In-context Learning

CSCI 601 471/671
NLP: Self-Supervised Models

https://self-supervised.cs.jhu.edu/sp2023/

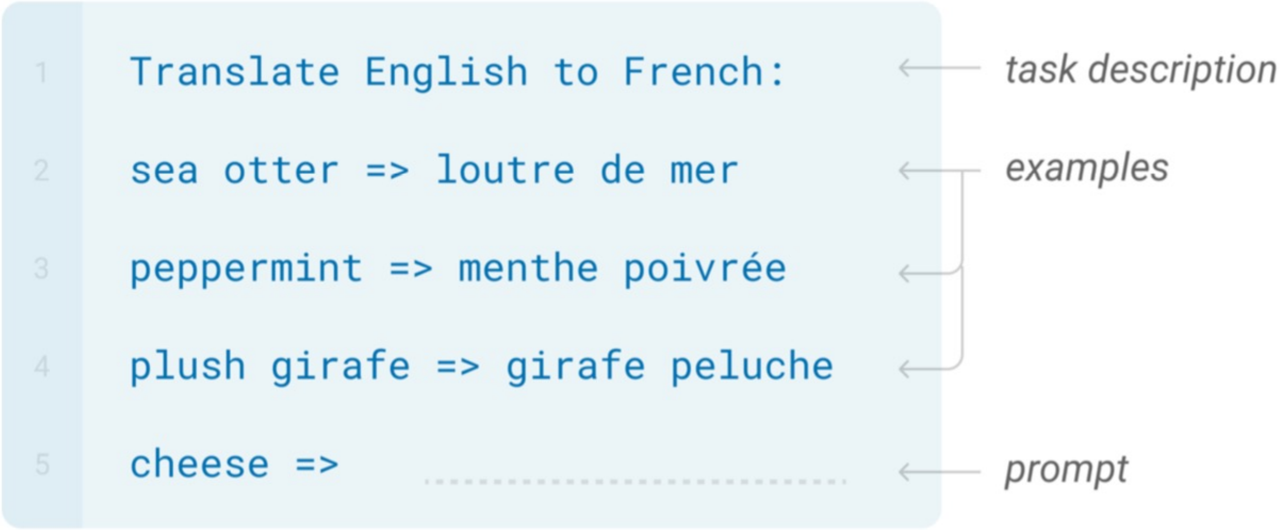
JOHNS HOPKINS UNIVERSITY

[Slide credit: Iz Beltagy, Arman Cohan, Robert Logan IV, Sewon Min, Sameer Singh, and many others]

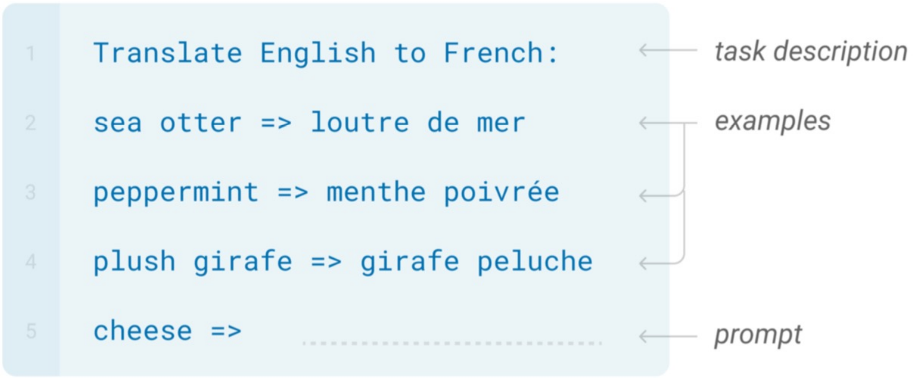
Logistics

- HW6: The Struggles … ☹
- HW7 will be the last homework! ☺
- Projects
  - Do you have an idea yet?
  - Do you have teammate(s) already?
  - Have you looked at the deck of ideas I shared yet?
- Project proposal deadline: Thu Mar 16
  - Should we push it back? Alternative: Tue Mar 28

In-Context Learning



In-Context Learning



- Learns to do a downstream task by conditioning on input-output examples!

- **No weight update** — our model is not **explicitly pre-trained** to learn from examples
    - The underlying models are quite general

- Today's focus:
    - How to use effectively in practice?
    - Fundamentally, why does it work?

Why Do We Care About Few-Shot Learning?

Practically Useful

Intellectually Intriguing

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

Practically Useful

Labeling data is costly

- May require domain expertise
  - Medical, legal, financial
- Inputs may be long/complex
  - Grammaticality
- Outputs may be complex
  - Semantic parsing

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh; quote credit: Colin Raffel]

Practically Useful

Labeling data is costly

You want to do best with what you have

- You don't want to get more data
- Emergent, time-sensitive scenarios
  - Something new happened
  - Need to react quickly!

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

Practically Useful

Labeling data is costly

You want to do best with what you have

Finetuning can unstable

- Training is sensitive to hyperparams
- Not enough validation data
- We don’t quite understand how finetuning works

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

Practically Useful

Labeling data is costly

You want to do best with what you have

Finetuning can be unstable

Finetuning large LMs is expensive

- Expensive to train, time and memory

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

Intellectually Intriguing

Potential test for "Intelligent Behavior"

- Generalization from few examples
  - Fundamental piece of intelligence
  - Often used in psychology
  - Quickly adjust to environment

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

Intellectually Intriguing

Potential test for "Intelligent Behavior"

"But… Deep learning is data hungry!"

- Long-standing criticism of DL
- Understand why it doesn't work here
  - Or does it?
- What are the new limitations of DL?

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

Intellectually Intriguing

Potential test for "Intelligent Behavior"

"But… Deep learning is data hungry!"

Insights into Language Modeling

- What does an LLM "know"?
- What are the biases/limitations of LLMs?
- …

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

Intellectually Intriguing

Potential test for "Intelligent Behavior"

"But… Deep learning is data hungry!"

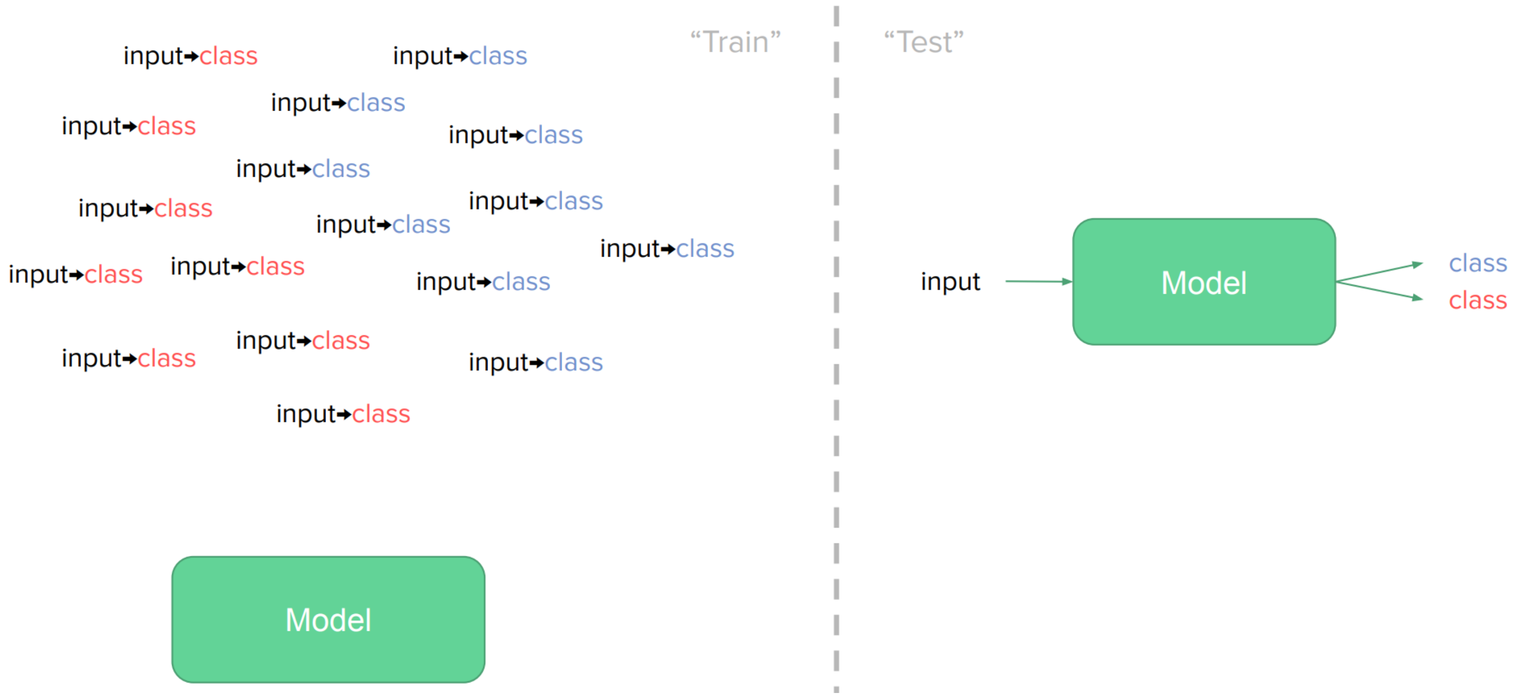
Insights into Language Modeling

Because LARGE language models

- Training/inference/access is tough
- What else can we do? ;)

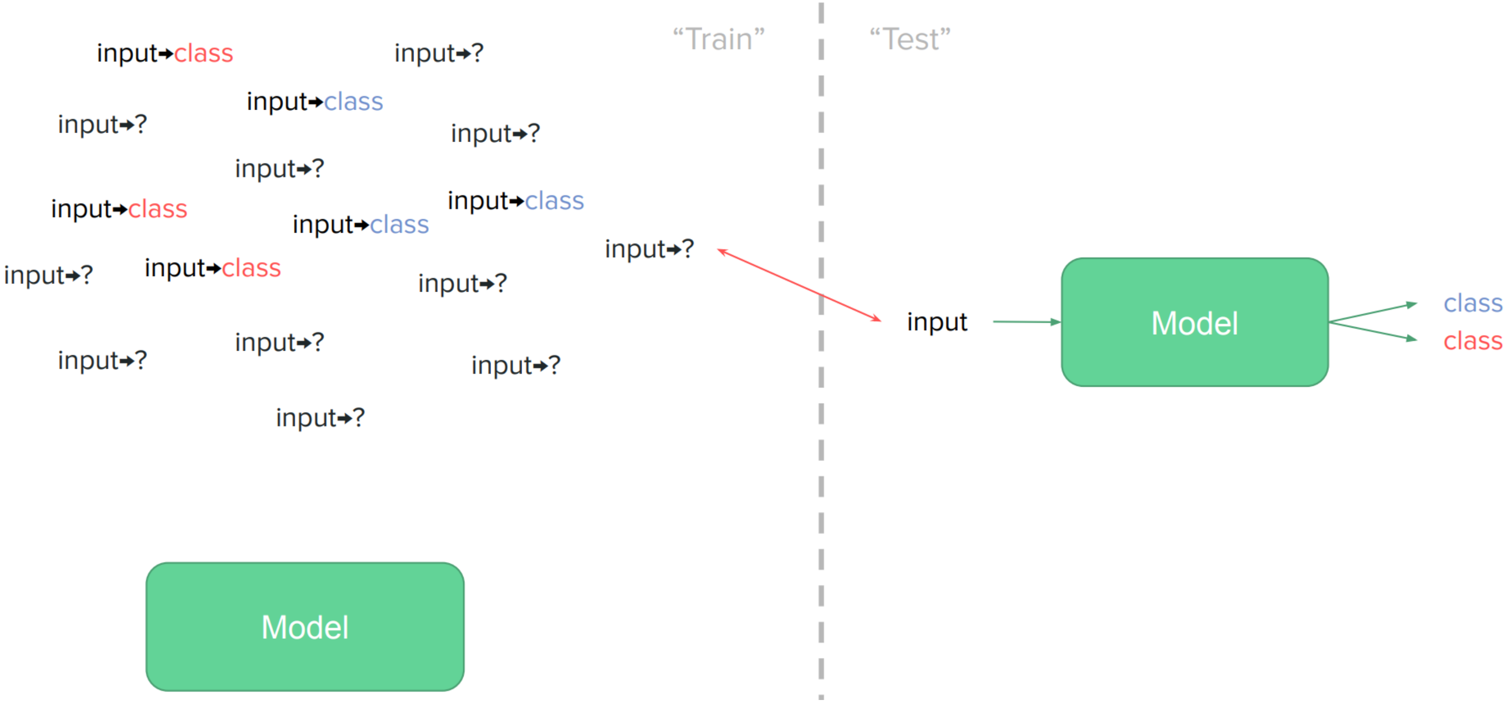
[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

The Broader Context: Supervised Learning



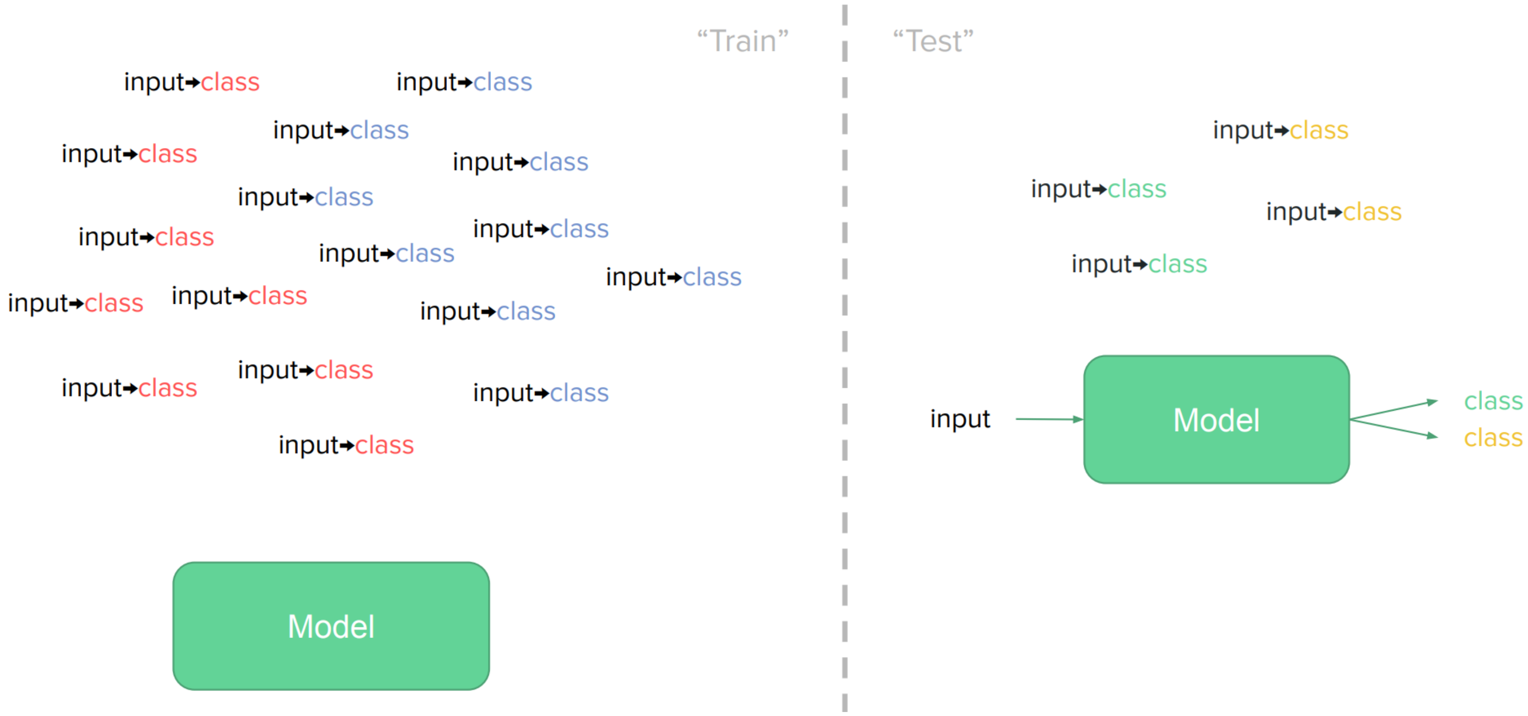
[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

The Broader Context: Semi-Supervised Learning



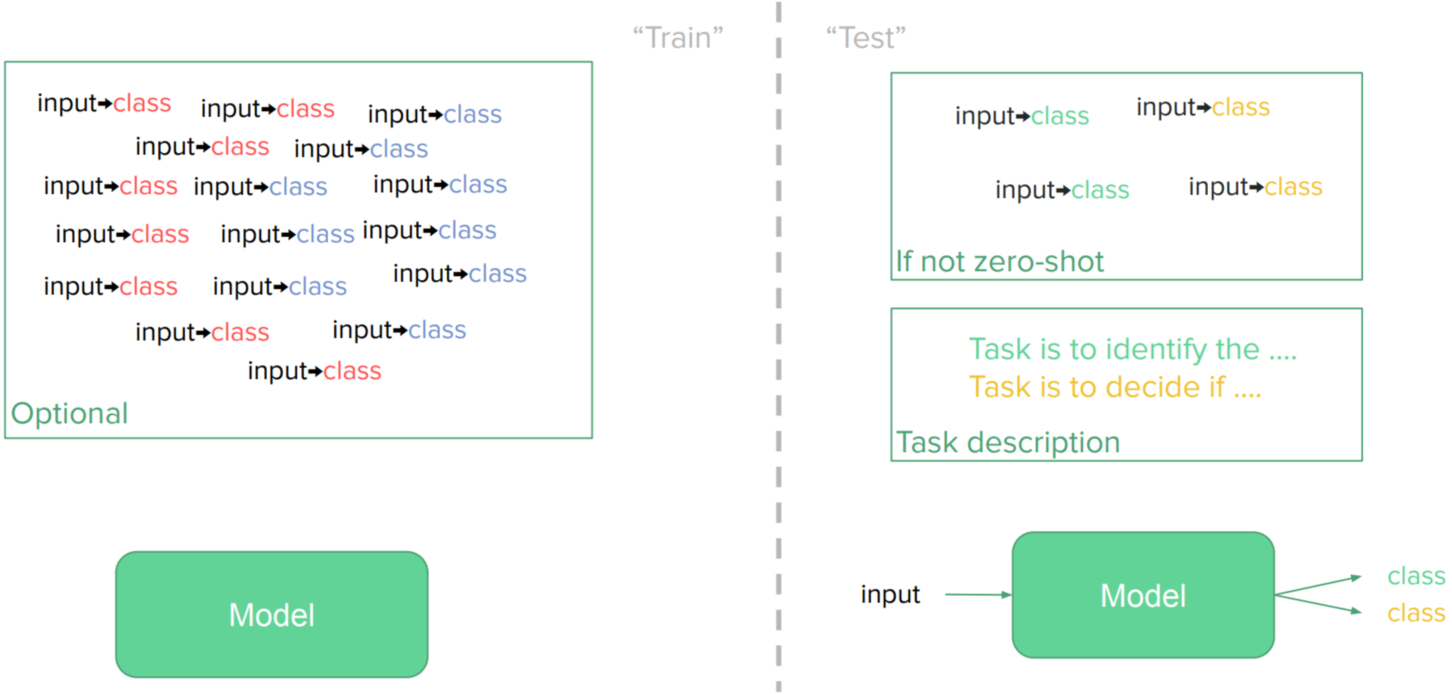
[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

The Broader Context: [Traditional] Few-shot Learning



[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

The Broader Context: [Modern] Few-shot Learning



[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

In-Context Prompting

Movie review dataset

Input: An effortlessly accomplished and richly resonant work.
Label: positive

Input: A mostly tired retread of several other mob tales.
Label: negative

An effortlessly accomplished and richly resonant work. It was great! A mostly tired retread of several other mob tales. It was terrible!

A three-hour cinema master class. It was ____________

Language Model

p1 = P(It was great! | 1st train input+output \n 2nd train input+output \n A three-hour cinema master class.)

p2 = P(It was terrible! | 1st train input+output \n 2nd train input+output \n A three-hour cinema master class.)

p1>p2 "positive"

p1<p2 "negative"

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

LM Prompting: Choices of Encoding

Prompt

Input: Subpar acting. Sentiment: negative

Input: Beautiful film. Sentiment: positive

Input: Amazing. Sentiment:

[Slide credit: Eric Wallace]

LM Prompting: Choices of Encoding

Prompt

Input: Subpar acting. Sentiment: negative

In

In

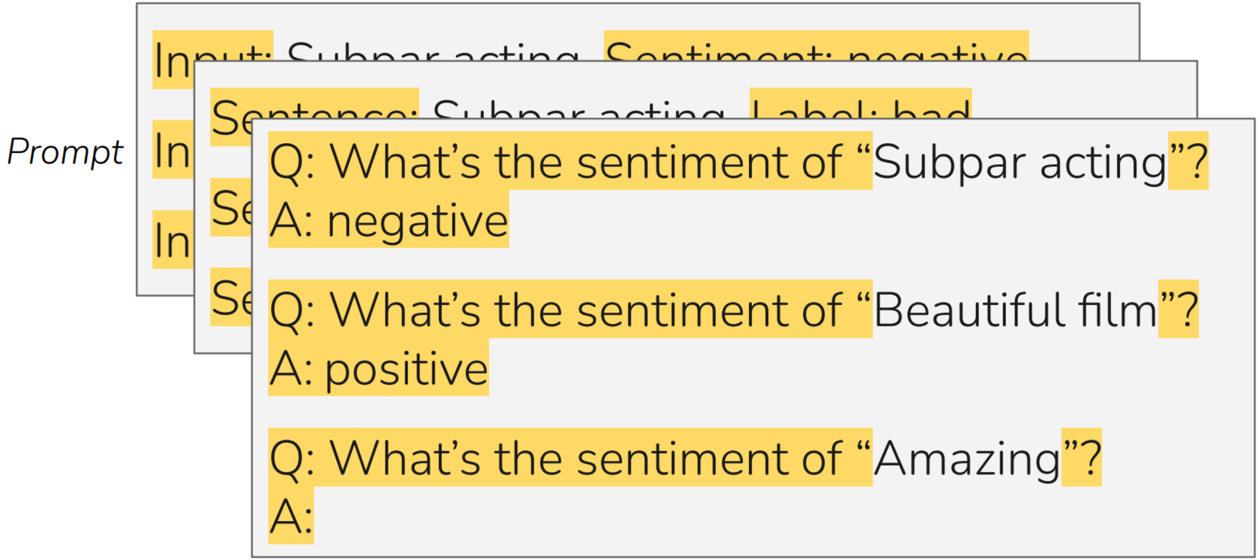
Sentence: Subpar acting. Label: bad

Sentence: Beautiful film. Label: good

Sentence: Amazing. Label:

[Slide credit: Eric Wallace]

LM Prompting: Choices of Encoding



LM Prompting: Choices of Encoding

- **Pattern:** A function that encodes the inputs.
- **Verbalizer:** A function that encodes the output.

Pattern: f(<x>) = "Input: <x>"
Verbalizer: v(<x>)) = "Label: <x>)"

Input: Subpar acting. Sentiment: negative

Input: Beautiful film. Sentiment: positive

Input: Amazing. Sentiment:

Pattern: f(<x>) = "Q: What is the sentiment of <x>"
Verbalizer: v(<x>) = "A: <x>"

Q: What's the sentiment of "Subpar acting"?
A: negative

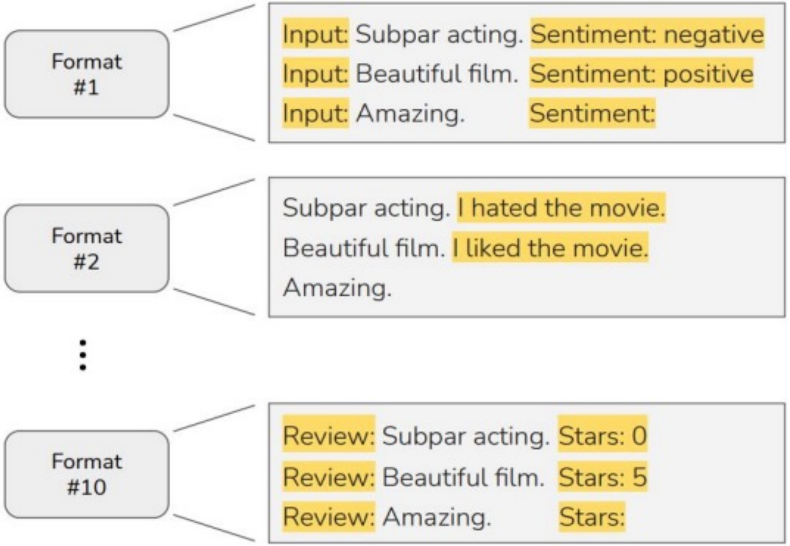
Q: What's the sentiment of "Beautiful film"?
A: positive

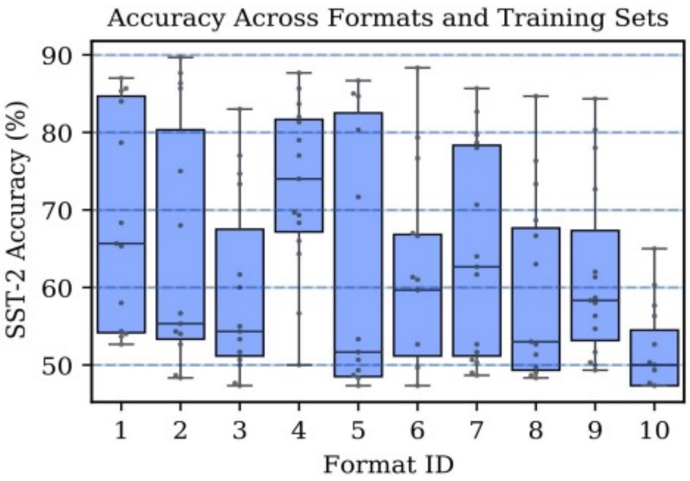
Q: What's the sentiment of "Amazing"?
A:

[Slide credit: Eric Wallace]

[Slide credit: Eric Wallace]

In-Context Learning: Sensitivity to Encoding





In-context learning is highly sensitive to prompt format (training sets and patterns/verbalizers)

[“Calibrate Before Use: Improving Few-Shot Performance of Language Models.” Zhao et al. 2021]

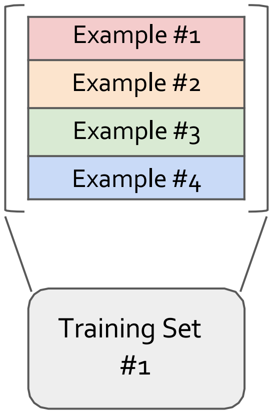
[Slide credit: Eric Wallace]

In-Context Learning: Sensitivity to Demo. Permutations

Training Set
#1

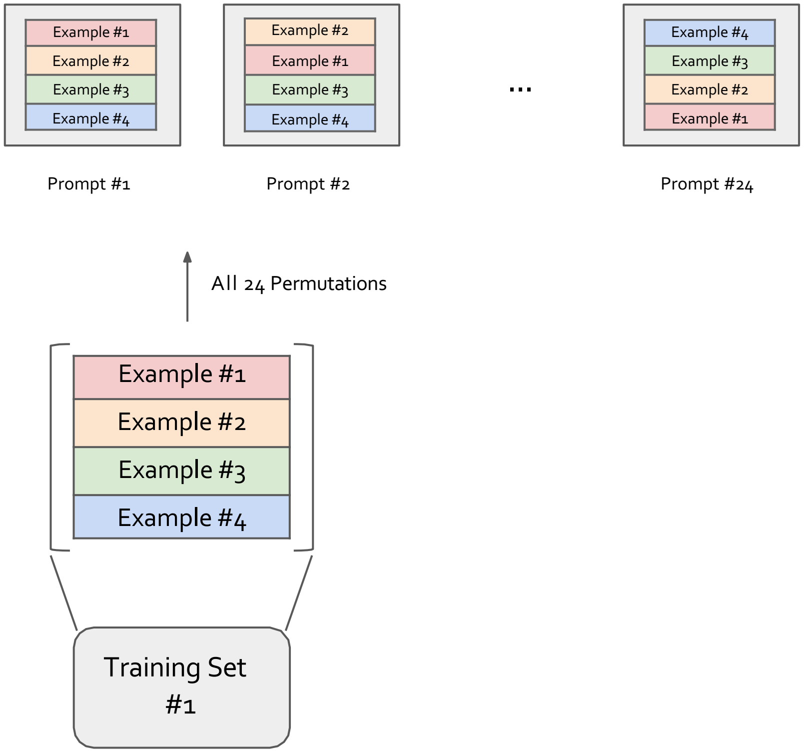
[Slide credit: Eric Wallace]

In-Context Learning: Sensitivity to Demo. Permutations

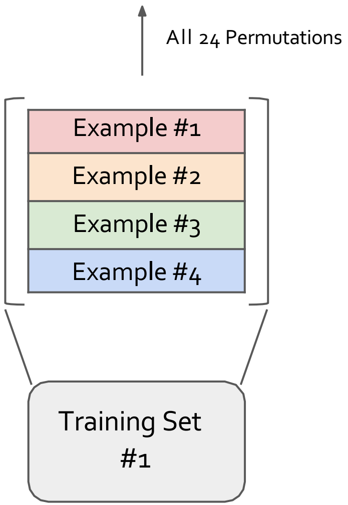


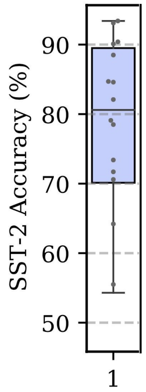
[Slide credit: Eric Wallace]

In-Context Learning: Sensitivity to Demo. Permutations



In-Context Learning: Sensitivity to Demo. Permutations



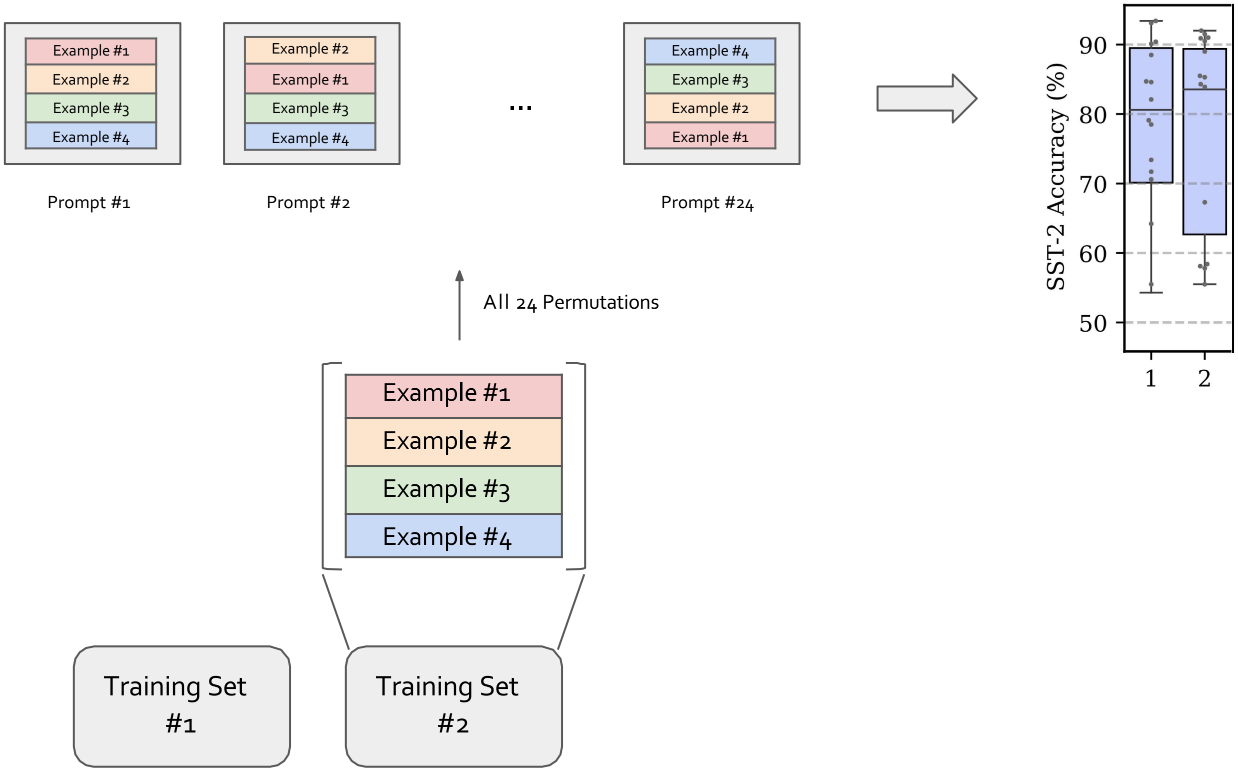


In-context learning is highly sensitive to prompt format (training sets and patterns/verbalizers)

[“Calibrate Before Use: Improving Few-Shot Performance of Language Models.” Zhao et al. 2021]

[Slide credit: Eric Wallace]

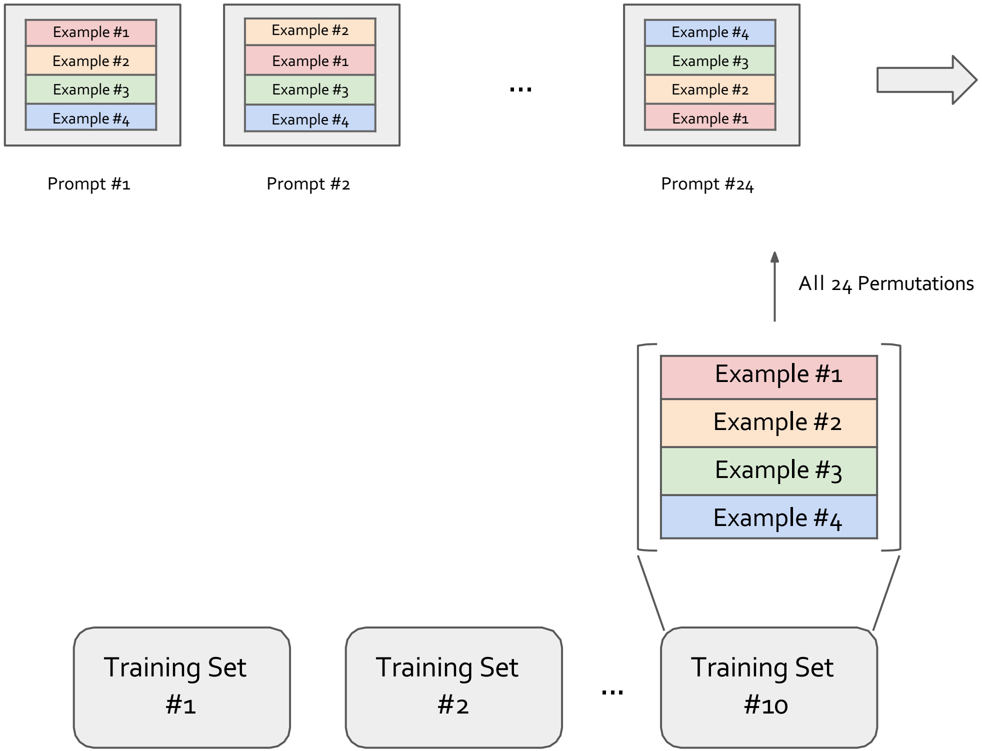
In-Context Learning: Sensitivity to Demo. Permutations

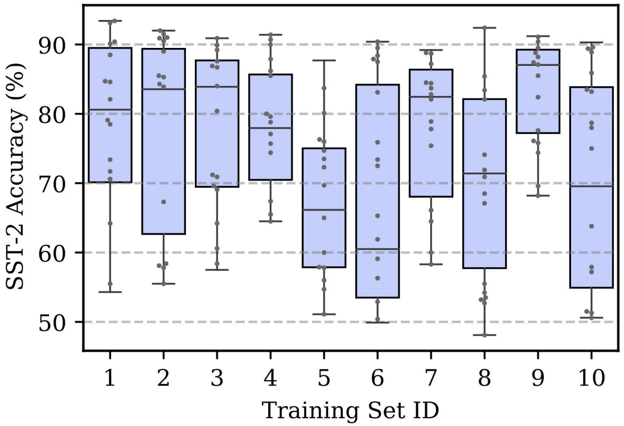


["Calibrate Before Use: Improving Few-Shot Performance of Language Models." Zhao et al. 2021]

[Slide credit: Eric Wallace]

In-Context Learning: Sensitivity to Demo. Permutations





The choice of demonstrations and their order is quite important.

[“Calibrate Before Use: Improving Few-Shot Performance of Language Models.” Zhao et al. 2021]

Sensitivity to Wording (Framing) of Prompts

- Framing of prompts matters a lot.

```
Craft a question that requires
commonsense to be answered. Based on the
given context, craft a common-sense
question, especially those that are LONG,
INTERESTING, and COMPLEX. The goal is to
write questions that are easy for humans
and hard for AI machines! To create such
questions, here are some suggestions: A.
What may (or may not) be the plausible
reason for an event? B. What may (or may
not) happen before (or after, or during)
an event?  ...
```

```
Generate questions such that you use
 - ‘what may happen’,
 - ‘will ...?’,
 - ‘why might’,
 - ‘what may have caused’,
 - ‘what may be true about’,
 - ‘what is probably true about’,
 - ‘what must’
and similar phrases in your question
based on the input context.
```

[“Reframing Instructional Prompts to GPTk's Language.” Mishra et al. 2021.]

Sensitivity to Wording (Framing) of Prompts

- Prompts can often be phrased in a language that are easier to be understood by language models.

- Generally, it is easier for LMs to follow shorter, crisp, itemized prompts.

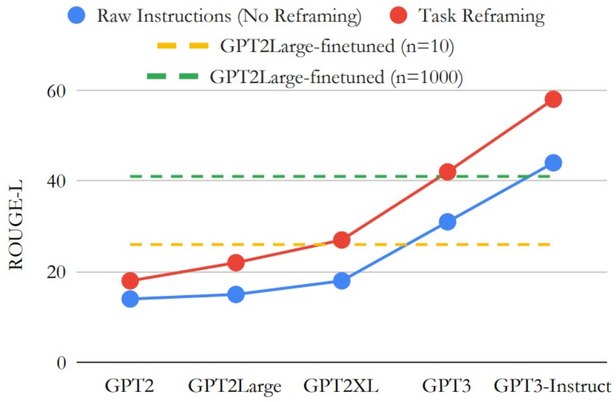


Figure 2: Across a variety of model sizes, reframed prompts consistently show considerable performance gain over raw task instructions (no reframing) in a few-shot learning setup. Since fine-tuning GPT3 is prohibitively expensive, we show the performance of fine-tuning smaller models (horizontal lines). This results indicates that *evaluating* reframed prompts on a large model like GPT3-instruct (red line) might be more effective that *fine-tuning* a smaller model like GPT2Large (green line) with 200× more data. Details of the experiments in §4.

["Reframing Instructional Prompts to GPTk's Language." Mishra et al. 2021.]

Summary Thus Far

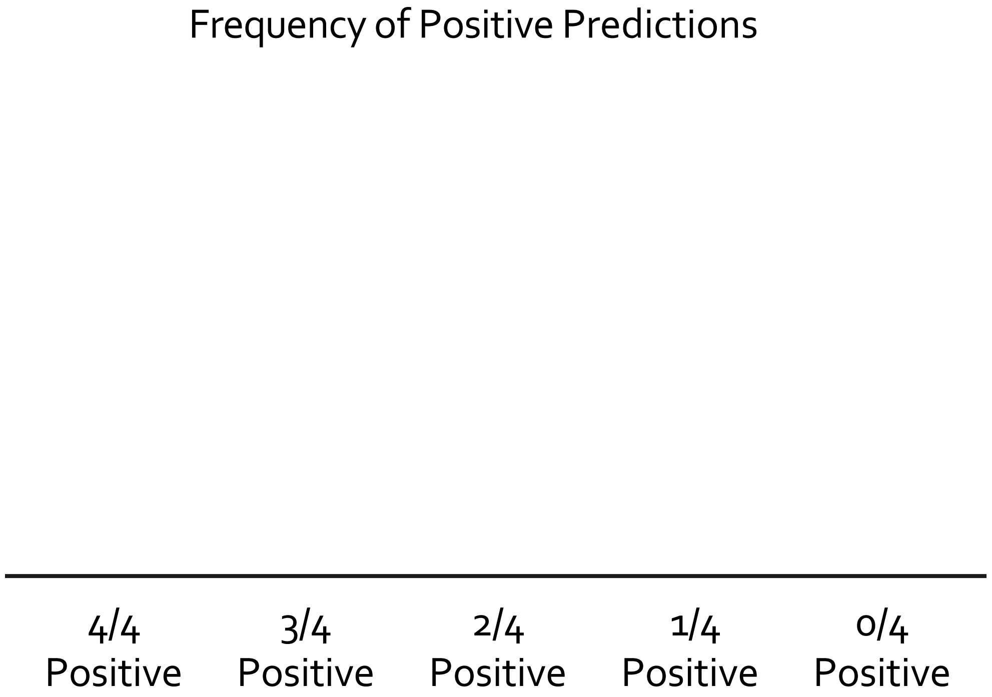
- There are many possible ways to encode in-context examples of a fixed task
  - Many possible patterns/verbalizers
  - The choice of demonstrations
  - Ordering of the examples
  - ....
- It turns out there is a huge variance in performance depending on the encoding.
  - You can treat them as hyper-parameters
  - You should **not** choose these encodings based on the test data.
- Generally, it is better to use encoding that makes the sequence closer to language modeling — closer to what is observed during pretraining.

What Causes These Variances?

- Here we will provide several justifications ...

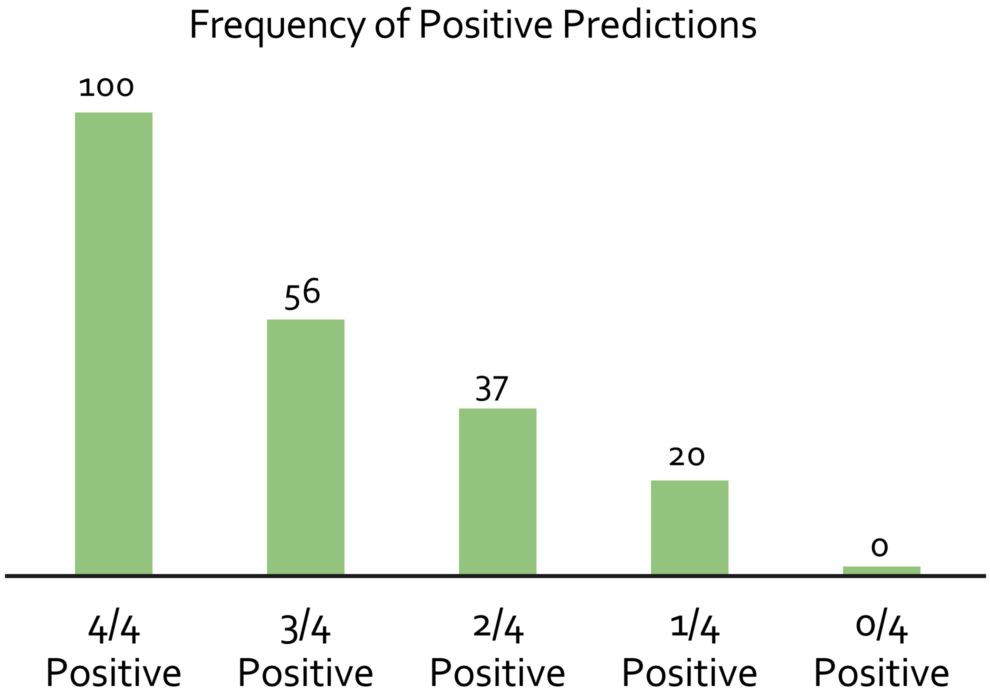
Majority Label Bias

Majority Label Bias



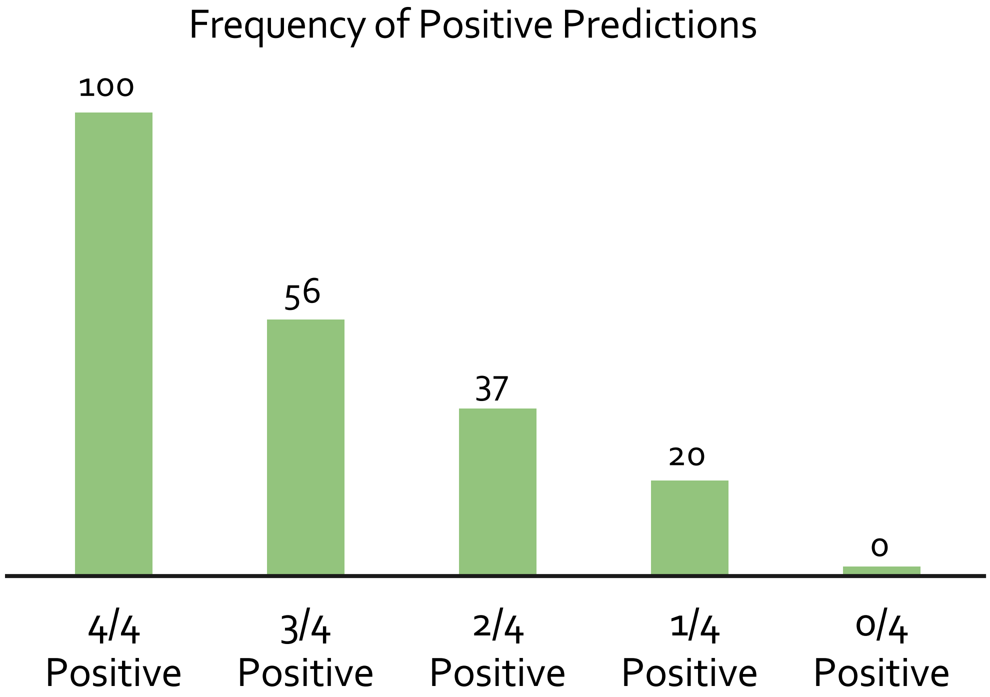
[“Calibrate Before Use: Improving Few-Shot Performance of Language Models.” Zhao et al. 2021]

Majority Label Bias



["Calibrate Before Use: Improving Few-Shot Performance of Language Models." Zhao et al. 2021]

Majority Label Bias



Majority label bias: frequent training answers dominate predictions
Explain some of the variances across example selections

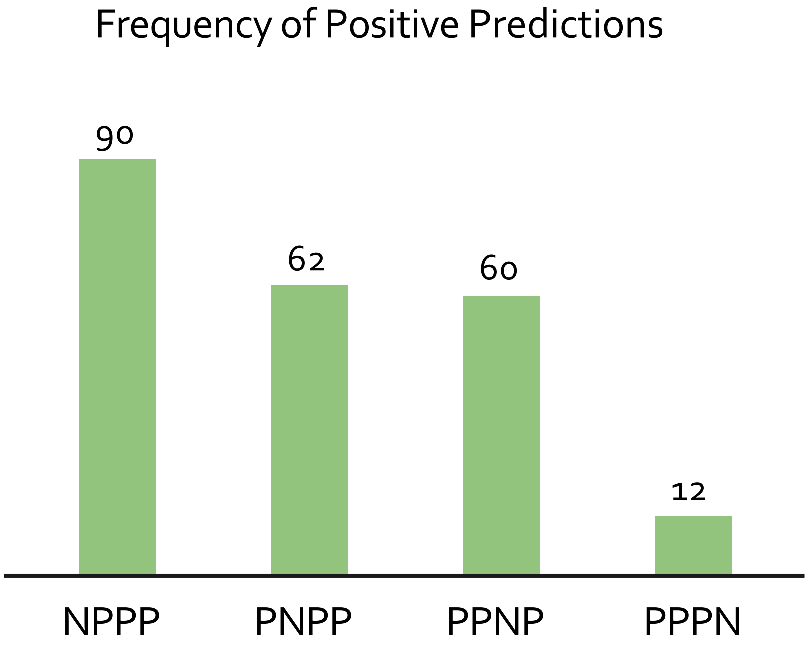
Recency Bias

Recency Bias

Frequency of Positive Predictions

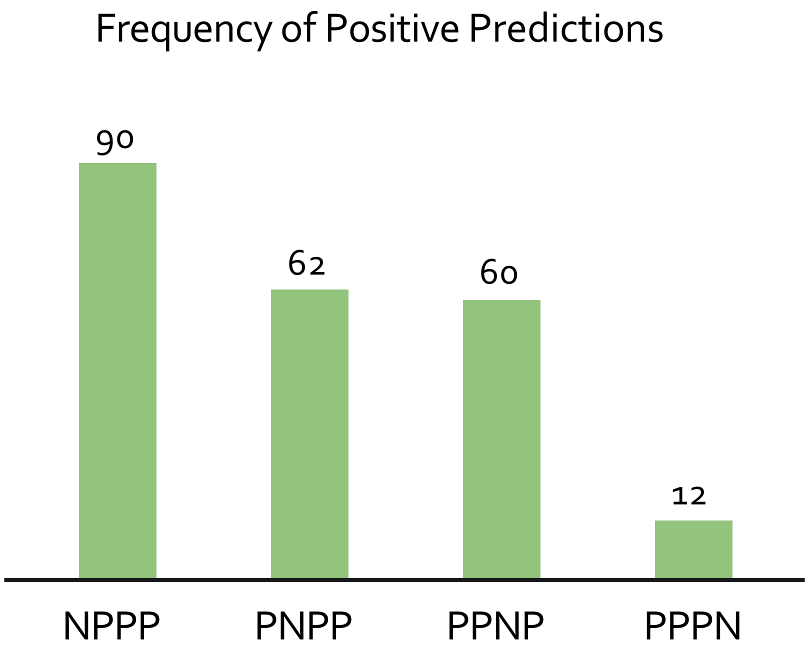
NPPP PNPP PPNP PPPN

Recency Bias



["Calibrate Before Use: Improving Few-Shot Performance of Language Models." Zhao et al. 2021]

Recency Bias

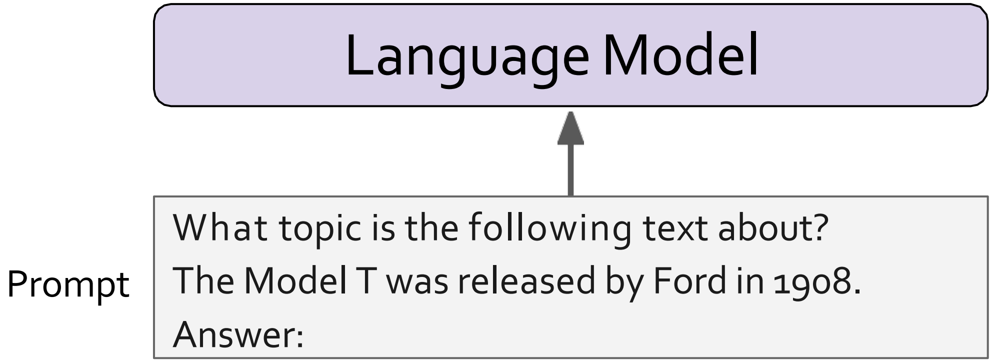


Recency bias: examples near end of prompt dominate predictions

- Explains variance across example permutations

Common Token Bias

Common Token Bias



Common Token Bias

Language Model

Prompt

What topic is the following text about?
The Model T was released by Ford in 1908.
Answer:

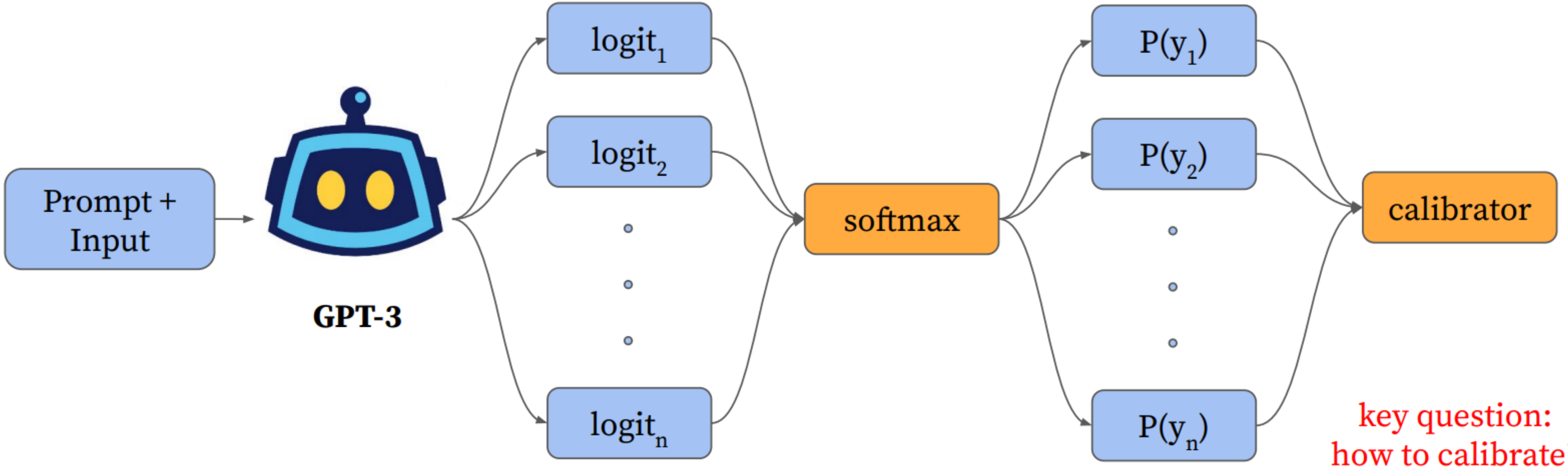
Token	Prob
book	0.35
transportation	0.23
school	0.11
village	0.03
company	0.02

Model is biased towards predicting the incorrect frequent token "book" even when both "book" and "transportation" are equally likely labels in the dataset

	Token	Web (%)	Label (%)	Prediction (%)
❌	*book*	0.026	9	29
✅	*transportation*	0.0000006	9	4

- Common token bias: common n-grams dominate predictions
  - helps explain variance across prompt formats

Calibrating LM Probabilities



key question:
how to calibrate?

[Slide credit: Howard Yen, Vishvak Murahari]

Calibrating LM Probabilities

Step 1: Estimate the bias

Calibrating LM Probabilities

Step 1: Estimate the bias

Insert “content-free” test input into prompt

```
Input: Subpar acting. Sentiment: negative
Input: Beautiful film.  Sentiment: positive
Input:                          Sentiment:
```

Calibrating LM Probabilities

Step 1: Estimate the bias

Insert "content-free" test input into prompt

Input: Subpar acting. Sentiment: negative
Input: Beautiful film. Sentiment: positive
Input: Sentiment:

Get model's prediction

positive	0.65
negative	0.35

Calibrating LM Probabilities

Step 1: Estimate the bias

Insert “content-free” test input into prompt

Input: Subpar acting. Sentiment: negative
Input: Beautiful film. Sentiment: positive
Input: Sentiment:

Get model’s prediction

positive	0.65
negative	0.35

Step 2: Counter the bias

Calibrating LM Probabilities

Step 1: Estimate the bias

Insert "content-free" test input into prompt

Input: Subpar acting. Sentiment: negative
Input: Beautiful film. Sentiment: positive
Input: Sentiment:

Get model's prediction

positive	0.65
negative	0.35

Step 2: Counter the bias ("Platt Scaling")

"Calibrate" predictions with affine transformation

$$\hat{\mathbf{q}} = \mathrm{softmax}(\mathbf{W}\hat{\mathbf{p}} + \mathbf{b})$$

["Calibrate Before Use: Improving Few-Shot Performance of Language Models." Zhao et al. 2021]

Calibrating LM Probabilities

Step 1: Estimate the bias

Insert "content-free" test input into prompt

> Input: Subpar acting. Sentiment: negative
> Input: Beautiful film. Sentiment: positive
> Input: Sentiment:

Get model's prediction

positive	0.65
negative	0.35

Step 2: Counter the bias ("Platt Scaling")

"Calibrate" predictions with affine transformation

$$\underbrace{\hat{\mathbf{q}}}_{\text{Calibrated probs}} = \mathrm{softmax}(\mathbf{W}\underbrace{\hat{\mathbf{p}}}_{\text{Original probs}} + \mathbf{b})$$

["Calibrate Before Use: Improving Few-Shot Performance of Language Models." Zhao et al. 2021]

Calibrating LM Probabilities

Step 1: Estimate the bias

Insert "content-free" test input into prompt

Input: Subpar acting. Sentiment: negative
Input: Beautiful film. Sentiment: positive
Input: Sentiment:

Get model's prediction

positive	0.65
negative	0.35

Step 2: Counter the bias ("Platt Scaling")

"Calibrate" predictions with affine transformation

$$\hat{\mathbf{q}} = \mathrm{softmax}(\mathbf{W}\hat{\mathbf{p}} + \mathbf{b})$$

Calibrated probs

Original probs

Fit $\mathbf{W}$ and $\mathbf{b}$ to so that outputs for the content-free input are uniform

["Calibrate Before Use: Improving Few-Shot Performance of Language Models." Zhao et al. 2021]

Calibrating LM Probabilities

Step 1: Estimate the bias

Note, this calibration does **not** require any labeled data.

- **Classification tasks:** normalized scores of labels
- **Generation tasks:** probabilities of the first token of the generation over the entire vocabulary

Get model's prediction

positive	0.65
negative	0.35

Step 2: Counter the bias ("Platt Scaling")

"Calibrate" predictions with affine transformation

$$\hat{\mathbf{q}} = \mathrm{softmax}(\mathbf{W}\hat{\mathbf{p}} + \mathbf{b})$$

Calibrated probs — Original probs

Fit $\mathbf{W}$ and $\mathbf{b}$ to so that outputs for the content-free input are uniform

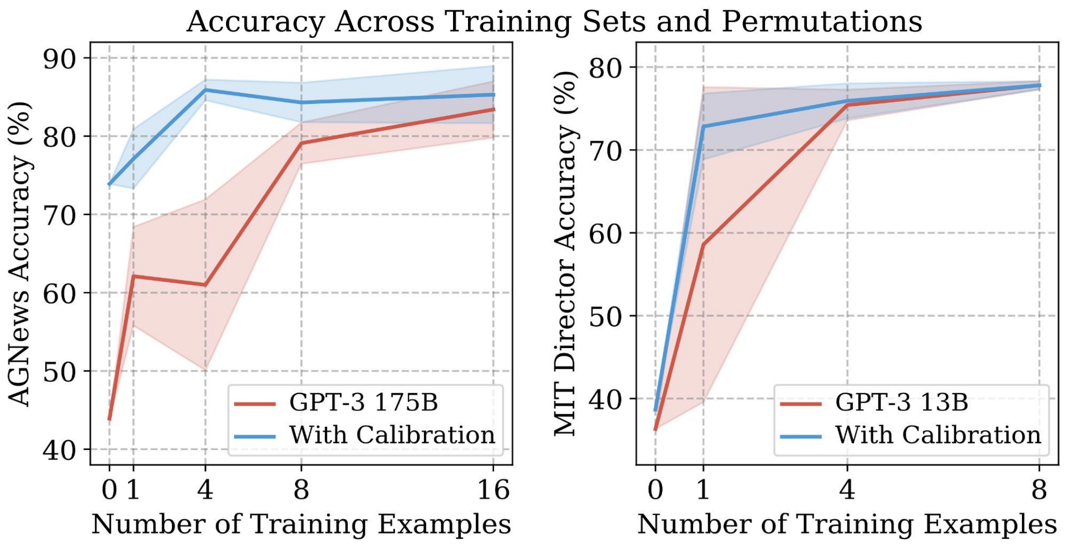
$$\mathbf{W} = \begin{pmatrix} \frac{1}{0.65} & 0 \\ 0 & \frac{1}{0.35} \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

W: diagonal matrix
b: bias set to zeros

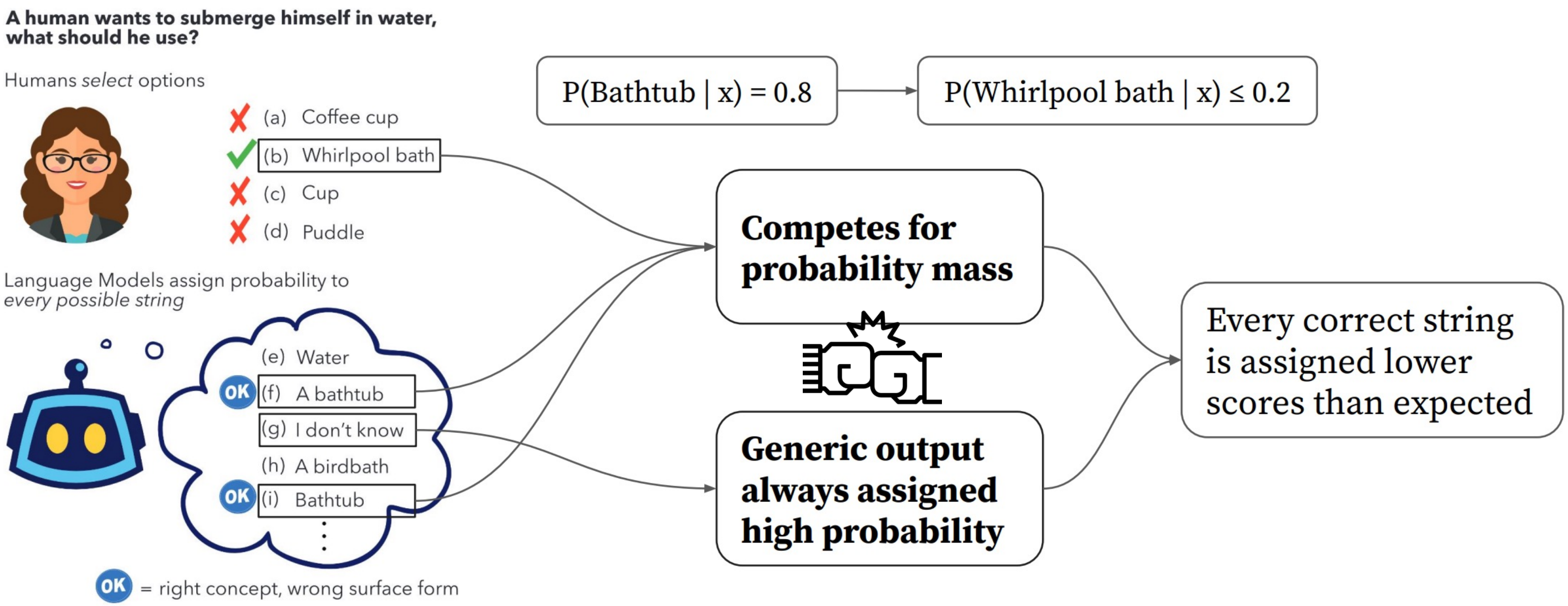
["Calibrate Before Use: Improving Few-Shot Performance of Language Models." Zhao et al. 2021]

Effect of Calibration

- Improves mean and worst-case accuracy
- Reduces variance across training sets and permutations



Surface Form Competition



Surface forms are competing for probability mass — skew the probabilities.
There are some ideas on how to calibrate for these issues (see Holtzman et al 2021).

[Slide credit: Howard Yen, Vishvak Murahari]

Summary Thus Far

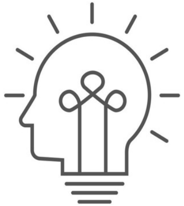
- LM prompting & In-context learning show promising results, but their performance is highly unstable/brittle.

- Better scoring: Calibration

- Other factors:
  - Better formation of demonstrations
  - Better choice of demonstrative examples
  - Better ordering of demonstrative examples

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

How/Why does In-context Learning Work?



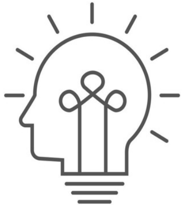
Any arbitrary task

Language Model

A few-shot learner

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

How/Why does In-context Learning Work?



Any arbitrary task

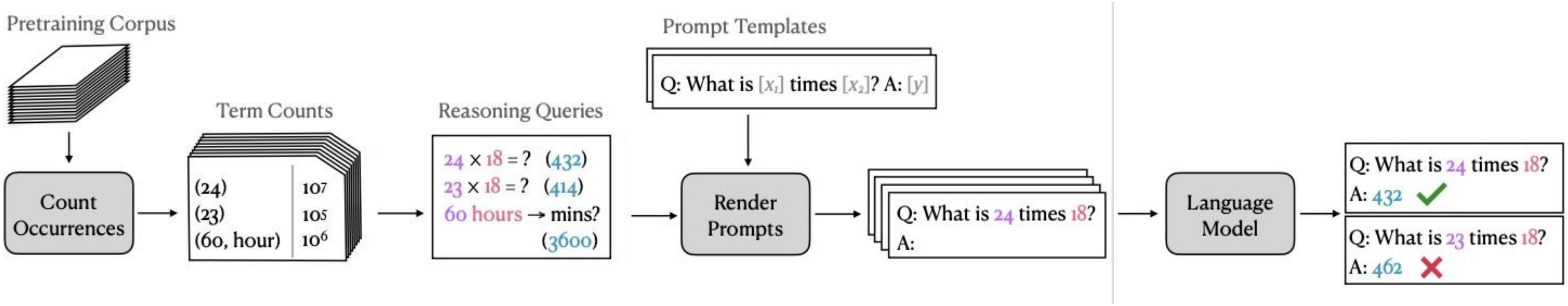
Language Model

A few-shot learner

[ACL 2022 Tutorial Beltagy, Cohan, Logan IV, Min and Singh]

Impact of Pretraining Term Frequencies

- For each task, identify relevant terms from each instance—numbers and units
- Count co-occurrences of these terms in the pretraining data (term pairs or triples within a fixed window)



["Impact of Pretraining Term Frequencies on Few-Shot Reasoning" Razeghi et al. 2022.]

Impact of Pretraining Term Frequencies

This may also indicate that, demonstrations do not teach a new task; instead, it is about locating an already-learned task during pretraining (Reynolds & McDonell, 2021)

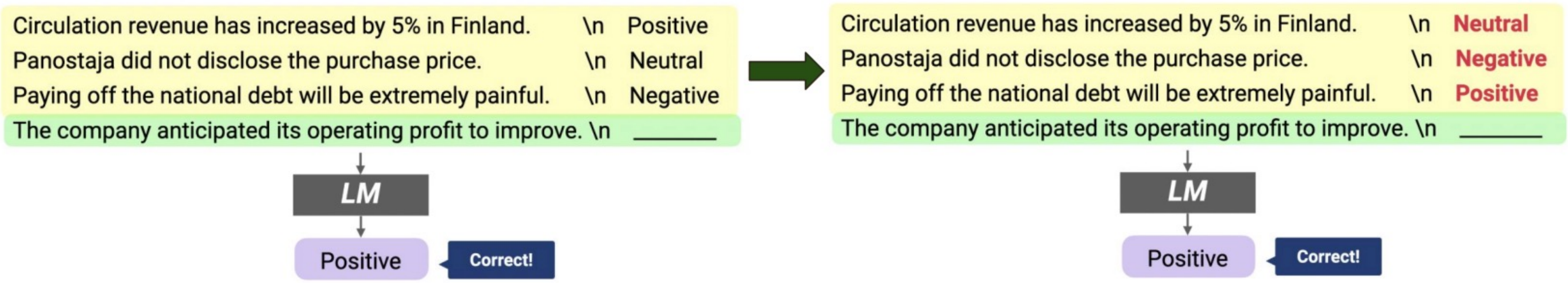
But that brings up the question of how much LMs **actually** reason when solving these tasks.🤔 Overlooking the impact of pretraining data can be misleading in evaluation!

In-context learning performance is highly correlated with term frequencies during pretraining

["Impact of Pretraining Term Frequencies on Few-Shot Reasoning" Razeghi et al. 2022.]

Impact of Input-Output Mapping

- Study the effect of randomizing labels in demonstrations.
  - Randomly sample a label from the correct label space



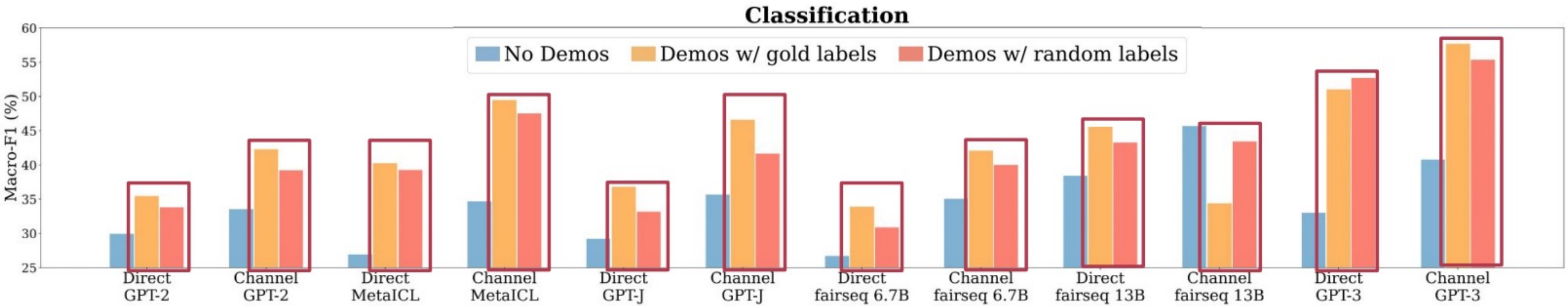
Prompt with true labels

Prompt with random labels

["Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?" Min et al. 2022.]

Impact of Input-Output Mapping

- Models see a small performance drop (0–5% absolute) with random labels



Comparisons between no-examples (blue), examples with ground truth outputs (yellow) and examples with random outputs (red)

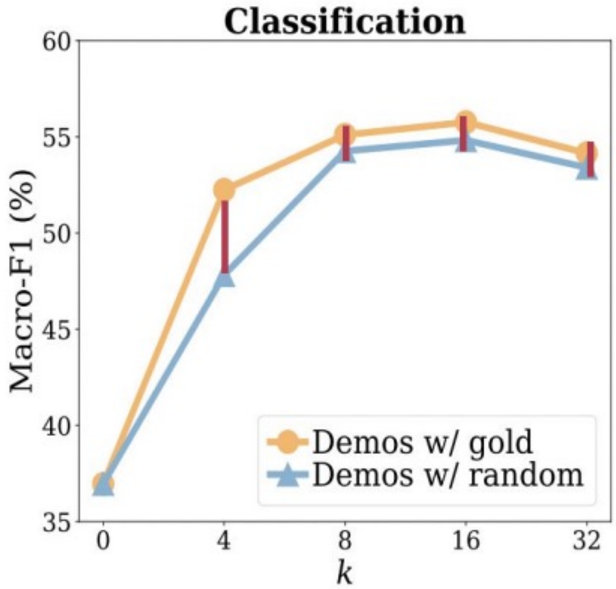
- **Takeaway:** ground truth input-label mapping in the prompt is not as important as we thought

["Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?" Min et al. 2022.]

Impact of Input-Output Mapping

- Vary number of demonstrations

- **Takeaway:**
  - Performance drop from using gold labels to using random labels is consistently small across varying k, ranging from 0.8–1.6%
  - Using small number of examples with random labels is better than no examples



["Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?" Min et al. 2022.]