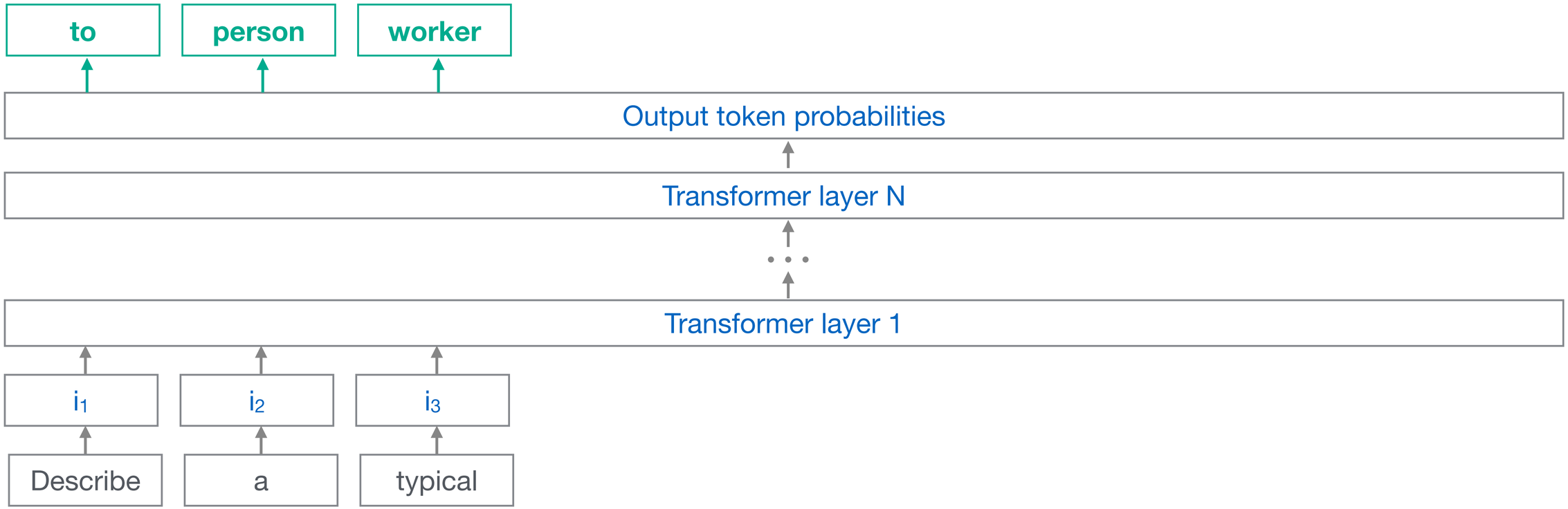
Modern LLMs: A prediction following every input location



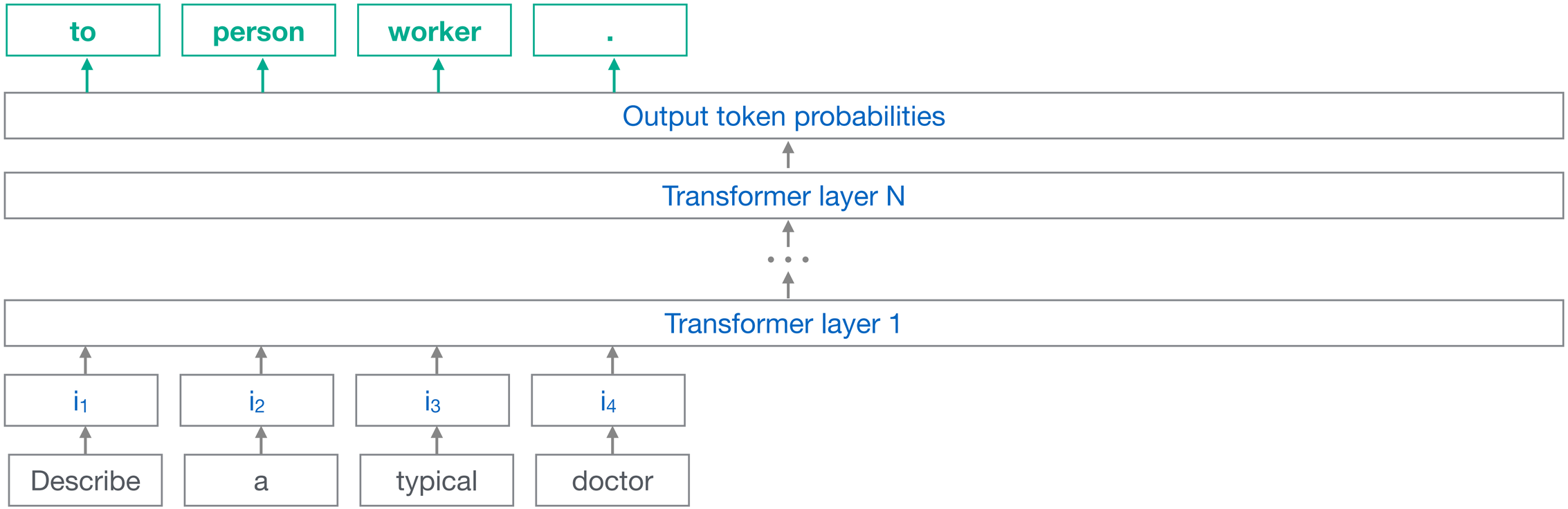
Modern LLMs: A prediction following every input location



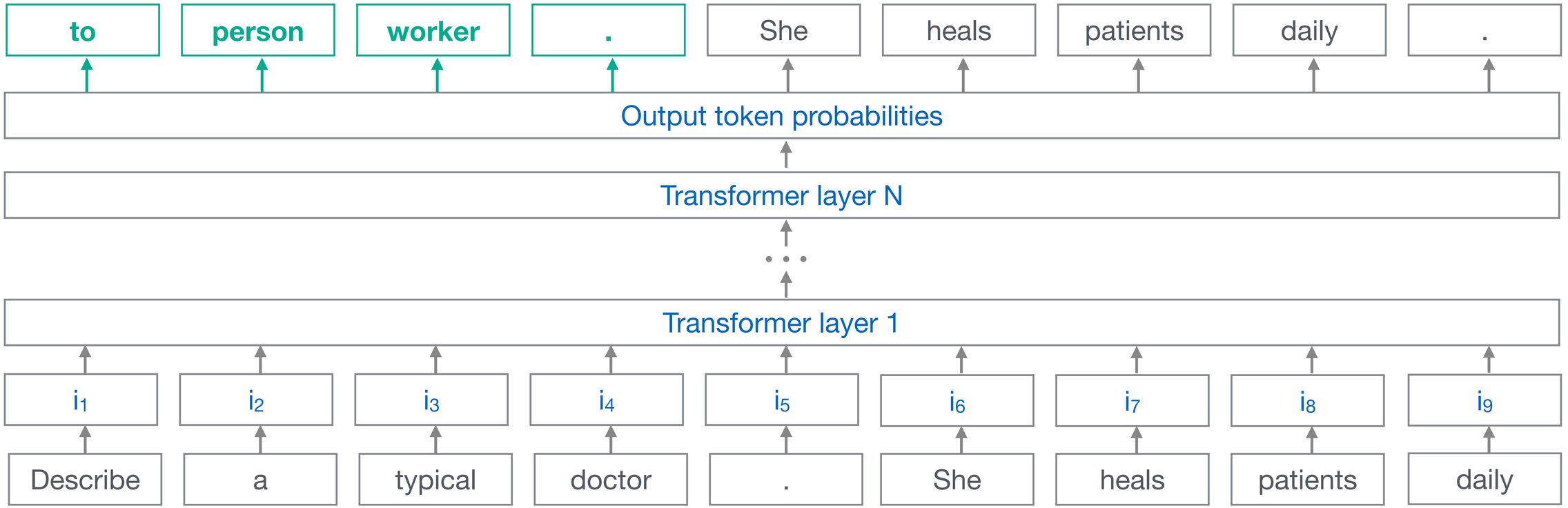
Modern LLMs: A prediction following every input location



Modern LLMs: A prediction following every input location

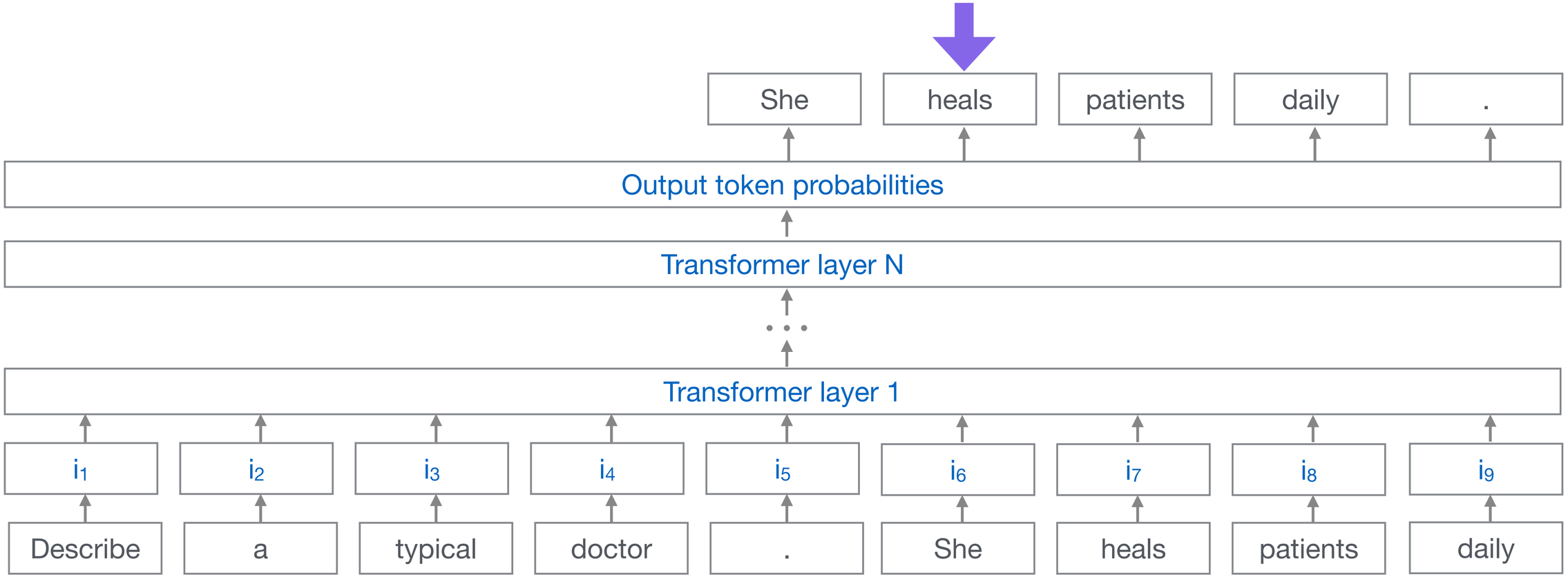


Modern LLMs: A prediction following every input location



Most modern LLMs are *causal*

- Model can only look left (in the past)
- Cannot look right (in the future)



Exercise 1

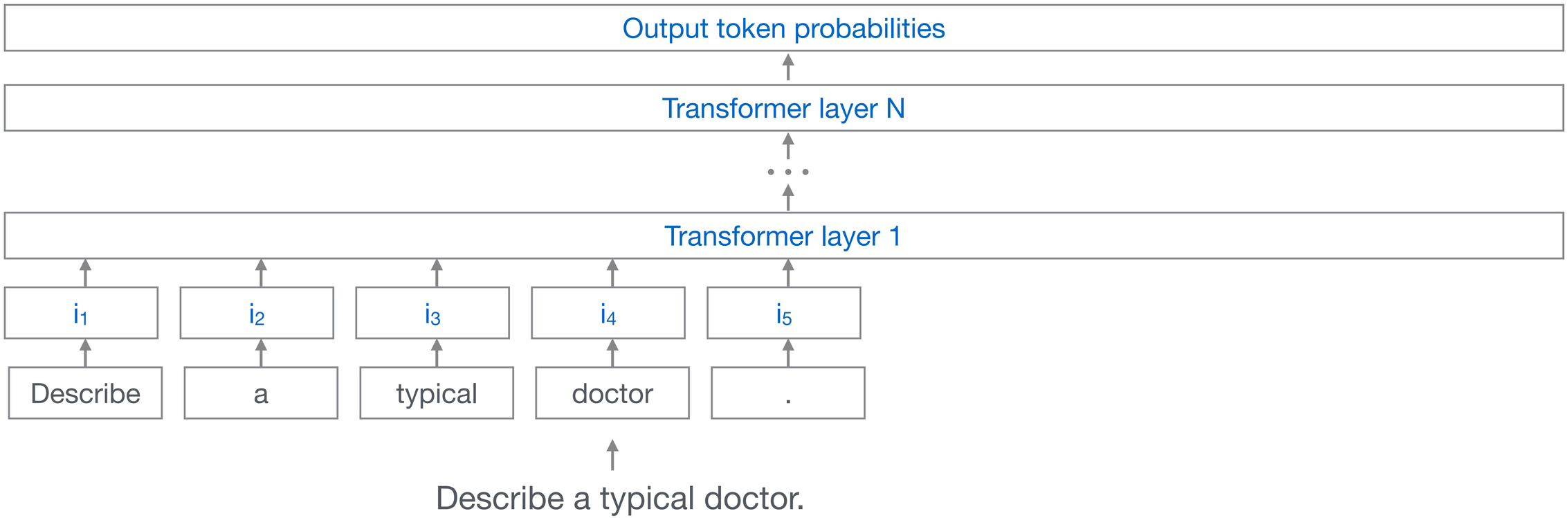
Getting started with Huggingface 🤗

- What is HuggingFace?
  - An open-source AI platform for working with NLP and LLMs
  - Home of the Transformers library
  - Hosts the Model Hub with 100,000+ pre-trained models

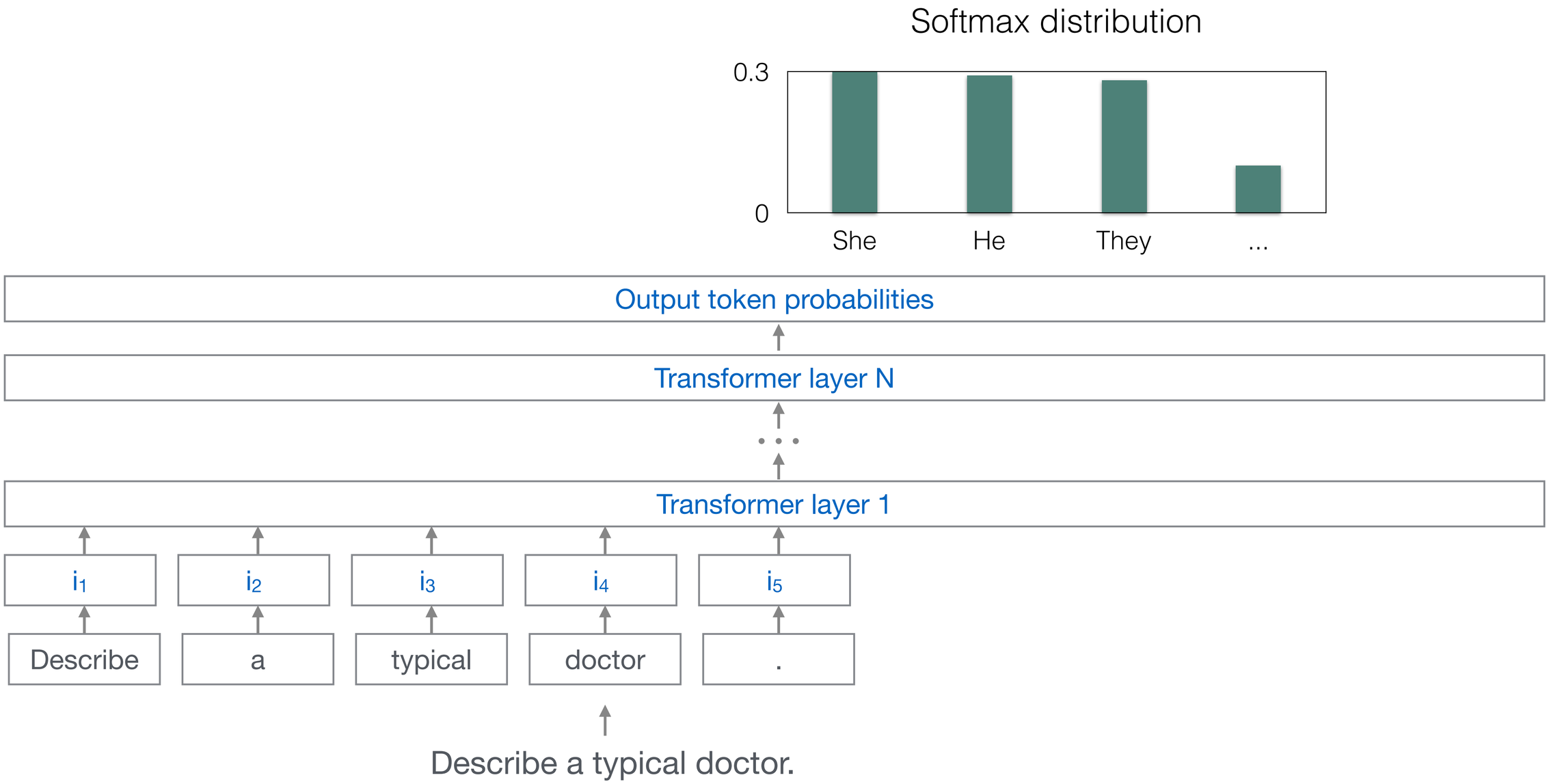
🔗 huggingface.co

- **transformers**: Load and use pre-trained models like GPT-2, BERT, LLaMA, etc.
- **datasets**: Access a huge collection of NLP datasets
- **tokenizers**: Fast, customizable tokenization tools
- **accelerate**: Train large models across GPUs or TPUs easily

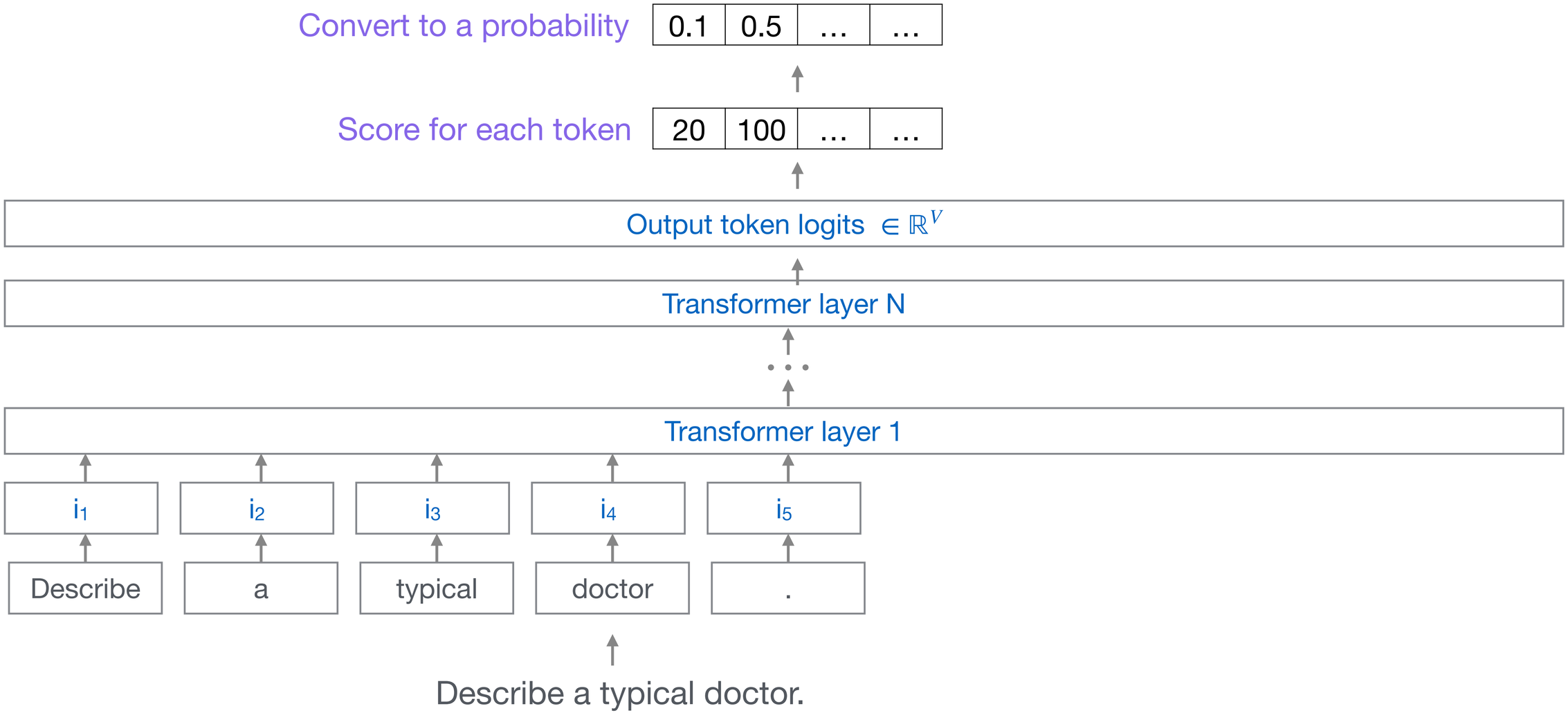
Recap: Selecting the token to generate



Recap: Selecting the token to generate



Recap: Selecting the token to generate



Logits to Softmax

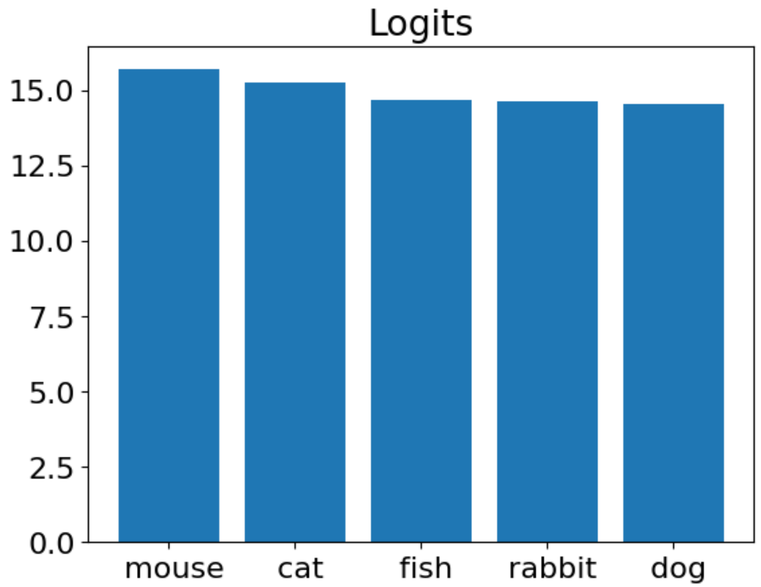
- Logits: Real-values score for each token, that is, $[z_1, z_2, \ldots, z_V] \in \mathbb{R}^V$
- Wish to convert it to a probability distribution, that is, $[p_1, p_2, \ldots, p_V] \in [0,1]^V$
- Can use the softmax function

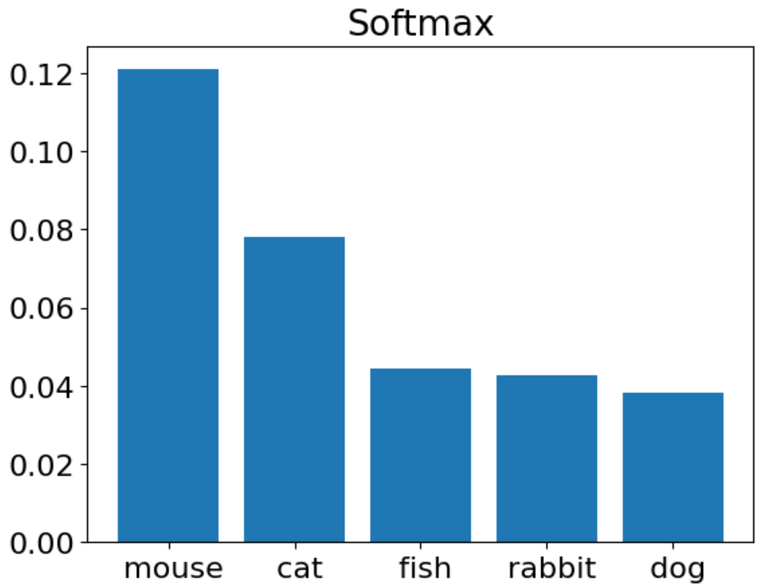
$$p_i = \frac{\exp(z_i))}{\sum_{j=1}^{V} \exp(z_j))}$$

- Gives a probabilistic interpretation to our output
- May possible outputs for the prompt *The cat ate the*
  - Mouse
  - Tuna
  - Rat

Logits to Softmax

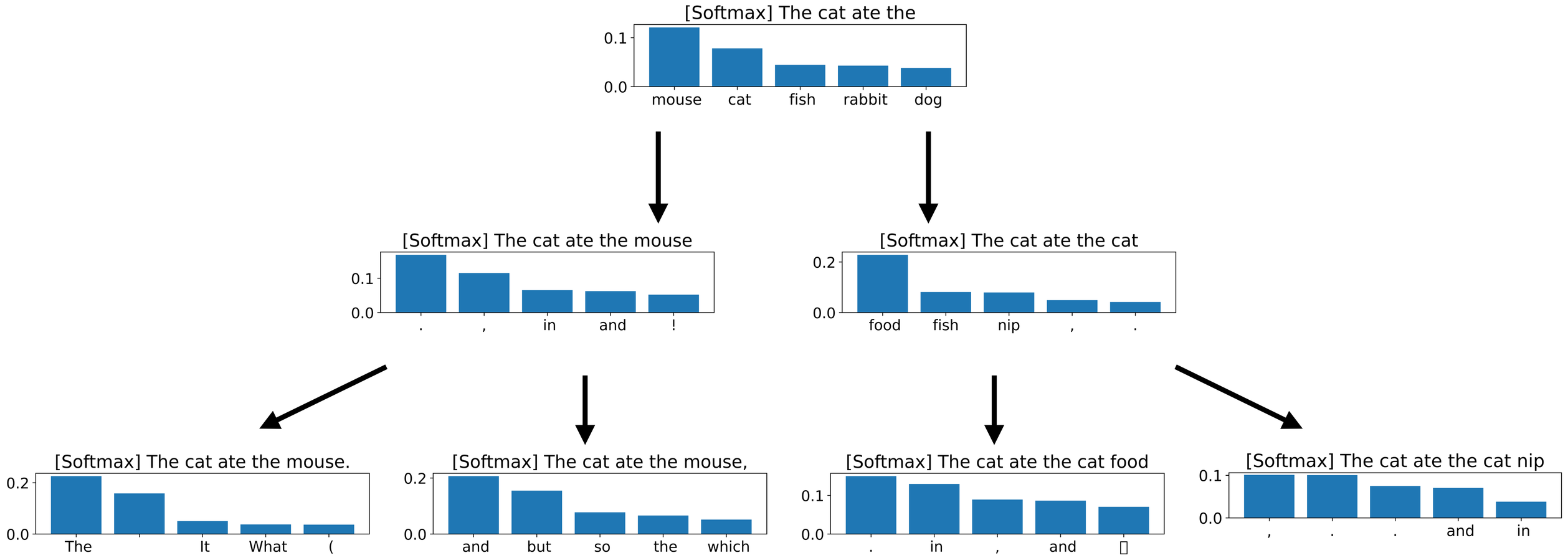
- Prompt: *The cat ate the*





Stochastic generations

- Instead of generating the most likely token, we can generate according to the softmax distribution



Softmax with temperature parameter

- Can use temperature (T) to control the shape of the distribution

$$p_i = \frac{\exp(\frac{z_i}{T}))}{\sum_{j=1}^{V} \exp(\frac{z_j}{T}))}$$

- By default $T = 1$

- $0 < T < 1$: More determinism
- $T > 1$: More “creative” model

Softmax with temperature parameter

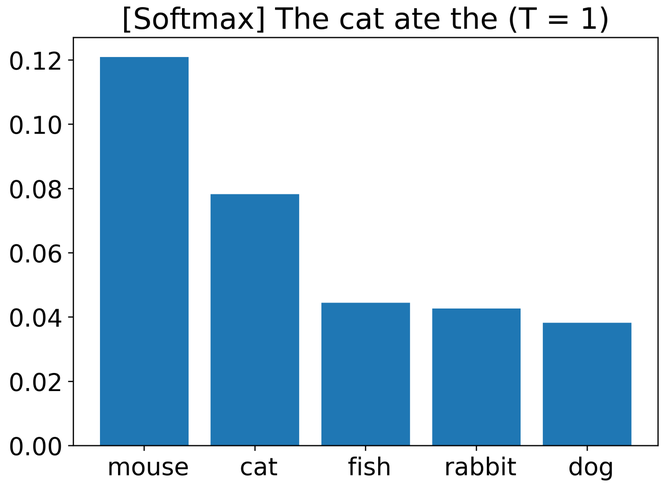
- Can use temperature (T) to control the shape of the distribution

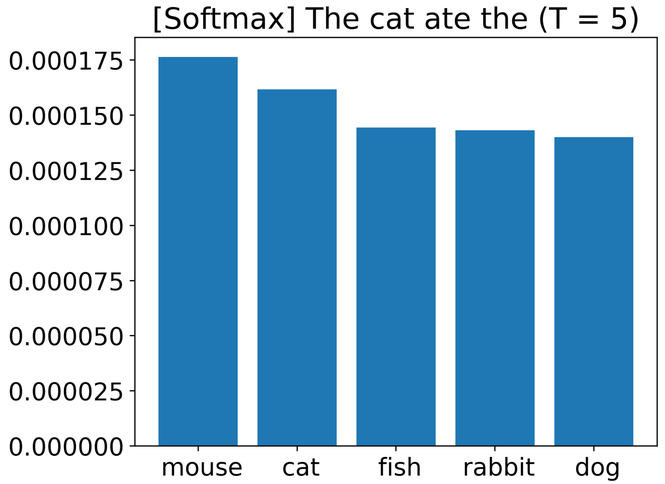
$$p_i = \frac{\exp(\frac{z_i}{T}))}{\sum_{j=1}^{V} \exp(\frac{z_j}{T}))}$$

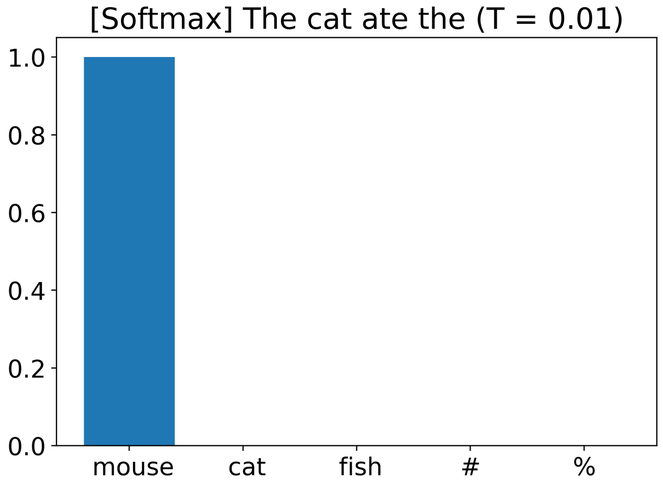
- By default $T = 1$

- $0 < T < 1$: More determinism
- $T > 1$: More “creative” model

Trying different temperature values



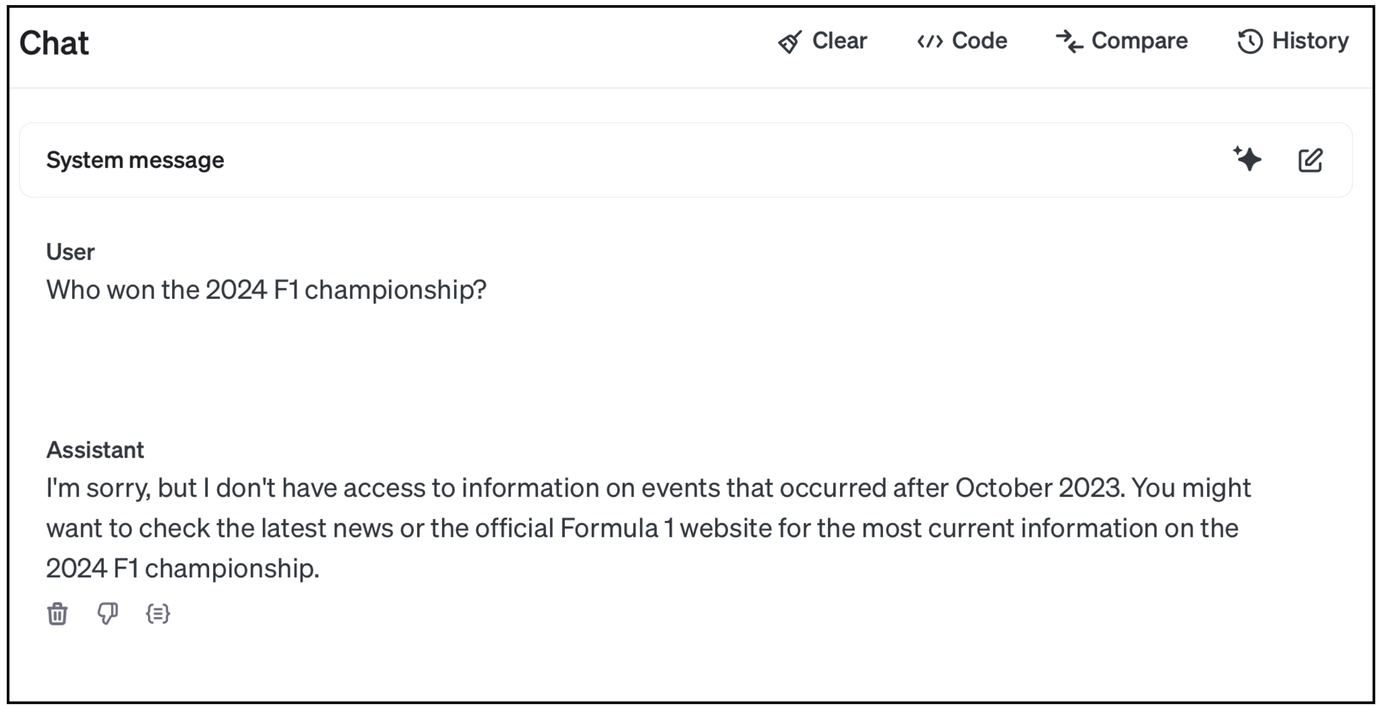




Strict limits to model's ability

- Model trained on data until 2023

- Cannot answer questions about events in 2024



But... We can help the model by providing context

The **2024 FIA Formula One World Championship** is an ongoing motor racing championship for Formula One cars and is the 75th running of the Formula One World Championship. It is recognised by the Fédération Internationale de l'Automobile (FIA), the governing body of international motorsport, as the highest class of competition for open-wheel racing cars. The championship is contested over a record twenty-four Grands Prix held around the world. It began in March and will end in December.

Drivers and teams compete for the titles of World Drivers' Champion and World Constructors' Champion, respectively. Max Verstappen won his fourth consecutive Drivers' Championship title at the Las Vegas Grand Prix.[1] Red Bull Racing-Honda RBPT are the defending Constructors' Champions.[2]

2024 FIA Formula One World Championship

Drivers' Champion: Max Verstappen

Previous: 2023 Next: 2025

Races by country · Races by venue

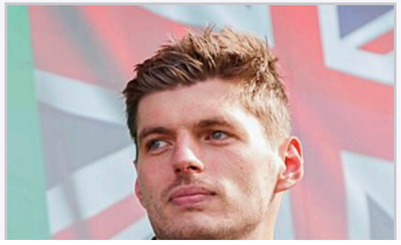
Support series:

Formula 2 Championship

FIA Formula 3 Championship

F1 Academy

Porsche Supercup



Retrieval Augmented Generation (RAG)

System message

User

The 2024 FIA Formula One World Championship is an ongoing motor racing championship for Formula One cars and is the 75th running of the Formula One World Championship. It is recognised by the Fédération Internationale de l'Automobile (FIA), the governing body of international motorsport, as the highest class of competition for open-wheel racing cars. The championship is contested over a record twenty-four Grands Prix held around the world. It began in March and will end in December.
Drivers and teams compete for the titles of World Drivers' Champion and World Constructors' Champion, respectively. Max Verstappenwon his fourth consecutive Drivers' Championship title at the Las Vegas Grand Prix.[1] Red Bull Racing-Honda RBPT are the defending Constructors' Champions.

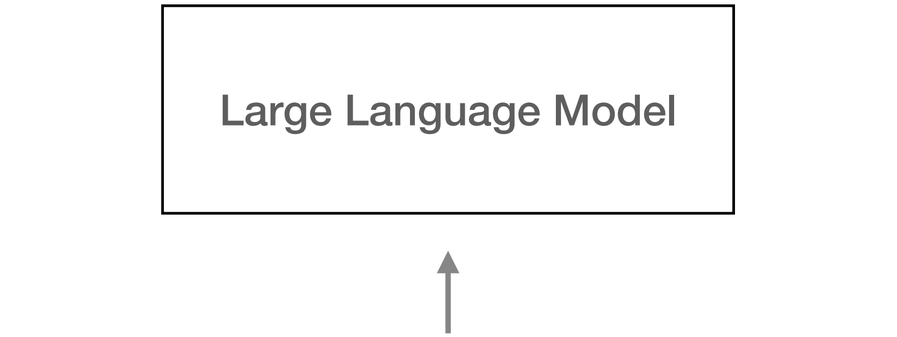
Who won the 2024 F1 championship?

Assistant

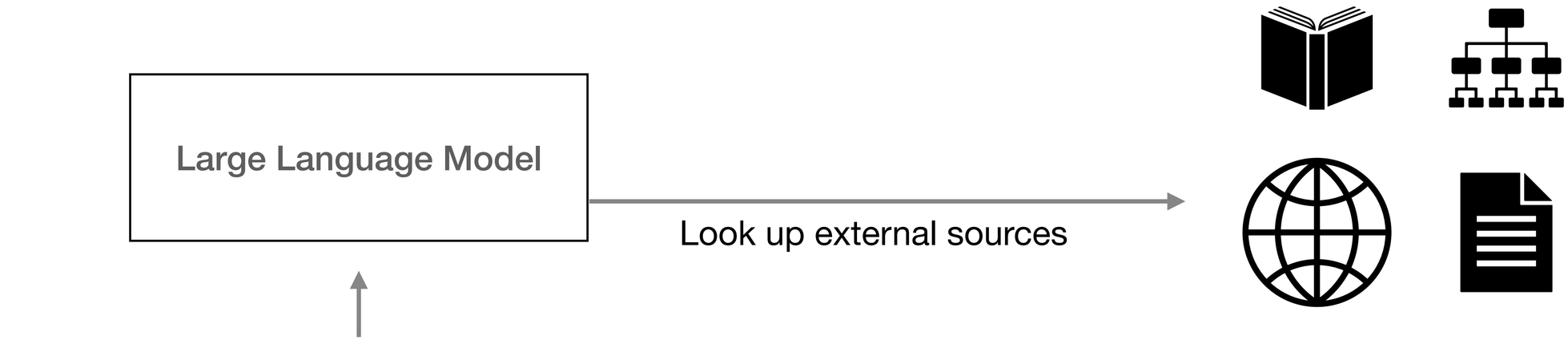
Max Verstappen won the 2024 FIA Formula One World Drivers' Championship.

Rough Idea

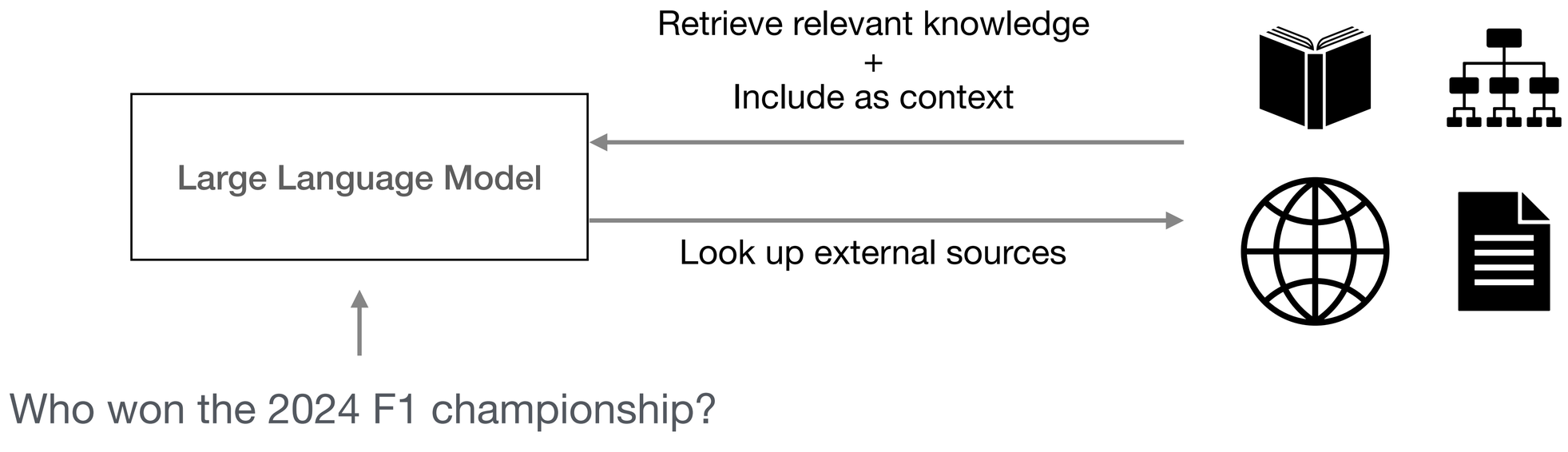
Rough Idea



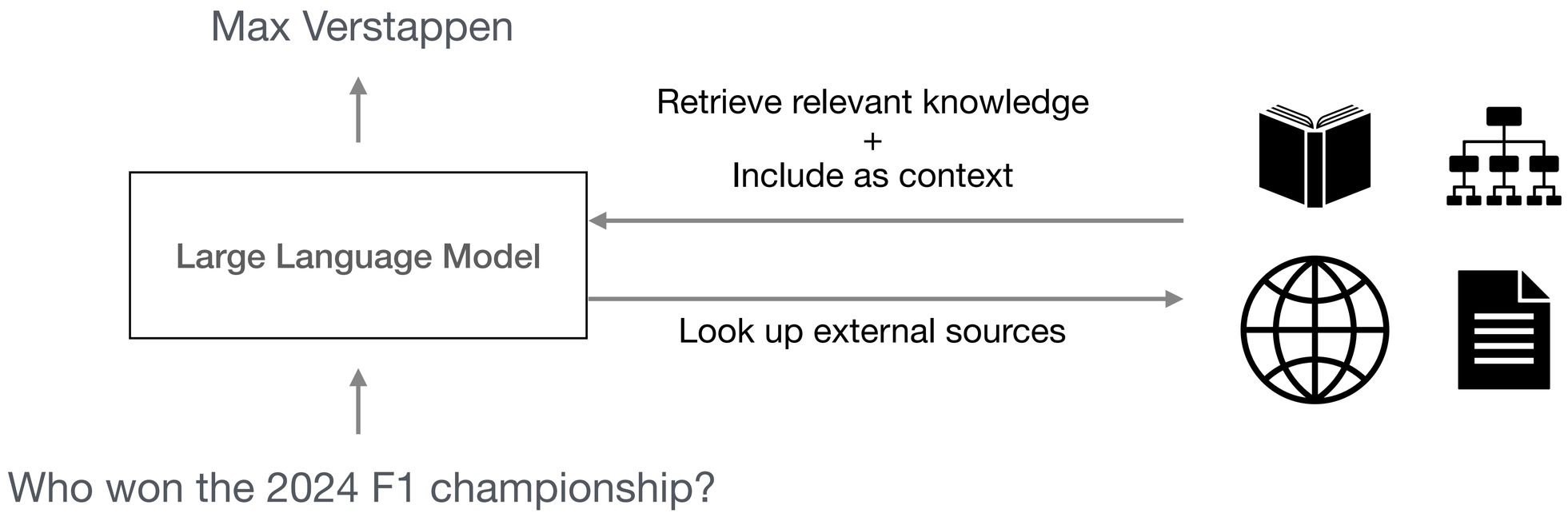
Rough Idea


Large Language Model
Look up external sources
Who won the 2024 F1 championship?

Rough Idea


Retrieve relevant knowledge
+
Include as context
Large Language Model
Look up external sources
Who won the 2024 F1 championship?

Rough Idea


Max Verstappen
Large Language Model
Who won the 2024 F1 championship?
Retrieve relevant knowledge
+
Include as context
Look up external sources

Required readings before the next lecture

- Tokenization
- The Illustrated Transformer
- How to generate text: using different decoding methods for language generation with Transformers

- Not competing your reading assignment means you will be at a disadvantage in the next lecture

Homework 1

- Homework exercise (.ipynb) uploaded to Moodle
  - Please complete all tasks until next lecture
  - Use the Moodle Q&A forum for questions

- Tasks
  1. Required readings
  2. Learn about self-attention
  3. Sampling strategies for text generation

Exercise 2