Reinforcement Learning for Language Model Training

Polina Tsvilodub

Calibration & RL evaluation



Language models Understanding mechanics

- what information in represented at different stages of processing?
- what information contributes to predicting the right answer?
- what (architectural) mechanisms extract important information?
- what (architectural) mechanisms are necessary for solving different tasks?
- how do we investigate systems involving RL fine-tuning?

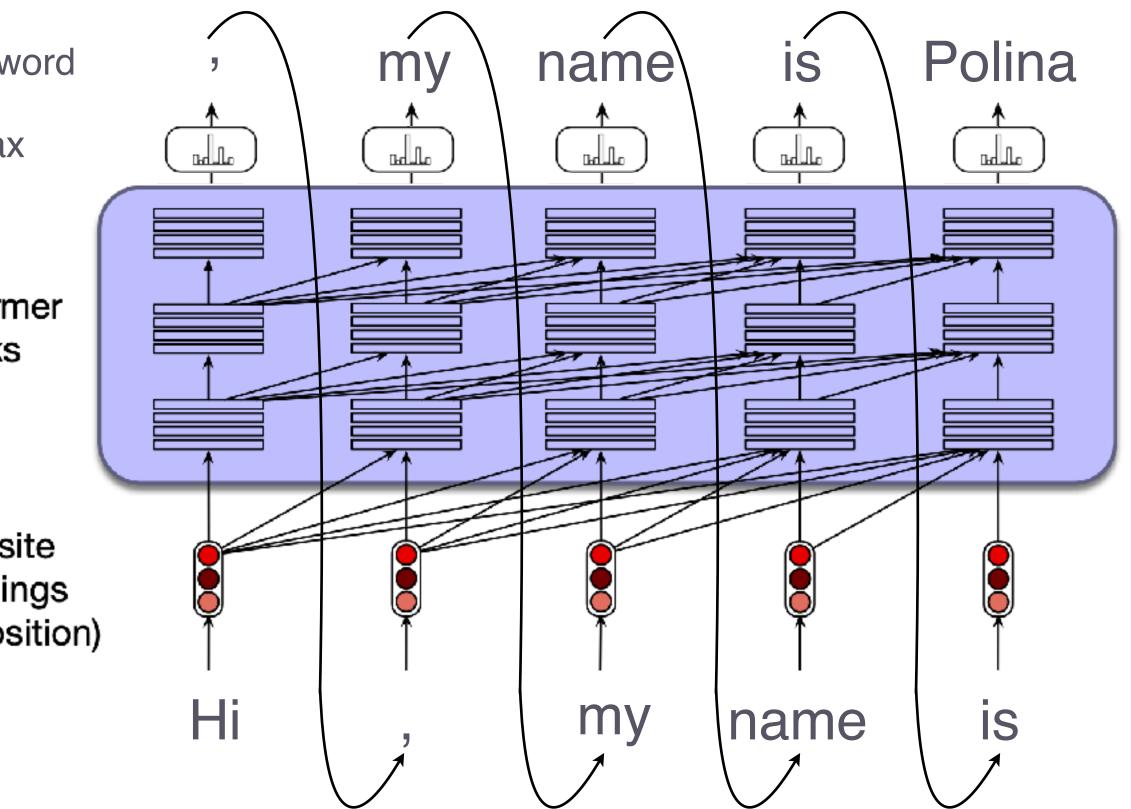
•

Sampled word

Softmax

Transformer Blocks

Composite Embeddings (input + position)

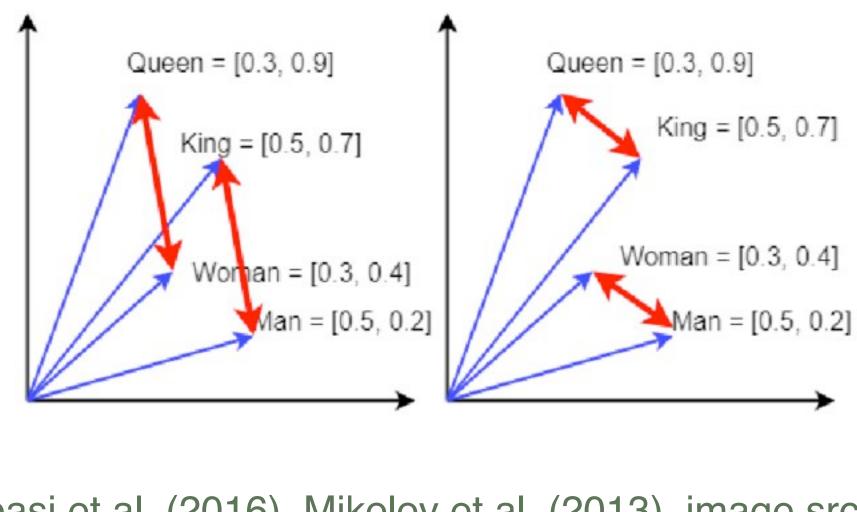


Embedding evaluation Doctor - man + woman = ?

- pretrained word embeddings have been evaluated as semantic representations
 - vector arithmetic

$$\cos(w_1, w_2) = \frac{w_1 \cdot w_2}{\|w_1\| \|w_2\|}$$

- current models are decoder-only and use sub-word embeddings
 - semantic tasks often solved few-shot

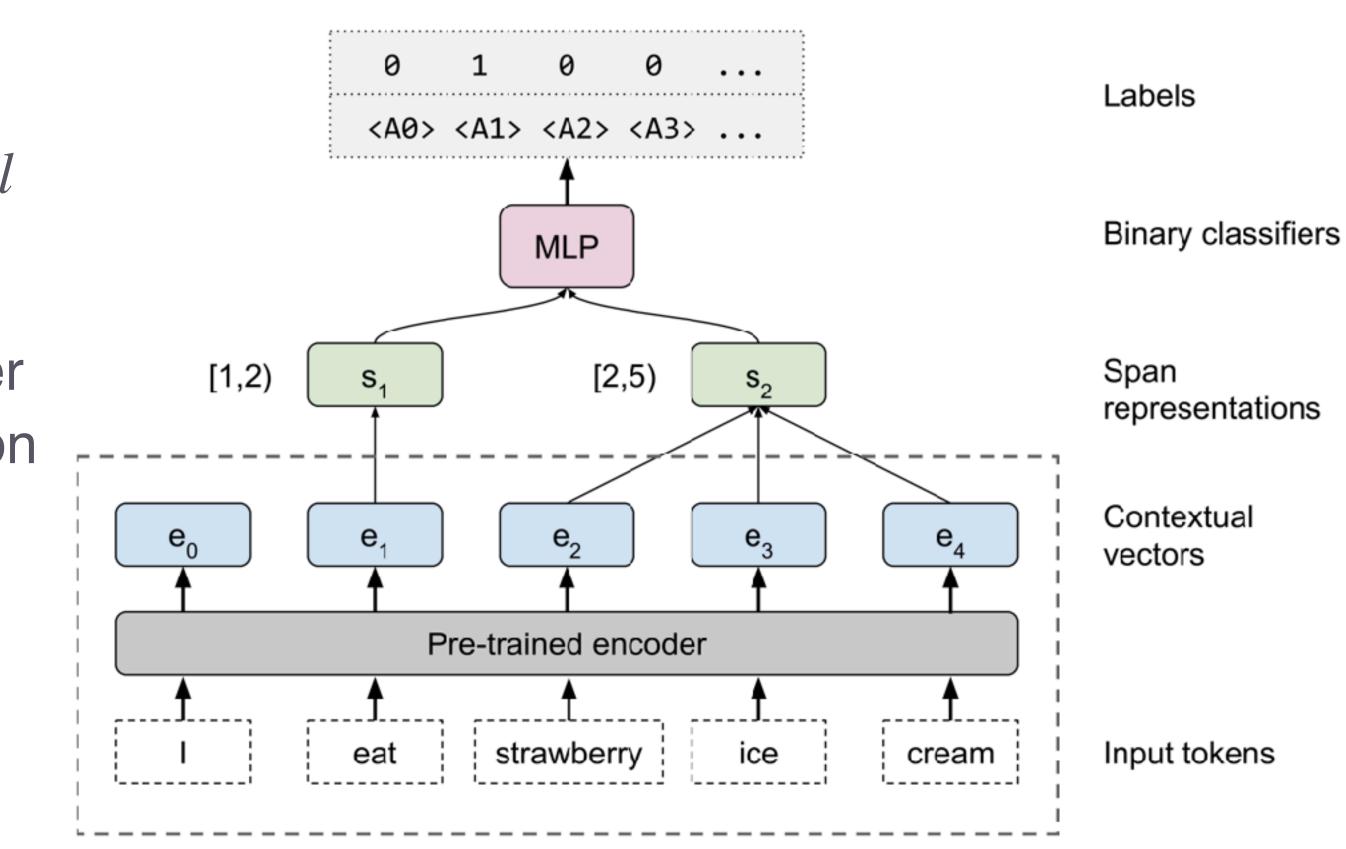


Bolukbasi et al. (2016), Mikolov et al. (2013), image src

Scalar mixing weights which layers to combine information from

- consider *L* layers of stacked embeddings $H^{(0)}, \ldots, H^{(L)}$, input w_1, \ldots, w_n , vector $\left[\mathbf{h}_0^{(l)}, \ldots, \mathbf{h}_n^{(l)}\right]$ of word embeddings at layer *l*
- train scalar mixing weights [s₀, ..., s_L] together with MLP classifiers for each layer to solve tasks (e.g., POS tagging) based on token representations:

$$\mathbf{h}_i = \sum_{l=0}^{L} s_l \mathbf{h}_i^{(l)}$$



Amnesic probing in neural networks Inferring functional roles of representations

- systematically intervene with the normal feedforward prediction of a trained model
 - check what happens to relevant task performance
 - interventions can take place at different locations
- sketch of amnestic operation:
 - train a sequence of linear classifiers (SVMs) for task T
 - iteratively remove information useable by classifier for the task
 - terminate when predictive accuracy is at chance level
- include controls (similar amount of deletion) but in more arbitrary direction)
 - information
 - selectivity

verb noun det $h_{ran}^{\neg POS}$ Property (POS)Amnesic (Remove POS) Operation standard probing noun det $\bigcirc\bigcirc\bigcirc\bigcirc$ h_{dog} $|h_{ran}|$... \mathbf{the} dog ran

Elazar et al., (2021), Rafvogel et al., (2020)

amnesic probing

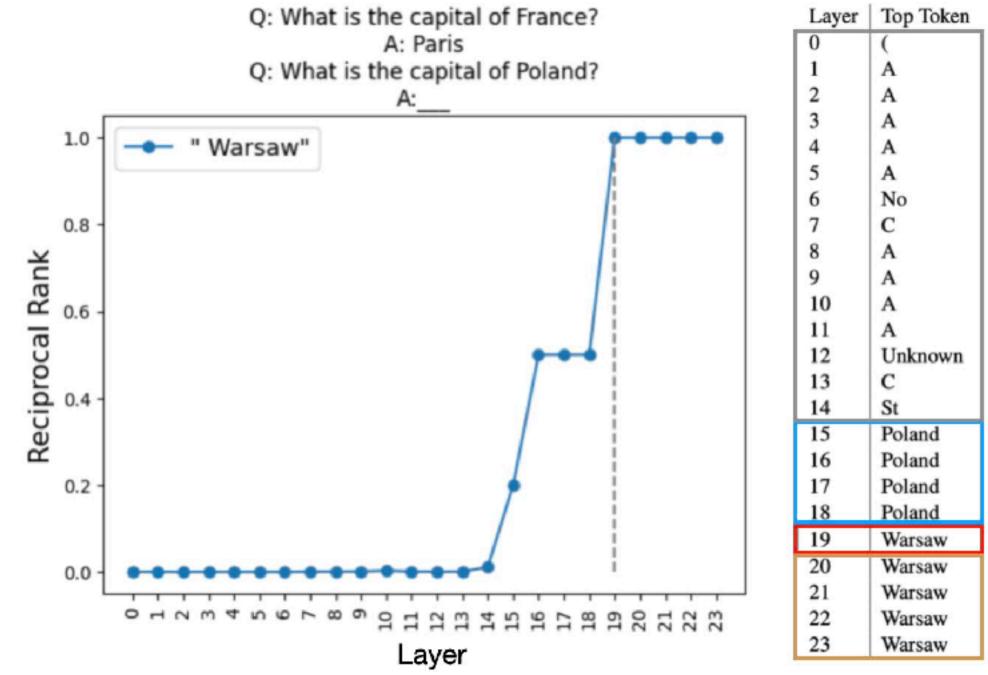




How do LLMs solve relational tasks? Merullo, Eickhoff & Pavlick (2023)

- new inputs
 - f I f(France) = Paris -> f(Poland) = Warsaw
- critical components for such tasks (capital identification, uppercasing, past tense mapping): transformer block FFN, residual stream
- early decoding used to identify that the FFN update retrieves the capital (=Warsaw) of a new argument (Poland)
 - applies the 'function get_capital(Poland)'
- interventions to check this role of the FFN
 - FFN update in other contexts
 - relevant for abstractive, but not extractive tasks

LLMs learn to solve relational tasks in-context by re-applying the example relation to



Reward collapse in RL fine-tuning Song et al. (2023)

- current reward model training objective (based on ranking of responses) leads to reward collapse
 - identical reward distributions for inputs where distinct distributions expected (open-ended vs. closedended tasks)

. problematic utility function: $U = \log \operatorname{sigmoid}(\frac{R_v}{-})$

- proposed mitigation: prompt-aware utility
 - $U_{closed} = x$ (polarized distribution)
 - . $U_{\text{open}} = \frac{-1}{x}$ (more uniform distribution)
- (artificial task) experiment with response length as reward

$$(\frac{w}{\sigma} - R_l)$$

of the second state of the

Reward collapse in RL fine-tuning Song et al. (2023)

- current reward model training objective (based on ranking of responses) leads to reward collapse
 - identical reward distributions for inputs where distinct distributions expected (open-ended vs. closedended tasks)

. problematic utility function: $U = \log \operatorname{sigmoid}(-1)$

- proposed mitigation: prompt-aware utility
 - $U_{closed} = x$ (polarized distribution)
 - . $U_{\text{open}} = \frac{-1}{x}$ (more uniform distribution)
- (artificial task) experiment with response length as reward
- could alleviate miscalibration of LLM responses

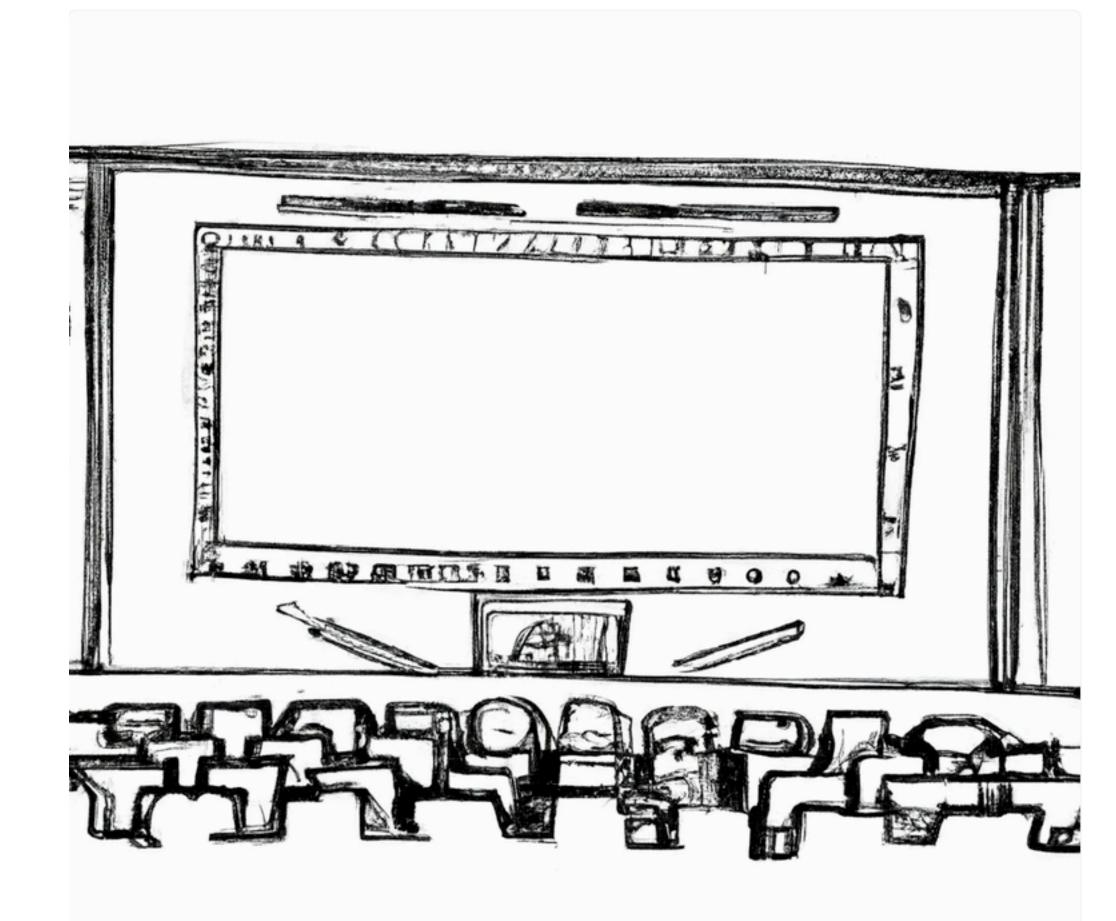
$$(\frac{R_w - R_l}{\sigma})$$

of functions

Presentations Your job

During the presentation, think about the following questions: What is honesty for you? Is it addressed with the proposed methodology?

- When) can we trust LLM outputs?



Calibration of natural text generation Evaluating Uncertainty in Neural Text Generators Against Human Production Variability

- main idea:
 - humans often express the same message in varying ways - NLG systems should capture the same variability
 - compare LLM and human production distributions
- methods:
 - comparison of productions via statistical similarity and different decoding schemes - lexical, syntactic, and semantic variability
 - distribution variability assessment and comparison via production probes
 - sample-based joint distribution approx. similarity between two outputs
 - similarity metrics:
 - unigram overlap (lexical)
 - POS bigram overlap (syntactic)
 - sentence embedding cosine similarity (semantic)



"Good" uncertainty in text generation **Production variability**

tasks

- machine translation
- text simplification
- story generation
- open domain dialogue
- decoding schemes:
 - unbiased samples
 - temperature scaling
 - top-k sampling
 - nucleus sampling
 - locally typical sampling
- models:
 - Transformer-Align
 - Flan-T5
 - GPT-2
 - DialoGPT

Dieleene eentert	Semantic variability	
Dialogue context		
It's very dark in here. Will you turn on the light?	0.6	
Okay. But our baby has fallen asleep.	0.5	
Then, turn on the lamp, please.	0.4	
But where's the switch?	0.3	
Humans	0.2	
• Don't you know where the switch is?	0.1	
• Switch is on the left side of the lamp.	0.0 0.5 1.0	
• Just press the second switch on the board.	Cosine similarity	
• Lamp is upon the study table and now you know	now where the switch is.	
• I will light up the torch, so you can find the sv		
DialoGPT-medium, nucleus $p = 0.9$		
• You don't have one. • I'm s	sorry.	
• Where's the button? • On n	my chest	
• It's on the top. • I'm o	on it!	
• Well, you'll want to turn it on. • Turn	ning on the switch	
	•	1
• Turn it on. • I hav	ve a few, try and figure it ou	
• Turn it on. • I hav	ve a few, try and figure it ou	

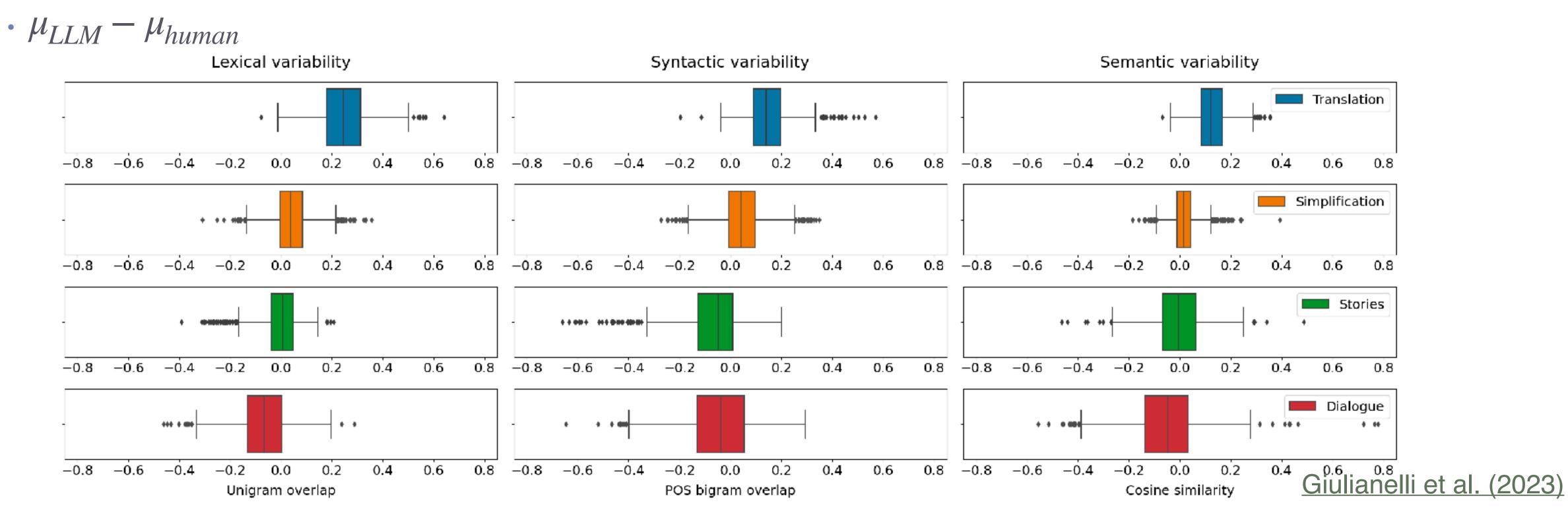




"Good" uncertainty in text generation **Production variability**

measures

- human variability
- LLM variability
- human-LLM cross-variability
- variability comparison:
 - μ_{human} : e.g. cos $sim(y_i, y_j)$ for y_i, y_j completions for prompt x





Evaluating RL agents

- goal of RL agent training: agent has learned to achieve a goal
 - LLMs: training helpful, harmless and honest agents
- evaluation aspects depend on the goals of the system, but generally:
 - performance of algorithm on standard environments like the OpenAI Gym(nasium) / Arcade
 - mean / median / cumulative training and test rewards / scores
 - relative to baseline, optimum or random behavior
 - downstream task performance
 - LLMs: comparative paradigm with pretrained LLMs
 - LLMs: evaluation of **alignment** via human annotations

alignment: agent's goals are congruent with human goals

- congruent ranking of outcomes (Askell et al., 2021)
- rewards don't provide information about how a goal should be achieved!
 - reward hacking / faulty reward functions: <u>example</u>



RL Gymnasium, RLiable blogpost

Human feedback in RL RLHF

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. Explain reinforcement learning to a 6 year old.

We give treats and punishments to teach...

This data is used to fine-tune GPT-3.5 with supervised learning.



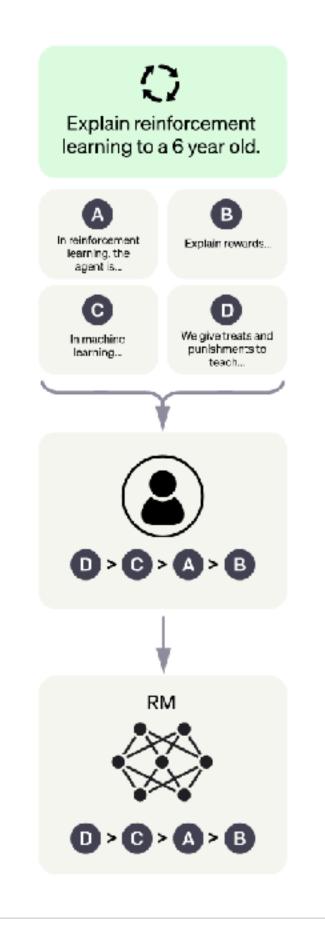
Step 2

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 the PPO reinforcement learning algorithm.

A new prompt is sampled from Write a story the dataset. about otters. PPO The PPO model is initialized from the supervised policy. The policy generates Once upon a time... an output. RM The reward model calculates a reward for the output. The reward is used to update the r_k policy using PPO.



Process-supervised reward models "Reasoning calibration"

- process (CoT)
 - model could be right for the wrong reasons! (hallucinations)
- ► set up:
 - train RM on MATH dataset with final solutions and human-annotated intermediate step solution evaluations (PRM800K for 12K problems)
 - evaluate accuracy of top-N response with highest reward (500 test problems)

The denominator of a fraction is 7 less than 3 times the nutlet the fraction? (Answer: $\boxed{14}$)
🙁 😐 😌 Let's call the numerator x.
🙁 😐 😌 So the denominator is 3x-7.
🙁 😐 😌 We know that x/(3x-7) = 2/5.
🙁 😐 😌 So 5x = 2(3x-7).
🙁 😐 😏 5x = 6x - 14.
🙁 😐 😎 So x = 7.



problem: standard (outcome-supervised) reward models only score the result of solution

idea: alleviate via process-supervised reward models which score the solution process

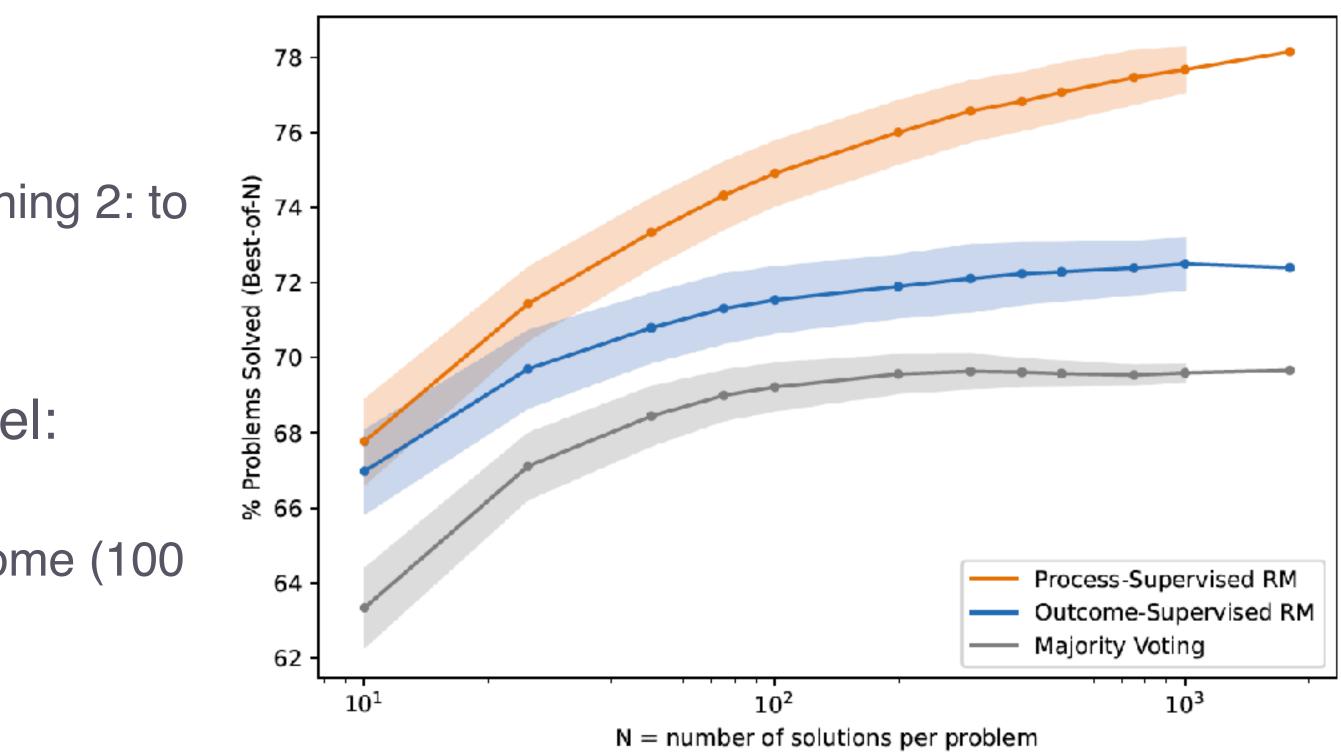
umerator. If the fraction is equivalent to 2/5, what is the numerator of





Process-supervised reward models "Reasoning calibration"

- fixed policy LLM (pretrained GPT-4)
- process-supervised reward model:
 - base pretrained GPT-4
 - fine-tuning 1: on MathMix (1.5B tokens); fine-tuning 2: to produce stepwise solutions
 - next-token prediction training up to first mistake
- outcome-supervised baseline reward model:
 - base pretrained GPT-4
 - trained on MATH to predict correctness of outcome (100) samples / problem from GPT-4)
- evaluation of data efficiency and OOD generalisation
- no evaluation of solution steps correctness!



Lightman et al. (2023)



Summary Calibration & RL evaluation

- calibration of LLMs reflects how well their probability predictions match 'true' outcome probabilities
 - approximates LLMs' 'knowledge' certainty
- natural language generation exhibits variability
 - LM generations' variability is not wellcalibrated wrt. human variability
- for RL fine-tuning, we might want to 'calibrate' the RMs' scores so as to reflect the solution process accuracy
 - idea: train process-supervised RMs



Feedback time

I would love to hear your feedback regarding the class!

https://forms.gle/hs4bFy4WVuJNimTM6

Please fill out the form by December 26th :)