

"Large Language Models" 1

Deep Learning for Computational Linguistics

C.M. Downey

Fall 2025

GPT-2 prompting

- Prompt: **I didn't ride my bike to campus today because of the snowy weather.**
- There is a lot of construction and I wanted to have a few hours of skiing. All that snow is a real pain in the ass. Good thing there was an LRT planned for that area. See you guys on campus in a few days! Photo credit: Tapto@archivosos.gr
– August 26, 2016: Windy Weather on the Mountain (photo credit: Diana Prat)
From Diana Prat, via tapto@archivosos.gr:
Ever since the camp that I organized to protest the modernisation of the Barranco, my beautiful alma mater, you know, going in for a press conference, I was asking why the Italian government

vs. ChatGPT



I didn't ride my bike to campus today because of the snowy weather.



It's understandable that you didn't ride your bike to campus today due to the snowy weather. Riding a bike in such conditions can be dangerous and difficult. It's important to prioritize your safety and well-being when making decisions like this. Is there anything else you would like to know or talk about?



Another example (GPT-2)

- Prompt: **What inspired Noam Chomsky to invent the light bulb?**
- He was, of course, inspired to do this by his lifelong fascination with electricity, especially its power and simplicity. He used to say that in the early days he had not really known how to work a light bulb

Another Example (ChatGPT)



You

What inspired Noam Chomsky to invent the light bulb?



ChatGPT

It seems like there might be a mix-up in your question! Noam Chomsky is a renowned linguist, philosopher, and cognitive scientist known for his work in linguistics, cognitive psychology, and political commentary. He didn't invent the light bulb; that credit goes to Thomas Edison, who patented the first practical incandescent light bulb in 1879.

Chomsky is famous for his theories on language acquisition and the structure of language, such as his groundbreaking work in generative grammar and the concept of a universal grammar. These ideas have had a significant impact on fields like linguistics, psychology, and artificial intelligence. If you're interested in Chomsky's contributions, I'd be happy to share more about his linguistic theories or any other aspect of his work!

Behavior change

Behavior change

- “Pure” language models will try to **complete a prompt**
 - i.e. act as if it were any other text, and try to **predict the next token(s)**

Behavior change

- “Pure” language models will try to **complete a prompt**
 - i.e. act as if it were any other text, and try to **predict the next token(s)**
- “Chatbot” LMs instead **act as a dialogue partner** (“interlocutor”)
 - How? We’ll cover the key techniques in this lecture
 - Much greater emphasis on **user-friendliness** (can be used easily by people without technical NLP knowledge)
 - Also leads to much greater **anthropomorphization of LMs**, which could be problematic

Tasks as Conditional Generation

Traditional Approach: Task Heads

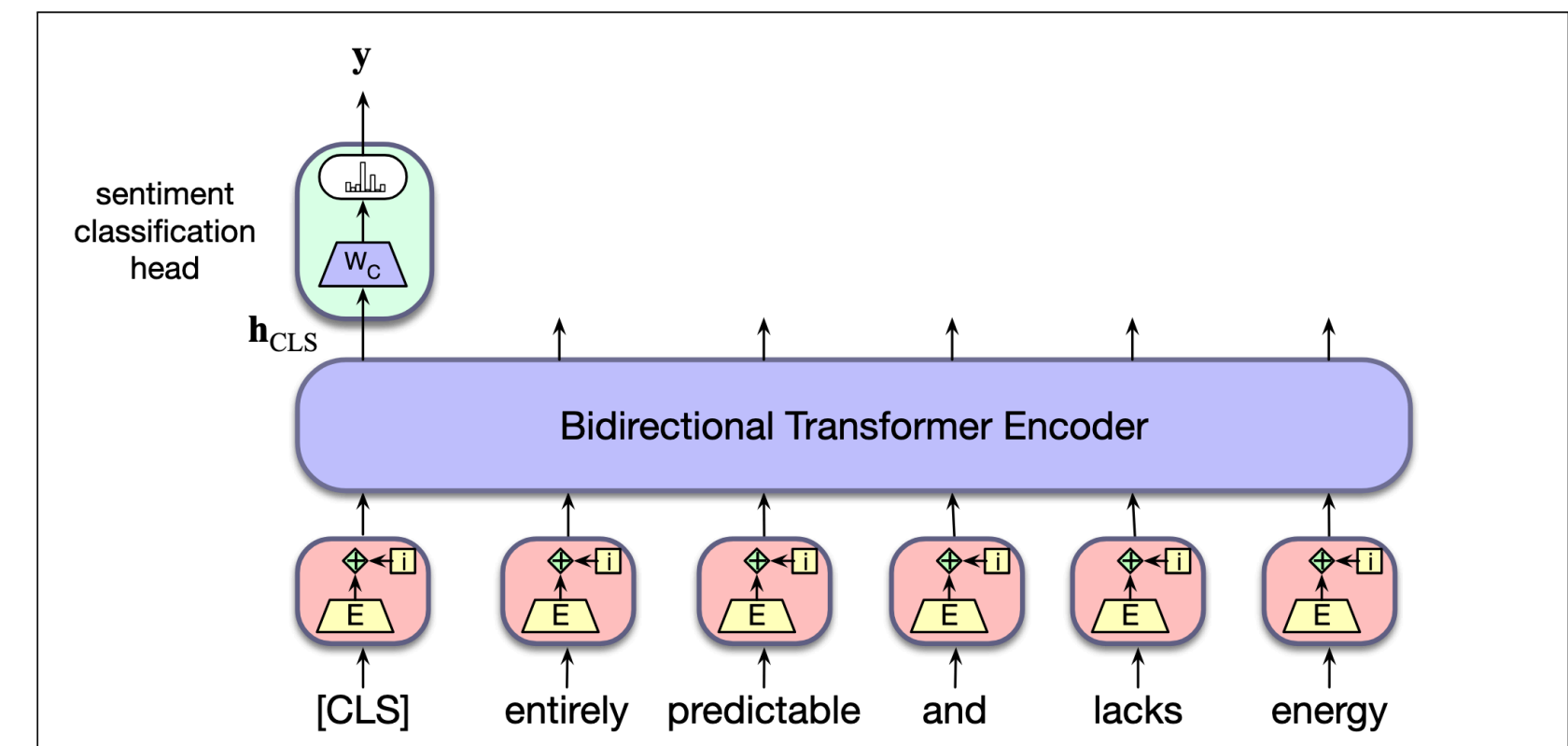


Figure 10.9 Sequence classification with a bidirectional transformer encoder. The output vector for the [CLS] token serves as input to a simple classifier.

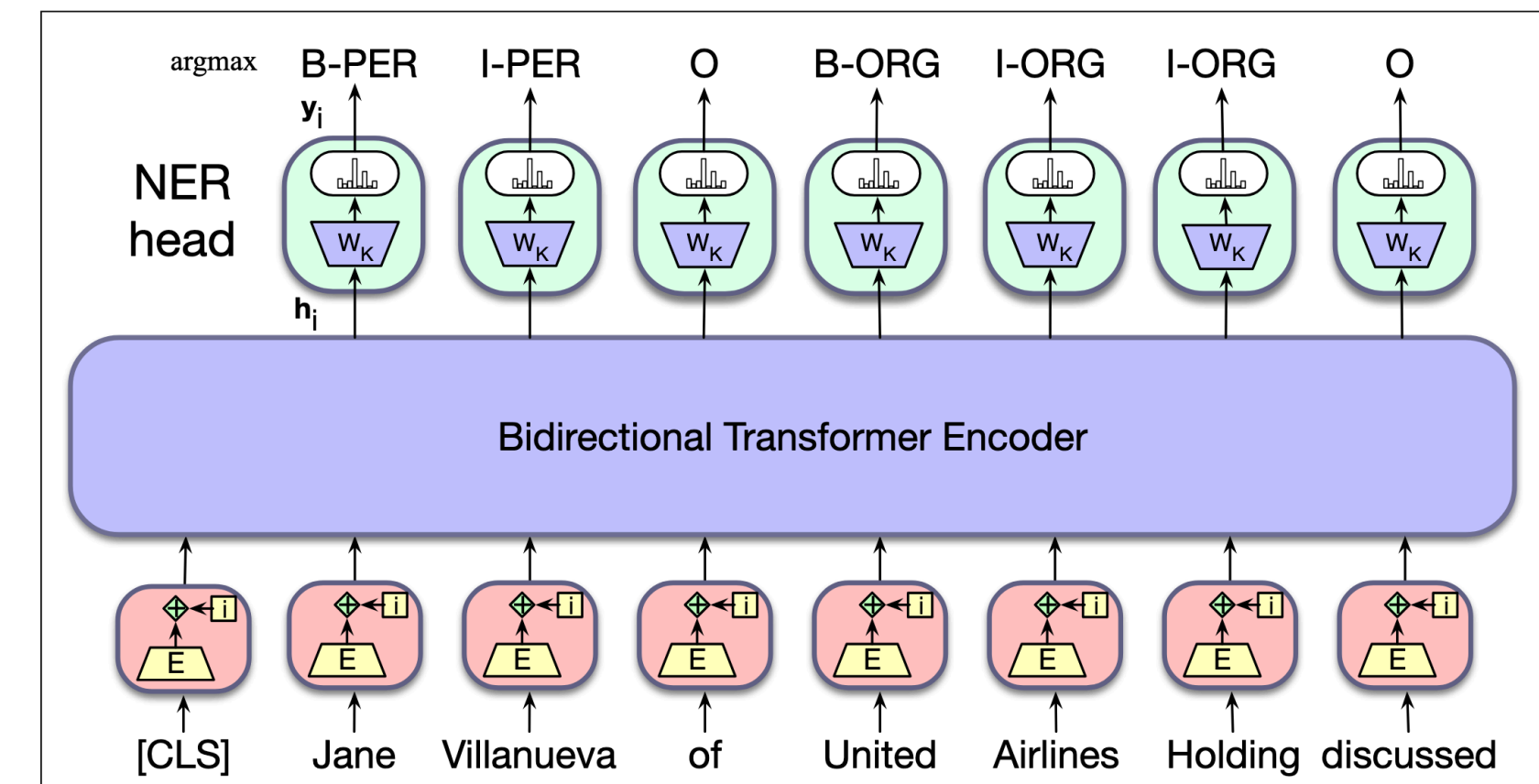


Figure 10.13 Sequence labeling for named entity recognition with a bidirectional transformer encoder. The output vector for each input token is passed to a simple k -way classifier.

Traditional Approach: Task Heads

- In the "BERT Era", classification tasks were done with **small task-specific layers** called "heads"

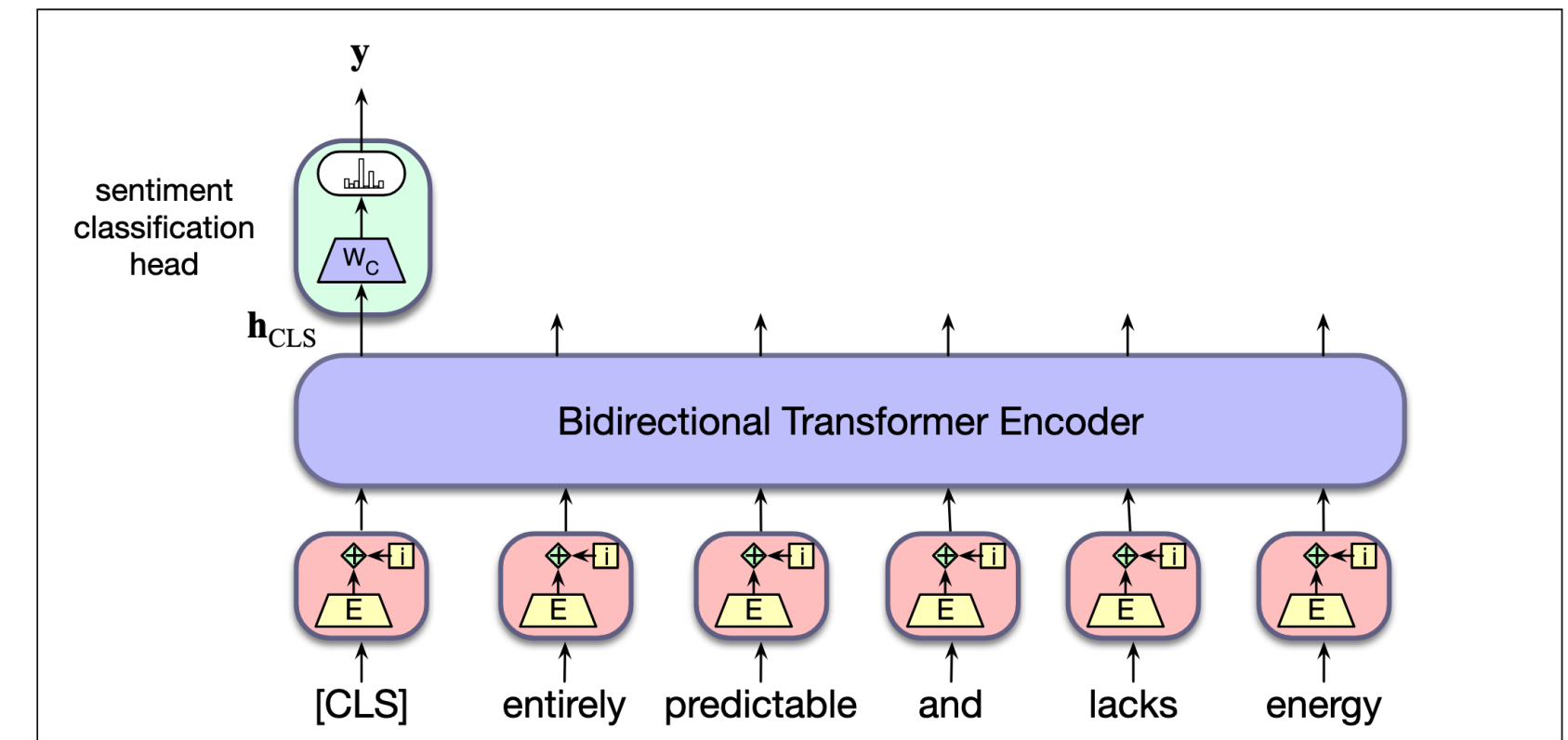


Figure 10.9 Sequence classification with a bidirectional transformer encoder. The output vector for the [CLS] token serves as input to a simple classifier.

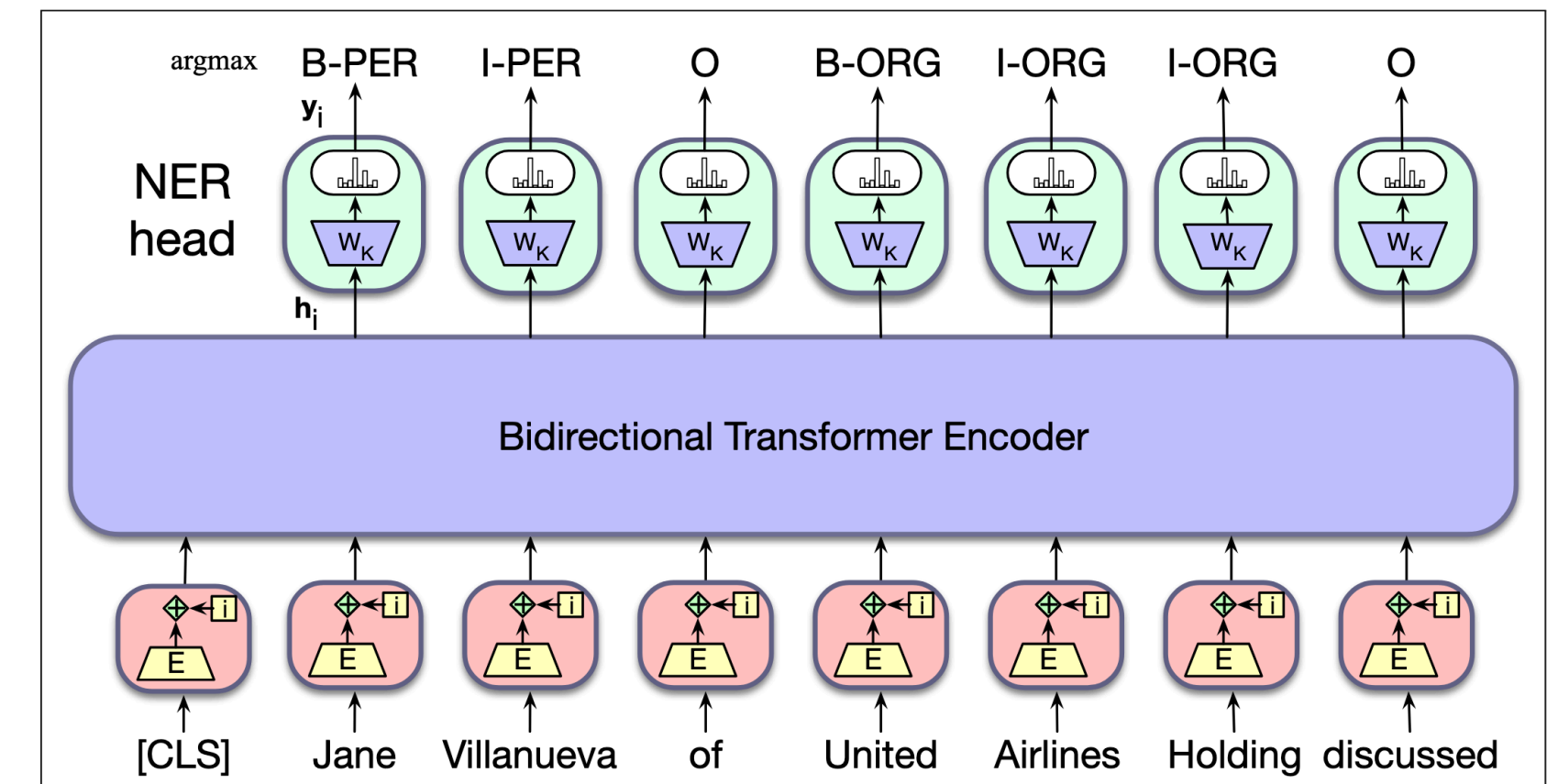


Figure 10.13 Sequence labeling for named entity recognition with a bidirectional transformer encoder. The output vector for each input token is passed to a simple k-way classifier.

Traditional Approach: Task Heads

- In the "BERT Era", classification tasks were done with **small task-specific layers** called "heads"
- E.g. linear layer + softmax for **five-way sentiment classification**

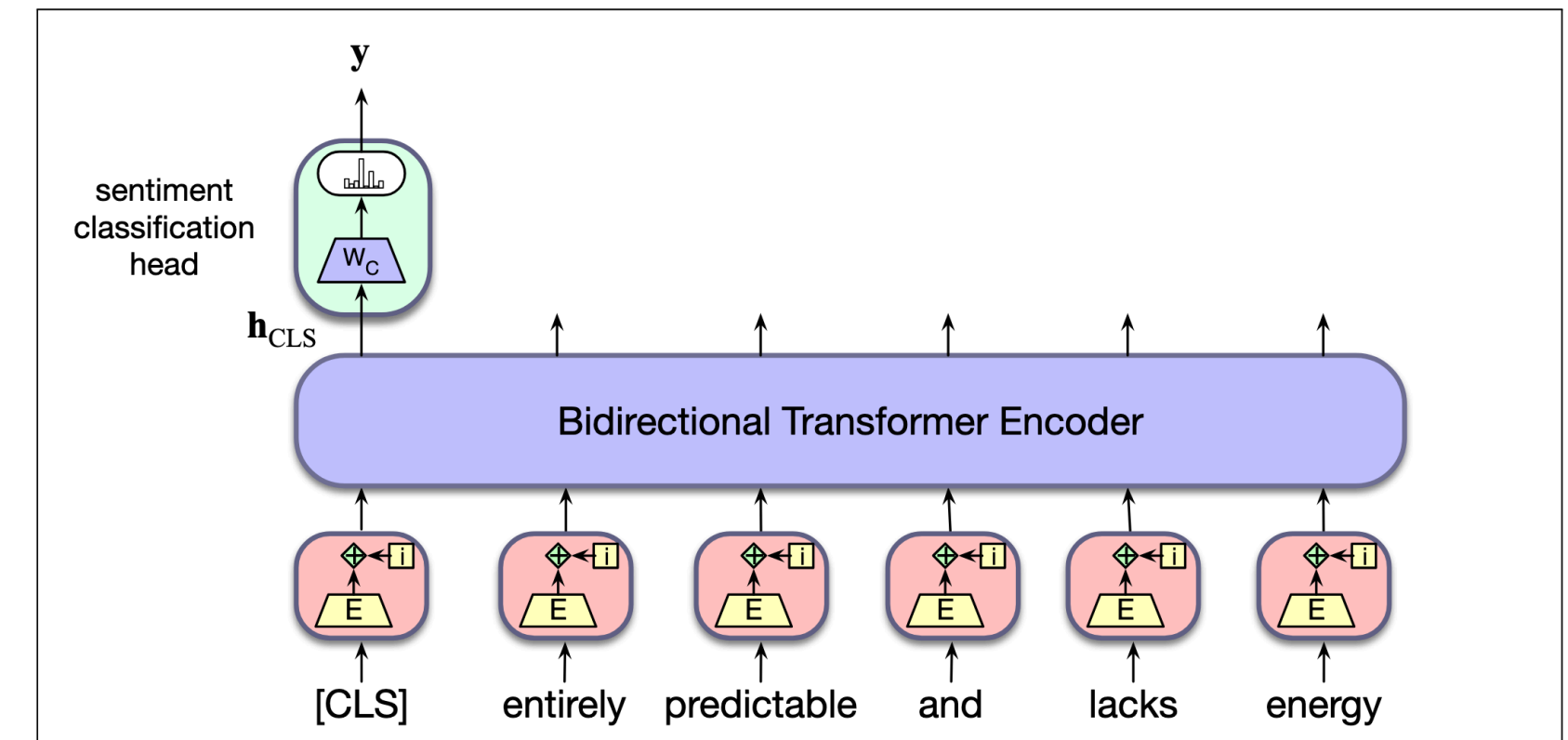


Figure 10.9 Sequence classification with a bidirectional transformer encoder. The output vector for the [CLS] token serves as input to a simple classifier.

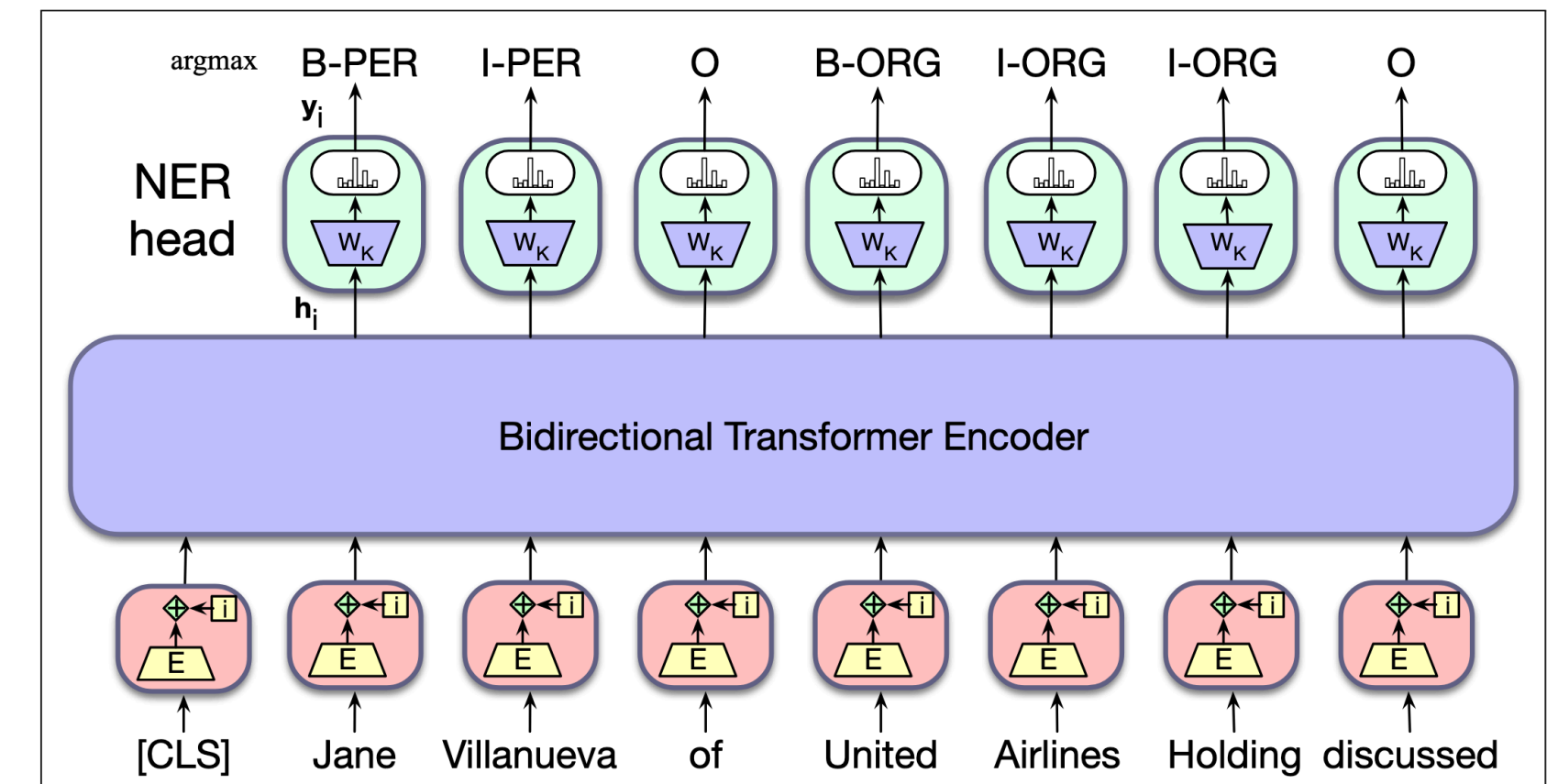


Figure 10.13 Sequence labeling for named entity recognition with a bidirectional transformer encoder. The output vector for each input token is passed to a simple k-way classifier.

Traditional Approach: Task Heads

- In the "BERT Era", classification tasks were done with **small task-specific layers** called "heads"
- E.g. linear layer + softmax for **five-way sentiment classification**
- Main model is sometimes **frozen** (only task head trains)

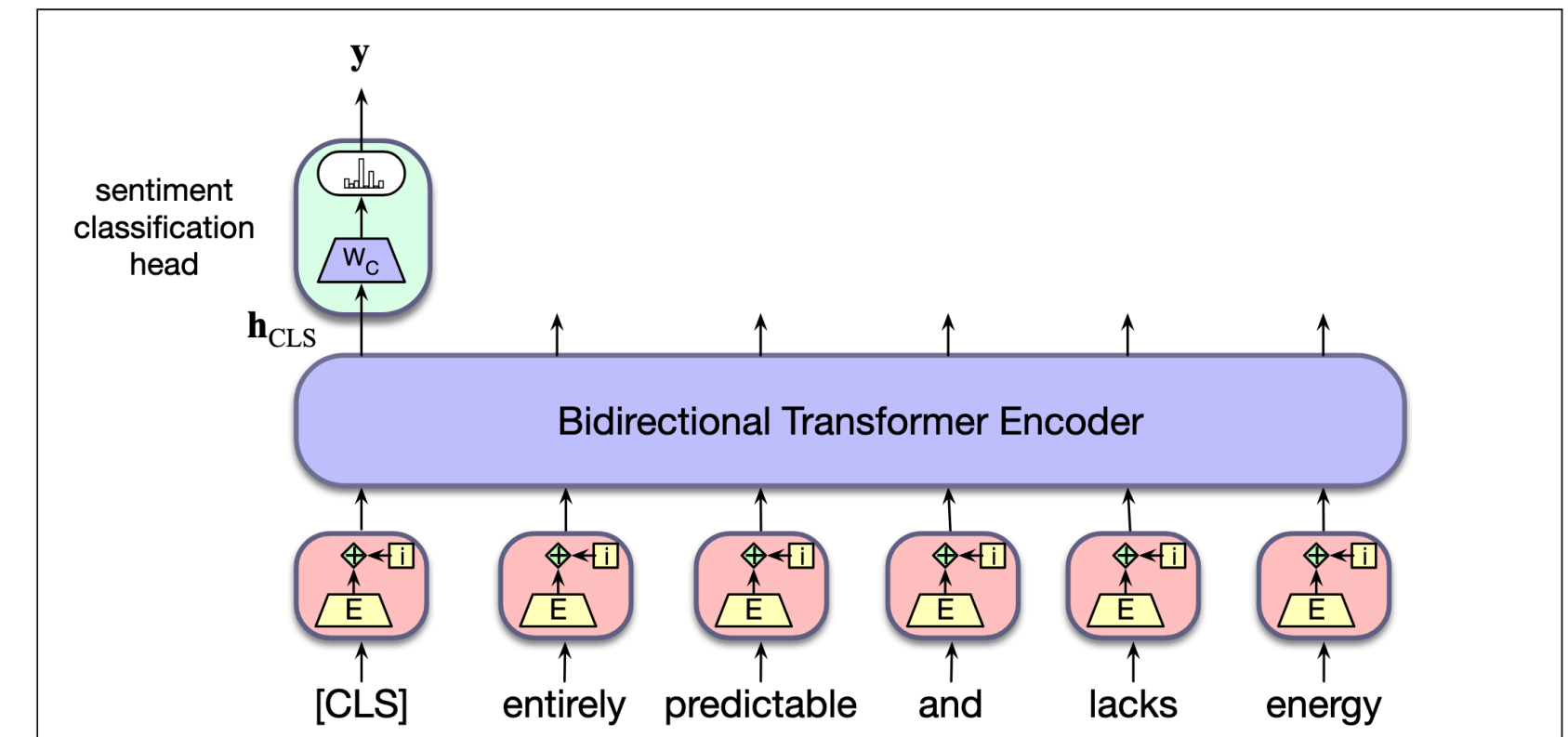


Figure 10.9 Sequence classification with a bidirectional transformer encoder. The output vector for the [CLS] token serves as input to a simple classifier.

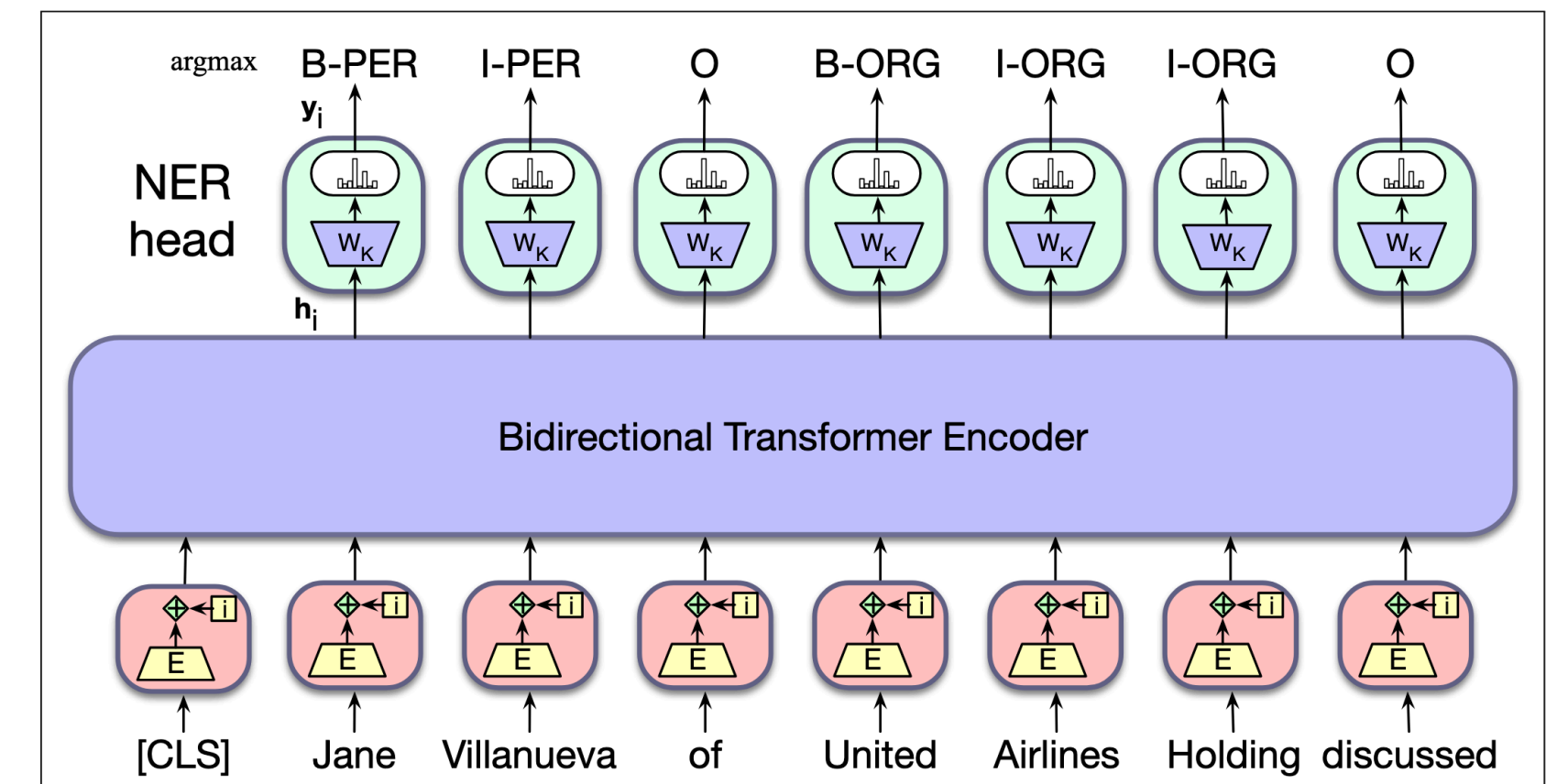


Figure 10.13 Sequence labeling for named entity recognition with a bidirectional transformer encoder. The output vector for each input token is passed to a simple k-way classifier.

Traditional Approach: Task Heads

- In the "BERT Era", classification tasks were done with **small task-specific layers** called "heads"
- E.g. linear layer + softmax for **five-way sentiment classification**
- Main model is sometimes **frozen** (only task head trains)
- Problem: each new task requires a **new task head**

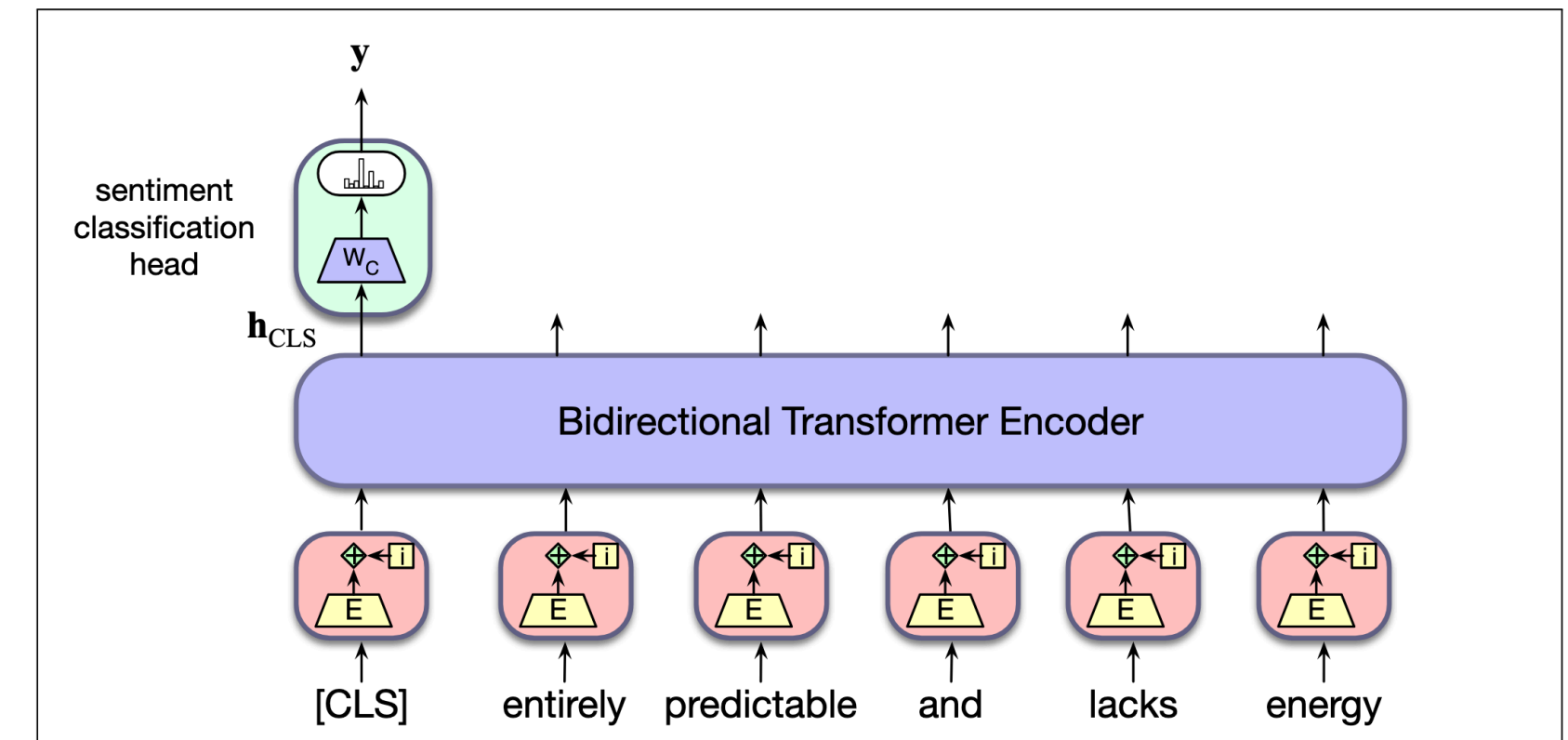


Figure 10.9 Sequence classification with a bidirectional transformer encoder. The output vector for the [CLS] token serves as input to a simple classifier.

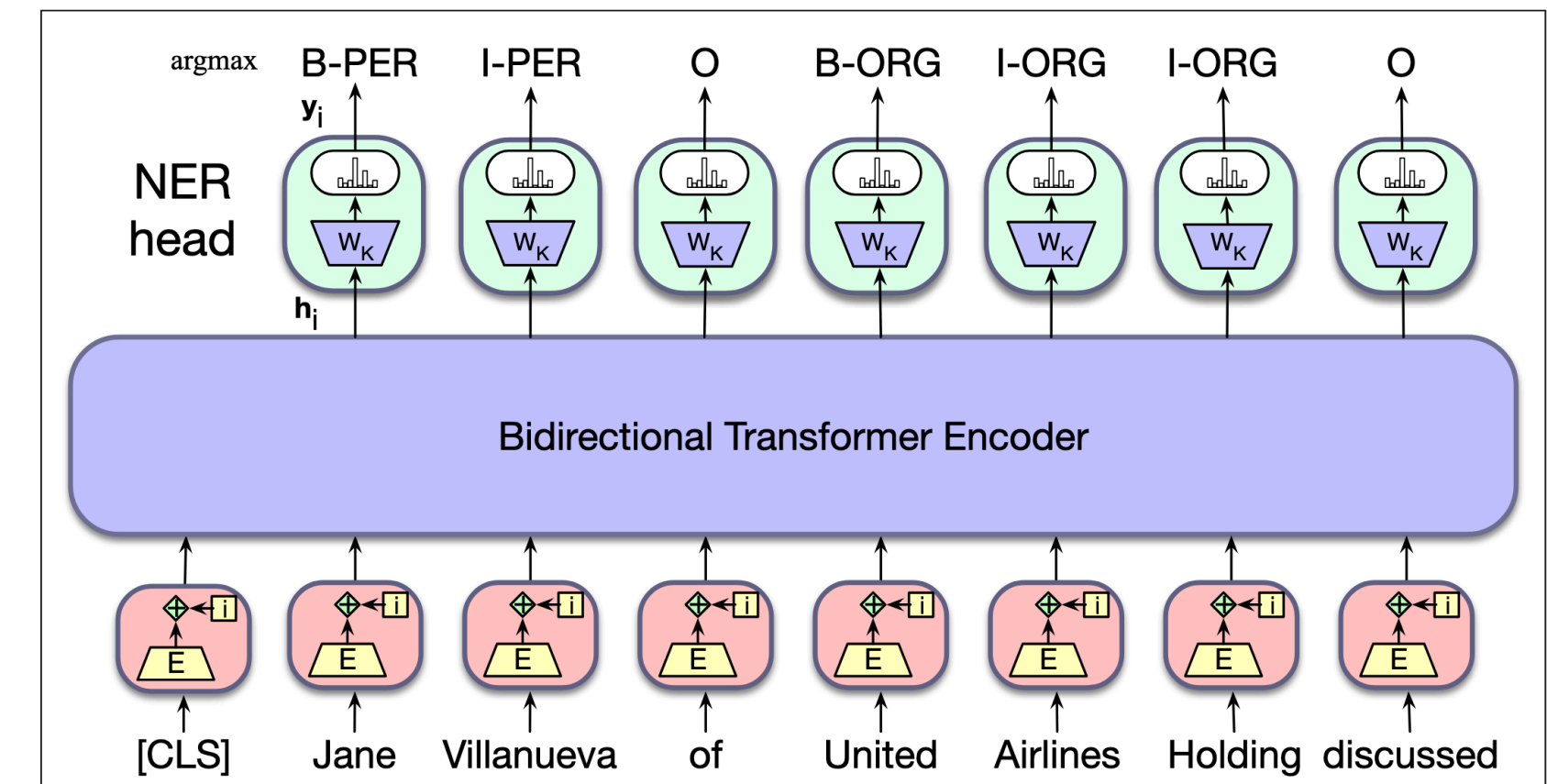
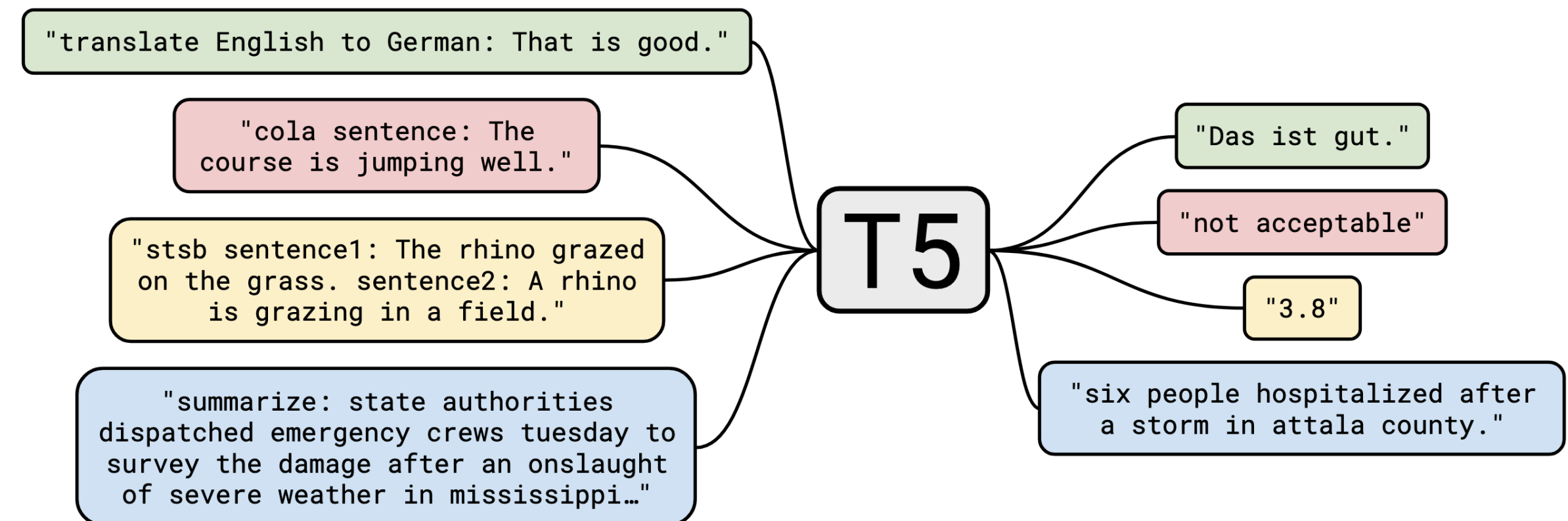


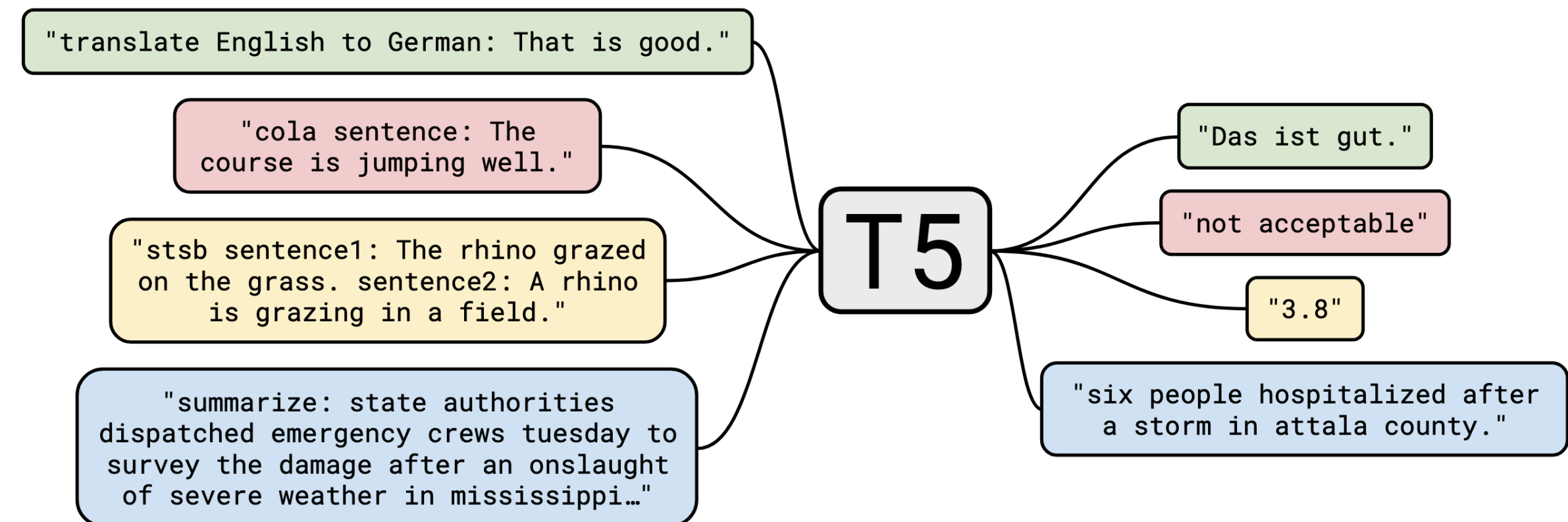
Figure 10.13 Sequence labeling for named entity recognition with a bidirectional transformer encoder. The output vector for each input token is passed to a simple k-way classifier.

Text-to-Text NLP



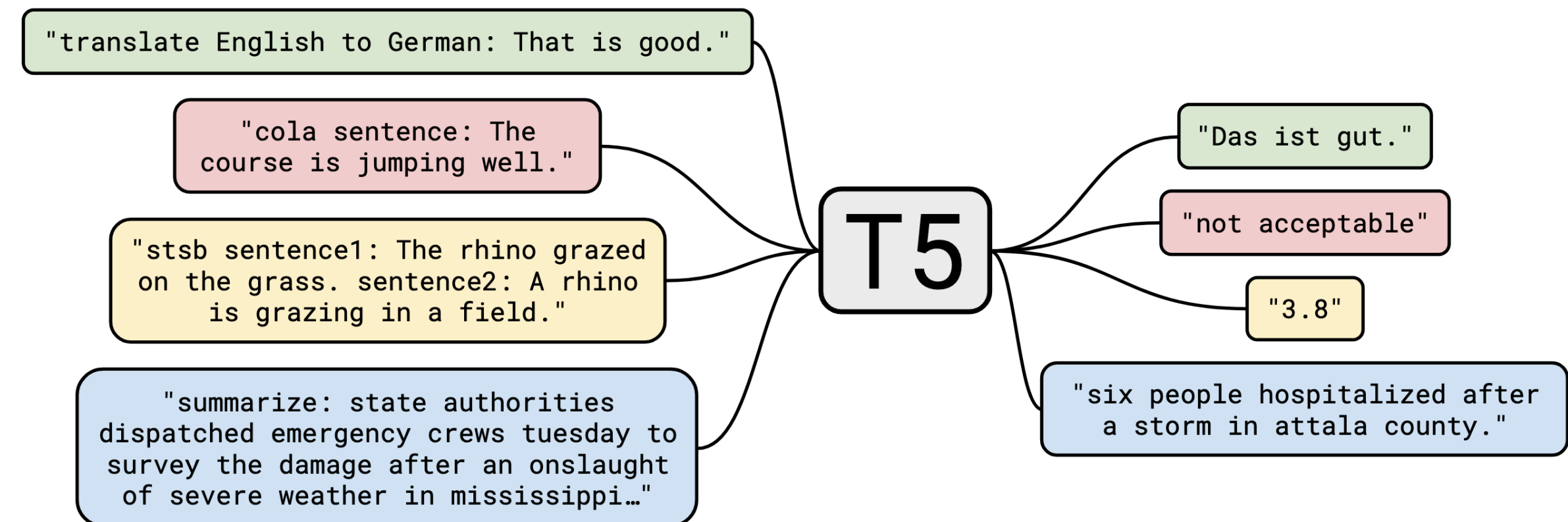
Text-to-Text NLP

- Train LMs to all tasks with **unified text output**
- **Bypasses** classification heads
- Well suited for **problems posed as text**



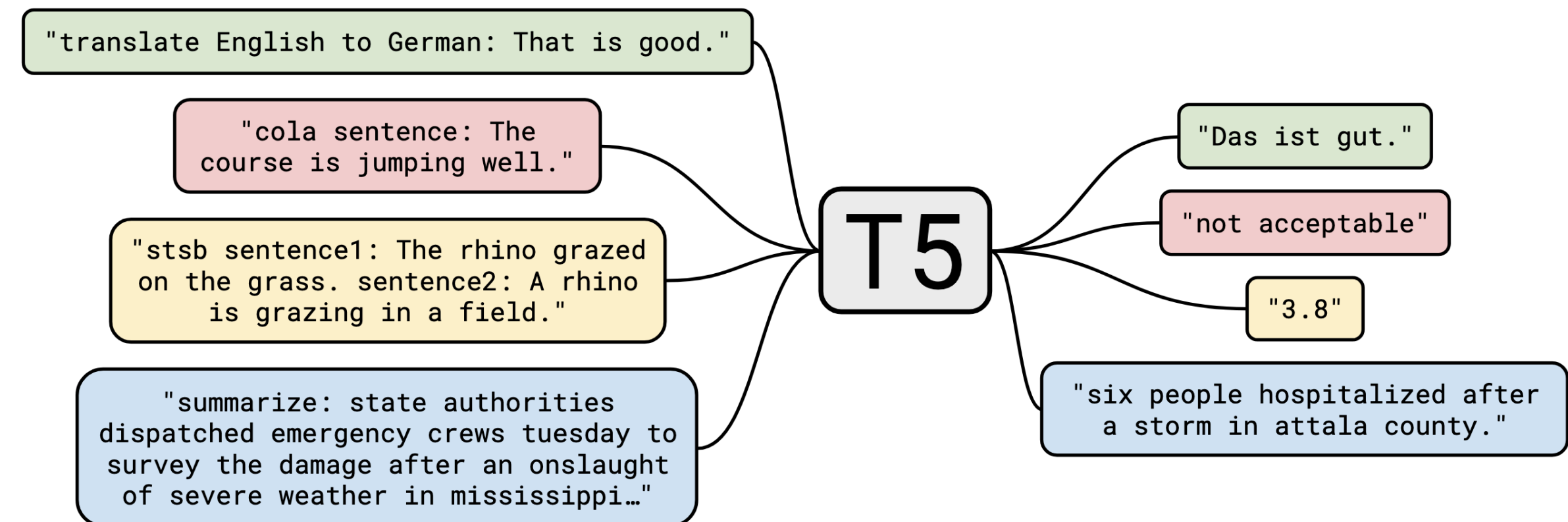
Text-to-Text NLP

- Train LMs to all tasks with **unified text output**
 - **Bypasses** classification heads
 - Well suited for **problems posed as text**
- **Explicit training** (does gradient updates)



Text-to-Text NLP

- Train LMs to all tasks with **unified text output**
 - **Bypasses** classification heads
 - Well suited for **problems posed as text**
- **Explicit training** (does gradient updates)
- Popularized by Google's **T5 model**
 - "Text-to-Text Transfer Transformer"



Converting Tasks to Language Modeling

D.2 RTE

Original input:

Sentence 1: A smaller proportion of Yugoslavia's Italians were settled in Slovenia (at the 1991 national census, some 3000 inhabitants of Slovenia declared themselves as ethnic Italians).

Sentence 2: Slovenia has 3,000 inhabitants.

Processed input: rte sentence1: A smaller proportion of Yugoslavia's Italians were settled in Slovenia (at the 1991 national census, some 3000 inhabitants of Slovenia declared themselves as ethnic Italians). sentence2: Slovenia has 3,000 inhabitants.

Original target: 1

Processed target: not_entailment

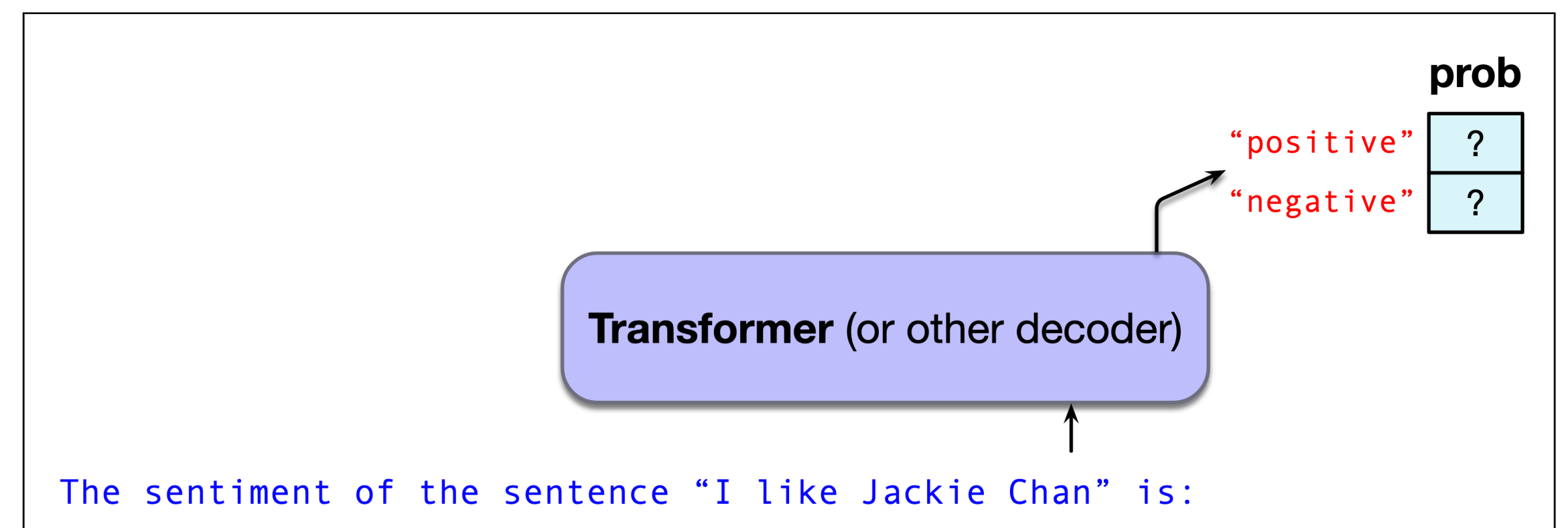


Figure 7.4 Computing the probabilities of the tokens positive and negative occurring after this prefix.

Converting Tasks to Language Modeling

- Take a supervised task and find a way to **pose it as text**

D.2 RTE

Original input:

Sentence 1: A smaller proportion of Yugoslavia's Italians were settled in Slovenia (at the 1991 national census, some 3000 inhabitants of Slovenia declared themselves as ethnic Italians).

Sentence 2: Slovenia has 3,000 inhabitants.

Processed input: rte sentence1: A smaller proportion of Yugoslavia's Italians were settled in Slovenia (at the 1991 national census, some 3000 inhabitants of Slovenia declared themselves as ethnic Italians). sentence2: Slovenia has 3,000 inhabitants.

Original target: 1

Processed target: not_entailment

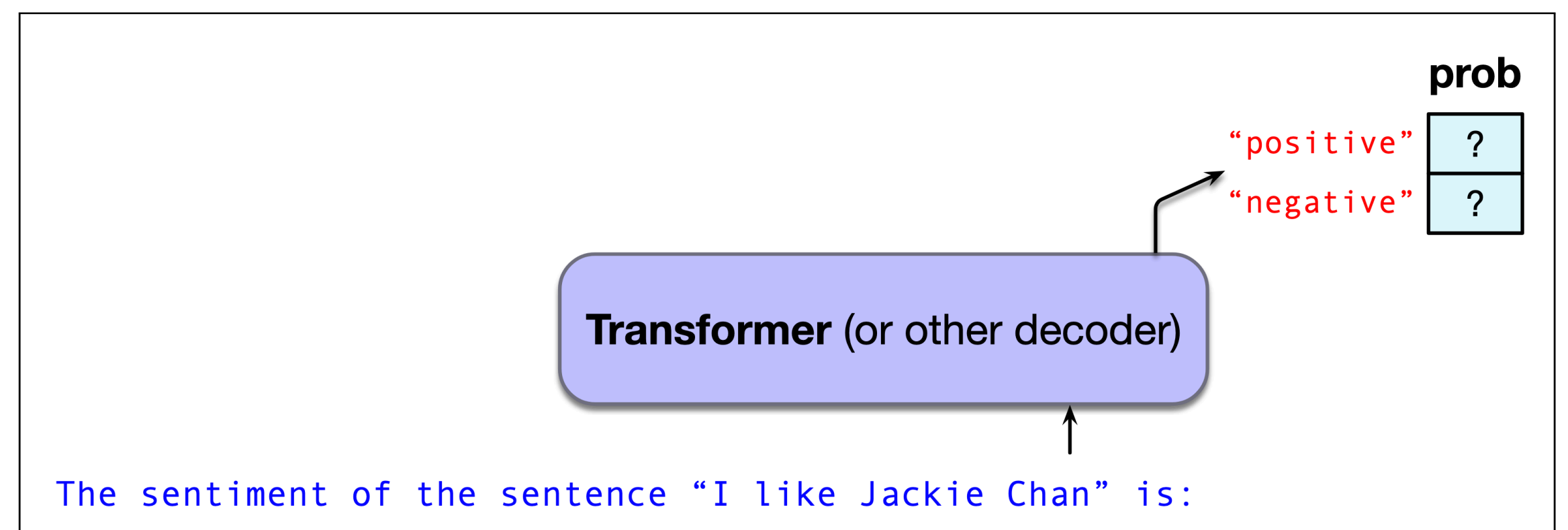


Figure 7.4 Computing the probabilities of the tokens positive and negative occurring after this prefix.

Converting Tasks to Language Modeling

- Take a supervised task and find a way to **pose it as text**
- For T5, the input is **prefixed with the task name** (e.g. `rte`)

D.2 RTE

Original input:

Sentence 1: A smaller proportion of Yugoslavia's Italians were settled in Slovenia (at the 1991 national census, some 3000 inhabitants of Slovenia declared themselves as ethnic Italians).

Sentence 2: Slovenia has 3,000 inhabitants.

Processed input: rte sentence1: A smaller proportion of Yugoslavia's Italians were settled in Slovenia (at the 1991 national census, some 3000 inhabitants of Slovenia declared themselves as ethnic Italians). sentence2: Slovenia has 3,000 inhabitants.

Original target: 1

Processed target: not_entailment

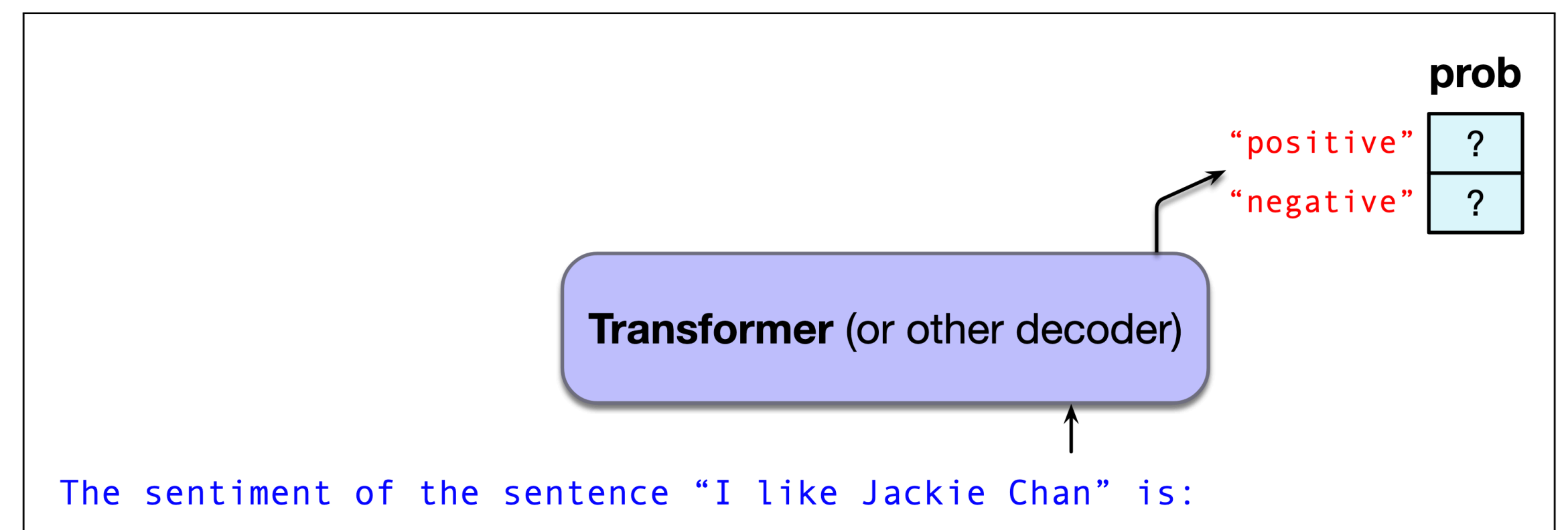


Figure 7.4 Computing the probabilities of the tokens positive and negative occurring after this prefix.

Converting Tasks to Language Modeling

- Take a supervised task and find a way to **pose it as text**
- For T5, the input is **prefixed with the task name** (e.g. `rte`)
- The task is now to **generate the tokens** of the **correct answer**

D.2 RTE

Original input:

Sentence 1: A smaller proportion of Yugoslavia's Italians were settled in Slovenia (at the 1991 national census, some 3000 inhabitants of Slovenia declared themselves as ethnic Italians).

Sentence 2: Slovenia has 3,000 inhabitants.

Processed input: rte sentence1: A smaller proportion of Yugoslavia's Italians were settled in Slovenia (at the 1991 national census, some 3000 inhabitants of Slovenia declared themselves as ethnic Italians). sentence2: Slovenia has 3,000 inhabitants.

Original target: 1

Processed target: not_entailment

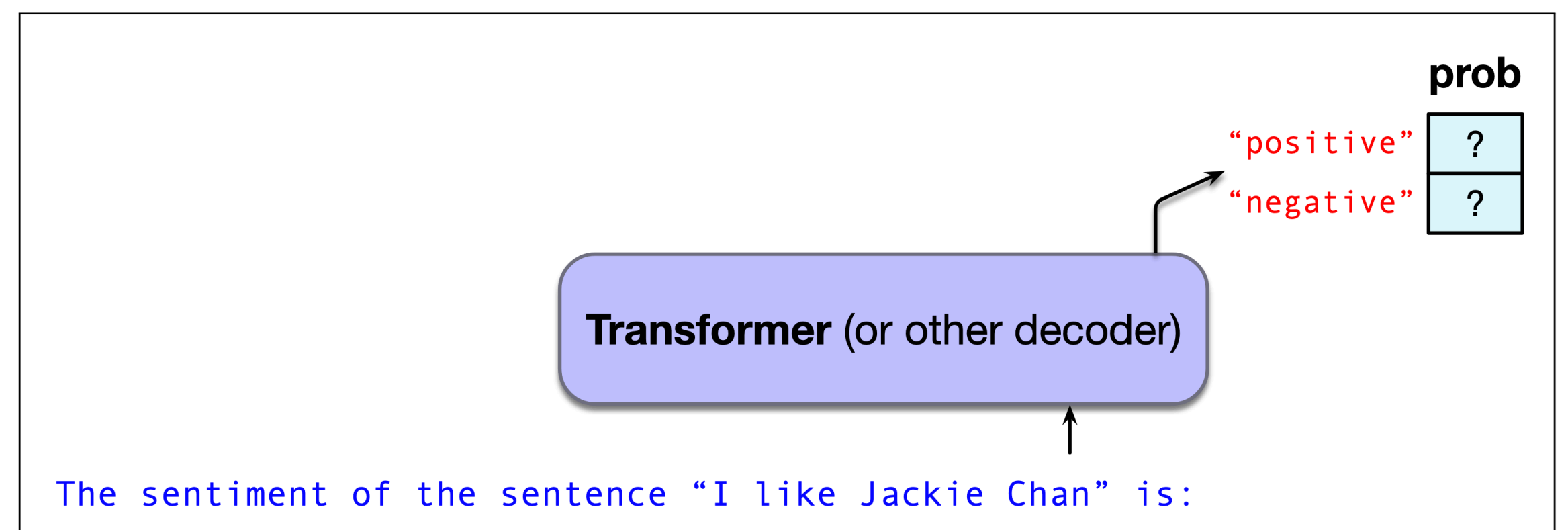


Figure 7.4 Computing the probabilities of the tokens positive and negative occurring after this prefix.

Converting Tasks to Language Modeling

- Take a supervised task and find a way to **pose it as text**
- For T5, the input is **prefixed with the task name** (e.g. `rte`)
- The task is now to **generate the tokens** of the **correct answer**
- Sometimes with **naturalistic prompting** like "the sentiment of the sentence is..."

D.2 RTE

Original input:

Sentence 1: A smaller proportion of Yugoslavia's Italians were settled in Slovenia (at the 1991 national census, some 3000 inhabitants of Slovenia declared themselves as ethnic Italians).

Sentence 2: Slovenia has 3,000 inhabitants.

Processed input: rte sentence1: A smaller proportion of Yugoslavia's Italians were settled in Slovenia (at the 1991 national census, some 3000 inhabitants of Slovenia declared themselves as ethnic Italians). sentence2: Slovenia has 3,000 inhabitants.

Original target: 1

Processed target: not_entailment

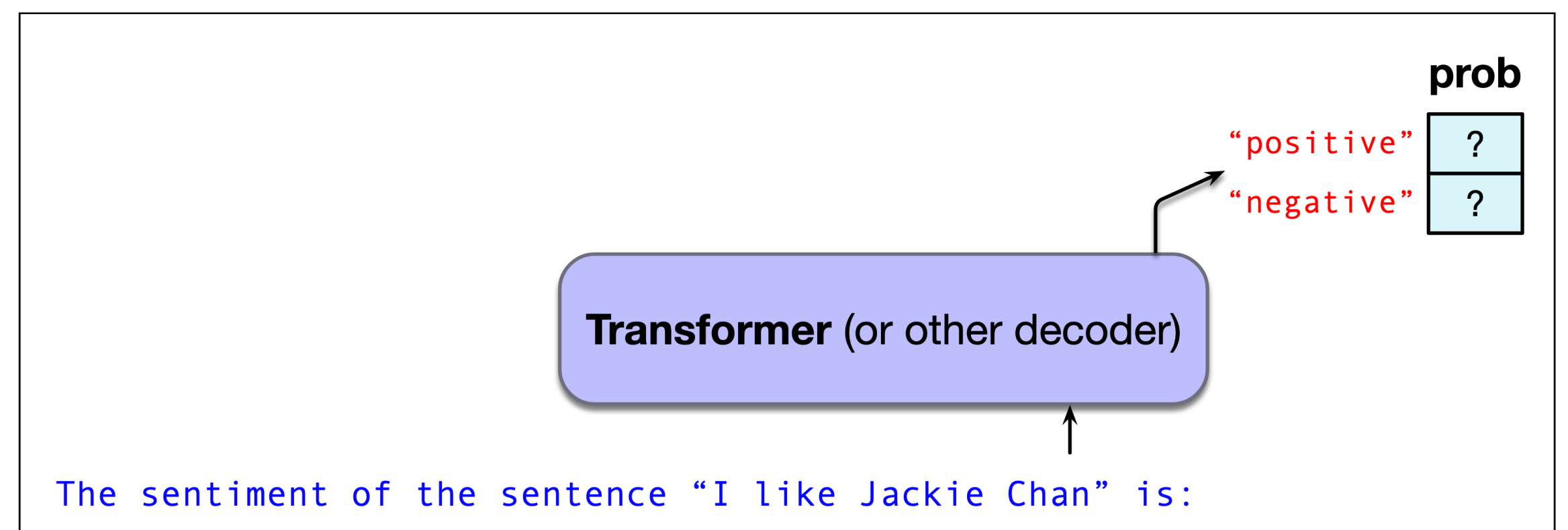


Figure 7.4 Computing the probabilities of the tokens positive and negative occurring after this prefix.

Converting Tasks to Language Modeling

- Take a supervised task and find a way to **pose it as text**
- For T5, the input is **prefixed with the task name** (e.g. `rte`)
- The task is now to **generate the tokens** of the **correct answer**
- Sometimes with **naturalistic prompting** like "the sentiment of the sentence is..."
- Other times this is with more "**meta**" text like `sentence1:` , `sentence2:`

D.2 RTE

Original input:

Sentence 1: A smaller proportion of Yugoslavia's Italians were settled in Slovenia (at the 1991 national census, some 3000 inhabitants of Slovenia declared themselves as ethnic Italians).

Sentence 2: Slovenia has 3,000 inhabitants.

Processed input: rte sentence1: A smaller proportion of Yugoslavia's Italians were settled in Slovenia (at the 1991 national census, some 3000 inhabitants of Slovenia declared themselves as ethnic Italians). sentence2: Slovenia has 3,000 inhabitants.

Original target: 1

Processed target: not_entailment

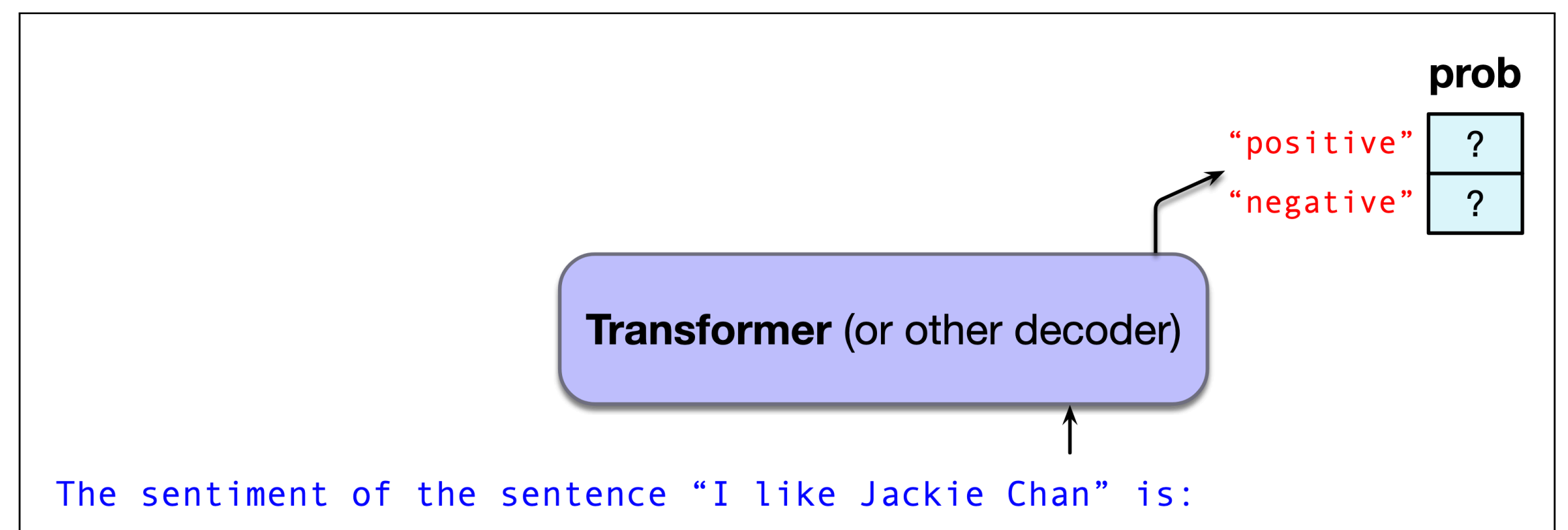
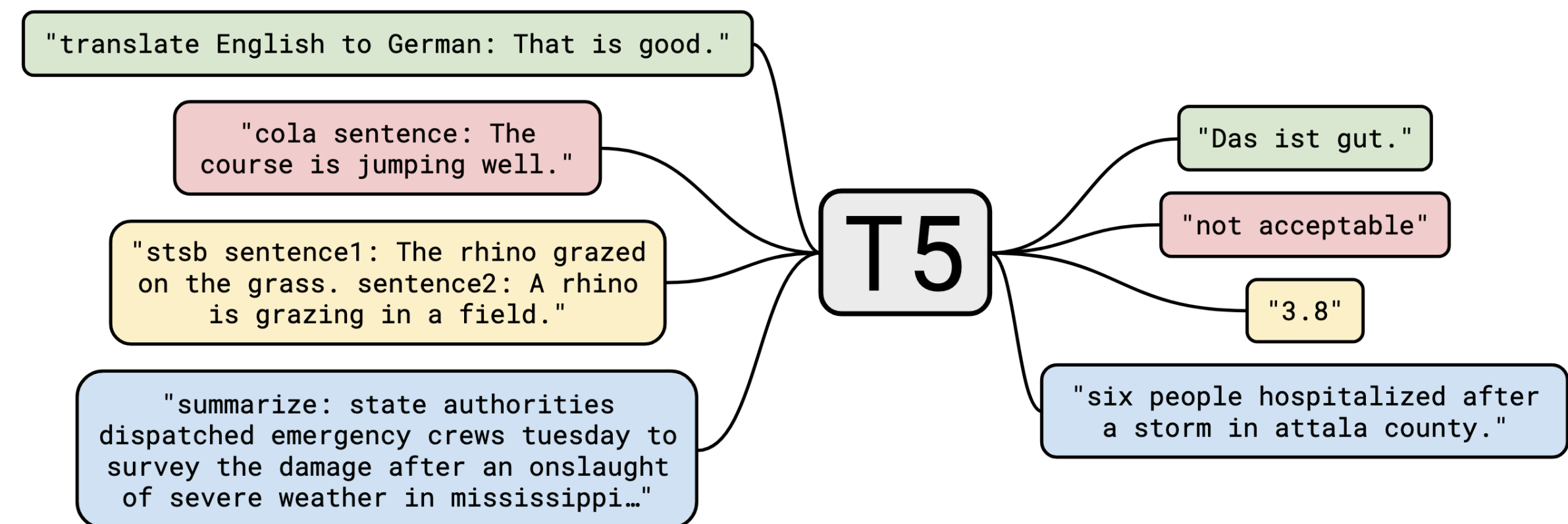


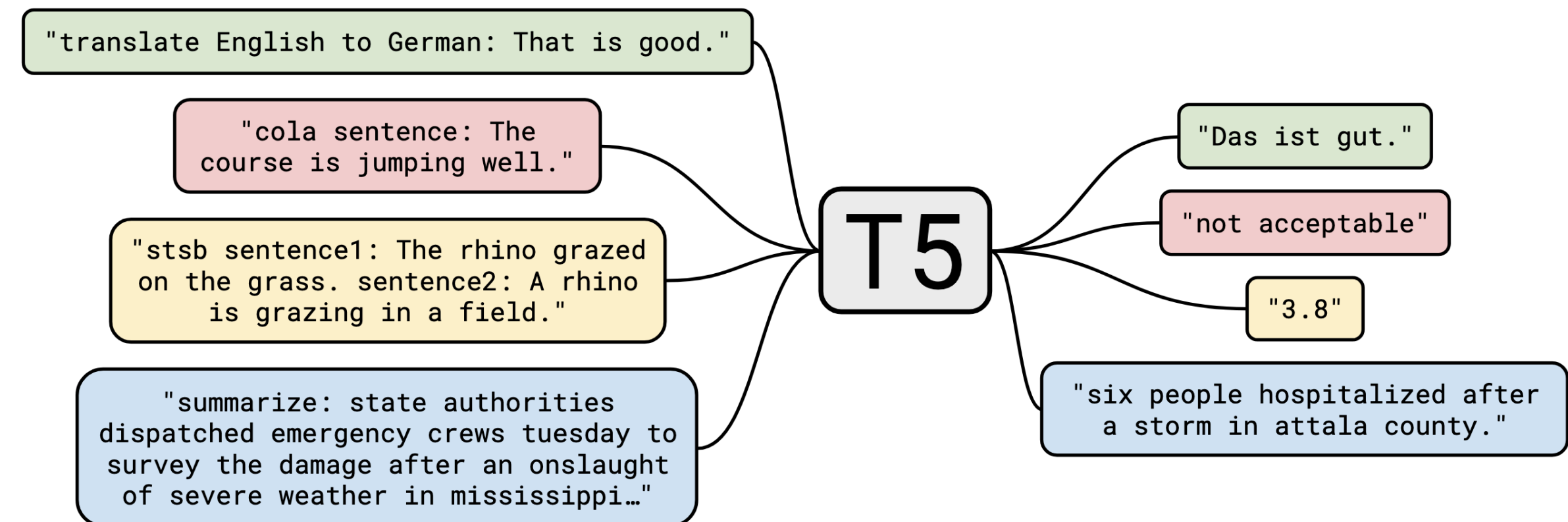
Figure 7.4 Computing the probabilities of the tokens positive and negative occurring after this prefix.

Why Text-to-Text?



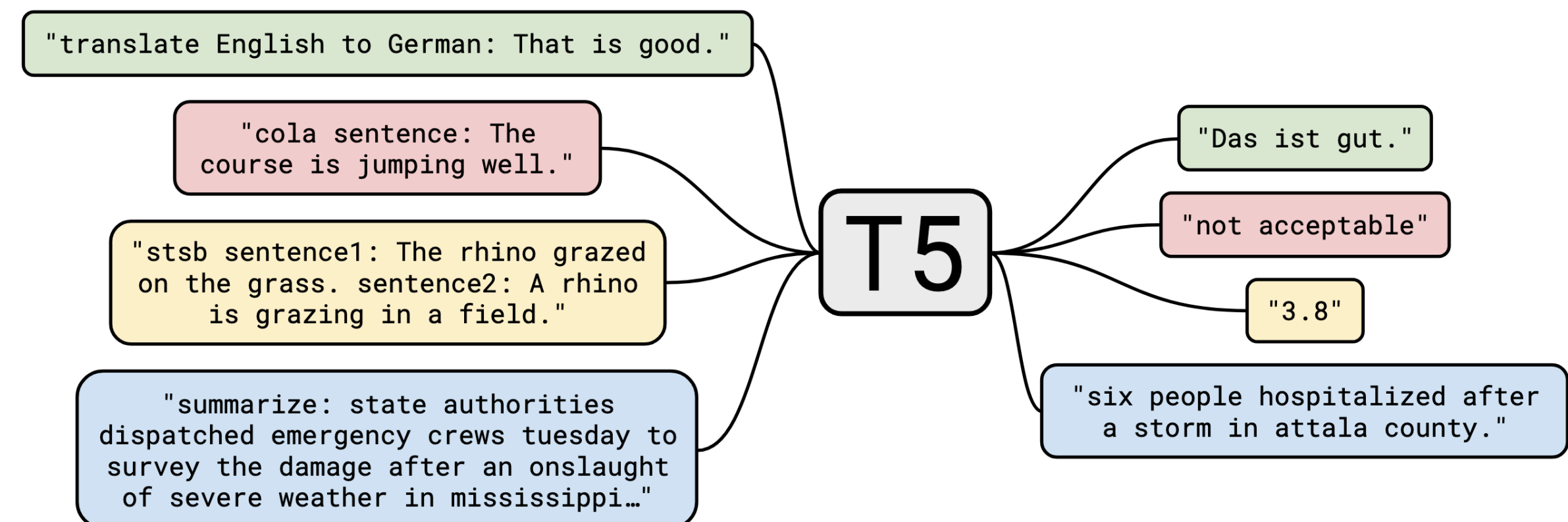
Why Text-to-Text?

- No need to train a **separate task head** for each new task



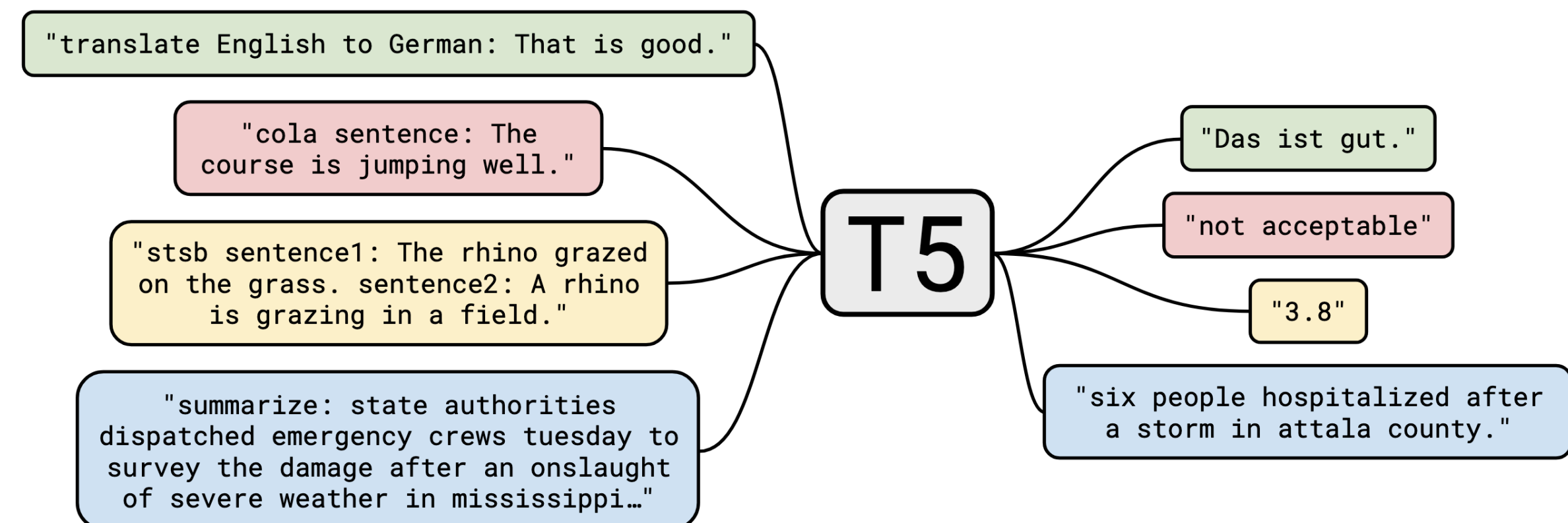
Why Text-to-Text?

- No need to train a **separate task head** for each new task
- **No specialized architecture at all!**



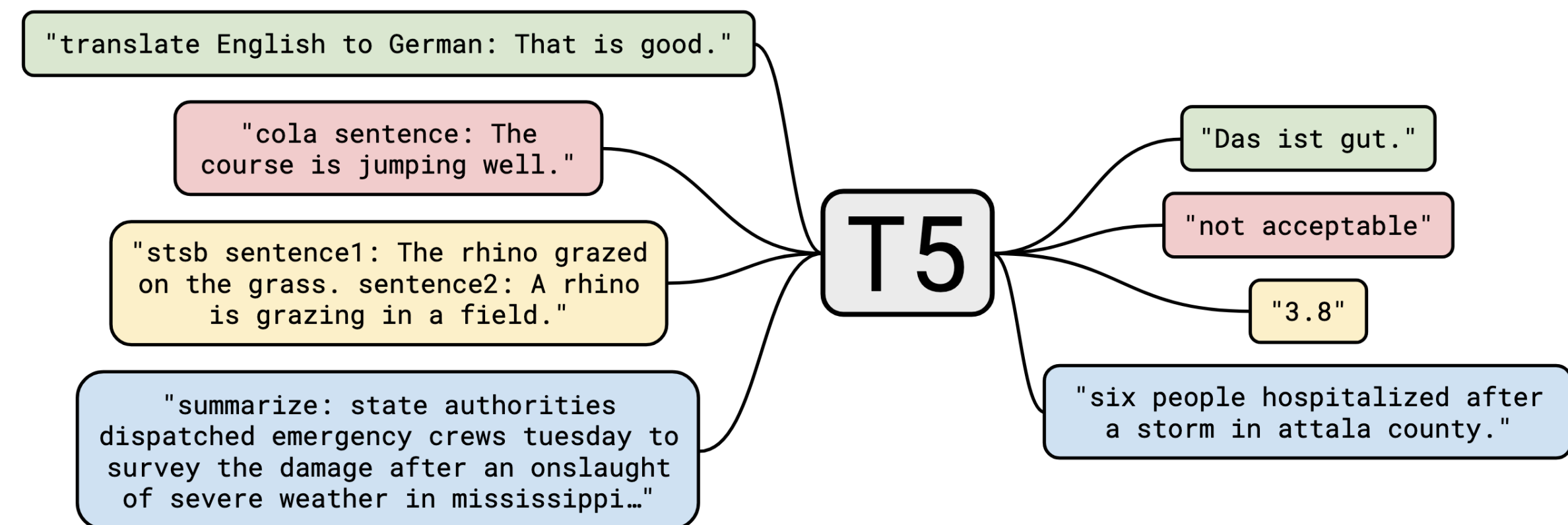
Why Text-to-Text?

- No need to train a **separate task head** for each new task
 - **No specialized architecture at all!**
- Builds on the model's **existing strength** in text prediction



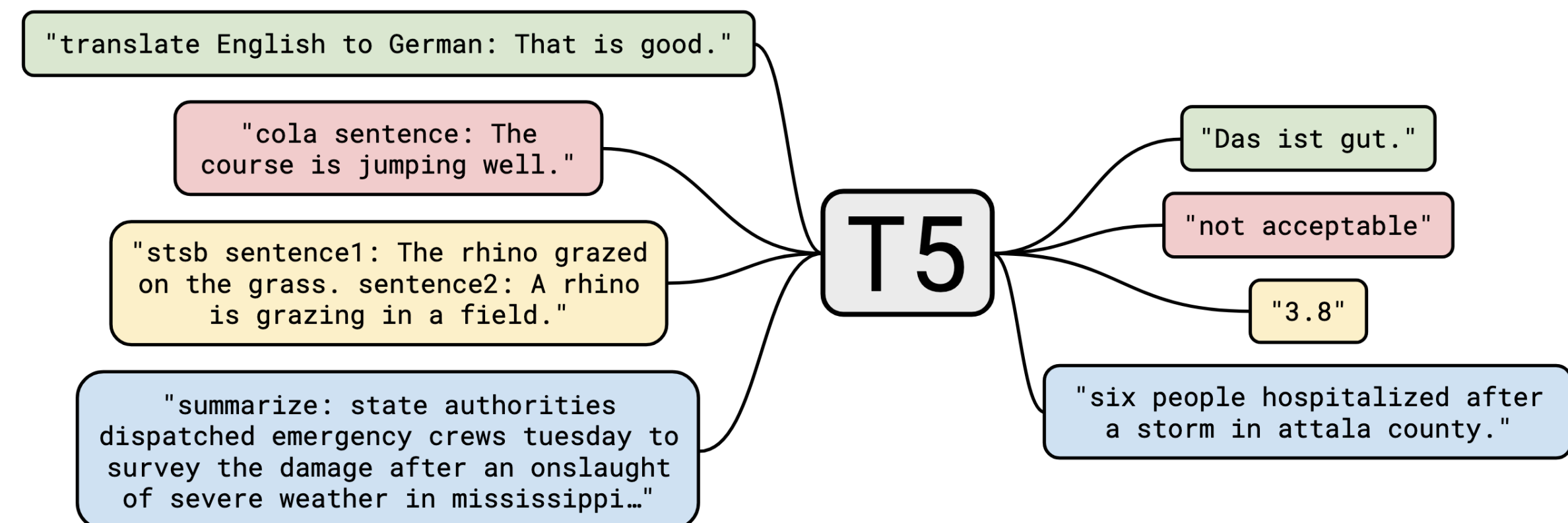
Why Text-to-Text?

- No need to train a **separate task head** for each new task
 - **No specialized architecture at all!**
- Builds on the model's **existing strength** in text prediction
- Allows models to **answer questions more naturalistically**



Why Text-to-Text?

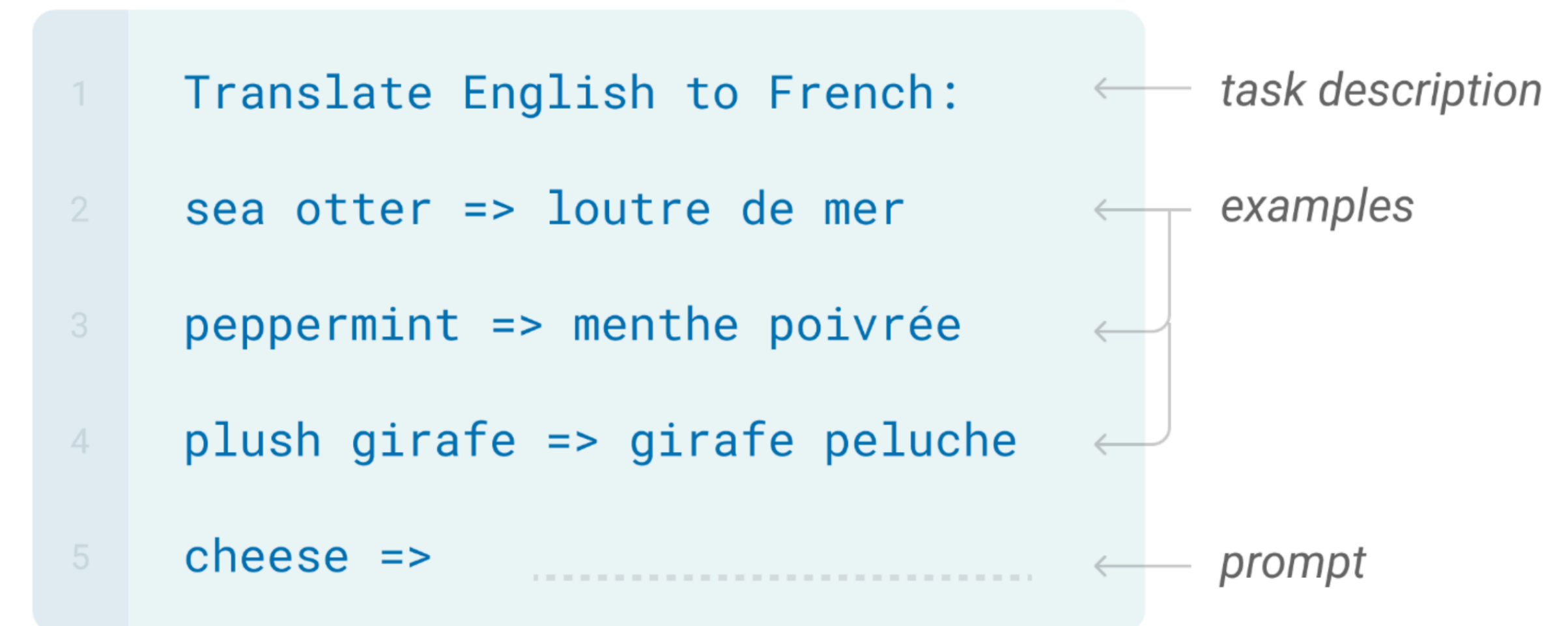
- No need to train a **separate task head** for each new task
 - **No specialized architecture at all!**
- Builds on the model's **existing strength** in text prediction
- Allows models to **answer questions more naturalistically**
- Prompts allow for **task specification and demonstration**



In-Context Learning

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

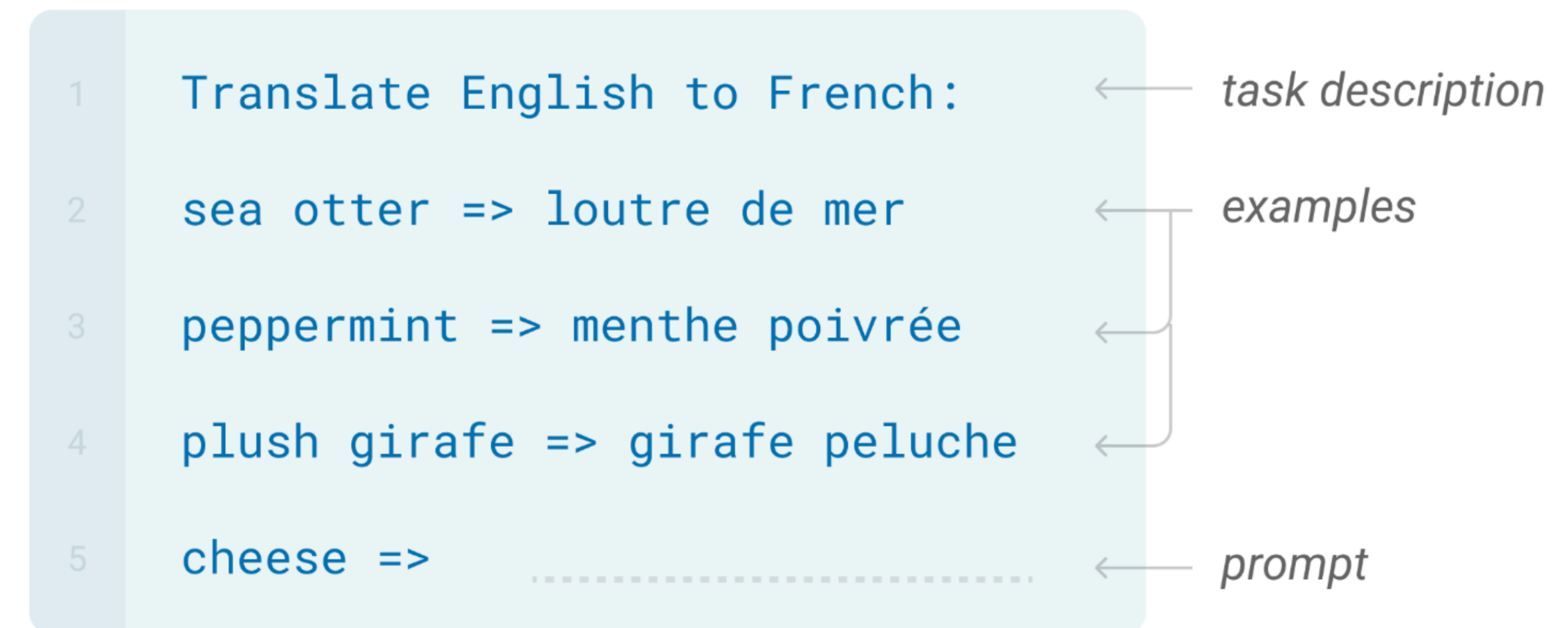


In-Context Learning

- Solve tasks by including a **description** and **examples** within a **natural language prompt**

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

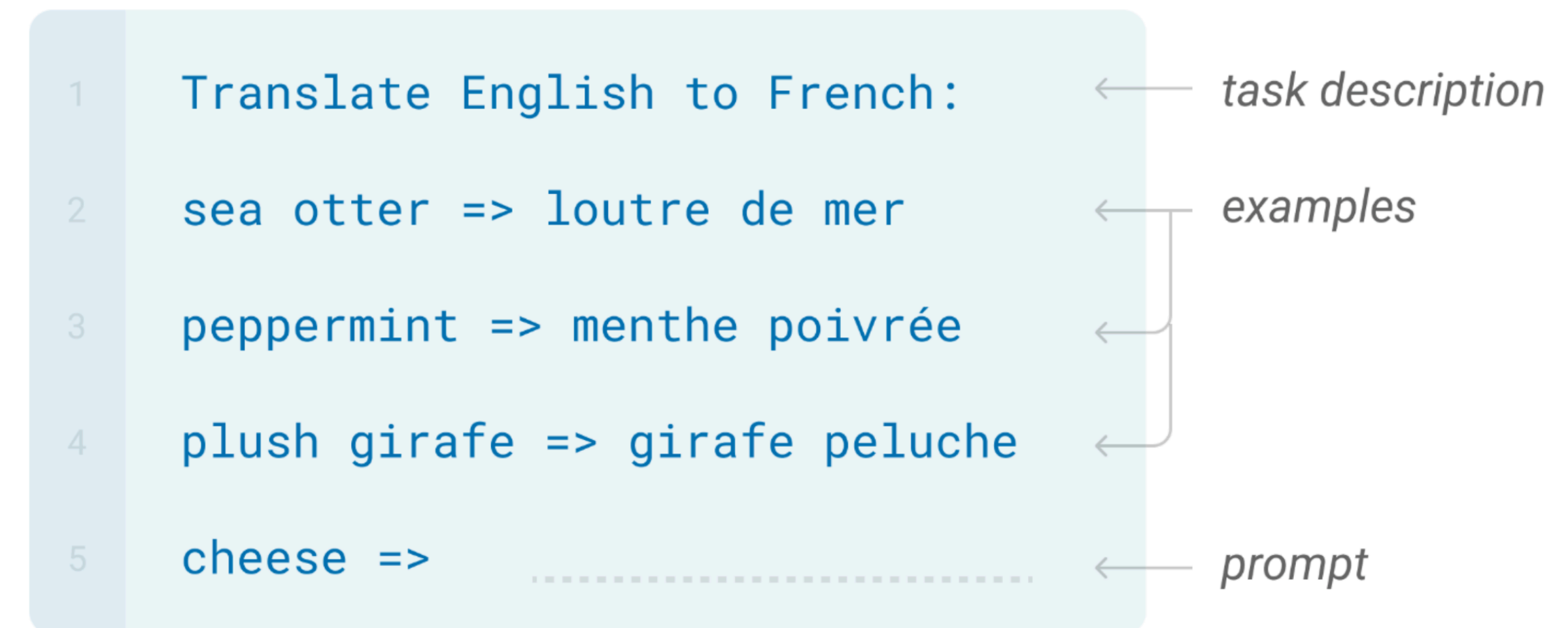


In-Context Learning

- Solve tasks by including a **description** and **examples** within a **natural language prompt**
- **No gradient updates** needed

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

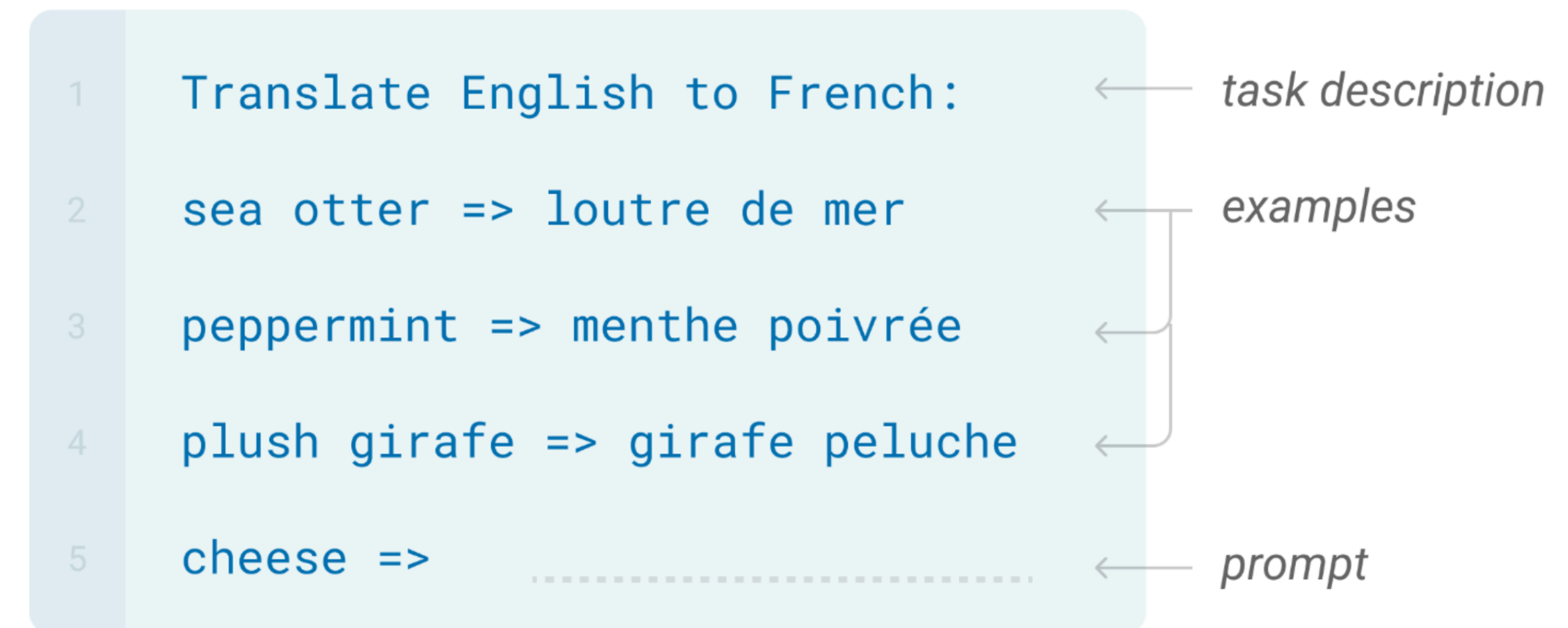


In-Context Learning

- Solve tasks by including a **description** and **examples** within a **natural language prompt**
- **No gradient updates** needed
- Popularized by the release of **GPT-3**

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

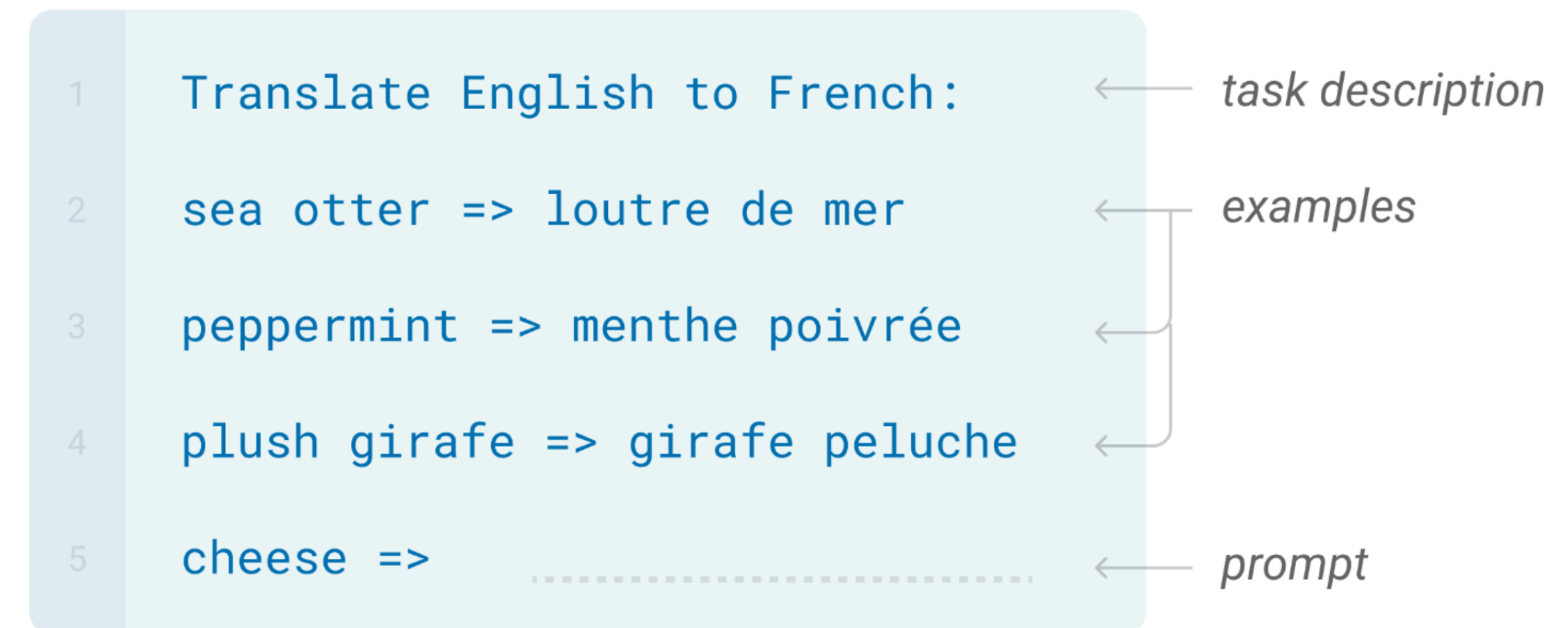


In-Context Learning

- Solve tasks by including a **description** and **examples** within a **natural language prompt**
- **No gradient updates** needed
- Popularized by the release of **GPT-3**
- **Few-shot**: examples of desired behavior/output provided

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

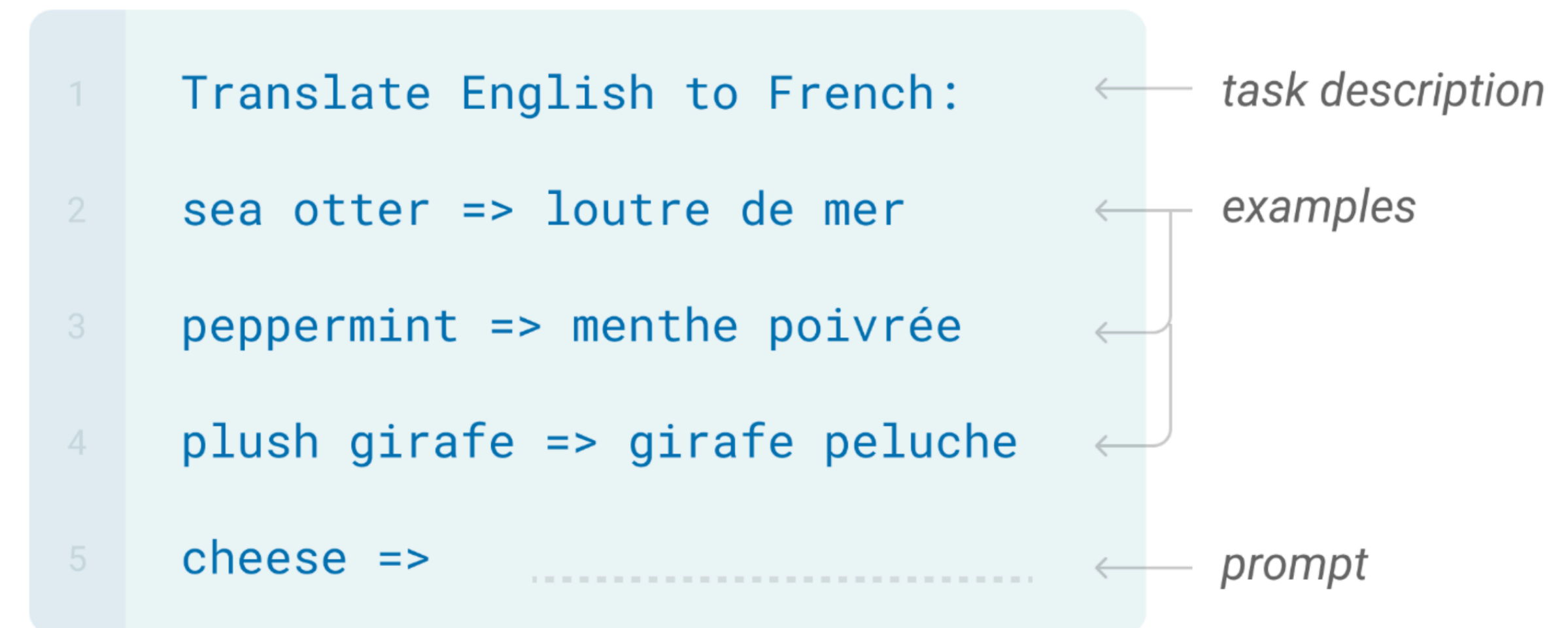


In-Context Learning

- Solve tasks by including a **description** and **examples** within a **natural language prompt**
- **No gradient updates** needed
- Popularized by the release of **GPT-3**
- **Few-shot**: examples of desired behavior/output provided
- **Zero-shot**: only task description and prompt given

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.



Prompt Engineering

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ❌

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✅

Prompt Engineering

- Variation in the prompt **significantly affects model behavior**
- Find the best way to **elicit desired behavior** from the model

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ❌

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✅

Prompt Engineering

- Variation in the prompt **significantly affects model behavior**
 - Find the best way to **elicit desired behavior** from the model
- Prominent example: “**Chain of Thought**” prompting
 - Prompt the model to generate **step-by-step** reasoning

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ❌

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✅

Prompt Engineering

- Variation in the prompt **significantly affects model behavior**
 - Find the best way to **elicit desired behavior** from the model
- Prominent example: **“Chain of Thought”** prompting
 - Prompt the model to generate **step-by-step** reasoning
- Huge area of NLP to this day

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ❌

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✅

Recipe for “Chatbot” LLMs

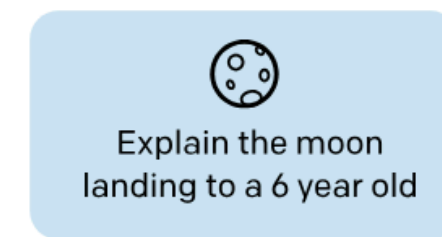
from [InstructGPT paper](#)

Overview

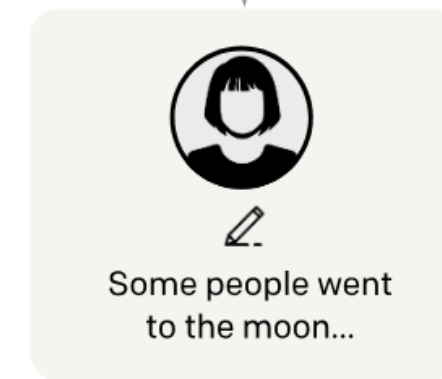
Step 1

Collect demonstration data, and train a supervised policy.

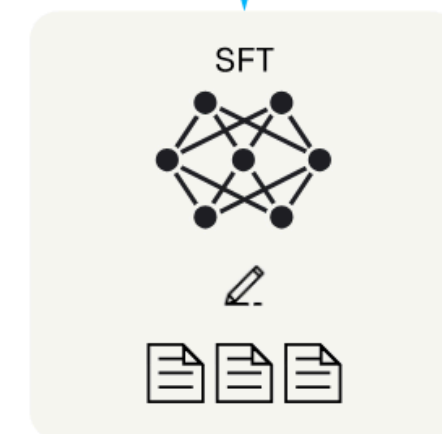
A prompt is sampled from our prompt dataset.



A labeler demonstrates the desired output behavior.



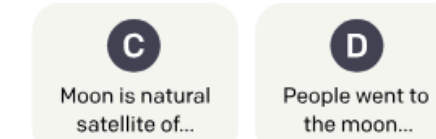
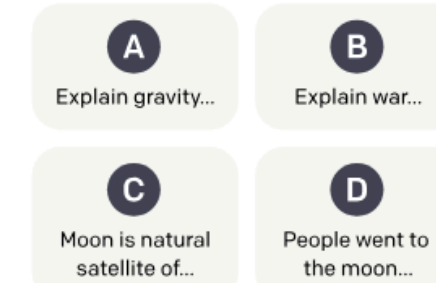
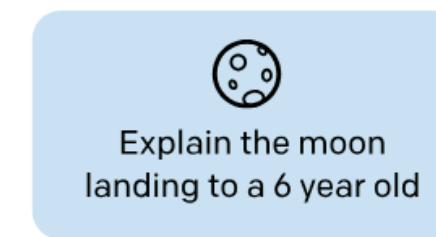
This data is used to fine-tune GPT-3 with supervised learning.



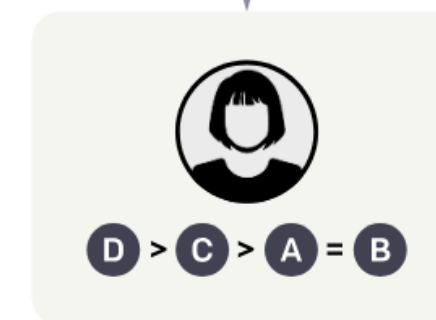
Step 2

Collect comparison data, and train a reward model.

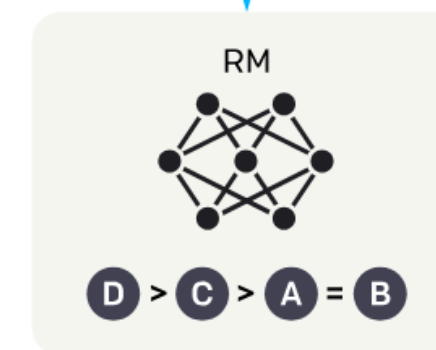
A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.



This data is used to train our reward model.



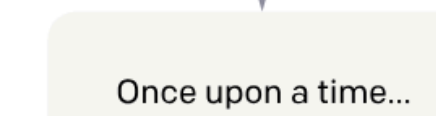
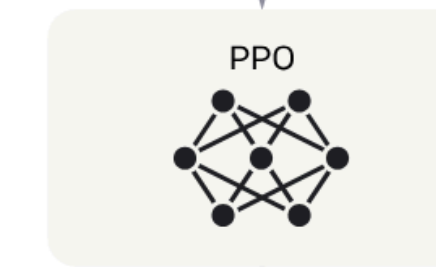
Step 3

Optimize a policy against the reward model using reinforcement learning.

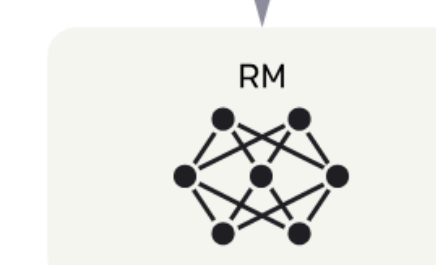
A new prompt is sampled from the dataset.



The policy generates an output.



The reward model calculates a reward for the output.



The reward is used to update the policy using PPO.



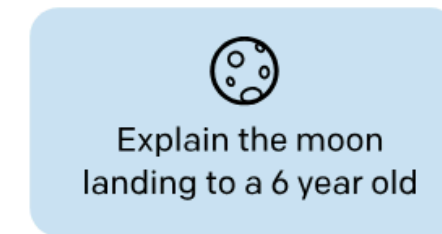
from [InstructGPT paper](#)

Overview

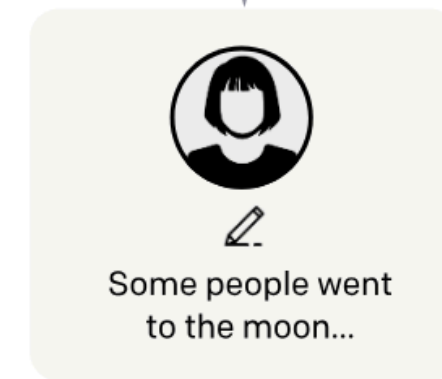
Step 1

Collect demonstration data, and train a supervised policy.

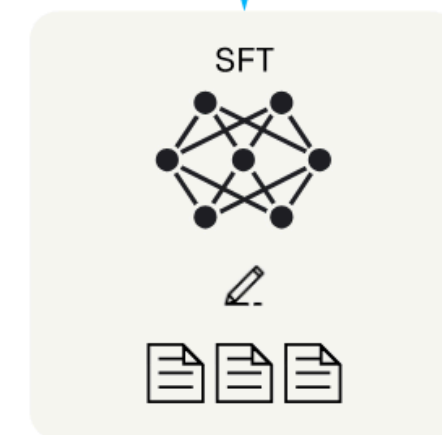
A prompt is sampled from our prompt dataset.



A labeler demonstrates the desired output behavior.



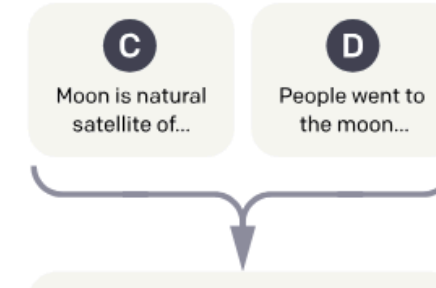
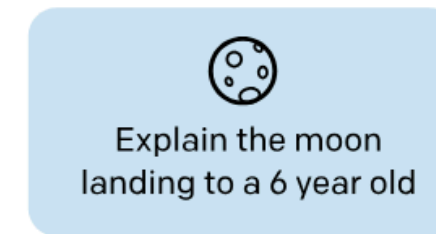
This data is used to fine-tune GPT-3 with supervised learning.



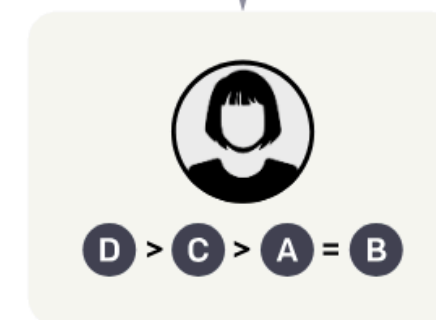
Step 2

Collect comparison data, and train a reward model.

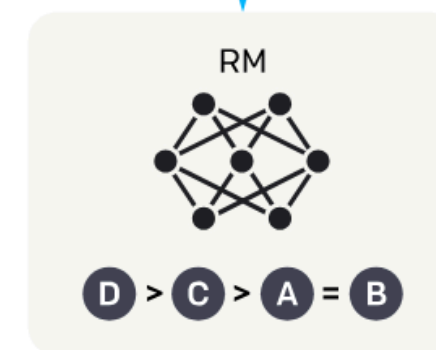
A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.



This data is used to train our reward model.



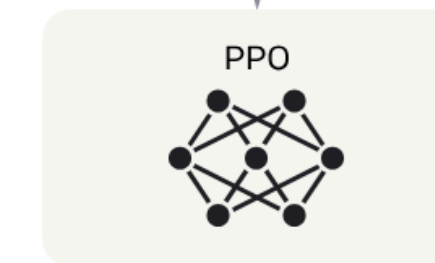
Step 3

Optimize a policy against the reward model using reinforcement learning.

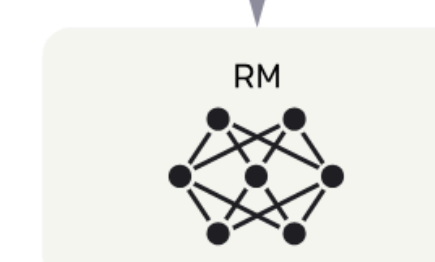
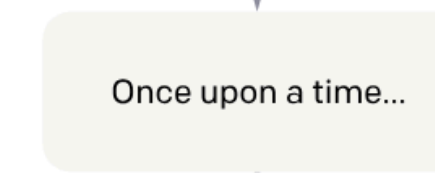
A new prompt is sampled from the dataset.



The policy generates an output.



The reward model calculates a reward for the output.



The reward is used to update the policy using PPO.



Instruction tuning

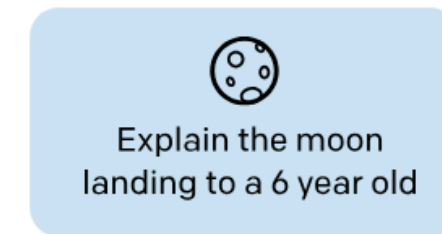
from [InstructGPT paper](#)

Overview

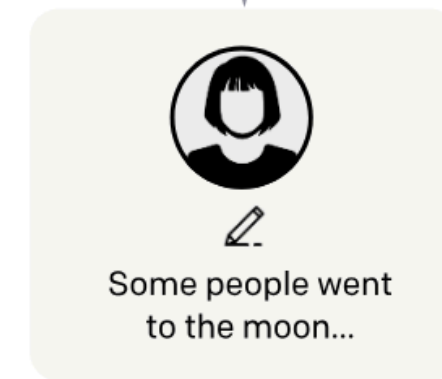
Step 1

Collect demonstration data, and train a supervised policy.

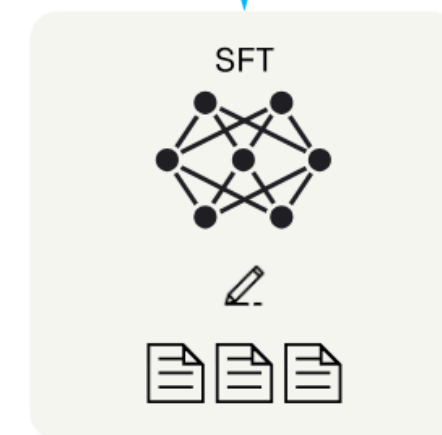
A prompt is sampled from our prompt dataset.



A labeler demonstrates the desired output behavior.



This data is used to fine-tune GPT-3 with supervised learning.

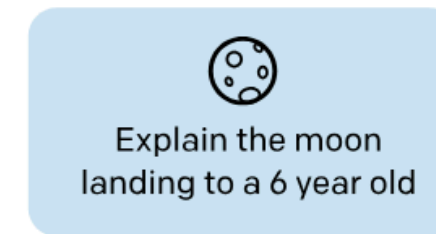


Instruction tuning

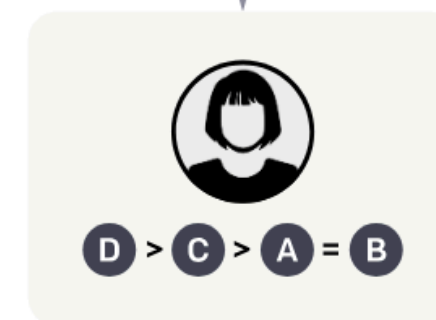
Step 2

Collect comparison data, and train a reward model.

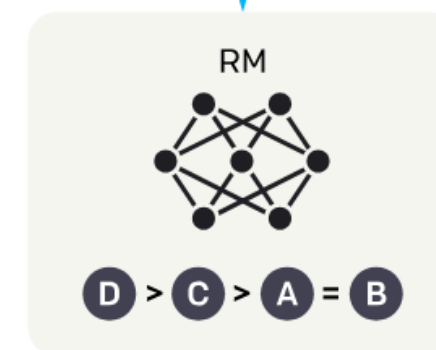
A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.



This data is used to train our reward model.



Preference data collection

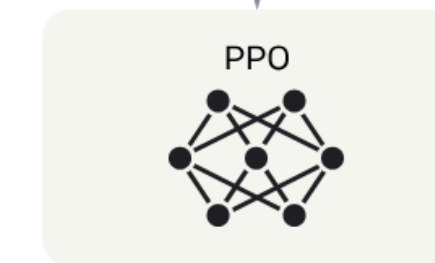
Step 3

Optimize a policy against the reward model using reinforcement learning.

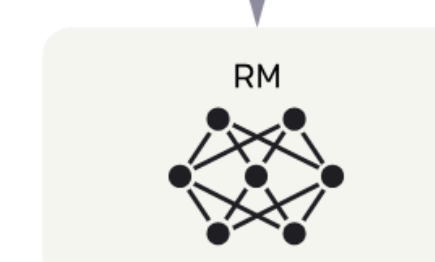
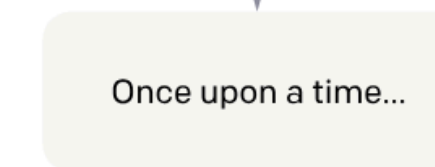
A new prompt is sampled from the dataset.



The policy generates an output.



The reward model calculates a reward for the output.



The reward is used to update the policy using PPO.



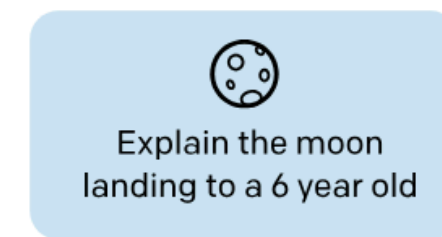
from [InstructGPT paper](#)

Overview

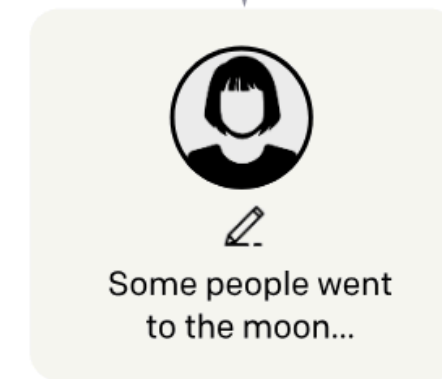
Step 1

Collect demonstration data, and train a supervised policy.

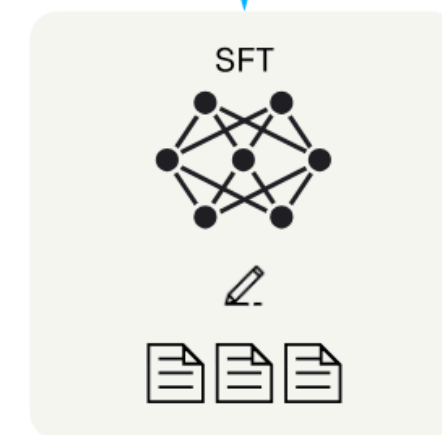
A prompt is sampled from our prompt dataset.



A labeler demonstrates the desired output behavior.



This data is used to fine-tune GPT-3 with supervised learning.

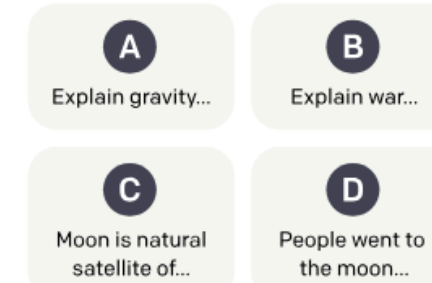
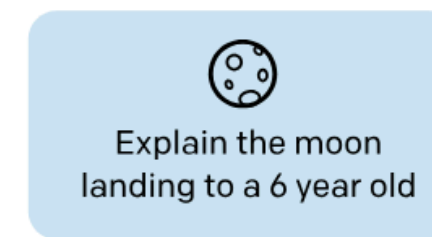


Instruction tuning

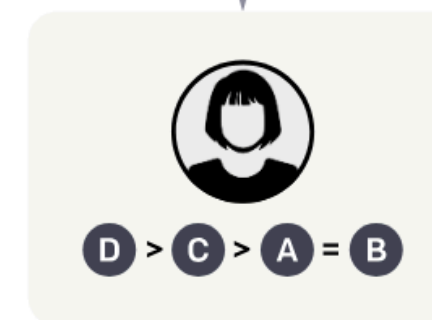
Step 2

Collect comparison data, and train a reward model.

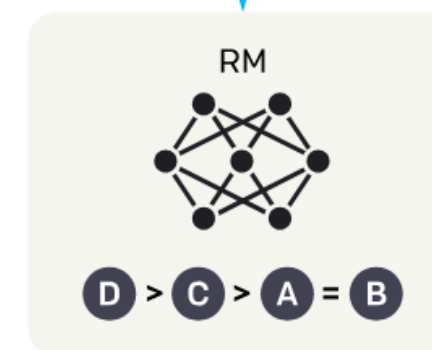
A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.



This data is used to train our reward model.



Preference data collection

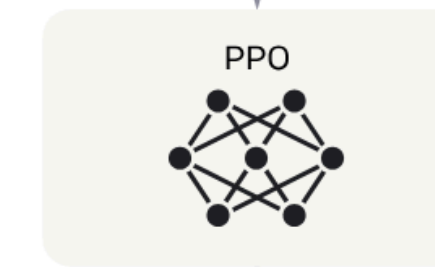
Step 3

Optimize a policy against the reward model using reinforcement learning.

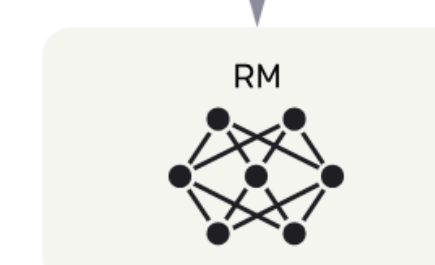
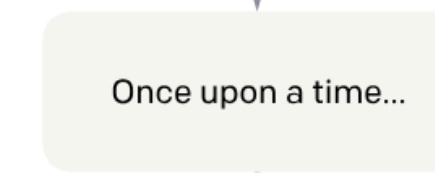
A new prompt is sampled from the dataset.



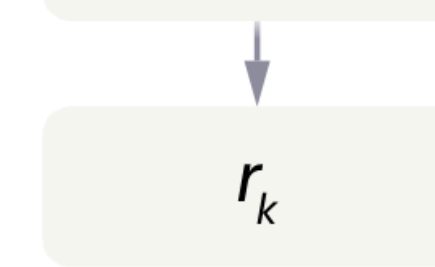
The policy generates an output.



The reward model calculates a reward for the output.



The reward is used to update the policy using PPO.



Reinforcement Learning from Human Feedback (RLHF)

Instruction Tuning

Finetune on many tasks (“instruction-tuning”)

<p>Input (Commonsense Reasoning)</p> <p>Here is a goal: Get a cool sleep on summer days.</p> <p>How would you accomplish this goal?</p> <p>OPTIONS:</p> <p>-Keep stack of pillow cases in fridge.</p> <p>-Keep stack of pillow cases in oven.</p> <p>Target</p> <p>keep stack of pillow cases in fridge</p>	<p>Input (Translation)</p> <p>Translate this sentence to Spanish:</p> <p>The new office building was built in less than three months.</p> <p>Target</p> <p>El nuevo edificio de oficinas se construyó en tres meses.</p>
---	--

Sentiment analysis tasks

Coreference resolution tasks

...

from [FLAN paper](#)

Inference on unseen task type

→

Input (Natural Language Inference)

Premise: At my age you will probably have learnt one lesson.

Hypothesis: It's not certain how many lessons you'll learn by your thirties.

Does the premise entail the hypothesis?

OPTIONS:

-yes -it is not possible to tell -no

FLAN Response

It is not possible to tell

Instruction Tuning

- Explicitly train on **textual formulations of tasks** (like T5)
 - Subtle differences from T5, including **generalization to unseen tasks**

Finetune on many tasks (“instruction-tuning”)

Input (Commonsense Reasoning) Here is a goal: Get a cool sleep on summer days. How would you accomplish this goal? OPTIONS: <input type="radio"/> -Keep stack of pillow cases in fridge. <input type="radio"/> -Keep stack of pillow cases in oven. Target keep stack of pillow cases in fridge	Input (Translation) Translate this sentence to Spanish: The new office building was built in less than three months. Target El nuevo edificio de oficinas se construyó en tres meses.
--	---

Sentiment analysis tasks
Coreference resolution tasks
...

from [FLAN paper](#)

Inference on unseen task type

→

Input (Natural Language Inference) Premise: At my age you will probably have learnt one lesson. Hypothesis: It's not certain how many lessons you'll learn by your thirties. Does the premise entail the hypothesis? OPTIONS: <input type="radio"/> -yes <input type="radio"/> -it is not possible to tell <input type="radio"/> -no FLAN Response It is not possible to tell
--

Instruction Tuning

- Explicitly train on **textual formulations of tasks** (like T5)
 - Subtle differences from T5, including **generalization to unseen tasks**
- Later: **demonstration data**
 - Have an annotator write out the **ideal response** to input from an end-user
 - Explicitly training the model to **act as an interlocutor**

Finetune on many tasks (“instruction-tuning”)

<p>Input (Commonsense Reasoning)</p> <p>Here is a goal: Get a cool sleep on summer days.</p> <p>How would you accomplish this goal?</p> <p>OPTIONS:</p> <p>-Keep stack of pillow cases in fridge.</p> <p>-Keep stack of pillow cases in oven.</p> <p>Target</p> <p>keep stack of pillow cases in fridge</p>	<p>Input (Translation)</p> <p>Translate this sentence to Spanish:</p> <p>The new office building was built in less than three months.</p> <p>Target</p> <p>El nuevo edificio de oficinas se construyó en tres meses.</p>
---	--

Sentiment analysis tasks

Coreference resolution tasks

...

from [FLAN paper](#)

Inference on unseen task type

→

Input (Natural Language Inference)

Premise: At my age you will probably have learnt one lesson.

Hypothesis: It's not certain how many lessons you'll learn by your thirties.

Does the premise entail the hypothesis?

OPTIONS:

-yes -it is not possible to tell -no

FLAN Response

It is not possible to tell

Instruction Tuning

- Explicitly train on **textual formulations of tasks** (like T5)
 - Subtle differences from T5, including **generalization to unseen tasks**
- Later: **demonstration data**
 - Have an annotator write out the **ideal response** to input from an end-user
 - Explicitly training the model to **act as an interlocutor**
- Confusingly called “**Supervised Fine-Tuning**” (SFT) sometimes

Finetune on many tasks (“instruction-tuning”)

Input (Commonsense Reasoning)
Here is a goal: Get a cool sleep on summer days.
How would you accomplish this goal?
OPTIONS:
-Keep stack of pillow cases in fridge.
-Keep stack of pillow cases in oven.
Target
keep stack of pillow cases in fridge

Input (Translation)
Translate this sentence to Spanish:
The new office building was built in less than three months.
Target
El nuevo edificio de oficinas se construyó en tres meses.

Sentiment analysis tasks
Coreference resolution tasks
...

from [FLAN paper](#)

Inference on unseen task type

Input (Natural Language Inference)
Premise: At my age you will probably have learnt one lesson.
Hypothesis: It's not certain how many lessons you'll learn by your thirties.
Does the premise entail the hypothesis?
OPTIONS:
-yes -it is not possible to tell -no

FLAN Response
It is not possible to tell

Building Instruction-Tuning Sets

por	Qual é a capital do estado da Bahia?	A capital do estado da Bahia, no Brasil, é Salvador. Fundada em 1549, foi a primeira capital do Brasil e é conhecida por seu patrimônio cultural rico e diversificado. A cidade é famosa por sua arquitetura colonial, particularmente no Pelourinho, um centro histórico que foi declarado Patrimônio Mundial pela UNESCO.
pes	در برج کبوترخانه میبد چند لانه کبوتر وجود دارد؟	کبوترخانه میبد برجی استوانه‌ای است که داخل آن چهار هزار لانه کبوتر وجود دارد و نوار سفید رنگی برای جذب کبوتران دورتادور برج کشیده شده است. این برج در برابر خطر حمله دیگر حیوانات به کبوترها بسیار ایمن بوده است.
msa	Apakah nasi lemak?	Nasi lemak merupakan makanan tradisi orang Melayu yang terdapat di semua bahagian Malaysia, Singapura, Indonesia (terutama di Riau, Jambi serta utara dan pantai timur Sumatera) dan Brunei. Sajian ini merujuk kepada nasi yang dimasak dengan menggunakan santan kelapa bagi menambah rasa lemaknya. Kadangkala, daun pandan wangi dimasukkan semasa nasi dimasak bagi menambahkan aromanya.
tam	செயற்கை நுண்ணறிவு என்றால் என்ன?	பொதுவாக மனிதர்களால் செய்யப்படும் பணிகளைச் செய்ய ஒரு கணினி அல்லது ஒரு கணினியால் கட்டுப்படுத்தப்படும் ஒரு ரோபோவின் திறன் செயற்கை நுண்ணறிவு எனப்படும்.

Figure 9.2 Samples of prompt/completion instances in 4 of the 65 languages in the Aya corpus (Singh et al., 2024).

Building Instruction-Tuning Sets

- How do we **create** this instruction-tuning data? Multiple options

por	Qual é a capital do estado da Bahia?	A capital do estado da Bahia, no Brasil, é Salvador. Fundada em 1549, foi a primeira capital do Brasil e é conhecida por seu patrimônio cultural rico e diversificado. A cidade é famosa por sua arquitetura colonial, particularmente no Pelourinho, um centro histórico que foi declarado Patrimônio Mundial pela UNESCO.
pes	در برج کبوترخانه میبد چند لانه کبوتر وجود دارد؟	کبوترخانه میبد برجی استوانه‌ای است که داخل آن چهار هزار لانه کبوتر وجود دارد و نوار سفید رنگی برای جذب کبوتران دورتادور برج کشیده شده است. این برج در برابر خطر حمله دیگر حیوانات به کبوترها بسیار ایمن بوده است.
msa	Apakah nasi lemak?	Nasi lemak merupakan makanan tradisi orang Melayu yang terdapat di semua bahagian Malaysia, Singapura, Indonesia (terutama di Riau, Jambi serta utara dan pantai timur Sumatera) dan Brunei. Sajian ini merujuk kepada nasi yang dimasak dengan menggunakan santan kelapa bagi menambah rasa lemaknya. Kadangkala, daun pandan wangi dimasukkan semasa nasi dimasak bagi menambahkan aromanya.
tam	செயற்கை நுண்ணறிவு என்றால் என்ன?	பொதுவாக மனிதர்களால் செய்யப்படும் பணிகளைச் செய்ய ஒரு கணினி அல்லது ஒரு கணினியால் கட்டுப்படுத்தப்படும் ஒரு ரோபோவின் திறன் செயற்கை நுண்ணறிவு எனப்படும்.

Figure 9.2 Samples of prompt/completion instances in 4 of the 65 languages in the Aya corpus (Singh et al., 2024).

Building Instruction-Tuning Sets

- How do we **create** this instruction-tuning data? Multiple options
- Option 1: **human-written data**
 - Use people (usually **crowd-workers**) to write **prompt-completion pairs**
 - **High-quality** data, but **expensive** and **time-consuming** to curate

por	Qual é a capital do estado da Bahia?	A capital do estado da Bahia, no Brasil, é Salvador. Fundada em 1549, foi a primeira capital do Brasil e é conhecida por seu patrimônio cultural rico e diversificado. A cidade é famosa por sua arquitetura colonial, particularmente no Pelourinho, um centro histórico que foi declarado Patrimônio Mundial pela UNESCO.
pes	در برج کیوتراخانه میبد چند لانه کیوتر وجود دارد؟	کیوتراخانه میبد برجی استوانه‌ای است که داخل آن چهار هزار لانه کیوتر وجود دارد و نوار سفید رنگی برای جذب کیوتران دورتادور برج کشیده شده است. این برج در برابر خطر حمله دیگر حیوانات به کیوترها بسیار ایمن بوده است.
msa	Apakah nasi lemak?	Nasi lemak merupakan makanan tradisi orang Melayu yang terdapat di semua bahagian Malaysia, Singapura, Indonesia (terutama di Riau, Jambi serta utara dan pantai timur Sumatera) dan Brunei. Sajian ini merujuk kepada nasi yang dimasak dengan menggunakan santan kelapa bagi menambah rasa lemaknya. Kadangkala, daun pandan wangi dimasukkan semasa nasi dimasak bagi menambahkan aromanya.
tam	செயற்கை நுண்ணறிவு என்றால் என்ன?	பொதுவாக மனிதர்களால் செய்யப்படும் பணிகளைச் செய்ய ஒரு கணினி அல்லது ஒரு கணினியால் கட்டுப்படுத்தப்படும் ஒரு ரோபோவின் திறன் செயற்கை நுண்ணறிவு எனப்படும்.

Figure 9.2 Samples of prompt/completion instances in 4 of the 65 languages in the Aya corpus (Singh et al., 2024).

Building Instruction-Tuning Sets

- How do we **create** this instruction-tuning data? Multiple options
- Option 1: **human-written data**
 - Use people (usually **crowd-workers**) to write **prompt-completion pairs**
 - **High-quality** data, but **expensive** and **time-consuming** to curate
- Example: **Aya Dataset**
 - Written by **recruited language experts**
 - 204k examples, 65 languages, 3000 experts

por	Qual é a capital do estado da Bahia?	A capital do estado da Bahia, no Brasil, é Salvador. Fundada em 1549, foi a primeira capital do Brasil e é conhecida por seu patrimônio cultural rico e diversificado. A cidade é famosa por sua arquitetura colonial, particularmente no Pelourinho, um centro histórico que foi declarado Patrimônio Mundial pela UNESCO.
pes	در برج کیوتراخانه میبد چند لانه کیوتر وجود دارد؟	کیوتراخانه میبد برجی استوانه‌ای است که داخل آن چهار هزار لانه کیوتر وجود دارد و نوار سفید رنگی برای جذب کیوتران دورتادور برج کشیده شده است. این برج در برابر خطر حمله دیگر حیوانات به کیوترها بسیار ایمن بوده است.
msa	Apakah nasi lemak?	Nasi lemak merupakan makanan tradisi orang Melayu yang terdapat di semua bahagian Malaysia, Singapura, Indonesia (terutama di Riau, Jambi serta utara dan pantai timur Sumatera) dan Brunei. Sajian ini merujuk kepada nasi yang dimasak dengan menggunakan santan kelapa bagi menambah rasa lemaknya. Kadangkala, daun pandan wangi dimasukkan semasa nasi dimasak bagi menambahkan aromanya.
tam	செயற்கை நுண்ணறிவு என்றால் என்ன?	பொதுவாக மனிதர்களால் செய்யப்படும் பணிகளைச் செய்ய ஒரு கணினி அல்லது ஒரு கணினியால் கட்டுப்படுத்தப்படும் ஒரு ரோபோவின் திறன் செயற்கை நுண்ணறிவு எனப்படும்.

Figure 9.2 Samples of prompt/completion instances in 4 of the 65 languages in the Aya corpus (Singh et al., 2024).

Building Instruction-Tuning Sets

Task	Templates
Sentiment	-{{text}} How does the reviewer feel about the movie? -The following movie review expresses what sentiment? {{text}} -{{text}} Did the reviewer enjoy the movie?
Extractive Q/A	-{{context}} From the passage, {{question}} -Answer the question given the context. Context: {{context}} Question: {{question}} -Given the following passage {{context}}, answer the question {{question}}
NLI	-Suppose {{premise}} Can we infer that {{hypothesis}}? Yes, no, or maybe? -{{premise}} Based on the previous passage, is it true that {{hypothesis}}? Yes, no, or maybe? -Given {{premise}} Should we assume that {{hypothesis}} is true? Yes, no, or maybe?

Figure 9.4 Instruction templates for sentiment, Q/A and NLI tasks.

Building Instruction-Tuning Sets

- Option 2: **convert existing supervised datasets**

Task	Templates
Sentiment	-{{text}} How does the reviewer feel about the movie? -The following movie review expresses what sentiment? {{text}} -{{text}} Did the reviewer enjoy the movie?
Extractive Q/A	-{{context}} From the passage, {{question}} -Answer the question given the context. Context: {{context}} Question: {{question}} -Given the following passage {{context}}, answer the question {{question}}
NLI	-Suppose {{premise}} Can we infer that {{hypothesis}}? Yes, no, or maybe? -{{premise}} Based on the previous passage, is it true that {{hypothesis}}? Yes, no, or maybe? -Given {{premise}} Should we assume that {{hypothesis}} is true? Yes, no, or maybe?

Figure 9.4 Instruction templates for sentiment, Q/A and NLI tasks.

Building Instruction-Tuning Sets

- Option 2: **convert existing supervised datasets**
- Automated, so much more **scalable**

Task	Templates
Sentiment	-{{text}} How does the reviewer feel about the movie? -The following movie review expresses what sentiment? {{text}} -{{text}} Did the reviewer enjoy the movie?
Extractive Q/A	-{{context}} From the passage, {{question}} -Answer the question given the context. Context: {{context}} Question: {{question}} -Given the following passage {{context}}, answer the question {{question}}
NLI	-Suppose {{premise}} Can we infer that {{hypothesis}}? Yes, no, or maybe? -{{premise}} Based on the previous passage, is it true that {{hypothesis}}? Yes, no, or maybe? -Given {{premise}} Should we assume that {{hypothesis}} is true? Yes, no, or maybe?

Figure 9.4 Instruction templates for sentiment, Q/A and NLI tasks.

Building Instruction-Tuning Sets

- Option 2: **convert existing supervised datasets**
 - Automated, so much more **scalable**
 - Write a **natural language template**, then fill-in the content automatically

Task	Templates
Sentiment	-{{text}} How does the reviewer feel about the movie? -The following movie review expresses what sentiment? {{text}} -{{text}} Did the reviewer enjoy the movie?
Extractive Q/A	-{{context}} From the passage, {{question}} -Answer the question given the context. Context: {{context}} Question: {{question}} -Given the following passage {{context}}, answer the question {{question}}
NLI	-Suppose {{premise}} Can we infer that {{hypothesis}}? Yes, no, or maybe? -{{premise}} Based on the previous passage, is it true that {{hypothesis}}? Yes, no, or maybe? -Given {{premise}} Should we assume that {{hypothesis}} is true? Yes,no, or maybe?

Figure 9.4 Instruction templates for sentiment, Q/A and NLI tasks.

Building Instruction-Tuning Sets

- Option 2: **convert existing supervised datasets**
 - Automated, so much more **scalable**
 - Write a **natural language template**, then fill-in the content automatically
 - **Repetitive prompts** can sometimes cause problems

Task	Templates
Sentiment	- <code>{{text}}</code> How does the reviewer feel about the movie? -The following movie review expresses what sentiment? <code>{{text}}</code> - <code>{{text}}</code> Did the reviewer enjoy the movie?
Extractive Q/A	- <code>{{context}}</code> From the passage, <code>{{question}}</code> -Answer the question given the context. Context: <code>{{context}}</code> Question: <code>{{question}}</code> -Given the following passage <code>{{context}}</code> , answer the question <code>{{question}}</code>
NLI	-Suppose <code>{{premise}}</code> Can we infer that <code>{{hypothesis}}</code> ? Yes, no, or maybe? - <code>{{premise}}</code> Based on the previous passage, is it true that <code>{{hypothesis}}</code> ? Yes, no, or maybe? -Given <code>{{premise}}</code> Should we assume that <code>{{hypothesis}}</code> is true? Yes,no, or maybe?

Figure 9.4 Instruction templates for sentiment, Q/A and NLI tasks.

Building Instruction-Tuning Sets

- Option 2: **convert existing supervised datasets**
 - Automated, so much more **scalable**
 - Write a **natural language template**, then fill-in the content automatically
 - **Repetitive prompts** can sometimes cause problems
- Example: **FLAN**

Task	Templates
Sentiment	- <code>{{text}}</code> How does the reviewer feel about the movie? -The following movie review expresses what sentiment? <code>{{text}}</code> - <code>{{text}}</code> Did the reviewer enjoy the movie?
Extractive Q/A	- <code>{{context}}</code> From the passage, <code>{{question}}</code> -Answer the question given the context. Context: <code>{{context}}</code> Question: <code>{{question}}</code> -Given the following passage <code>{{context}}</code> , answer the question <code>{{question}}</code>
NLI	-Suppose <code>{{premise}}</code> Can we infer that <code>{{hypothesis}}</code> ? Yes, no, or maybe? - <code>{{premise}}</code> Based on the previous passage, is it true that <code>{{hypothesis}}</code> ? Yes, no, or maybe? -Given <code>{{premise}}</code> Should we assume that <code>{{hypothesis}}</code> is true? Yes, no, or maybe?

Figure 9.4 Instruction templates for sentiment, Q/A and NLI tasks.

Building Instruction-Tuning Sets

- Option 2: **convert existing supervised datasets**
 - Automated, so much more **scalable**
 - Write a **natural language template**, then fill-in the content automatically
 - **Repetitive prompts** can sometimes cause problems
- Example: **FLAN**
 - 60 datasets converted via templating

Task	Templates
Sentiment	- <code>{{text}}</code> How does the reviewer feel about the movie? -The following movie review expresses what sentiment? <code>{{text}}</code> - <code>{{text}}</code> Did the reviewer enjoy the movie?
Extractive Q/A	- <code>{{context}}</code> From the passage, <code>{{question}}</code> -Answer the question given the context. Context: <code>{{context}}</code> Question: <code>{{question}}</code> -Given the following passage <code>{{context}}</code> , answer the question <code>{{question}}</code>
NLI	-Suppose <code>{{premise}}</code> Can we infer that <code>{{hypothesis}}</code> ? Yes, no, or maybe? - <code>{{premise}}</code> Based on the previous passage, is it true that <code>{{hypothesis}}</code> ? Yes, no, or maybe? -Given <code>{{premise}}</code> Should we assume that <code>{{hypothesis}}</code> is true? Yes, no, or maybe?

Figure 9.4 Instruction templates for sentiment, Q/A and NLI tasks.

Templating Detail

Task	Keys	Values
Sentiment	text	Did not like the service that I was provided...
	label	0
Sentiment	text	It sounds like a great plot, the actors are first grade, and...
	label	1
NLI	premise	No weapons of mass destruction found in Iraq yet.
	hypothesis	Weapons of mass destruction found in Iraq.
NLI	label	2
	premise	Jimmy Smith... played college football at University of Colorado.
NLI	hypothesis	The University of Colorado has a college football team.
	label	0
Extractive Q/A	context	Beyoncé Giselle Knowles-Carter is an American singer...
	question	When did Beyoncé start becoming popular?
	answers	{ text: ['in the late 1990s'], answer_start: 269 }

Figure 9.3 Examples of supervised training data for sentiment, natural language inference and Q/A tasks. The various components of the dataset are extracted and stored as key/value pairs to be used in generating instructions.

Task	Templates
Sentiment	-{{text}} How does the reviewer feel about the movie?
	-The following movie review expresses what sentiment? {{text}}
	-{{text}} Did the reviewer enjoy the movie?
Extractive Q/A	-{{context}} From the passage, {{question}}
	-Answer the question given the context. Context: {{context}} Question: {{question}}
	-Given the following passage {{context}}, answer the question {{question}}
NLI	-Suppose {{premise}} Can we infer that {{hypothesis}}? Yes, no, or maybe?
	-{{premise}} Based on the previous passage, is it true that {{hypothesis}}? Yes, no, or maybe?
	-Given {{premise}} Should we assume that {{hypothesis}} is true? Yes, no, or maybe?

Figure 9.4 Instruction templates for sentiment, Q/A and NLI tasks.

Building Instruction-Tuning Sets

Sample Extended Instruction

- **Definition:** This task involves creating answers to complex questions, from a given passage. Answering these questions, typically involve understanding multiple sentences. Make sure that your answer has the same type as the "answer type" mentioned in input. The provided "answer type" can be of any of the following types: "span", "date", "number". A "span" answer is a continuous phrase taken directly from the passage or question. You can directly copy-paste the text from the passage or the question for span type answers. If you find multiple spans, please add them all as a comma separated list. Please restrict each span to five words. A "number" type answer can include a digit specifying an actual value. For "date" type answers, use DD MM YYYY format e.g. 11 Jan 1992. If full date is not available in the passage you can write partial date such as 1992 or Jan 1992.
- **Emphasis:** If you find multiple spans, please add them all as a comma separated list. Please restrict each span to five words.
- **Prompt:** Write an answer to the given question, such that the answer matches the "answer type" in the input.
Passage: { passage }
Question: { question }

Figure 9.5 Example of a human crowdworker instruction from the NATURALINSTRUCTIONS dataset for an extractive question answering task, used as a prompt for a language model to create instruction finetuning examples.

Building Instruction-Tuning Sets

- Option 3: leverage LLMs

Sample Extended Instruction

- **Definition:** This task involves creating answers to complex questions, from a given passage. Answering these questions, typically involve understanding multiple sentences. Make sure that your answer has the same type as the "answer type" mentioned in input. The provided "answer type" can be of any of the following types: "span", "date", "number". A "span" answer is a continuous phrase taken directly from the passage or question. You can directly copy-paste the text from the passage or the question for span type answers. If you find multiple spans, please add them all as a comma separated list. Please restrict each span to five words. A "number" type answer can include a digit specifying an actual value. For "date" type answers, use DD MM YYYY format e.g. 11 Jan 1992. If full date is not available in the passage you can write partial date such as 1992 or Jan 1992.
- **Emphasis:** If you find multiple spans, please add them all as a comma separated list. Please restrict each span to five words.
- **Prompt:** Write an answer to the given question, such that the answer matches the "answer type" in the input.
Passage: { passage }
Question: { question }

Figure 9.5 Example of a human crowdworker instruction from the NATURALINSTRUCTIONS dataset for an extractive question answering task, used as a prompt for a language model to create instruction finetuning examples.

Building Instruction-Tuning Sets

- Option 3: leverage LLMs
 - (Wasn't an option before LLMs)

Sample Extended Instruction

- **Definition:** This task involves creating answers to complex questions, from a given passage. Answering these questions, typically involve understanding multiple sentences. Make sure that your answer has the same type as the "answer type" mentioned in input. The provided "answer type" can be of any of the following types: "span", "date", "number". A "span" answer is a continuous phrase taken directly from the passage or question. You can directly copy-paste the text from the passage or the question for span type answers. If you find multiple spans, please add them all as a comma separated list. Please restrict each span to five words. A "number" type answer can include a digit specifying an actual value. For "date" type answers, use DD MM YYYY format e.g. 11 Jan 1992. If full date is not available in the passage you can write partial date such as 1992 or Jan 1992.
- **Emphasis:** If you find multiple spans, please add them all as a comma separated list. Please restrict each span to five words.
- **Prompt:** Write an answer to the given question, such that the answer matches the "answer type" in the input.
Passage: { passage }
Question: { question }

Figure 9.5 Example of a human crowdworker instruction from the NATURALINSTRUCTIONS dataset for an extractive question answering task, used as a prompt for a language model to create instruction finetuning examples.

Building Instruction-Tuning Sets

- Option 3: leverage LLMs
 - (Wasn't an option before LLMs)
 - Can use **original instructions for human annotators**, and have LLM generate annotated examples

Sample Extended Instruction

- **Definition:** This task involves creating answers to complex questions, from a given passage. Answering these questions, typically involve understanding multiple sentences. Make sure that your answer has the same type as the "answer type" mentioned in input. The provided "answer type" can be of any of the following types: "span", "date", "number". A "span" answer is a continuous phrase taken directly from the passage or question. You can directly copy-paste the text from the passage or the question for span type answers. If you find multiple spans, please add them all as a comma separated list. Please restrict each span to five words. A "number" type answer can include a digit specifying an actual value. For "date" type answers, use DD MM YYYY format e.g. 11 Jan 1992. If full date is not available in the passage you can write partial date such as 1992 or Jan 1992.
- **Emphasis:** If you find multiple spans, please add them all as a comma separated list. Please restrict each span to five words.
- **Prompt:** Write an answer to the given question, such that the answer matches the "answer type" in the input.
Passage: { passage }
Question: { question }

Figure 9.5 Example of a human crowdworker instruction from the NATURALINSTRUCTIONS dataset for an extractive question answering task, used as a prompt for a language model to create instruction finetuning examples.

Building Instruction-Tuning Sets

- Option 3: leverage LLMs
 - (Wasn't an option before LLMs)
 - Can use **original instructions for human annotators**, and have LLM generate annotated examples
 - Can also use to **generate paraphrases** of existing examples/templates (helps avoid repetitive examples)

Sample Extended Instruction

- **Definition:** This task involves creating answers to complex questions, from a given passage. Answering these questions, typically involve understanding multiple sentences. Make sure that your answer has the same type as the "answer type" mentioned in input. The provided "answer type" can be of any of the following types: "span", "date", "number". A "span" answer is a continuous phrase taken directly from the passage or question. You can directly copy-paste the text from the passage or the question for span type answers. If you find multiple spans, please add them all as a comma separated list. Please restrict each span to five words. A "number" type answer can include a digit specifying an actual value. For "date" type answers, use DD MM YYYY format e.g. 11 Jan 1992. If full date is not available in the passage you can write partial date such as 1992 or Jan 1992.
- **Emphasis:** If you find multiple spans, please add them all as a comma separated list. Please restrict each span to five words.
- **Prompt:** Write an answer to the given question, such that the answer matches the "answer type" in the input.
Passage: { passage }
Question: { question }

Figure 9.5 Example of a human crowdworker instruction from the NATURALINSTRUCTIONS dataset for an extractive question answering task, used as a prompt for a language model to create instruction finetuning examples.

Task Inventories



Figure 2: Compared to other datasets, SUP-NATINST covers a more diverse range of task types. InstructGPT reports a very coarse categorization of their task types. Bubble size represents the number of tasks of each type in log scale.

Adding Exemplars and CoT

Without chain-of-thought

With chain-of-thought

Instruction
without
exemplars

Answer the following
yes/no question.

→ yes

Can you write a whole
Haiku in a single tweet?

Answer the following yes/no question
by reasoning step-by-step.

→

Can you write a whole Haiku in a
single tweet?

A haiku is a japanese
three-line poem.
That is short enough
to fit in 280
characters. The
answer is yes.

Instruction
with exemplars

Q: Answer the following
yes/no question.
Could a dandelion suffer
from hepatitis?

A: no

→ yes

Q: Answer the following
yes/no question.
Can you write a whole Haiku
in a single tweet?
A:

Q: Answer the following yes/no question by
reasoning step-by-step.

Could a dandelion suffer from hepatitis?

A: Hepatitis only affects organisms with livers.
Dandelions don't have a liver. The answer is no.

→

Q: Answer the following yes/no question by
reasoning step-by-step.
Can you write a whole Haiku in a single tweet?
A:

A haiku is a japanese
three-line poem.
That is short enough
to fit in 280
characters. The
answer is yes.

Evaluating Instruction-Tuning

Finetune on many tasks (“instruction-tuning”)

<u>Input (Commonsense Reasoning)</u>	<u>Input (Translation)</u>
Here is a goal: Get a cool sleep on summer days. How would you accomplish this goal? OPTIONS: <input type="radio"/> -Keep stack of pillow cases in fridge. <input type="radio"/> -Keep stack of pillow cases in oven.	Translate this sentence to Spanish: The new office building was built in less than three months.
<u>Target</u> keep stack of pillow cases in fridge	<u>Target</u> El nuevo edificio de oficinas se construyó en tres meses.

Sentiment analysis tasks
Coreference resolution tasks
...

Inference on unseen task type

→

<u>Input (Natural Language Inference)</u>
Premise: At my age you will probably have learnt one lesson. Hypothesis: It's not certain how many lessons you'll learn by your thirties. Does the premise entail the hypothesis? OPTIONS: <input type="radio"/> -yes <input type="radio"/> -it is not possible to tell <input type="radio"/> -no
<u>FLAN Response</u> It is not possible to tell

Evaluating Instruction-Tuning

- Instruction-tuned models emphasize **generalization to unseen tasks**

Finetune on many tasks (“instruction-tuning”)

<u>Input (Commonsense Reasoning)</u>	<u>Input (Translation)</u>
Here is a goal: Get a cool sleep on summer days. How would you accomplish this goal? OPTIONS: <input type="checkbox"/> -Keep stack of pillow cases in fridge. <input type="checkbox"/> -Keep stack of pillow cases in oven.	Translate this sentence to Spanish: The new office building was built in less than three months.
<u>Target</u> keep stack of pillow cases in fridge	<u>Target</u> El nuevo edificio de oficinas se construyó en tres meses.

Sentiment analysis tasks
Coreference resolution tasks
...

Inference on unseen task type

→

<u>Input (Natural Language Inference)</u>
Premise: At my age you will probably have learnt one lesson. Hypothesis: It's not certain how many lessons you'll learn by your thirties. Does the premise entail the hypothesis? OPTIONS: <input type="checkbox"/> -yes <input type="checkbox"/> -it is not possible to tell <input type="checkbox"/> -no
<u>FLAN Response</u> It is not possible to tell

Evaluating Instruction-Tuning

- Instruction-tuned models emphasize **generalization to unseen tasks**
- Often evaluated by "**leave one out**"
 - Train on everything **except** NLI
 - Whole **task types** are left out, since there may be more than one NLI set, e.g.

Finetune on many tasks ("instruction-tuning")

Input (Commonsense Reasoning)

Here is a goal: Get a cool sleep on summer days.
How would you accomplish this goal?
OPTIONS:
-Keep stack of pillow cases in fridge.
-Keep stack of pillow cases in oven.

Target

keep stack of pillow cases in fridge

Input (Translation)

Translate this sentence to Spanish:
The new office building was built in less than three months.

Target

El nuevo edificio de oficinas se construyó en tres meses.

Sentiment analysis tasks

Coreference resolution tasks

...

Inference on unseen task type

Input (Natural Language Inference)

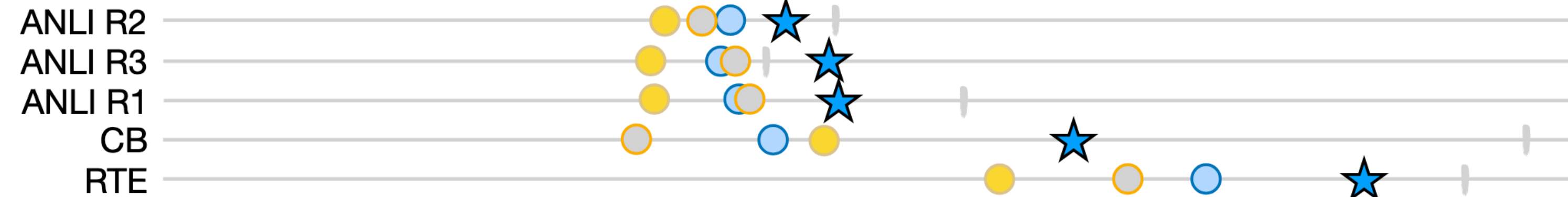
Premise: At my age you will probably have learnt one lesson.
Hypothesis: It's not certain how many lessons you'll learn by your thirties.
Does the premise entail the hypothesis?
OPTIONS:
-yes -it is not possible to tell -no

FLAN Response

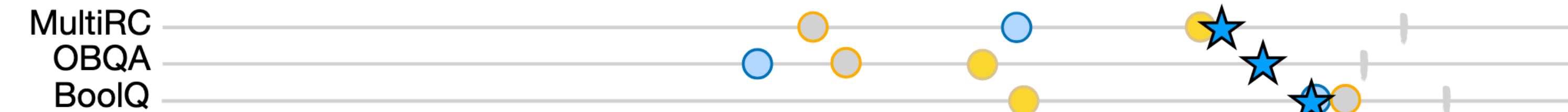
It is not possible to tell

FLAN Improvements

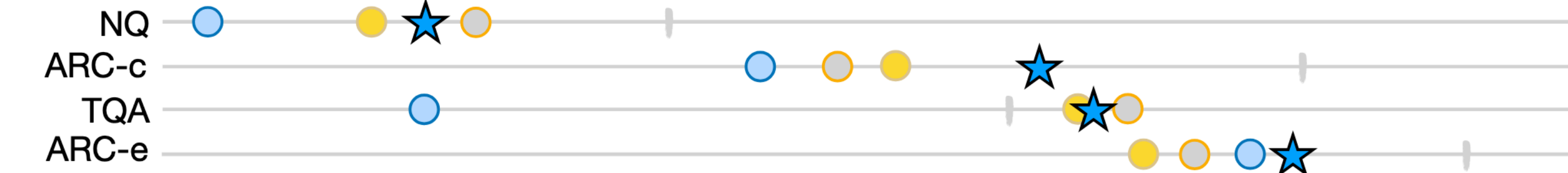
Natural language inference



Reading comprehension



Closed-book QA

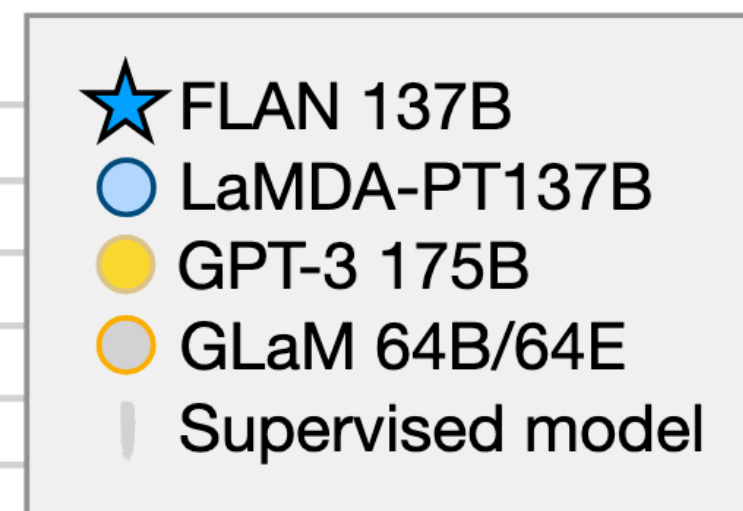


Translation



0 20 40 60 80 100

Zero-shot performance



Compute Cost

Params	Model	Architecture	Pre-training Objective	Pre-train FLOPs	Finetune FLOPs	% Finetune Compute
80M	Flan-T5-Small	encoder-decoder	span corruption	1.8E+20	2.9E+18	1.6%
250M	Flan-T5-Base	encoder-decoder	span corruption	6.6E+20	9.1E+18	1.4%
780M	Flan-T5-Large	encoder-decoder	span corruption	2.3E+21	2.4E+19	1.1%
3B	Flan-T5-XL	encoder-decoder	span corruption	9.0E+21	5.6E+19	0.6%
11B	Flan-T5-XXL	encoder-decoder	span corruption	3.3E+22	7.6E+19	0.2%
8B	Flan-PaLM	decoder-only	causal LM	3.7E+22	1.6E+20	0.4%
62B	Flan-PaLM	decoder-only	causal LM	2.9E+23	1.2E+21	0.4%
540B	Flan-PaLM	decoder-only	causal LM	2.5E+24	5.6E+21	0.2%
62B	Flan-cont-PaLM	decoder-only	causal LM	4.8E+23	1.8E+21	0.4%
540B	Flan-U-PaLM	decoder-only	prefix LM + span corruption	2.5E+23	5.6E+21	0.2%

Compute Cost

Params	Model	Architecture	Pre-training Objective	Pre-train FLOPs	Finetune FLOPs	% Finetune Compute
80M	Flan-T5-Small	encoder-decoder	span corruption	1.8E+20	2.9E+18	1.6%
250M	Flan-T5-Base	encoder-decoder	span corruption	6.6E+20	9.1E+18	1.4%
780M	Flan-T5-Large	encoder-decoder	span corruption	2.3E+21	2.4E+19	1.1%
3B	Flan-T5-XL	encoder-decoder	span corruption	9.0E+21	5.6E+19	0.6%
11B	Flan-T5-XXL	encoder-decoder	span corruption	3.3E+22	7.6E+19	0.2%
8B	Flan-PaLM	decoder-only	causal LM	3.7E+22	1.6E+20	0.4%
62B	Flan-PaLM	decoder-only	causal LM	2.9E+23	1.2E+21	0.4%
540B	Flan-PaLM	decoder-only	causal LM	2.5E+24	5.6E+21	0.2%
62B	Flan-cont-PaLM	decoder-only	causal LM	4.8E+23	1.8E+21	0.4%
540B	Flan-U-PaLM	decoder-only	prefix LM + span corruption	2.5E+23	5.6E+21	0.2%

Upshot: Instruction Tuning is **relatively cheap in compute**

Prompting Expanded

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ❌

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✅

Prompting Expanded

- Instruction tuning **enhances models' abilities to follow prompts**

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ❌

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✅

Prompting Expanded

- Instruction tuning **enhances models' abilities to follow prompts**
- Goes **beyond "in-context learning"** (e.g. GPT-3). Model is explicitly trained on prompts

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ❌

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✓

Prompting Expanded

- Instruction tuning **enhances models' abilities to follow prompts**
 - Goes **beyond "in-context learning"** (e.g. GPT-3). Model is explicitly trained on prompts
 - This allows for the creation of **more elaborate prompts**

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ❌

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✅

Prompting Expanded

- Instruction tuning **enhances models' abilities to follow prompts**
 - Goes **beyond "in-context learning"** (e.g. GPT-3). Model is explicitly trained on prompts
 - This allows for the creation of **more elaborate prompts**
 - Already seen examples of **Chain-of-Thought** prompting, which has proven powerful

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ❌

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✅

System Prompts

Claude should give concise responses to very simple questions, but provide thorough responses to complex and open-ended questions.

Claude is able to explain difficult concepts or ideas clearly. It can also illustrate its explanations with examples, thought experiments, or metaphors.

Claude does not provide information that could be used to make chemical or biological or nuclear weapons

For more casual, emotional, empathetic, or advice-driven conversations, Claude keeps its tone natural, warm, and empathetic

Claude cares about people's well-being and avoids encouraging or facilitating self-destructive behavior

If Claude provides bullet points in its response, it should use markdown, and each bullet point should be at least 1-2 sentences long unless the human requests otherwise

System Prompts

- Chatbots like ChatGPT and Claude usually have elaborate **system prompts**

Claude should give concise responses to very simple questions, but provide thorough responses to complex and open-ended questions.

Claude is able to explain difficult concepts or ideas clearly. It can also illustrate its explanations with examples, thought experiments, or metaphors.

Claude does not provide information that could be used to make chemical or biological or nuclear weapons

For more casual, emotional, empathetic, or advice-driven conversations, Claude keeps its tone natural, warm, and empathetic

Claude cares about people's well-being and avoids encouraging or facilitating self-destructive behavior

If Claude provides bullet points in its response, it should use markdown, and each bullet point should be at least 1-2 sentences long unless the human requests otherwise

System Prompts

- Chatbots like ChatGPT and Claude usually have elaborate **system prompts**
- Prompts that are **unseen by the user**, but characterize **desired model behavior**

Claude should give concise responses to very simple questions, but provide thorough responses to complex and open-ended questions.

Claude is able to explain difficult concepts or ideas clearly. It can also illustrate its explanations with examples, thought experiments, or metaphors.

Claude does not provide information that could be used to make chemical or biological or nuclear weapons

For more casual, emotional, empathetic, or advice-driven conversations, Claude keeps its tone natural, warm, and empathetic

Claude cares about people's well-being and avoids encouraging or facilitating self-destructive behavior

If Claude provides bullet points in its response, it should use markdown, and each bullet point should be at least 1-2 sentences long unless the human requests otherwise

System Prompts

- Chatbots like ChatGPT and Claude usually have elaborate **system prompts**
- Prompts that are **unseen by the user**, but characterize **desired model behavior**
- These can control things like **style, brevity, persona, "safety", etc.**

Claude should give concise responses to very simple questions, but provide thorough responses to complex and open-ended questions.

Claude is able to explain difficult concepts or ideas clearly. It can also illustrate its explanations with examples, thought experiments, or metaphors.

Claude does not provide information that could be used to make chemical or biological or nuclear weapons

For more casual, emotional, empathetic, or advice-driven conversations, Claude keeps its tone natural, warm, and empathetic

Claude cares about people's well-being and avoids encouraging or facilitating self-destructive behavior

If Claude provides bullet points in its response, it should use markdown, and each bullet point should be at least 1-2 sentences long unless the human requests otherwise

System Prompts

- Chatbots like ChatGPT and Claude usually have elaborate **system prompts**
- Prompts that are **unseen by the user**, but characterize **desired model behavior**
- These can control things like **style, brevity, persona, "safety", etc.**
- Included here: Claude's system prompt

Claude should give concise responses to very simple questions, but provide thorough responses to complex and open-ended questions.

Claude is able to explain difficult concepts or ideas clearly. It can also illustrate its explanations with examples, thought experiments, or metaphors.

Claude does not provide information that could be used to make chemical or biological or nuclear weapons

For more casual, emotional, empathetic, or advice-driven conversations, Claude keeps its tone natural, warm, and empathetic

Claude cares about people's well-being and avoids encouraging or facilitating self-destructive behavior

If Claude provides bullet points in its response, it should use markdown, and each bullet point should be at least 1-2 sentences long unless the human requests otherwise

Next Time

- LLMs continued
- Learning from **Human-Preferences**
 - Reinforcement Learning generally
 - Crowd-sourcing preferences
 - DPO
- Test-time Compute Scaling
- Ethical considerations