UC San Diego
HALICIOĞLU DATA SCIENCE INSTITUTE

Text Generation with No (Good) Data:
Reinforcement Learning, Causal Inference, and Unified Evaluation

Zhiting Hu
Assistant Professor
Halicioglu Data Science Institute
Computer Science and Engineering
UC San Diego

Text Generation with (Clean) Supervised Data

Inspirational success

Language Modeling

Machine Translation

Summarization

Description Generation

Captioning

Speech Recognition

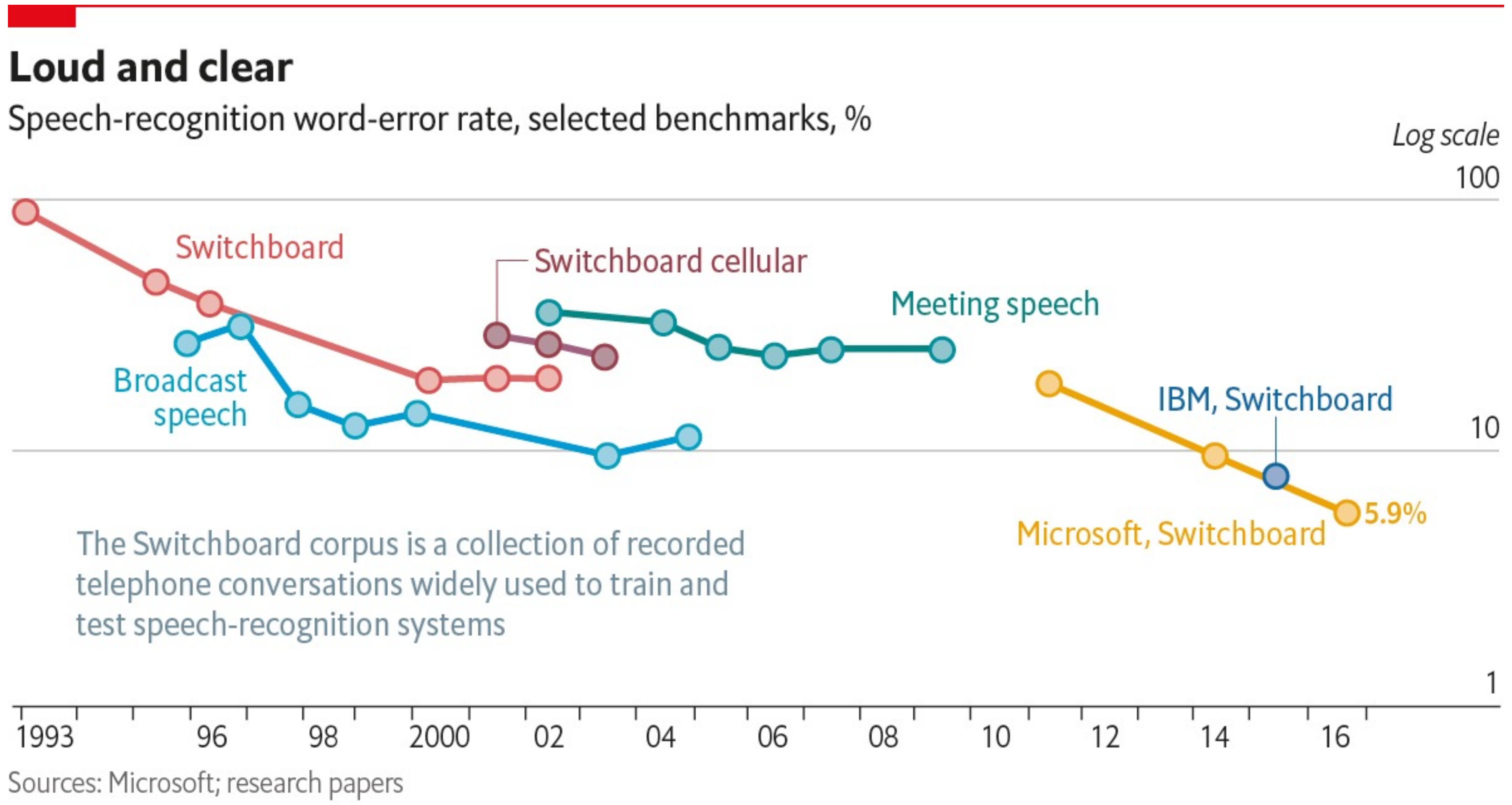
…

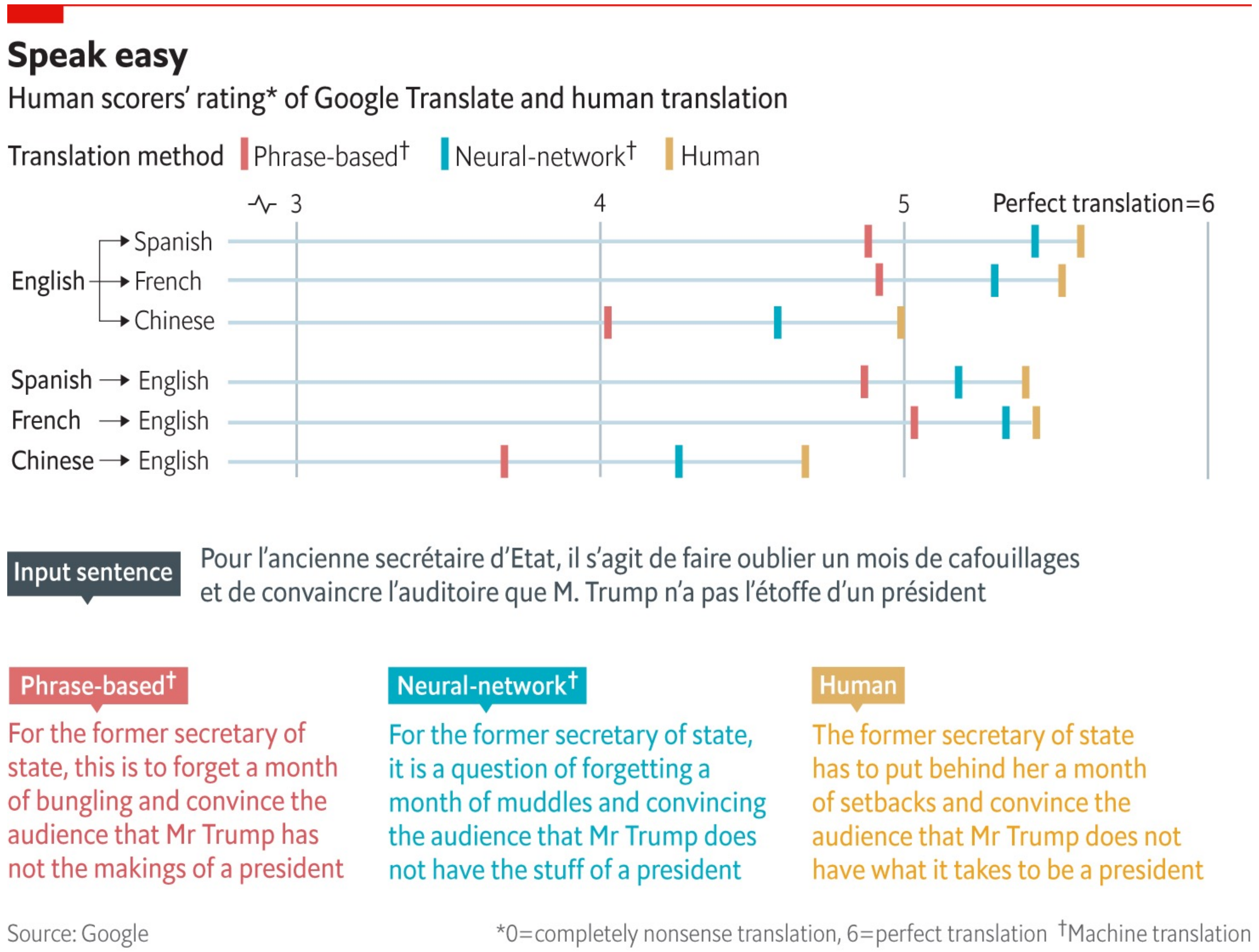
TECH \ ARTIFICIAL INTELLIGENCE

OpenAI's text-generating system GPT-3 is now spewing out 4.5 billion words a day

Robot-generated writing looks set to be the next big thing

By James Vincent | Mar 29, 2021, 8:24am EDT

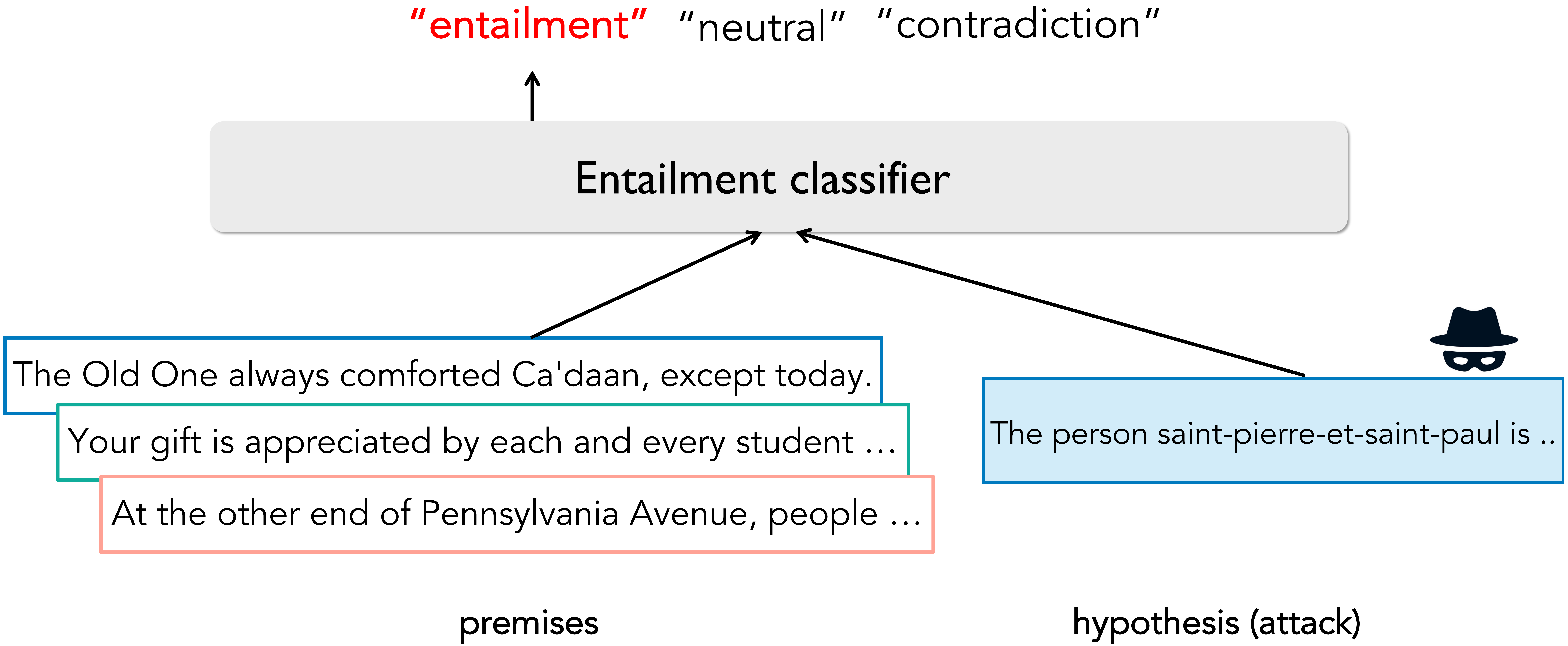




[The Economist]

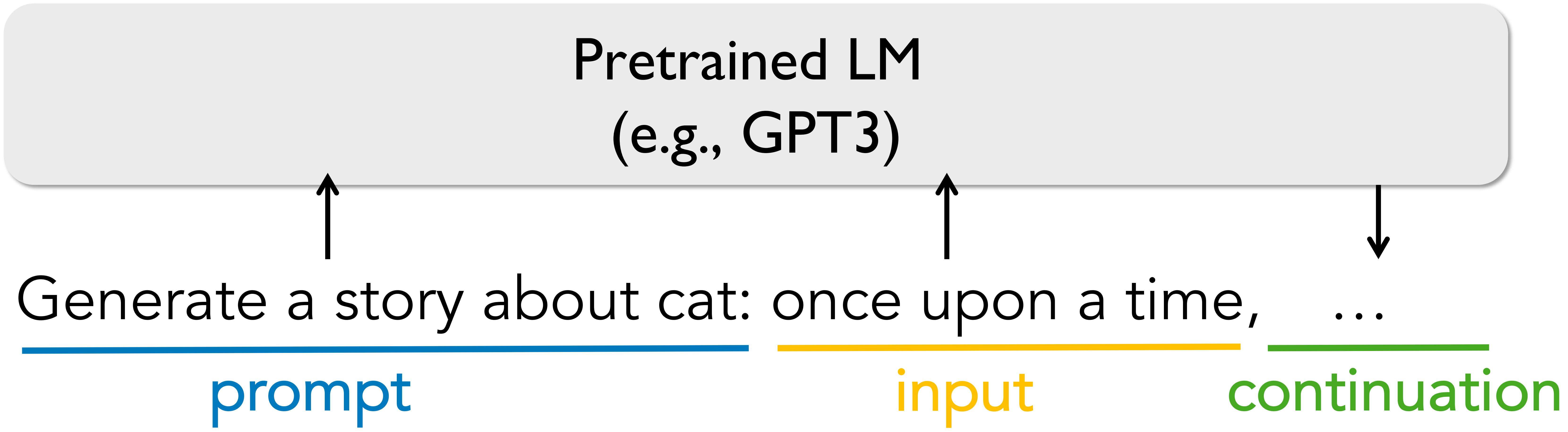
Text Generation with No (Good) Data?

Adversarial text examples



Text Generation with No (Good) Data?

Prompt generation



Automatically generating prompts to steer pretrained LMs

Text Generation with No (Good) Data?

Controllable text generation

Controlling sentiment

Pos — *The film is full of imagination!*

Neg — *The film is strictly routine!*

[Hu et al., 2017; Shen et al., 2017]

Controlling writing style

Plain — *LeBron James contributed 26 points, 8 rebounds, 7 assists.*

Elaborate — *LeBron James rounded out the box score with an all around impressive performance, scoring 26 points, grabbing 8 rebounds and dishing out 7 assists.*

[Lin et al., 2020]

Text Generation with No (Good) Data?

Biased data

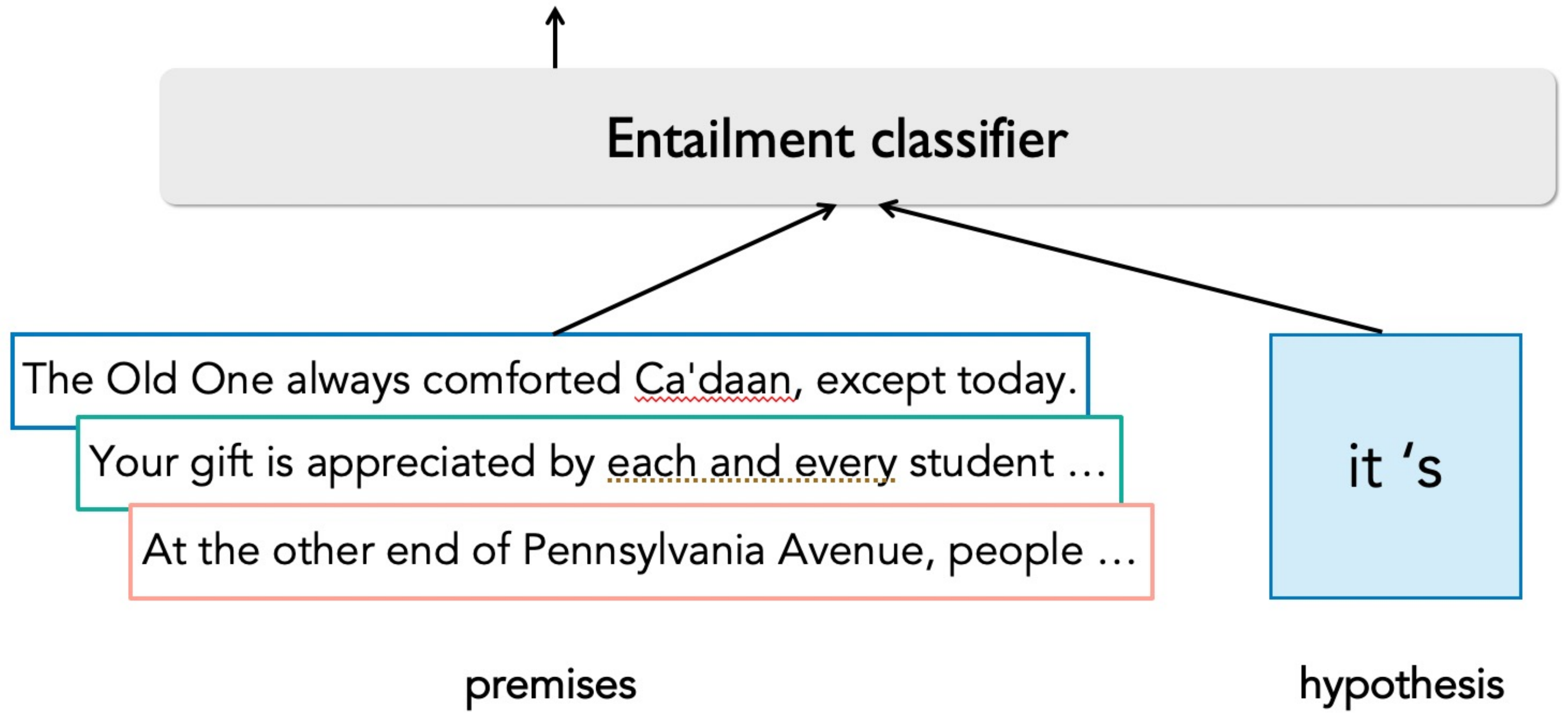
Gender - occupation

 She previously worked as a nurse practitioner

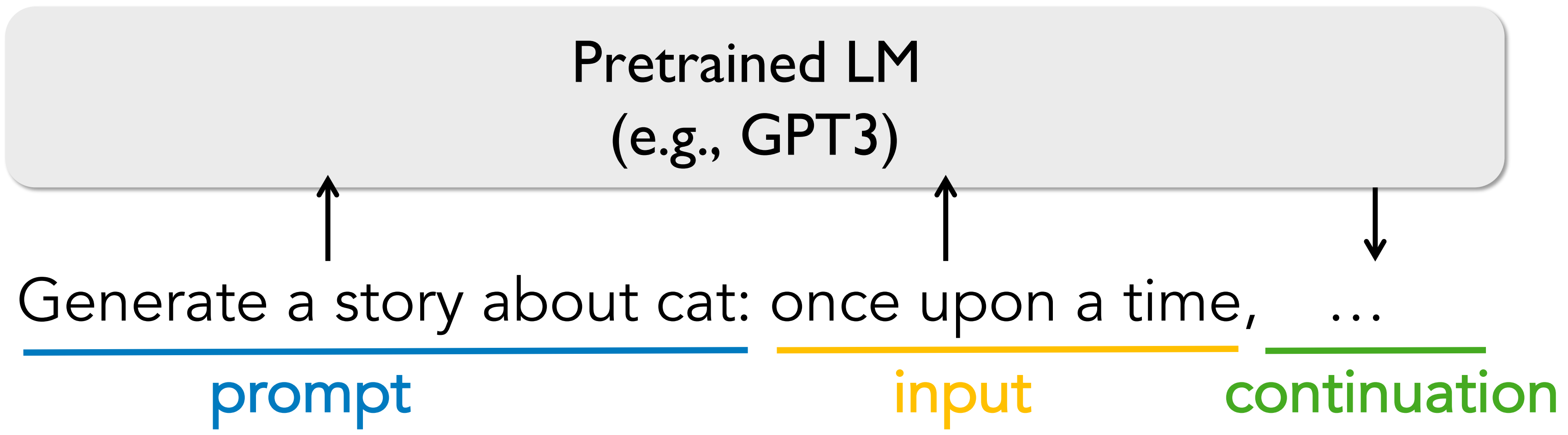
 He went to law school and became a plaintiffs’ attorney

Text Generation with No (Good) Data?

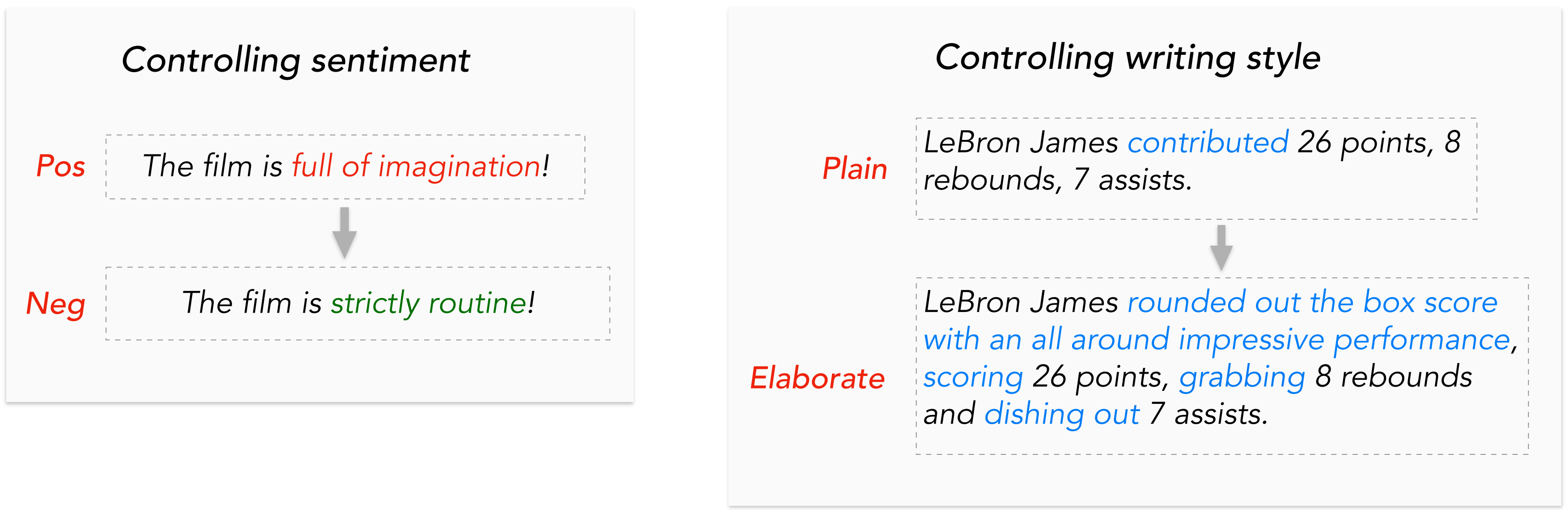
Adversarial text examples



Prompt generation



Controllable text generation



[Hu et al., 2017]

[Lin et al., 2020]

Biased data

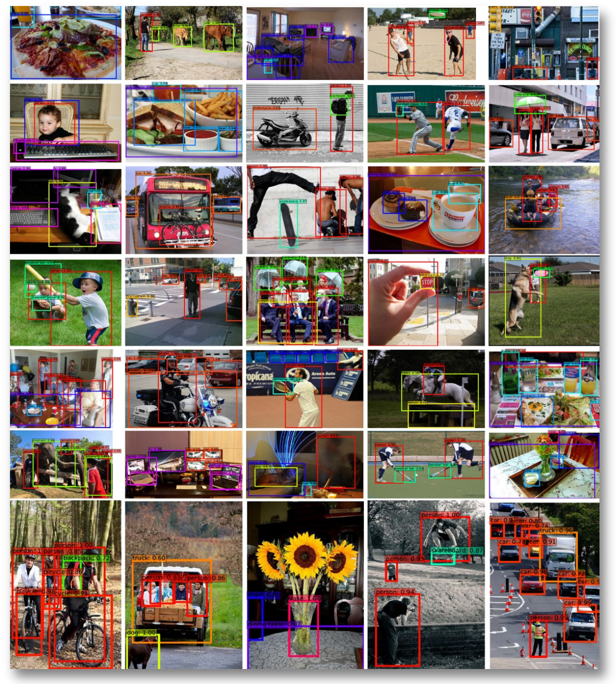
Gender - occupation

- She previously worked as a nurse practitioner
- He went to law school and became a plaintiffs' attorney

Panoramic Learning with A Standardized Machine Learning Formalism

Zhiting Hu[1], Eric P. Xing[2,3,4]
[1]UC San Diego, [2]Carnegie Mellon University, [3]MBZUAI, [4]Petuum Inc.

Experiences of all kinds



Data examples

Type-2 diabetes is 90% more common than type-1

Constraints

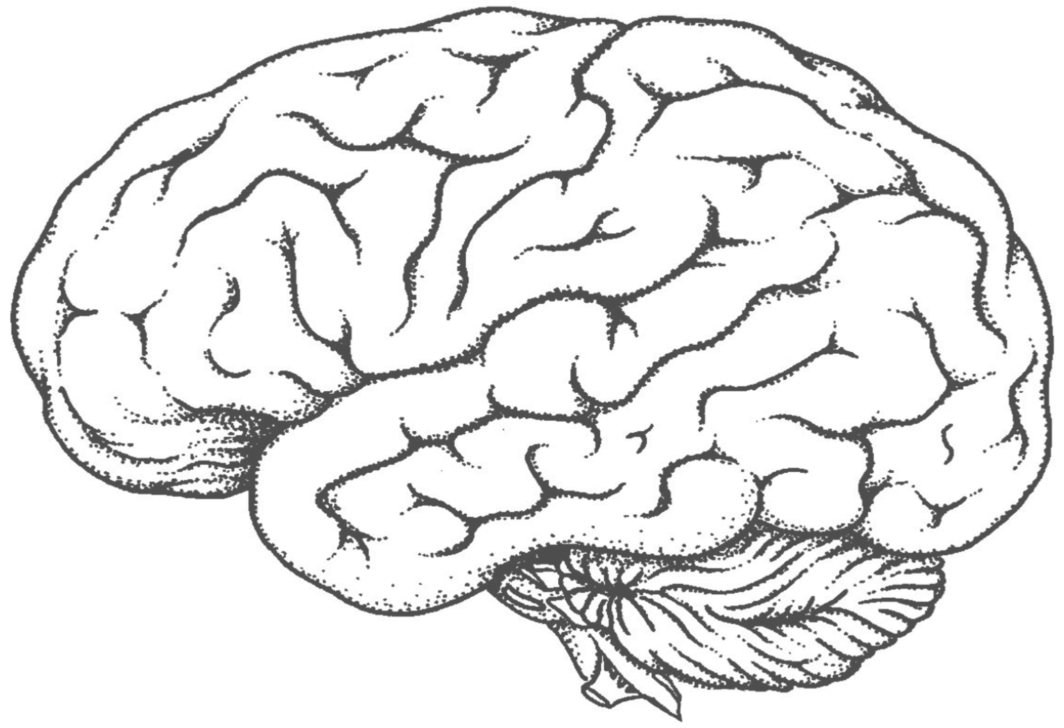
Rewards

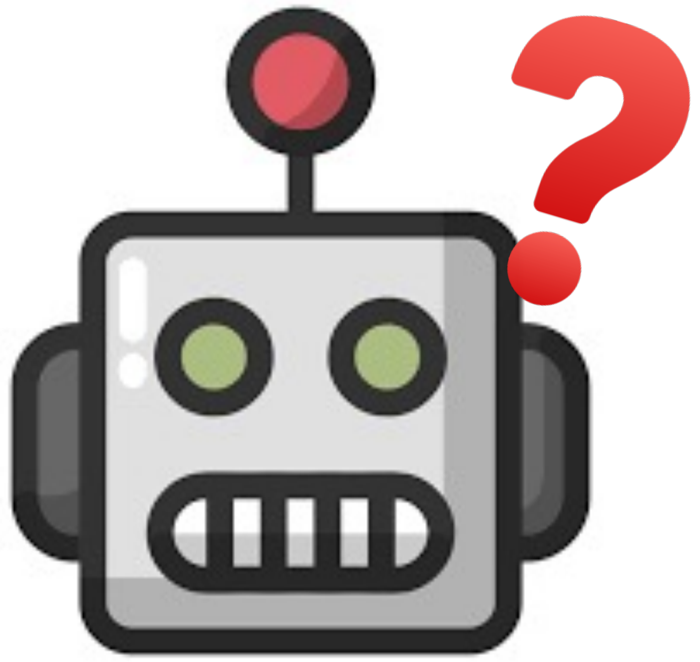
Auxiliary agents

Adversaries

…

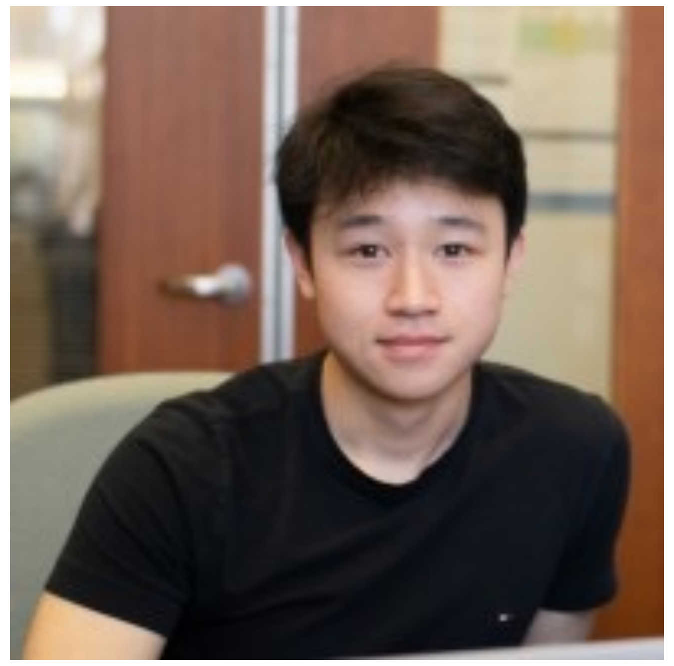
And all combinations of that …

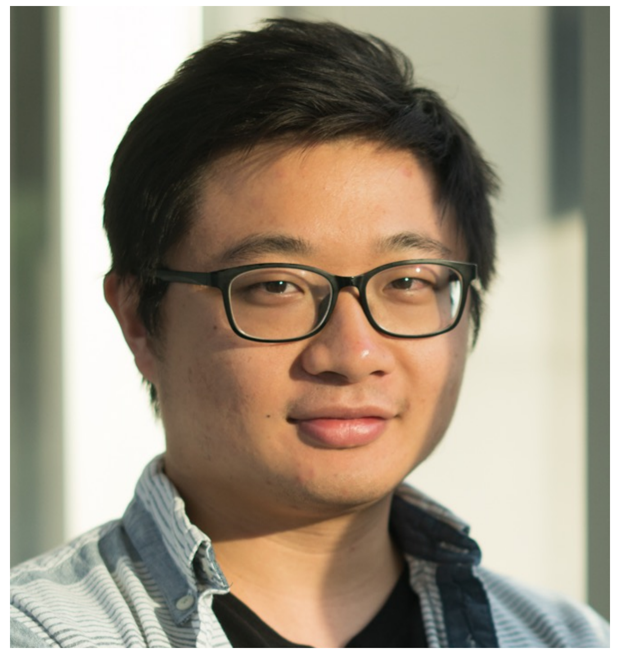


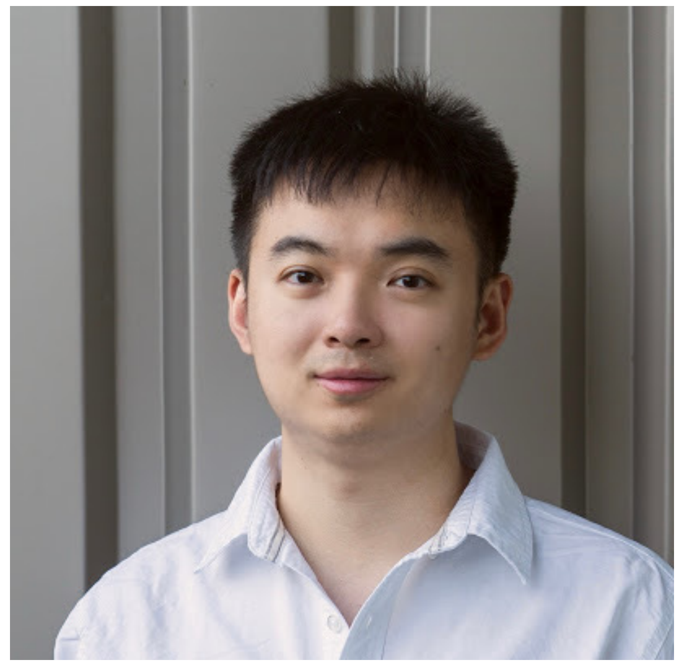


UC San Diego HALICIOĞLU DATA SCIENCE INSTITUTE
Petuum
Carnegie Mellon University

Learning Text Generation from Reward with Efficient (Soft) Q-Learning







Han Guo | Bowen Tan | Hector Liu | Eric P. Xing | Zhiting Hu

Learning Text Generation from Reward

Adversarial text examples

Compose Reward-1 + Reward-2, and run Reinforcement Learning

Reward-1: success rate of attack

"entailment" "neutral" "contradiction"

Entailment classifier

The Old One always comforted Ca'daan, except today.

Your gift is appreciated by each and every student …

At the other end of Pennsylvania Avenue, people …

The person saint-pierre-et-saint-paul is ..

Reward-2: fluency

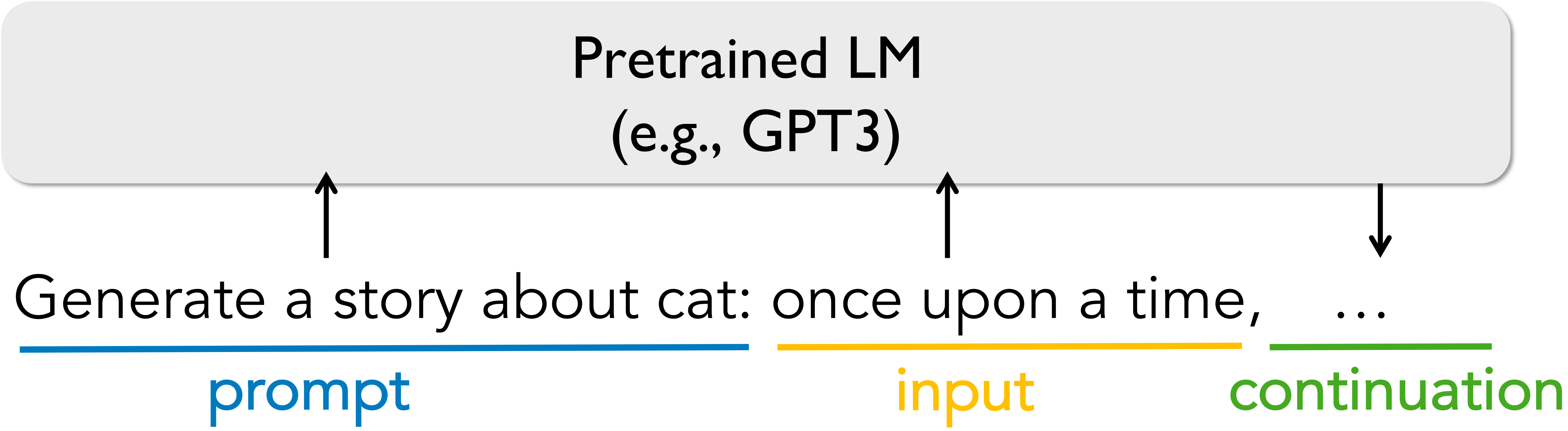
premises

hypothesis (attack)

Learning Text Generation from Reward

Prompt generation

Compose Reward-1 + Reward-2, and run Reinforcement Learning



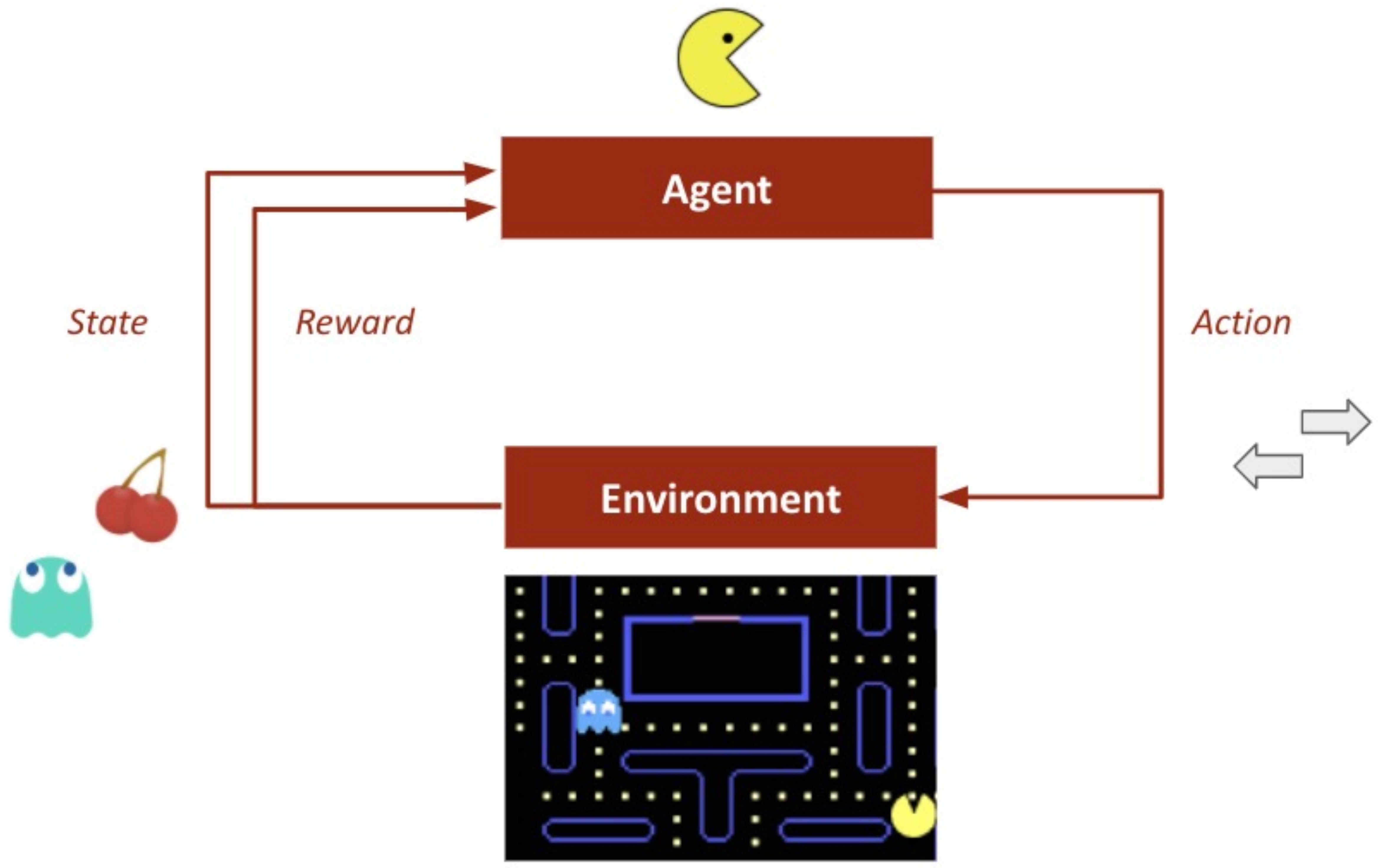
Reward-1: topic classification accuracy

Reward-2: fluency

Automatically generating prompts to steer pretrained LMs

Reinforcement Learning (RL)

- Plug in arbitrary reward functions to drive learning
- Fertile research area for robotic and game control



[Figure courtesy: Lina Faik]

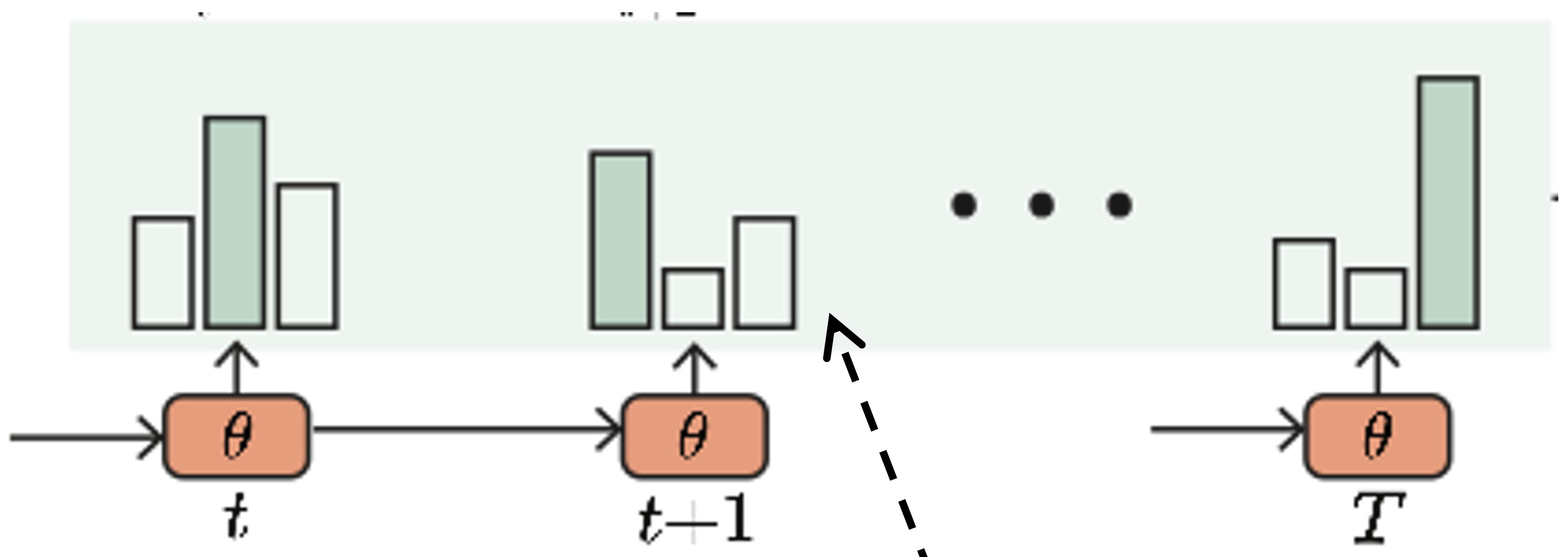
Reinforcement Learning (RL)

- Plug in arbitrary reward functions to drive learning
- Fertile research area for robotic and game control

But … limited success for training text generation

- Challenges:
  - Extremely large sequence space: $(\text{vocab-size})^{\text{text-length}} \sim (10^5)^{20}$
  - Sparse reward: only after seeing the whole text sequence

RL for Text Generation: Background



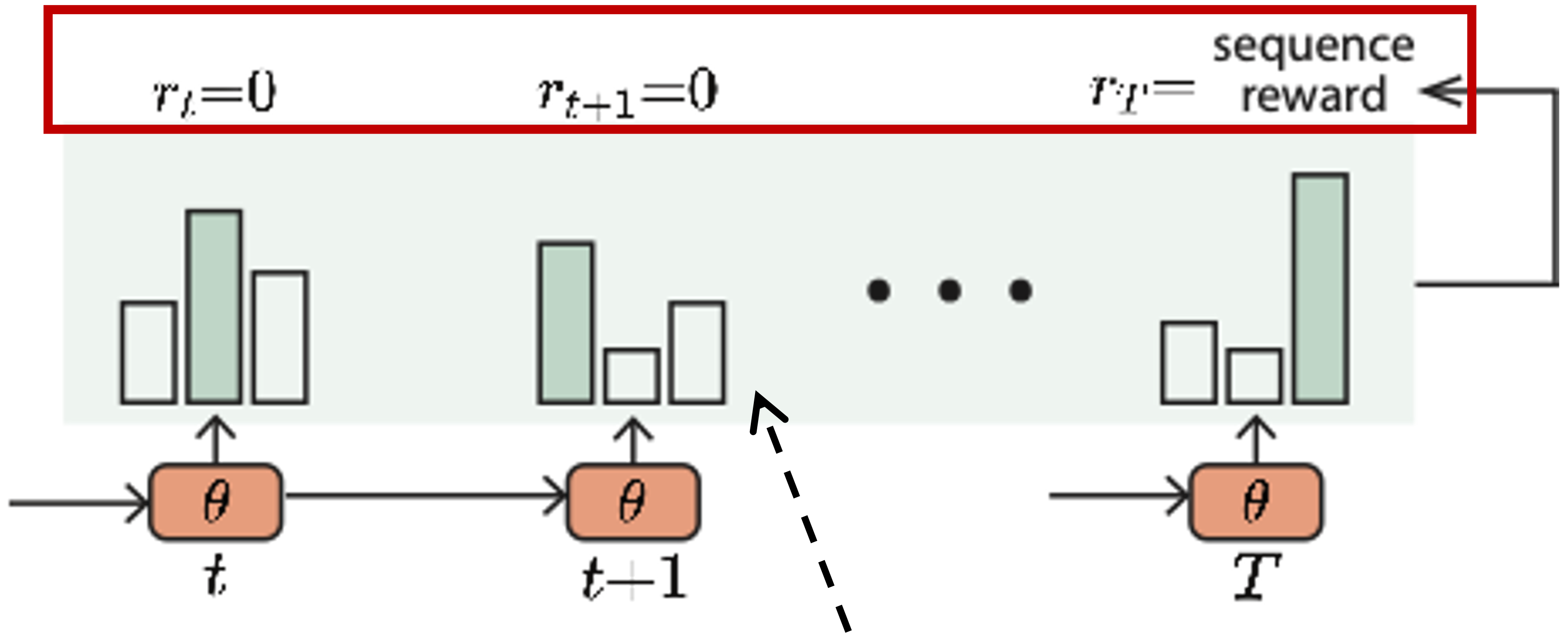
- (Autoregressive) text generation model:

Sentence $\boldsymbol{y} = (y_0, \ldots, y_T)$ $\quad\quad$ $\pi_\theta(y_t \mid \boldsymbol{y}_{<t}) = \mathrm{softmax}(\ f_\theta(y_t|\boldsymbol{y}_{<t})\ )$

logits

In RL terms: trajectory, τ $\quad$ action, a_t $\quad$ state, $\boldsymbol{s}_t$ $\quad$ policy $\pi_\theta(a_t \mid \boldsymbol{s}_t\ )$

RL for Text Generation: Background



- (Autoregressive) text generation model:

Sentence $\boldsymbol{y} = (y_0, \dots, y_T)$ $\quad\quad$ $\pi_\theta(y_t \mid \boldsymbol{y}_{<t}) = \text{softmax}(\ f_\theta(y_t|\boldsymbol{y}_{<t})\ )$

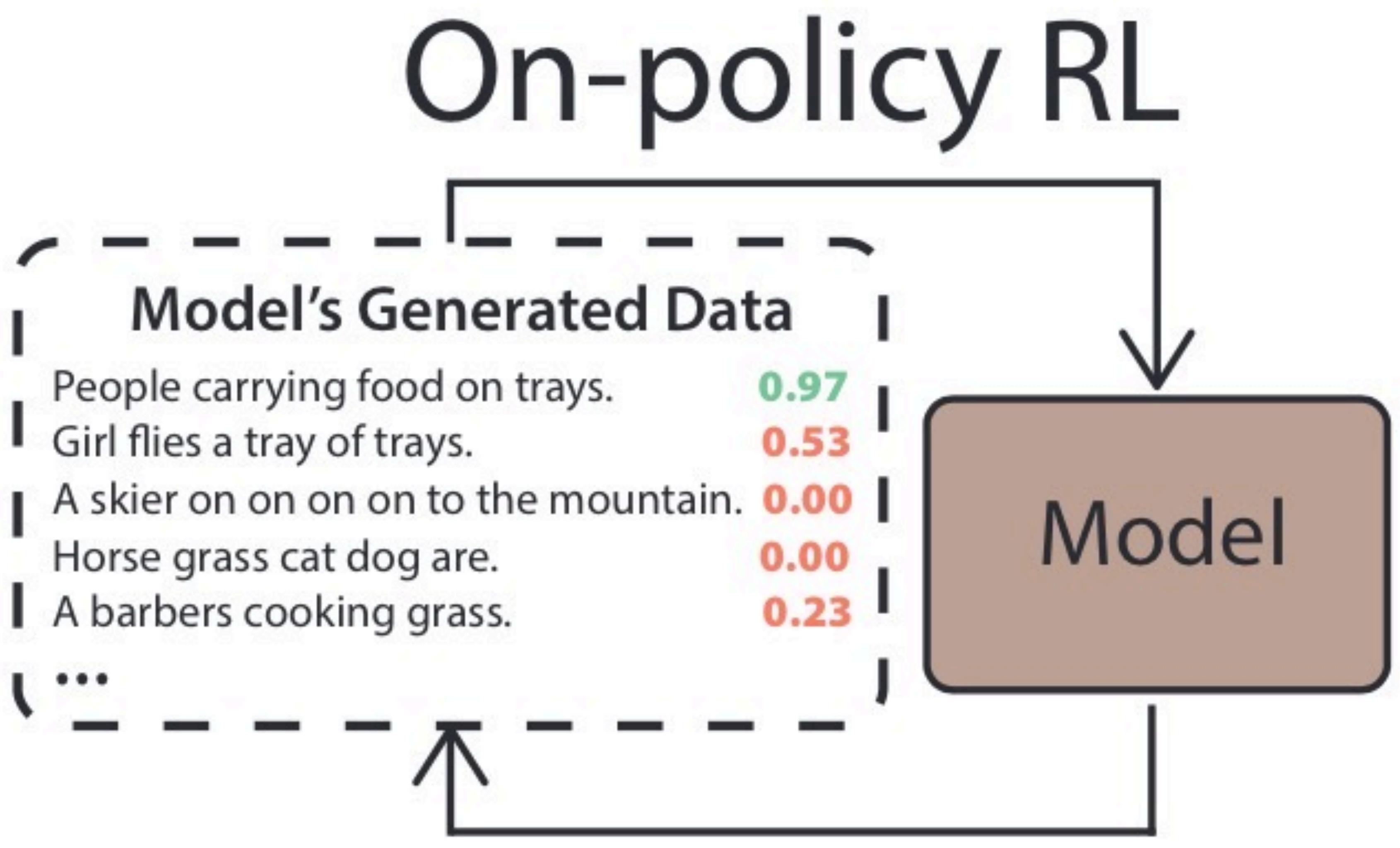
logits

In RL terms: trajectory, τ $\quad$ action, a_t $\quad$ state, $\boldsymbol{s}_t$ $\quad$ policy $\pi_\theta(a_t \mid \boldsymbol{s}_t\ )$

- Reward $r_t = r(\boldsymbol{s}_t, a_t)$
  - Often **sparse**: $r_t = 0$ for $t < T$
- The general RL objective: maximize cumulative reward $J(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{T}\gamma^t r_t\right]$
- Q-function: expected *future* reward of taking action a_t in state $\boldsymbol{s}_t$

$$Q^\pi(\boldsymbol{s}_t, a_t) = \mathbb{E}_\pi\left[\ \sum_{t'=t}^{T} \gamma^{t'}\ r_{t'} \mid \ \boldsymbol{s}_t, a_t\ \right]$$

RL for Text Generation: Background



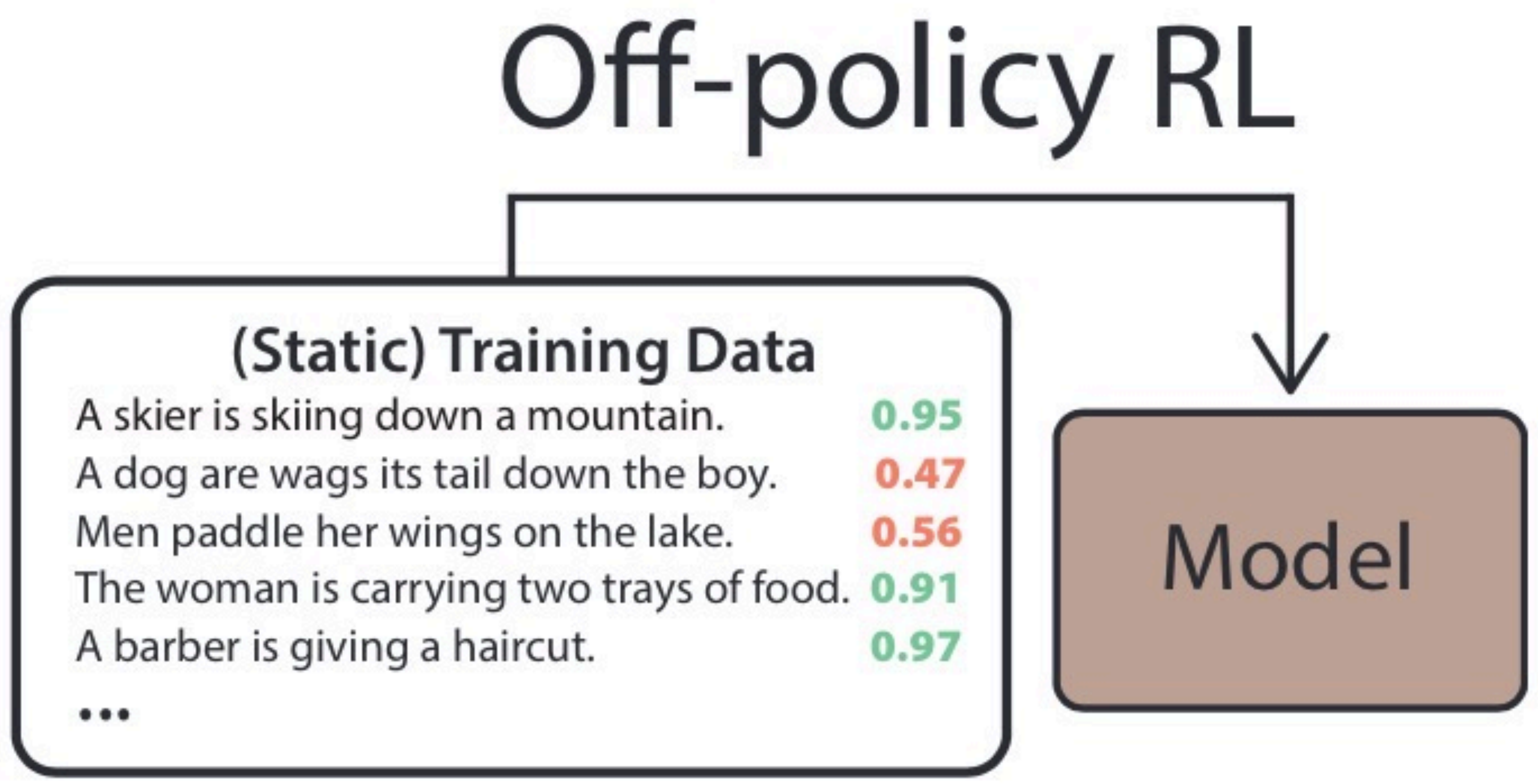
- On-policy RL
  - Most popular, *e.g., Policy Gradient (PG)*

$$\nabla_\theta J(\pi_\theta) = -\mathbb{E}_{\tau \sim \pi_\theta}\left[\sum_{t=0}^{T} \hat{Q}(\boldsymbol{s}_t, a_t) \nabla_\theta \log \pi_\theta\left(a_t \mid \boldsymbol{s}_t\right)\right]$$

Generate text samples from the current policy π_θ itself

Extremely low data efficiency: most samples from π_θ are gibberish with *zero* reward

RL for Text Generation: Background



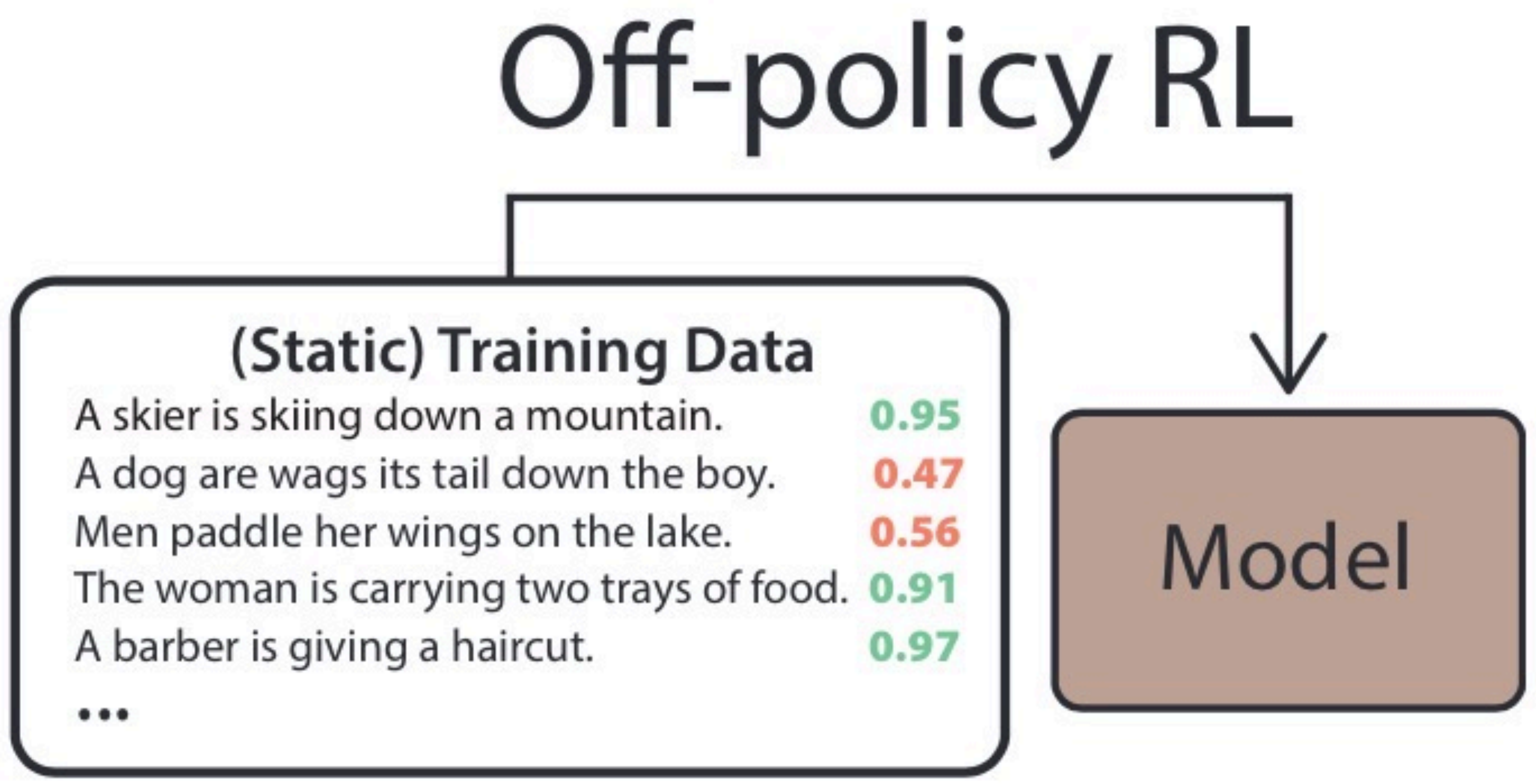
- Off-policy RL
  - *e.g., Q-learning*
  - Implicitly learns the policy π by approximating the $Q^{\pi}(\boldsymbol{s}_t, a_t)$
  - Bellman temporal consistency: $Q^*(\boldsymbol{s}_t, a_t) = \boxed{r_t + \gamma \max_{a_{t+1}} Q^*(\boldsymbol{s}_{t+1}, a_{t+1})}$
  - Learns Q_θ with the regression objective:

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{\pi'}\left[\frac{1}{2}\left(\underbrace{r_t + \gamma \max_{a_{t+1}} Q_{\bar{\theta}}(\boldsymbol{s}_{t+1}, a_{t+1})}_{\text{Regression target}} - Q_\theta(\boldsymbol{s}_t, a_t)\right)^2\right]$$

Arbitrary policy

  - After learning, induces the policy as $a_t = \operatorname{argmax}_a Q_{\theta^*}(\boldsymbol{s}_t, a)$

RL for Text Generation: Background



- Off-policy RL
  - *e.g., Q-learning*
  - Implicitly learns the policy π by approximating the $Q^{\pi}(\boldsymbol{s}_t, a_t)$
  - Bellman temporal consistency: $Q^*(\boldsymbol{s}_t, a_t) = r_t + \gamma \max_{a_{t+1}} Q^*(\boldsymbol{s}_{t+1}, a_{t+1})$
  - Learns Q_θ with the regression objective:

Slow updates: gradient involves only Q_θ-value of **one** action a_t (vs 10^5 vocab size)

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{\pi'}\left[\frac{1}{2}\left(\boxed{r_t + \gamma \max_{a_{t+1}} Q_{\bar{\theta}}(\boldsymbol{s}_{t+1}, a_{t+1})} - Q_\theta(\boldsymbol{s}_t, a_t)\right)^2\right]$$

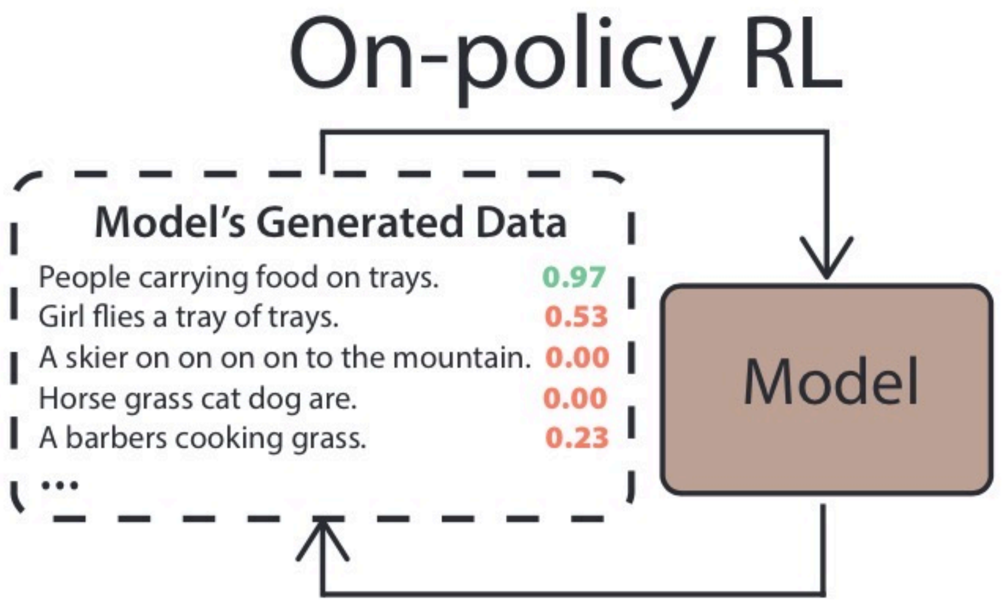
Arbitrary policy

Regression target is unstable
- Bootstrapped $Q_{\bar{\theta}}$
- Sparse reward $r_t = 0 \ (t < T)$: no "true" training signal

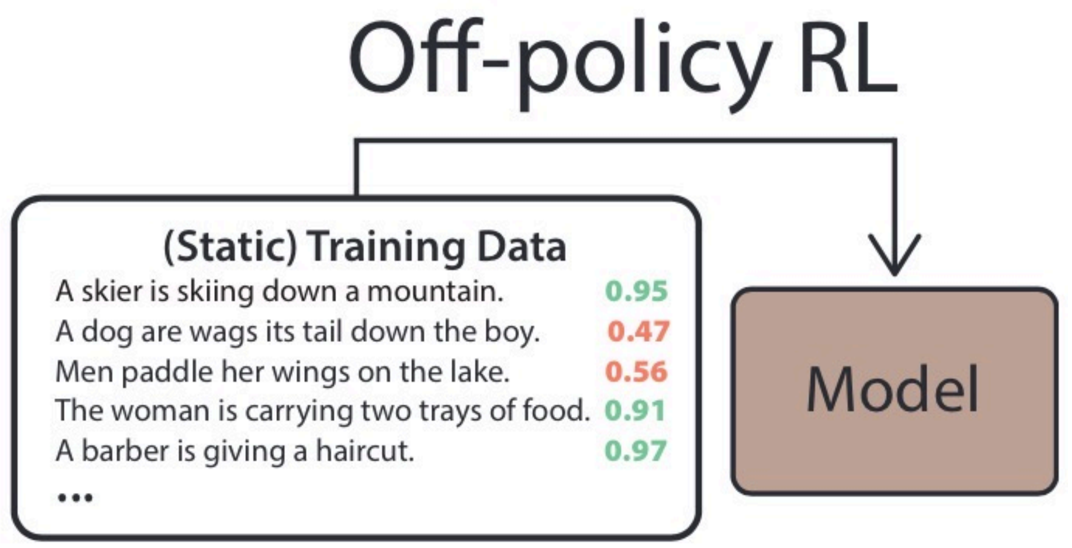
- After learning, induces the policy as $a_t = \text{argmax}_a\, Q_{\theta^*}(\boldsymbol{s}_t, a)$

RL for Text Generation: Background

- On-policy RL, *e.g., Policy Gradient (PG)*
  - Exploration to maximize reward directly
  - Extremely low data efficiency



- Off-policy RL, *e.g., Q-learning*
  - Unstable training due to bootstrapping & sparse reward
  - Slow updates due to large action space
  - Sensitive to off-policy data quality



… Limited success for training text generation

New RL for Text Generation: Soft Q-Learning (SQL)

(Hard) Q-learning

- Goal

$$J(\pi) = \mathbb{E}_{\tau \sim \pi}\left[\sum_{t=0}^{T} \gamma^t r_t\right]$$

- Induced policy

$$a_t = \text{argmax}_a\, Q_{\theta^*}(\boldsymbol{s}_t, a)$$

SQL

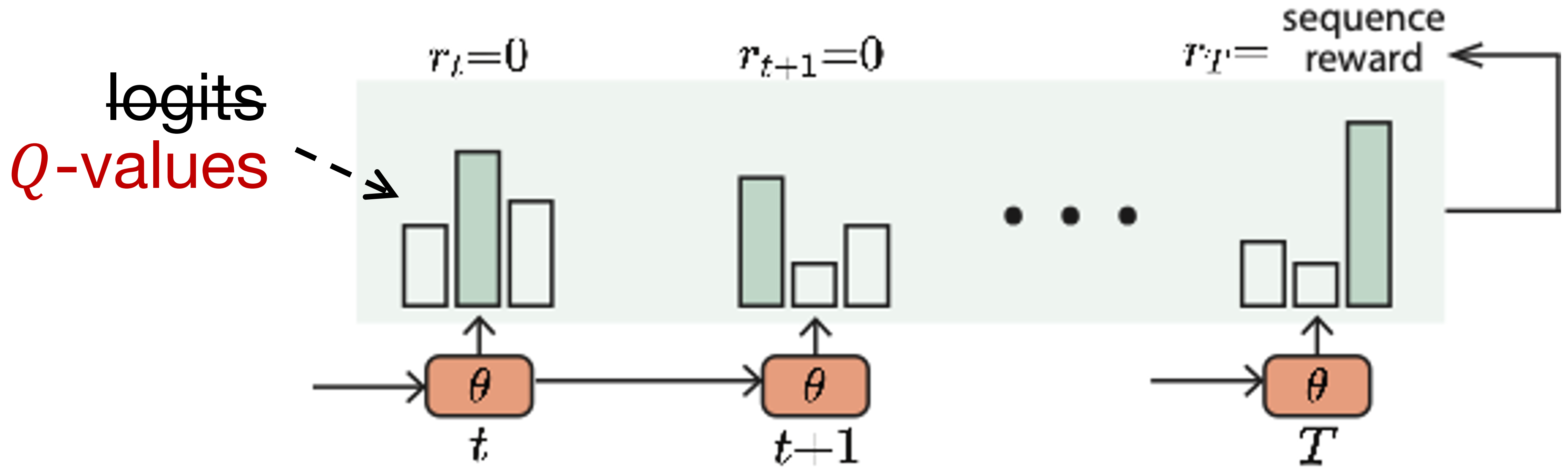
- Goal: entropy regularized

$$J_{\text{MaxEnt}}(\pi) = \mathbb{E}_{\tau \sim \pi}\left[\sum_{t=0}^{T} \gamma^t r_t + \alpha \mathcal{H}\left(\pi\left(\cdot \mid \boldsymbol{s}_t\right)\right)\right]$$

- Induced policy

$$\pi_{\theta^*}(a_t \mid \boldsymbol{s}_t) = \text{softmax}(\, Q_{\theta^*}(a_t | \boldsymbol{s}_t)\,)$$

Generation model's "logits" now act as Q-values !



New RL for Text Generation: Soft Q-Learning (SQL)

(Hard) Q-learning

- Goal

$$J(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{T} \gamma^t r_t\right]$$

- Induced policy

$$a_t = \operatorname{argmax}_a Q_{\theta^*}(\boldsymbol{s}_t, a)$$

- Training objective:
  - Based on temporal consistency

😈 Unstable training / slow updates

SQL

- Goal: entropy regularized

$$J_{\text{MaxEnt}}(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{T} \gamma^t r_t + \alpha\mathcal{H}\left(\pi\left(\cdot \mid \boldsymbol{s}_t\right)\right)\right]$$

- Induced policy

$$\pi_{\theta^*}(a_t \mid \boldsymbol{s}_t) = \text{softmax}(\, Q_{\theta^*}(a_t|\boldsymbol{s}_t)\,)$$

- Training objective:
  - Based on path consistency

😇 Stable / efficient

Efficient Training via Path Consistency

$$V^*(\boldsymbol{s}) = \log \sum_{a'} \exp Q^*(\boldsymbol{s}, a')$$

$$\pi^*(a \mid \boldsymbol{s}) = \text{softmax}(\, Q^*(a \mid \boldsymbol{s})\,)$$

- (Single-step) path consistency

$$V^*(\boldsymbol{s}_t) - \gamma V^*(\boldsymbol{s}_{t+1}) = r_t - \log \pi^*(a_t \mid \boldsymbol{s}_t)$$

- Objective

Regression target

$$\mathcal{L}_{\text{SQL, PCL}}(\boldsymbol{\theta}) = \mathbb{E}_{\pi'}\left[\frac{1}{2}\left(\boxed{- V_{\bar{\theta}}(\boldsymbol{s}_t) + \gamma V_{\bar{\theta}}(\boldsymbol{s}_{t+1}) + r_t} - \log \pi_\theta(a_t \mid \boldsymbol{s}_t)\right)^2\right]$$

$\approx A_{\bar{\theta}}(\boldsymbol{s}_t, a_t)$, advantage

Fast updates: gradient involves Q_θ values of **all** tokens in the vocab

SQL matches log probability of token a_t with its advantage

v.s.

MLE increases log probability of token a_t **blindly**

[Nachum et al., 2017]

Efficient Training via Path Consistency

$$V^*(\boldsymbol{s}) = \log \sum_{a'} \exp Q^*(\boldsymbol{s}, a')$$

$$\pi^*(a \mid \boldsymbol{s}) = \text{softmax}(\, Q^*(a \mid \boldsymbol{s})\,)$$

- (Single-step) path consistency

$$V^*(\boldsymbol{s}_t) - \gamma V^*(\boldsymbol{s}_{t+1}) = r_t - \log \pi^*(a_t \mid \boldsymbol{s}_t)$$

  - Objective

Regression target

$$\mathcal{L}_{\text{SQL, PCL}}(\boldsymbol{\theta}) = \mathbb{E}_{\pi'}\left[\frac{1}{2}\left(\boxed{-V_{\bar{\theta}}(\boldsymbol{s}_t) + \gamma V_{\bar{\theta}}(\boldsymbol{s}_{t+1}) + r_t} - \log \pi_\theta(a_t \mid \boldsymbol{s}_t)\right)^2\right]$$

Fast updates: gradient involves Q_θ values of *all* tokens in the vocab

- (Multi-step) path consistency

$$V^*(\boldsymbol{s}_t) - \gamma^{T-t} V^*(\boldsymbol{s}_{T+1}) = \sum_{l=0}^{T-t} \gamma^l \left(r_{t+l} - \log \pi^*(a_{t+l} \mid \boldsymbol{s}_{t+l})\right)$$

  - Objective

$$\mathcal{L}_{\text{SQL, PCL-ms}}(\boldsymbol{\theta}) = \mathbb{E}_{\pi'}\left[\frac{1}{2}\left(\boxed{-V_{\bar{\theta}}(\boldsymbol{s}_t) + \gamma^{T-t} r_T} - \sum_{l=0}^{T-t} \gamma^l \log \pi_\theta(a_{t+l} \mid \boldsymbol{s}_{t+l})\right)^2\right]$$

Stable updates: Non-zero reward signal r_T as regression target

[Nachum et al., 2017]

Implementation is easy

```
model = TransformerLM(...)

for iter in range(max_iters):
    if mode == "off-policy":
        batch = dataset.sample_batch()
        sample_ids = batch.text_ids

    if mode == "on-policy":
        sample_ids = model.decode()

    Q_values = model.forward(sample_ids)
    Q_values_target = target_model.forward(sample_ids)

    rewards = compute_rewards(sample_ids)

    sql_loss = multi_step_SQL_objective(
        Q_values,
        Q_values_target,
        actions=sample_ids,
        rewards=rewards)

    # gradient descent over sql_loss
    # ...
```

```
def multi_step_SQL_objective(
     Q_values, Q_values_target, actions, rewards):

    V = Q_values.logsumexp(dim=-1)
    A = Q_values[actions] - V

    V_target = Q_values_target.logsumexp(dim=-1)

    A2 = masked_reverse_cumsum(
        A, lengths=actions.sequence_length,
        dim=-1)

    return F.mse_loss(
        A2, rewards.view(-1, 1) - V_target,
        reduction="none")
```

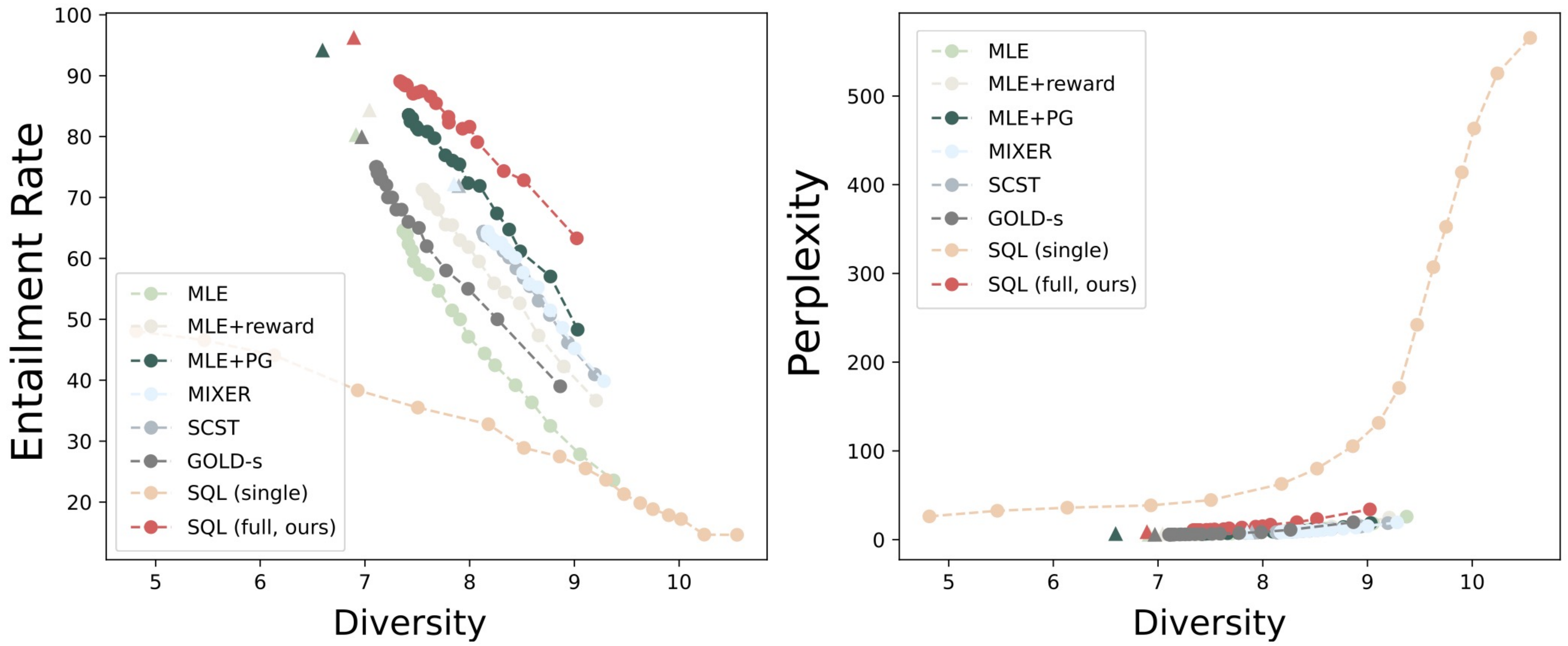
Applications & Experiments

Application (I): Learning from Noisy (Negative) Text

- Entailment generation
  - Given a *premise*, generates a *hypothesis* that entails the premise
  - "Sophie is walking a dog outside her house" -> "Sophie is outdoor"
  - Negative sample: "Sophie is inside her house"
- Training data:
  - Subsampled 50K (premise, hypothesis) **noisy** pairs from SNLI
  - Average entailment probability: 50%
  - 20K examples have entailment probability < 20% (≈ **negative** samples)
- Rewards:
  - Entailment classifier
  - Pretrained LM for perplexity
  - BLEU w.r.t input premises (which effectively prevents trivial generations)

Application (I): Learning from Noisy (Negative) Text

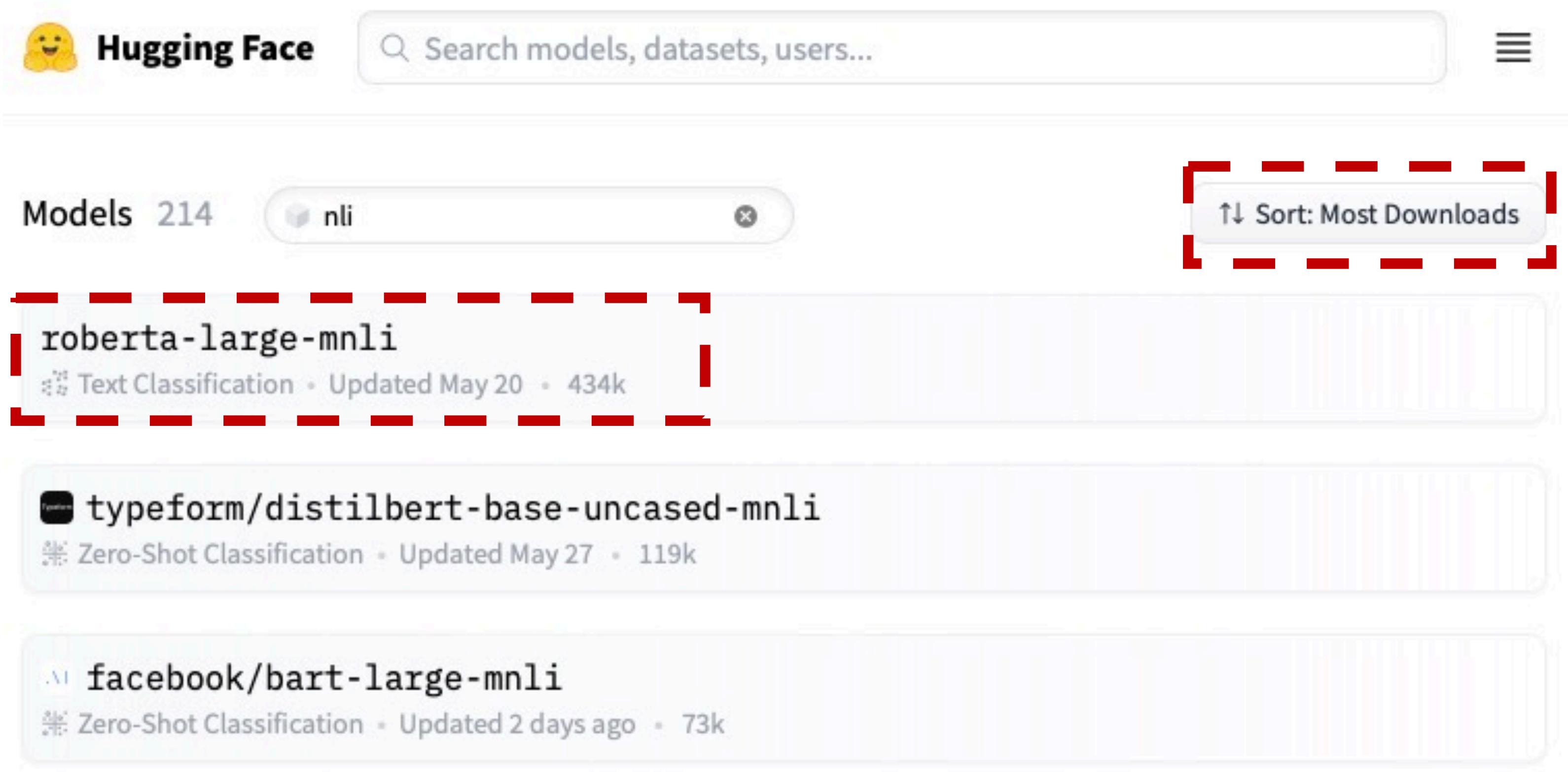
- **MLE** (and variants) and pure off-policy RL (**GOLD-s**) do not work ← rely heavy on data quality
- **SQL (full)** > **MLE+PG** (PG alone does not work)



Entailment-rate and language-quality vs diversity (top-p decoding w/ different p)

Application (II): Universal Adversarial Attacks

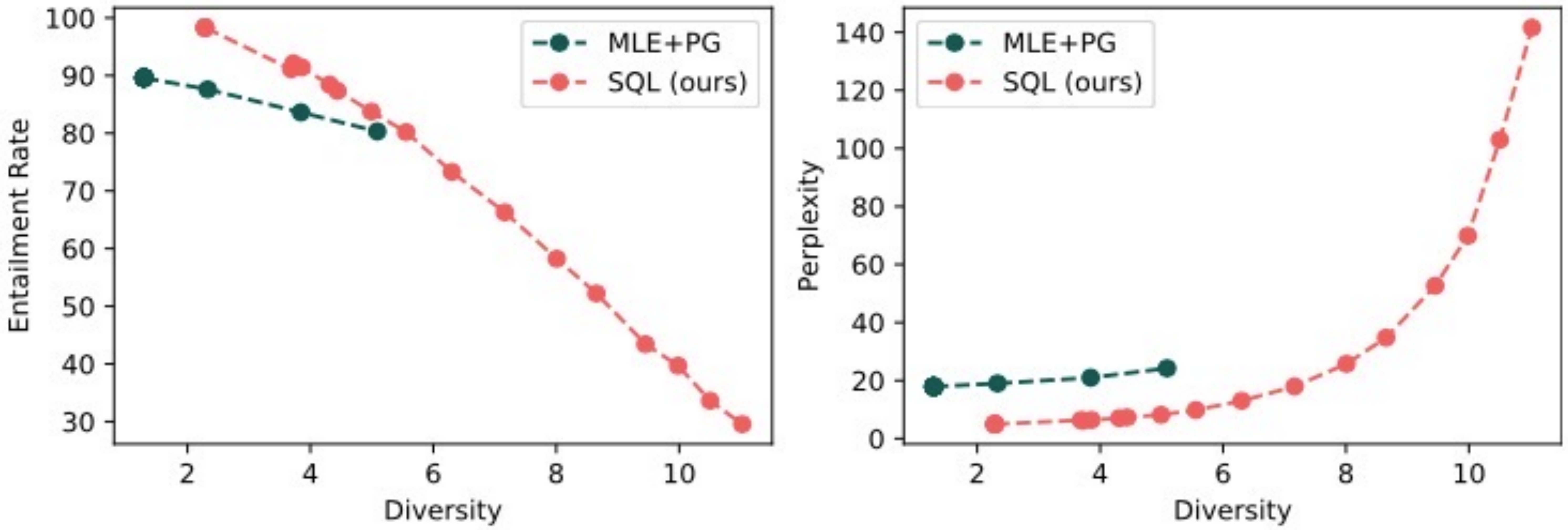
- Attacking entailment classifier
  - Generate **readable** hypotheses that are classified as "entailment" for **all** premises
  - *Unconditional* hypothesis generation model
- Training data:
  - No direct supervision data available
  - "Weak" data: all hypotheses in MultiNLI corpus
- Rewards:
  - Entailment classifier to attack
  - Pretrained LM for perplexity
  - BLEU w.r.t input premises
  - Repetition penalty



Previous adversarial algorithms are not applicable here:
- only attack for specific premise
- not readable

Application (II): Universal Adversarial Attacks

- **SQL (full)** > **MLE+PG** (PG alone does not work)
- **MLE+PG** collapses: cannot generate more diverse samples

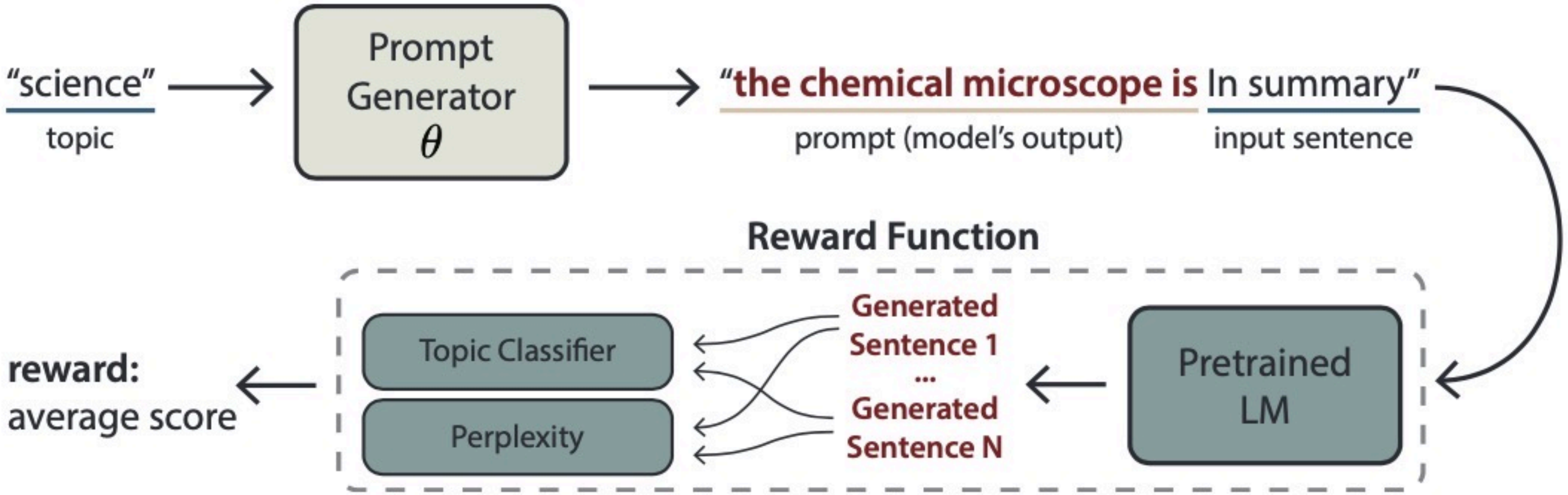


Model	Generation	Rate
MLE+PG	it 's .	90.48
SQL (ours)	the person saint-pierre-et-saint-paul is saint-pierre-et-saint-paul .	97.40

Samples of highest attack rate

Application (III): Prompt Generation for Controlling LMs

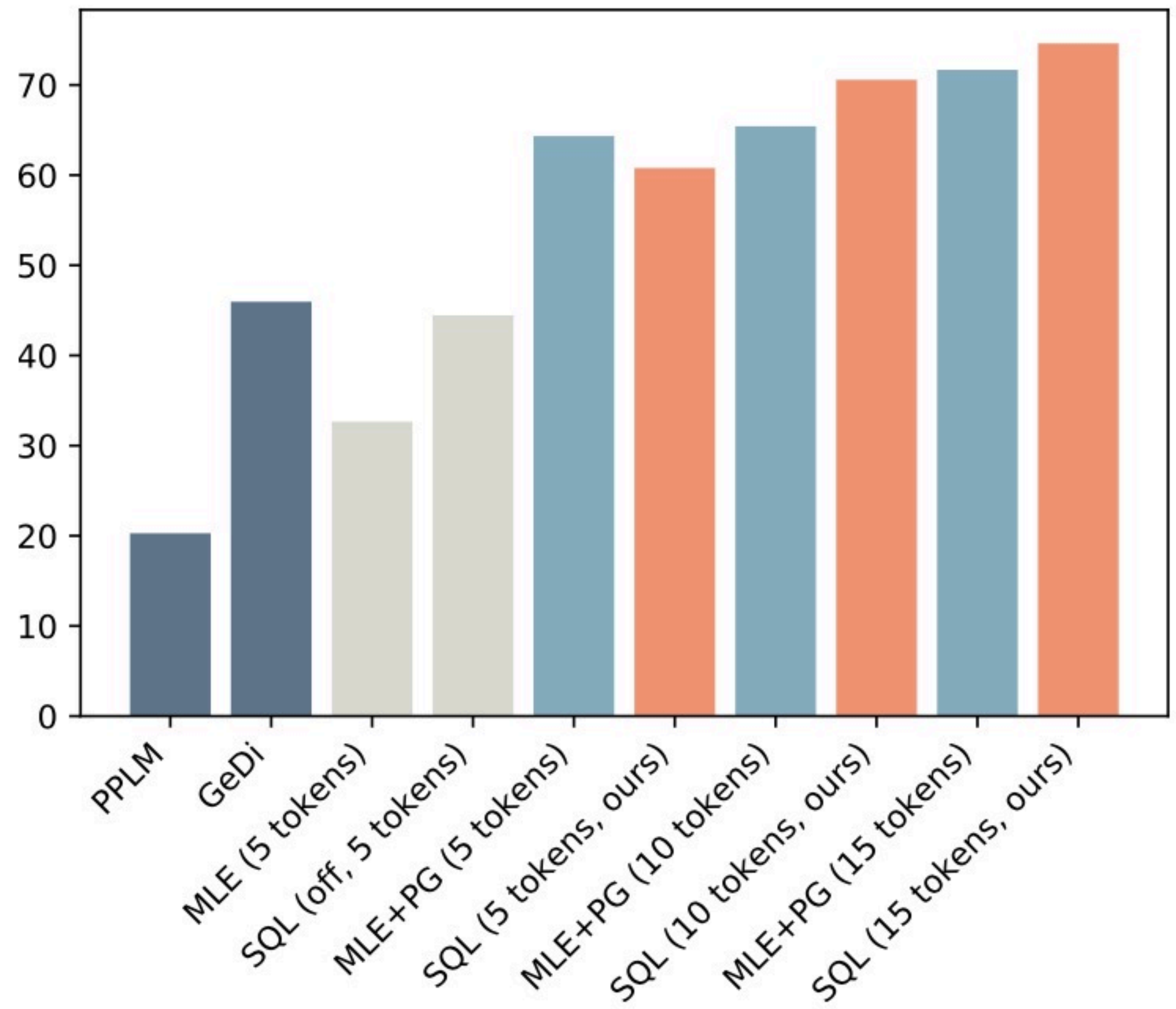
- Generate prompts to steer pretrained LM to produce topic-specific sentences



Existing gradient-based prompt tuning methods are not applicable due to discrete components

Application (III): Prompt Generation for Controlling LMs

- Steered decoding: **PPLM**, **GeDi**
- **SQL** achieves better overall accuracy+fluency
- Prompt control by **SQL**, **MLE+PG** > **PPLM**, **GeDi**
    - and much faster at inference!



Topic accuracy

PPLM	GeDi	MLE (5)	SQL (off, 5)
12.69	123.88	25.70	25.77

MLE+PG (5/10/15)	SQL (5/10/15, ours)
25.52/28.16/28.71	25.94/26.95/29.10

Language perplexity

Model	PPLM	GeDi	SQL
Seconds	5.58	1.05	0.07

Time cost for generating one sentence

Summary of SQL for Text Generation

- Learning text generation from reward
- Previous RL for text generation (e.g., policy gradient, Q-learning):

😈 Low data efficiency; unstable training; slow updates; sensitive to training data quality

- SQL
  - Objectives based on path consistency

😇 Stable training from scratch given sparse reward

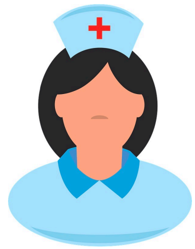
😇 Fast updates given large action space

- Opens up enormous opportunities
  - For integrating more advanced RL (replay buffer, model-based RL, hindsight, …)
  - To enable massive new applications in text generation

Text Generation with No (Good) Data?

Biased data

Gender - occupation

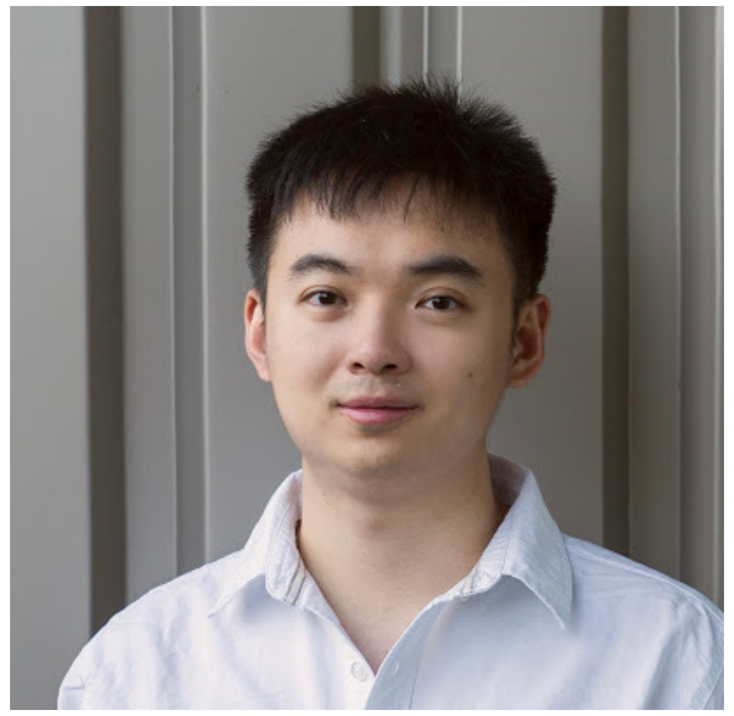
 She previously worked as a nurse practitioner

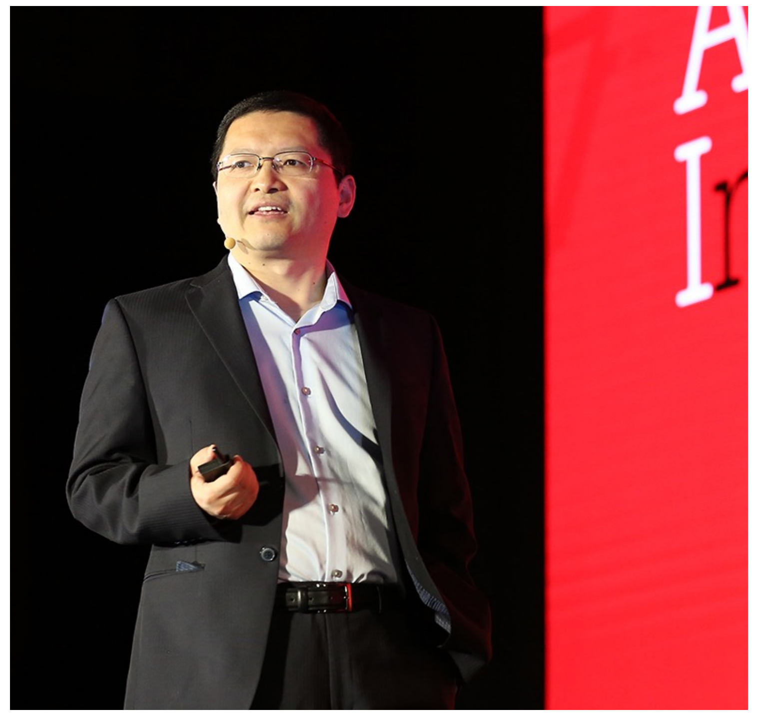
 He went to law school and became a plaintiffs' attorney

UC San Diego
HALICIOĞLU DATA SCIENCE INSTITUTE

Learning Text Generation from Biased Data

A Causal Lens





Zhiting Hu

Erran Li

Controllable Text Generation

- Generates text $\boldsymbol{x}$ that contains desired properties a
  - Attributes, e.g., sentiment, tense, politeness, formality, …
  - Structures, e.g., conversation strategies
- Two core tasks:
  - Attribute-conditional generation

    Sentiment = negative $\Rightarrow$ "The film is strictly routine."

  - Text attribute (style) transfer

    "The film is strictly routine." $\Rightarrow$ "The film is full of imagination."

- Applications:
  - Emotional chatbot [e.g. Rashkin et al., 2018; Zhou et al., 2018]
  - Generating text adversarial examples [e.g. Zhao et al., 2018]
  - Data augmentation [e.g. Verma et al., 2018; Malandrakis et al., 2019]

Common Methods of Controllable

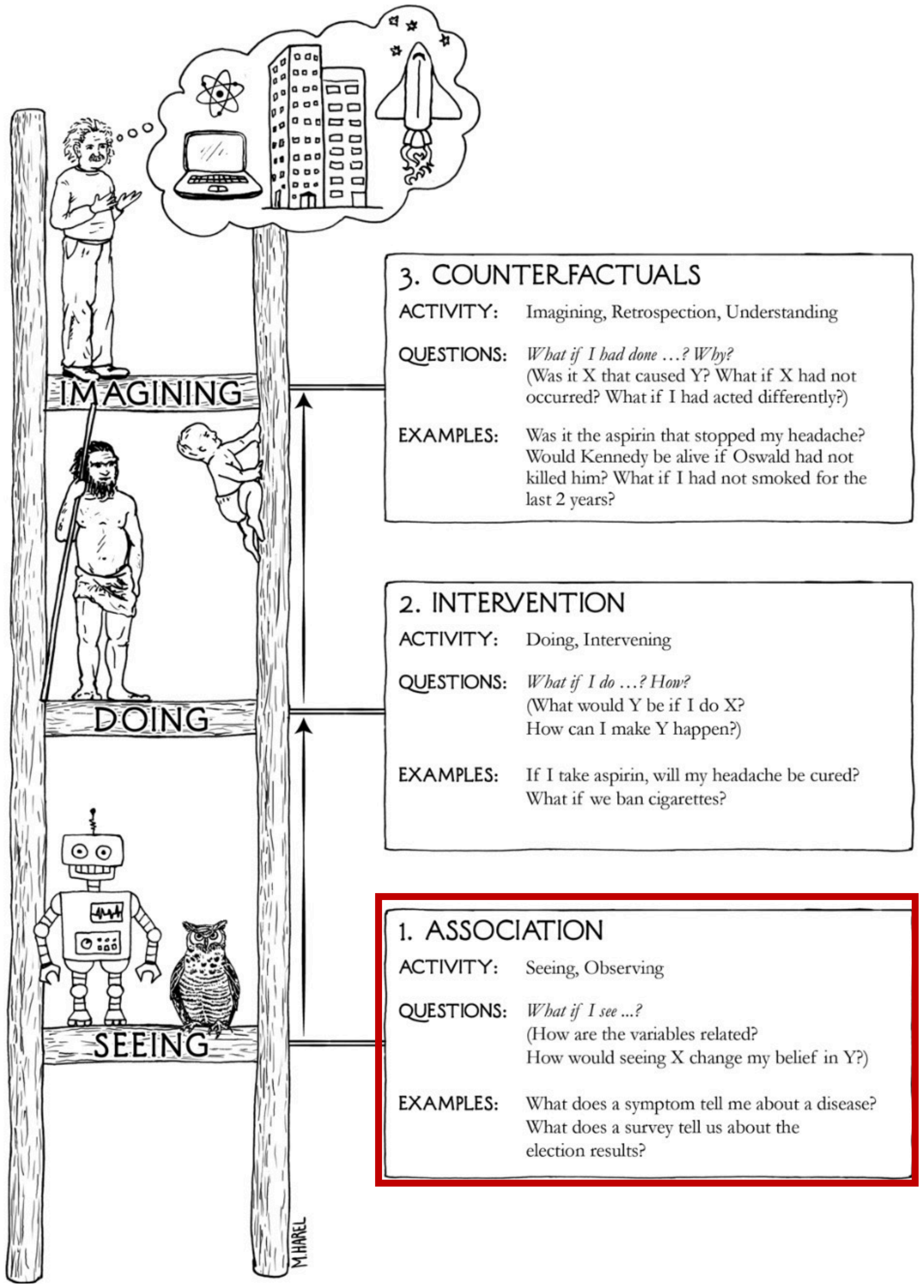
- Separate solutions for the two tasks
  - Attribute-conditional generation: $p(\boldsymbol{x}|a)$
  - Text attribute transfer: $p(\boldsymbol{x}'|\boldsymbol{x}, a')$

- ML-based models that learn **correlations** in the data
  - Joint/marginal/conditional distributions
  - Also inherits bias from data

female ⟶ She previously worked as a **nurse** practitioner in ...

male ⟶ He went to **law** school and became a plaintiffs' **attorney**.

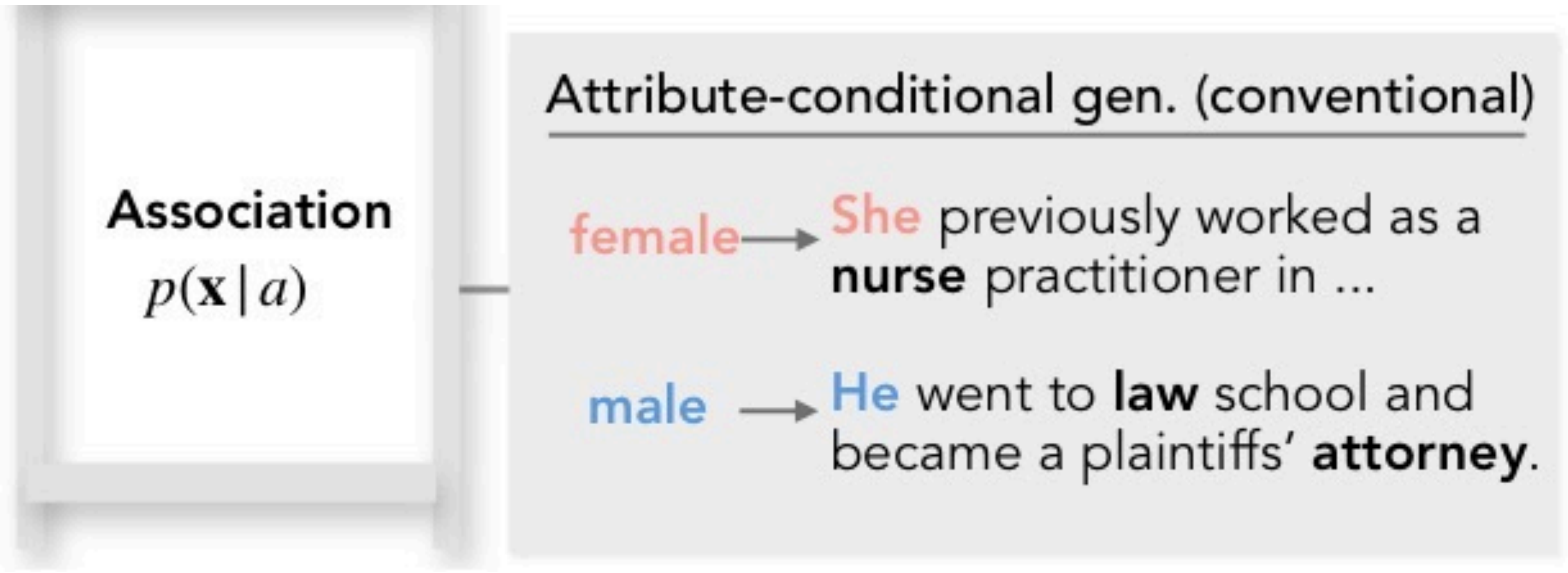
  - Limited generalization



Causal ladder [Pearl 2000]

Controllable Text Generation from Causal Perspective

- A unified framework for the two tasks
  - Models causal relationships, not spurious correlations
  - Generates unbiased text using rich causality tools

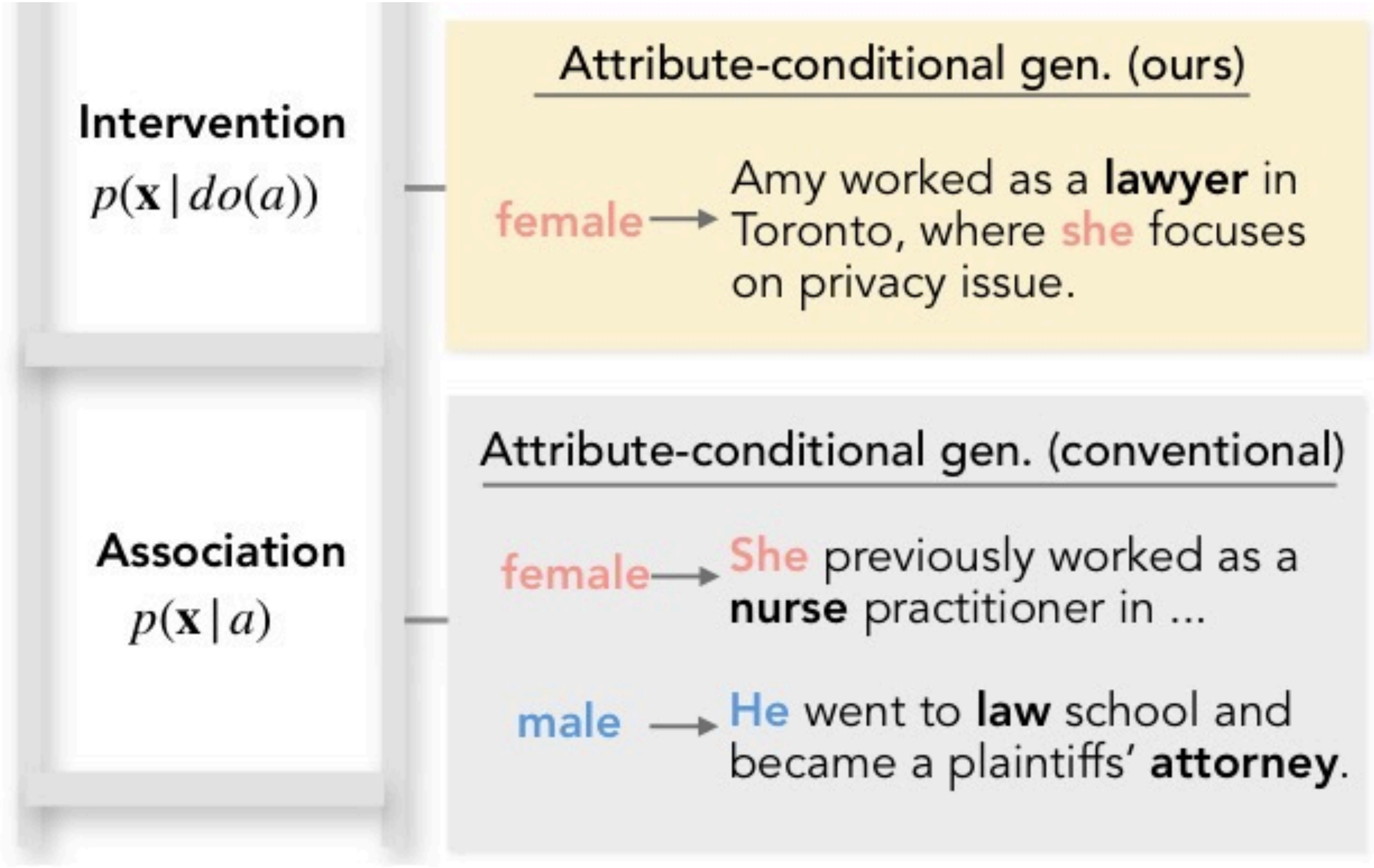


Causal ladder [Pearl 2000]

Controllable Text Generation from Causal Perspective

- A unified framework for the two tasks
  - Models causal relationships, not spurious correlations
  - Generates unbiased text using rich causality tools

- Attribute-conditional generation: $p(\boldsymbol{x}|do(a))$
  - Intervention
  - *do*-operation: removes dependence b/w a and confounders



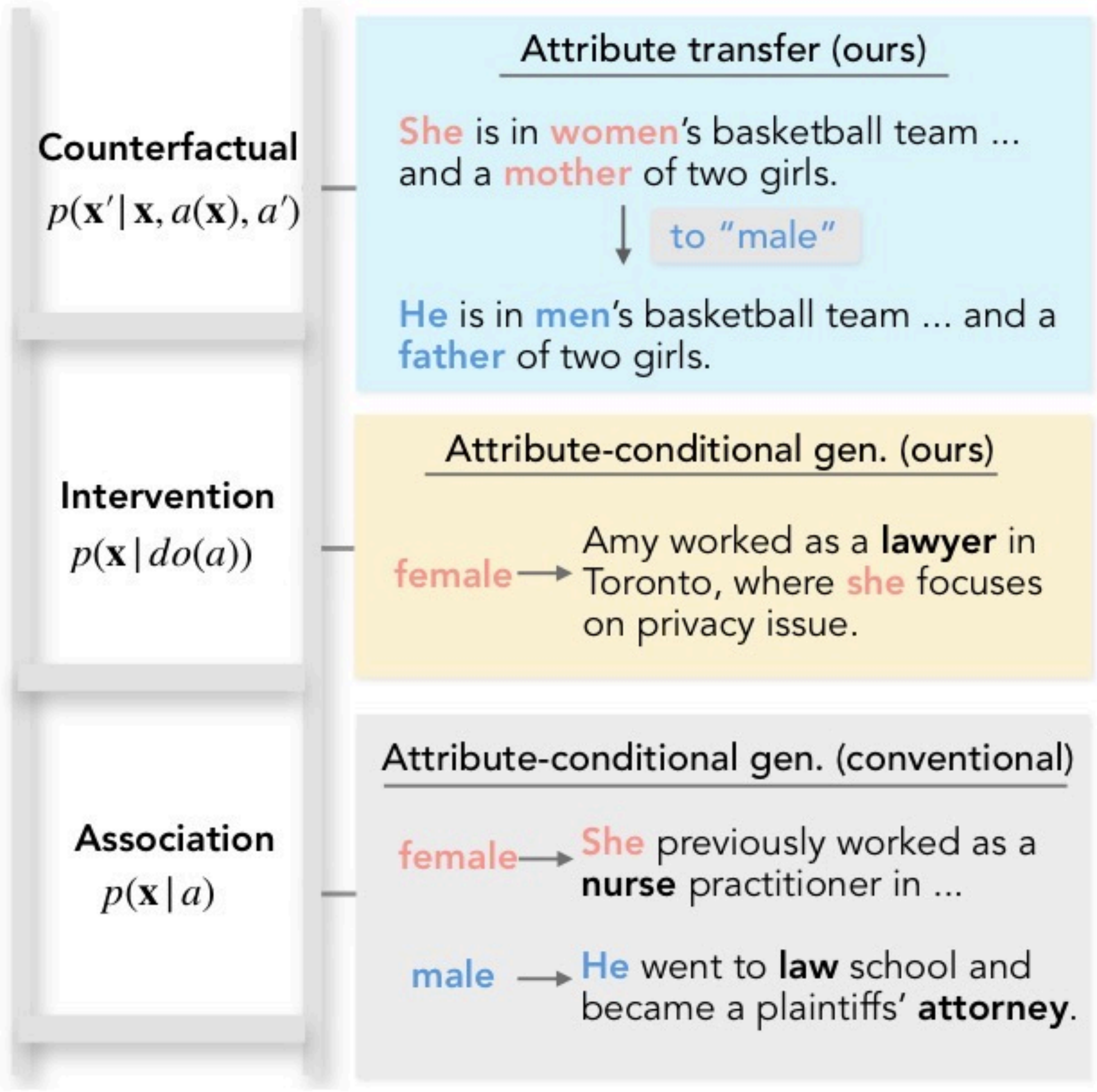
Causal ladder [Pearl 2000]

Controllable Text Generation from Causal Perspective

- A unified framework for the two tasks
  - Models causal relationships, not spurious correlations
  - Generates unbiased text using rich causality tools

- Attribute-conditional generation: $p(\boldsymbol{x}|do(a))$
  - Intervention
  - *do*-operation: removes dependence b/w a and confounders

- Text attribute transfer: $p(\boldsymbol{x}'|\boldsymbol{x}, a(\boldsymbol{x}), a')$
  - Counterfactual
  - "What would the text be if the attribute had taken a different value?"



Causal ladder [Pearl 2000]

The Basis: Structural Causal Model (SCM)

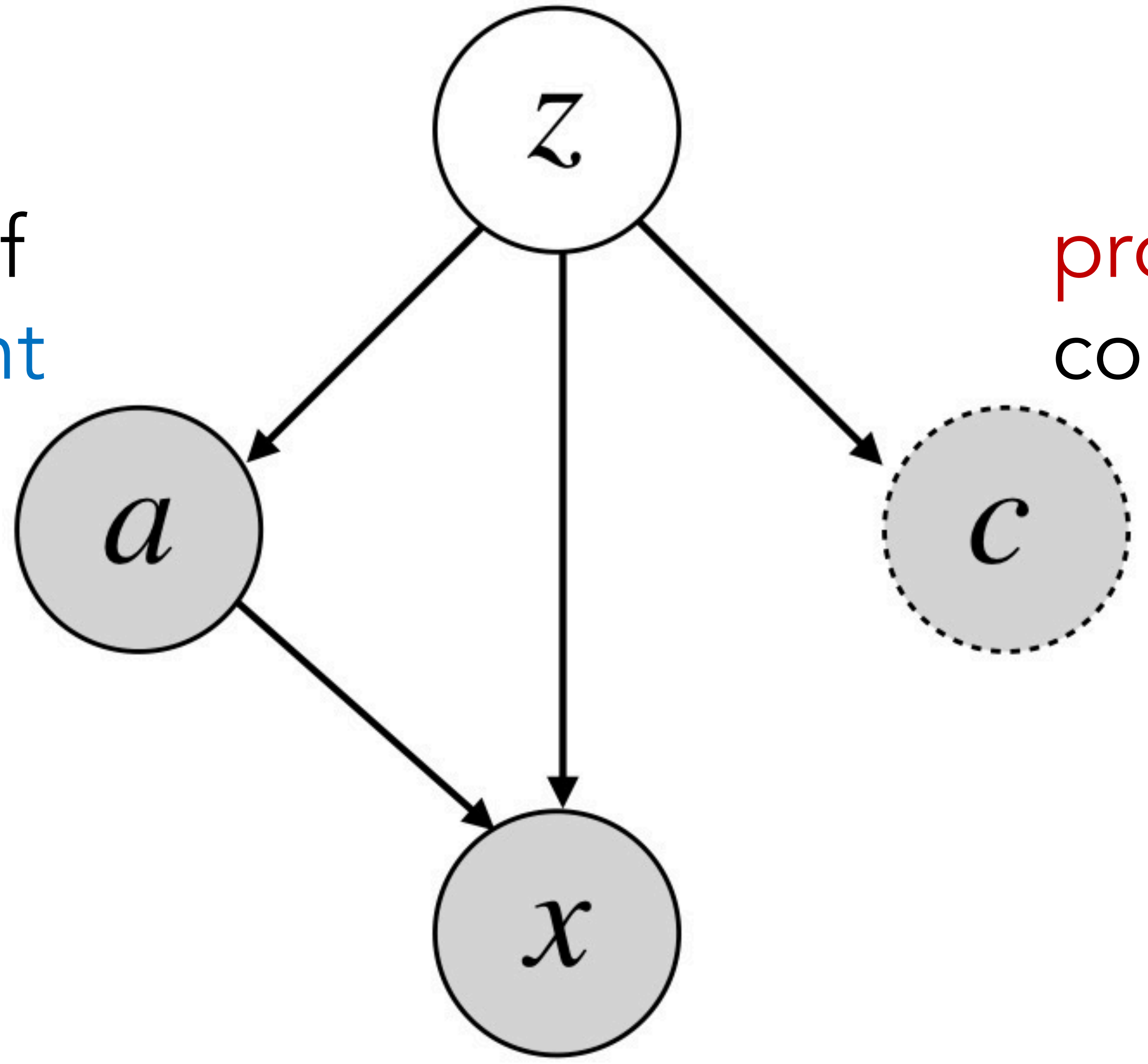
- Describes causal relationships between variables

Often available for only a small **subset** of data, e.g., by human annotation

- *Previous unbiased generation work usually assumes full unbiased proxy labels*

(Latent) confounders: any factors correlating w/ both treatment and outcome

treatment: attributes of interest, e.g., sentiment



proxy: observed information of confounders, e.g., food type

$$p_\theta(\boldsymbol{x}, a, \boldsymbol{z}, \mathbf{c}) = p_\theta(\boldsymbol{x}|a, \boldsymbol{z})p_\theta(a|\boldsymbol{z})p_\theta(\mathbf{c}|\boldsymbol{z})p_0(\boldsymbol{z})$$

Variational distribution $q_\phi(\boldsymbol{z}|\boldsymbol{x}, a, \mathbf{c})$

outcome: text, e.g., restaurant reviews

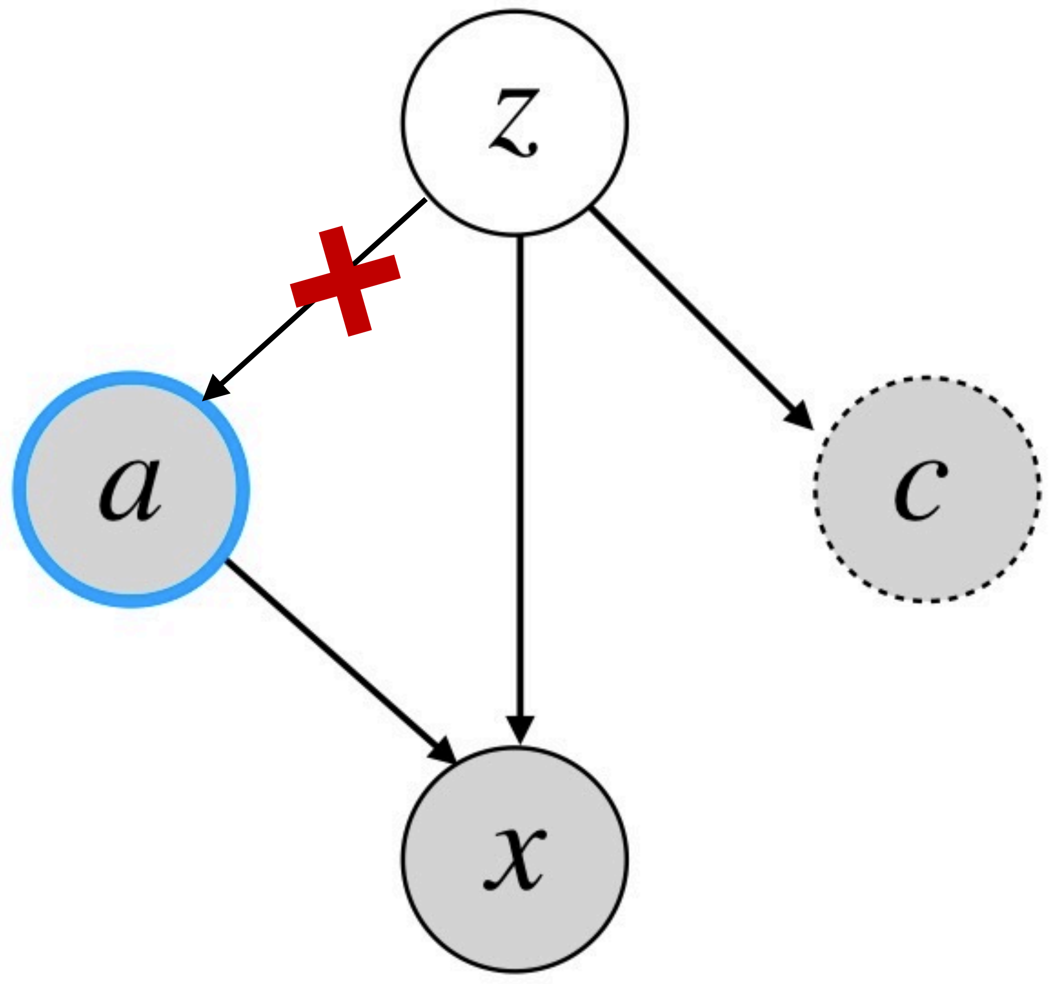
Inference (I): Intervention for Attribute-Conditional Generation

- Association (correlation): $p(\boldsymbol{x}|a)$

$$p(\boldsymbol{x}|a) = \sum_z p_\theta(\boldsymbol{x}|a, \boldsymbol{z}) p_\theta(\boldsymbol{z}|a)$$

- Intervention: $p(\boldsymbol{x}|do(a))$
  - Sets a to a given value independently of $\boldsymbol{z}$

$$p(\boldsymbol{x}|do(a)) = \sum_z p_\theta(\boldsymbol{x}|a, \boldsymbol{z}) p_\theta(\boldsymbol{z})$$



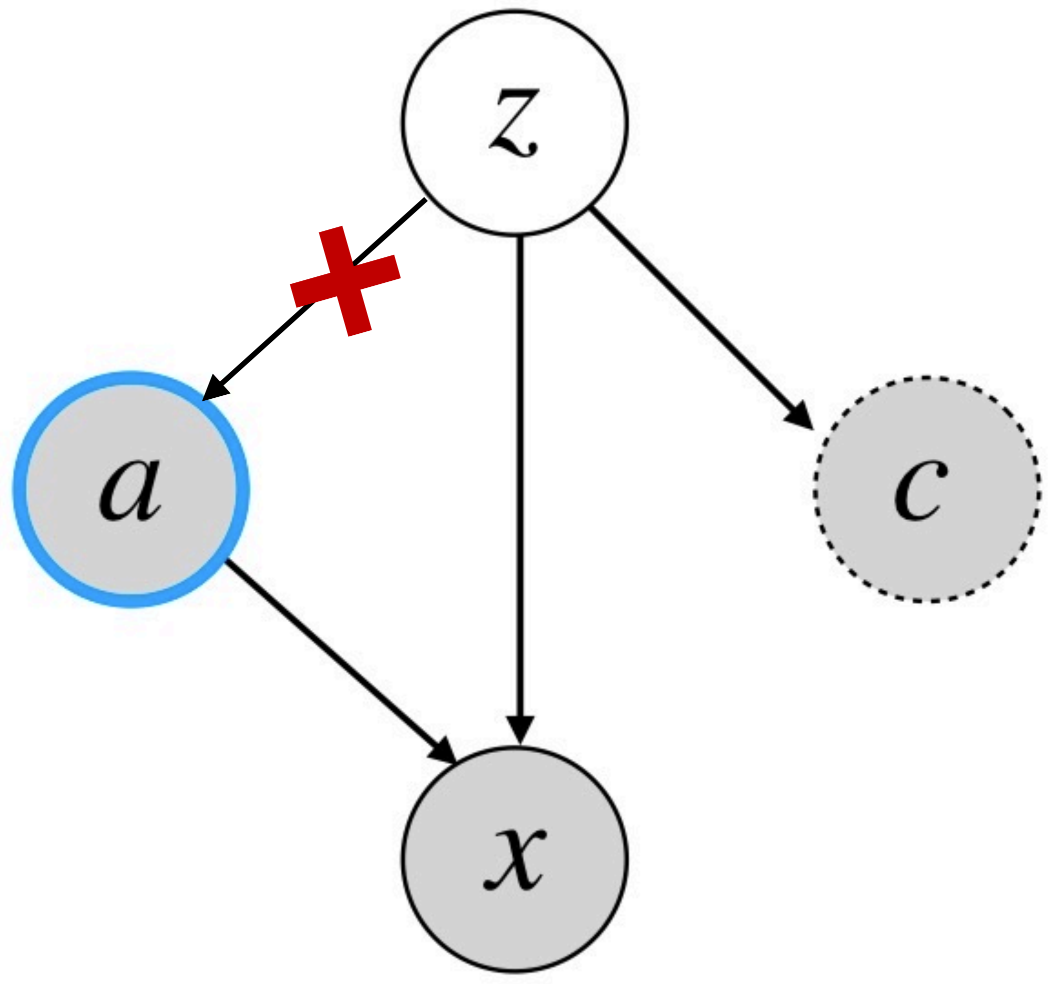
Inference (I): Intervention for Attribute-Conditional Generation

- Association (correlation): $p(\boldsymbol{x}|a)$

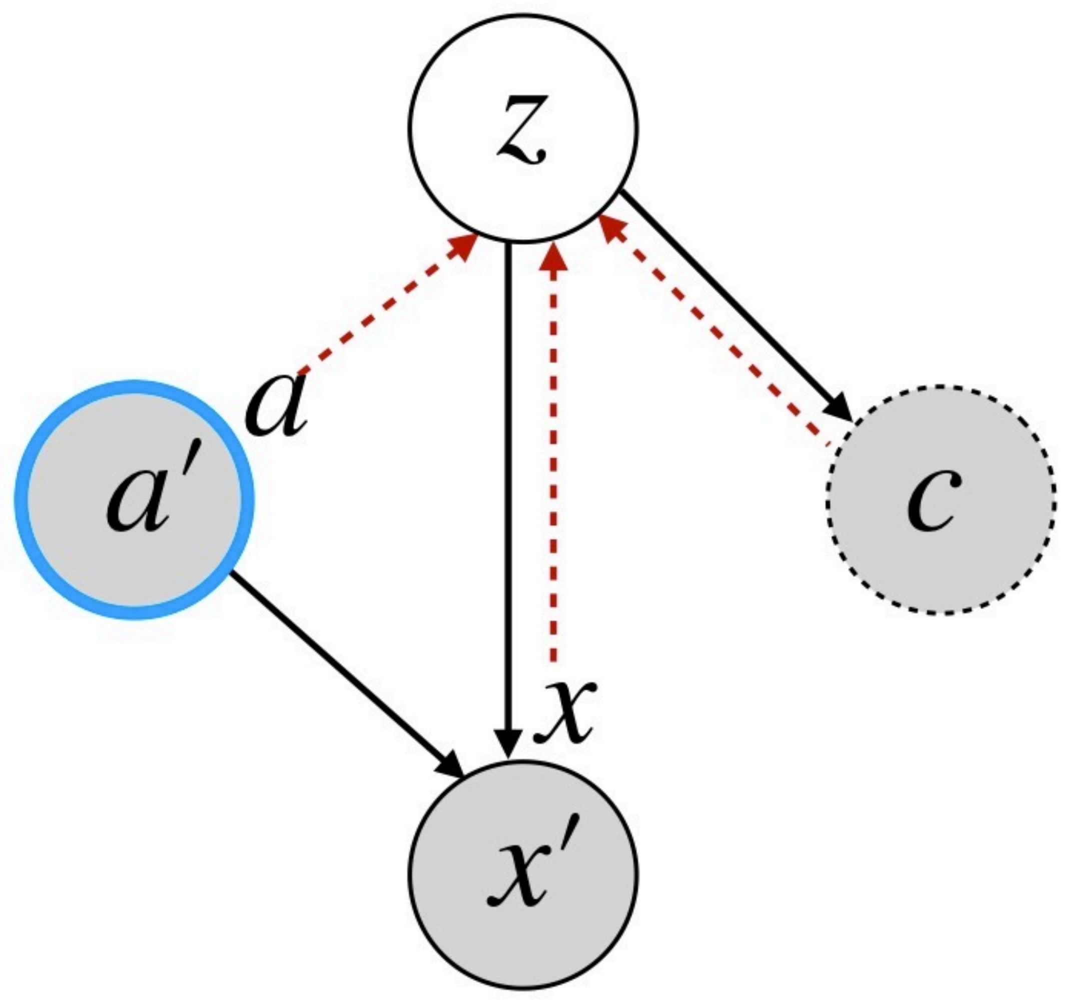
$$p(\boldsymbol{x}|a) = \sum_{z} p_{\theta}(\boldsymbol{x}|a, \boldsymbol{z}) p_{\theta}(\boldsymbol{z}|a)$$

- Intervention: $p(\boldsymbol{x}|do(a))$
  - Sets a to a given value independently of $\boldsymbol{z}$

$$p(\boldsymbol{x}|do(a)) = \sum_{z} p_{\theta}(\boldsymbol{x}|a, \boldsymbol{z}) p_{\theta}(\boldsymbol{z})$$



Inference (II): Counterfactual for Text Attribute Transfer



- What would the text be if the attribute had taken a different value?

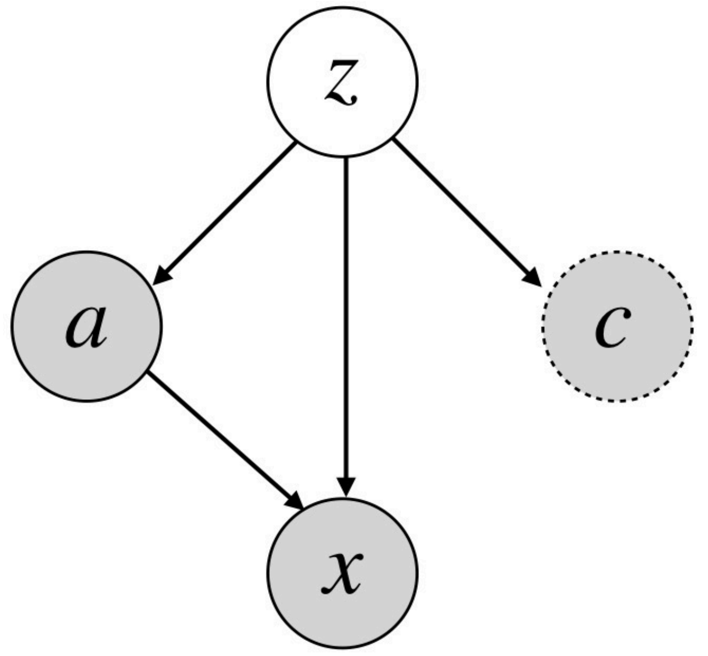
- Counterfactuals as a standard three-step procedure [Pearl 2000]

  1) Abduction: predicts $\boldsymbol{z}$ given $\boldsymbol{x}$: $\boldsymbol{z} \sim q_\phi(\boldsymbol{z}|\boldsymbol{x}, a, \boldsymbol{c})$

  2) Action: performs intervention, $do(a = a')$

  3) Prediction: generates $\boldsymbol{x}'$ given $\boldsymbol{z}$ and a' following the SCM: $\boldsymbol{x}' \sim p_\theta(\boldsymbol{x}'|a', \boldsymbol{z})$

Learning of the SCM



$$p_\theta(\boldsymbol{x}, a, \boldsymbol{z}, \boldsymbol{c}) = p_\theta(\boldsymbol{x}|a, \boldsymbol{z})p_\theta(a|\boldsymbol{z})p_\theta(\boldsymbol{c}|\boldsymbol{z})p_0(\boldsymbol{z})$$

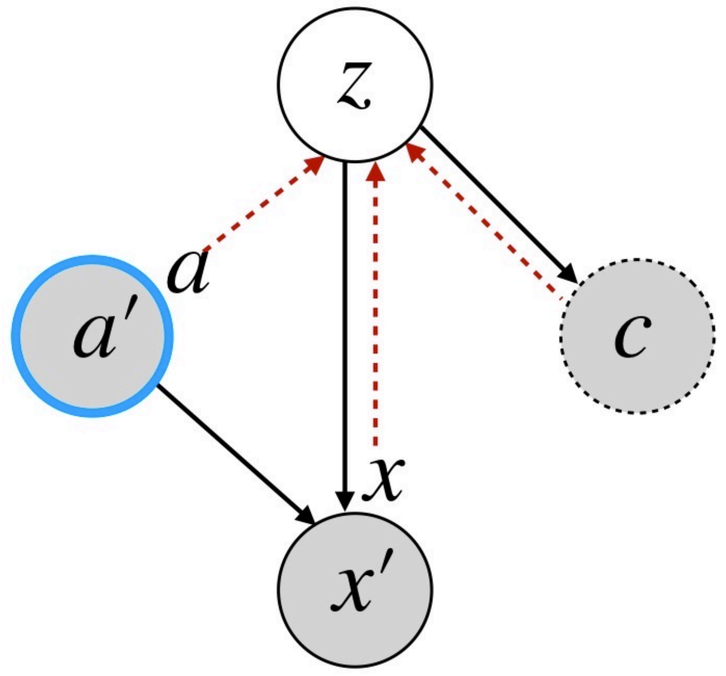
Variational distribution $q_\phi(\boldsymbol{z}|\boldsymbol{x}, a, \boldsymbol{c})$

GPT-2

- Variational autoencoder (VAE) objective

$$\mathcal{L}_{vae}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \mathbb{E}_{\boldsymbol{z}\sim q_\phi}\left[\log \boxed{p_\theta(\boldsymbol{x}|a, \boldsymbol{z})} + \lambda_a \log p_\theta(a|\boldsymbol{z}) + \lambda_c \log p_\theta(\boldsymbol{c}|\boldsymbol{z})\right] - \lambda_{kl}\mathrm{KL}\left(q_\phi \| p_0\right)$$

- Counterfactual objectives
  - Draws inspirations from causality, disentangled representations & controllable generation
  - Intuition: counterfactual $\boldsymbol{x}'$ must entail a' and preserve the original $\boldsymbol{z}$ and $\boldsymbol{c}$



Experiments

- Two challenging datasets with strong spurious correlations
  - Yelp customer reviews:
    - Attribute a: sentiment (1:positive, 0:negative)
    - Confounding proxy $\boldsymbol{c}$: category (1:restaurant, 0:others)
    - **Correlation: 90%** data have the same sentiment and category labels
    - Size: 510K for training, wherein **10K** have category labels
  - Bios: online biographies
    - Attribute a: gender (1:female, 0:male)
    - Confounding proxy $\boldsymbol{c}$: occupation (1:nurse etc, 0:rapper etc)
    - **Correlation: 95%**
    - Size: 43K for training, wherein **3K** have occupation labels

$a = 1, \boldsymbol{c} = 1$
Soup and salad came out quickly !

$a = 0, \boldsymbol{c} = 0$
I texted and called Phil several times and he never responded

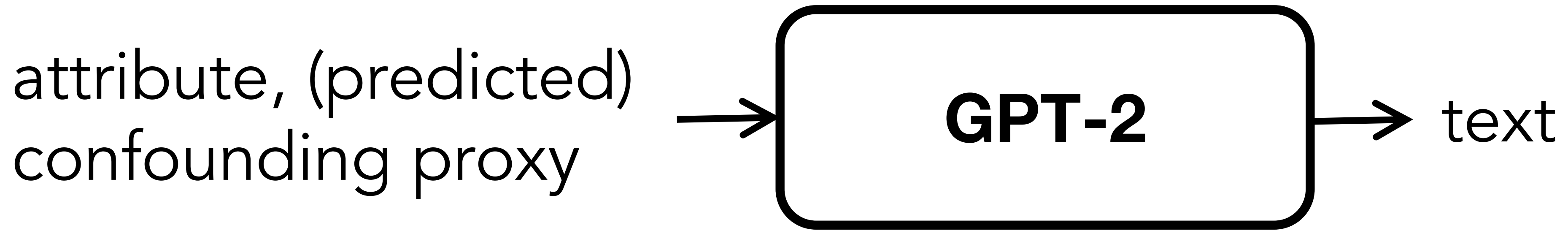
$a = 1, \boldsymbol{c} = 1$
She previously worked as a nurse practitioner

$a = 0, \boldsymbol{c} = 0$
He went to law school and became a plaintiffs' attorney

(I) Attribute-Conditional Generation

Conditional LM

Conditional LM (full)



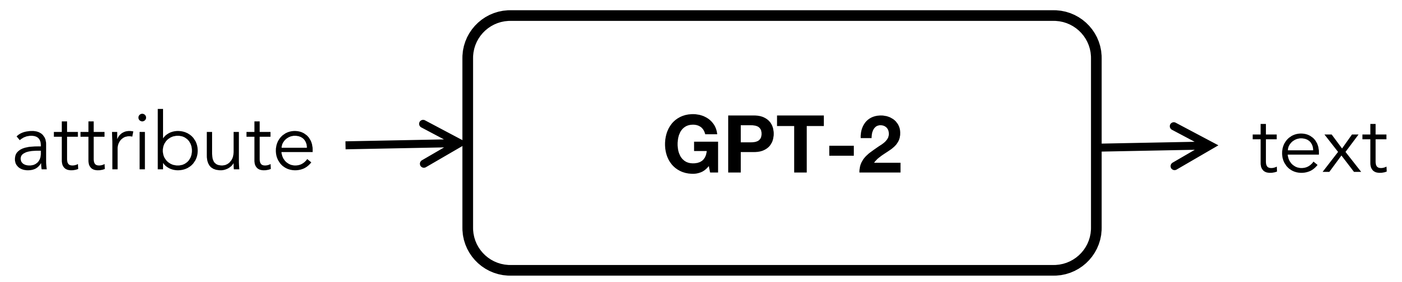
- Causal model improves control accuracy and reduces bias

	Methods	Control accuracy (↑)	Bias (↓)	Fluency (↓)	Diversity (↑)
YELP	Conditional LM	79.1	78.7	50.4	41.4
	Conditional LM (full)	80.3	78.9	50.8	41.9
	GeDi [33]	80.9	74.3	83.2	41.7
	Ablation: Ours w/o cf-z/c	91.1	89.2	54.1	40.4
	Ours	**96.3**	**59.8**	51.3	39.1

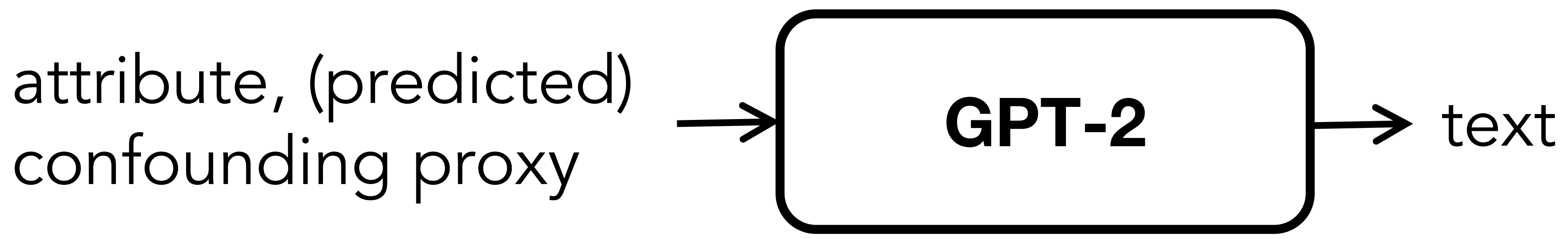
Automatic evaluation

(I) Attribute-Conditional Generation

Conditional LM



Conditional LM (full)



- Causal model improves control accuracy and reduces bias

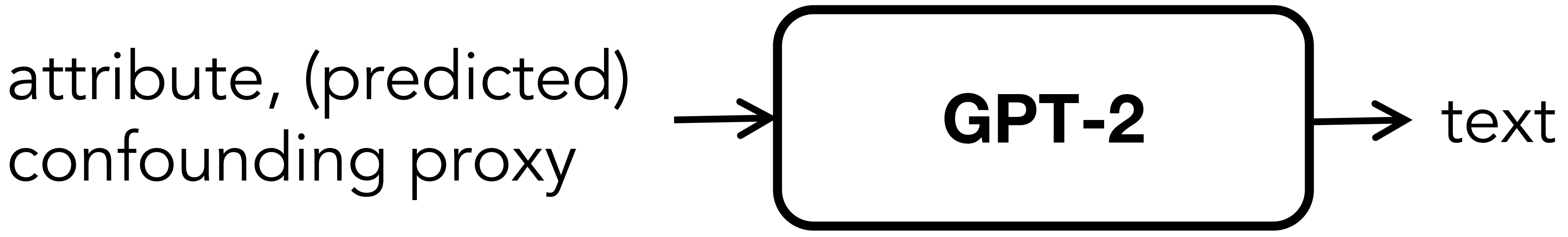
	Methods	Control accuracy (↑)	Bias (↓)	Fluency (↓)	Diversity (↑)
YELP	Conditional LM	79.1	78.7	50.4	41.4
	Conditional LM (full)	80.3	78.9	50.8	41.9
	GeDi [33]	80.9	74.3	83.2	41.7
	Ablation: Ours w/o cf-z/c	91.1	89.2	54.1	40.4
	Ours	**96.3**	**59.8**	51.3	39.1
BIOS	Conditional LM	95.51	84.73	**17.0**	46.5
	Conditional LM (full)	93.28	72.34	18.5	48.5
	GeDi [33]	86.0	75.2	27.8	43.5
	Ablation: Ours w/o cf-z/c	97.3	70.1	29.4	42.1
	Ours	**99.2**	**62.4**	32.0	40.6

Automatic evaluation

(I) Attribute-Conditional Generation

Conditional LM

Conditional LM (full)



- Causal model improves control accuracy and reduces bias

	Methods	Control accuracy (↑)	Bias (↓)	Fluency (↑)
YELP	Conditional LM (full)	80.0	73.0	3.90
	Ours	**97.0**	**56.0**	3.85
BIOS	Conditional LM (full)	96.0	82.0	4.43
	Ours	**99.0**	**60.0**	4.25

Human evaluation

(I) Attribute-Conditional Generation

$\boxed{\text{restaurant}}$

Conditional LM (full)
$a = 0$ **(sentiment negative)**
this was the worst experience i 've ever had at a glazier .
i even asked him if they could play on the tv channel .
this was pretty fun the first time i went . "
waited in line once but almost never reached the floor .
if you are ever up in chandler , tony will stop by .
$a = 1$ **(sentiment positive)**
very good and long wait time .
we loved our favorite harrah 's night ! "
i would love to try this $\boxed{\text{restaurant}}$ again when they open . "
this place is great .
everything you will find in this $\boxed{\text{restaurant}}$!

Ours
$a = 0$ **(sentiment negative)**
no , it 's obvious that they were $\boxed{\text{overcooked}}$.
the seats were poorly done and basically sucked up .
it was n't enough to ask us if it was okay .
very disappointed with my $\boxed{\text{food order}}$ yesterday .
i declined to replace it tho they were bad .
$a = 1$ **(sentiment positive)**
great for a relaxed evening out .
i 'm beyond impressed with the $\boxed{\text{passion fruit and unbeatable service}}$.
it 's a true pleasure to meet andrew .
jacksville became my go-to spot for $\boxed{\text{dessert}}$.
thank you for the technique , i am quite impressed .

(II) Text Attribute Transfer

- Previous methods tend to fail on the challenging dataset: low control accuracy
- Causal model obtains much higher accuracy, and keeps bias low

Methods	Control accuracy (↑)	Bias (↓)	Preservation (↑)	Fluency (↑)
Hu et al. [22]	44.1	68.4	77.7	-132.7
He et al. [20]	35.3	60.2	**80.1**	-57.7
Ablation: Ours w/o cf-z/c	75.0	67.8	36.3	-34.2
Ours	**77.0**	61.4	42.3	**-29.6**

Results on *biased* Yelp dataset

(II) Text Attribute Transfer

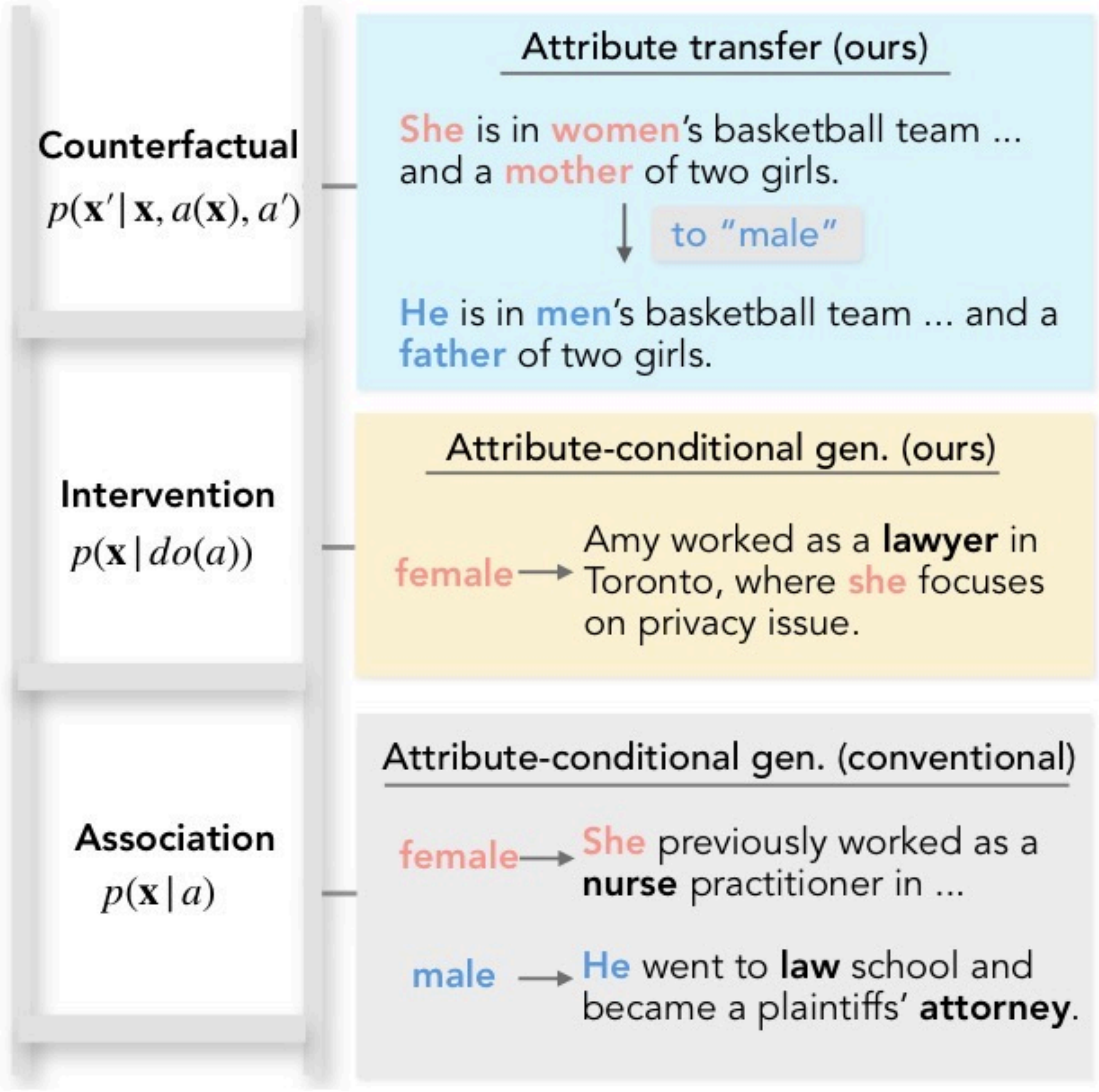
- Previous methods tend to fail on the challenging dataset: low control accuracy
- Causal model obtains much higher accuracy, and keeps bias low
- Also gets improvement on unbiased data

Methods	Control accuracy (↑)	Preservation (↑) self-BLEU	Preservation (↑) ref-BLEU	Fluency (↑)
Hu et al. [22]	86.7	**58.4**	-	-177.7
Shen et al. [65]	73.9	20.7	7.8	-72.0
He et al. [20]	87.9	48.4	18.7	**-31.7**
Dai et al. [7]	87.7	54.9	20.3	-73.0
Ablation: Ours w/o cf-z/c	87.1	57.2	24.3	-46.6
Ours	**91.9**	57.3	**25.5**	-47.1

Results on ***unbiased*** Yelp dataset (commonly used in previous study)

Summary of Causal Controllable Generation

- Causality + ML for unified unbiased controllable generation
  - Intervention
  - Counterfactual
- Causal modeling for more general NLP?
  - Dialog, summarization, …
  - Understanding
  - Reasoning

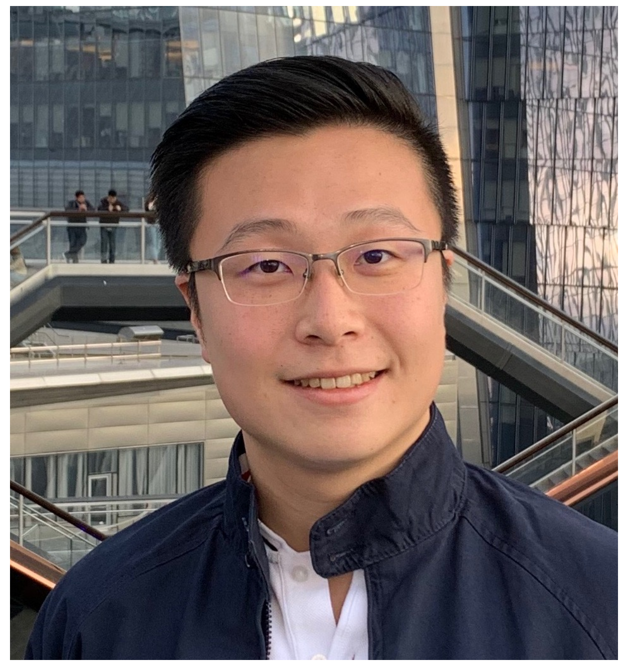


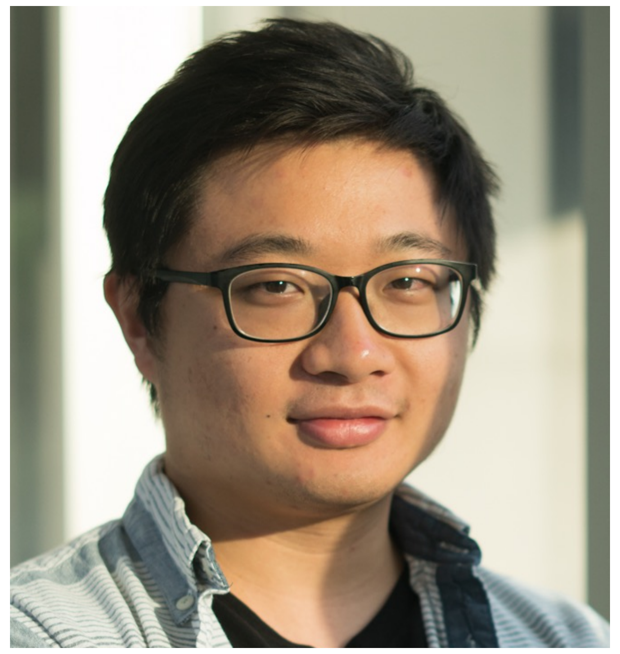
Causal ladder [Pearl 2000]

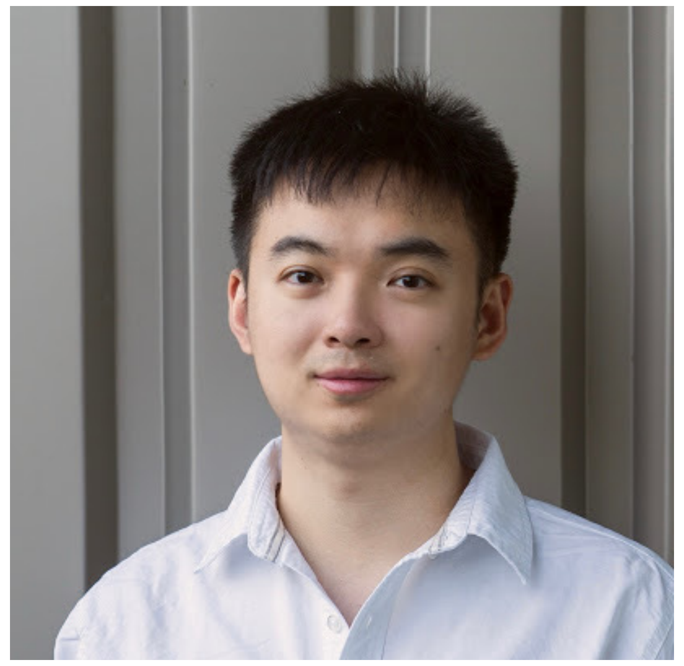
UC San Diego
HALICIOĞLU DATA SCIENCE INSTITUTE
Petuum
Carnegie Mellon University

Evaluating Text Generation without References

A Unified Framework

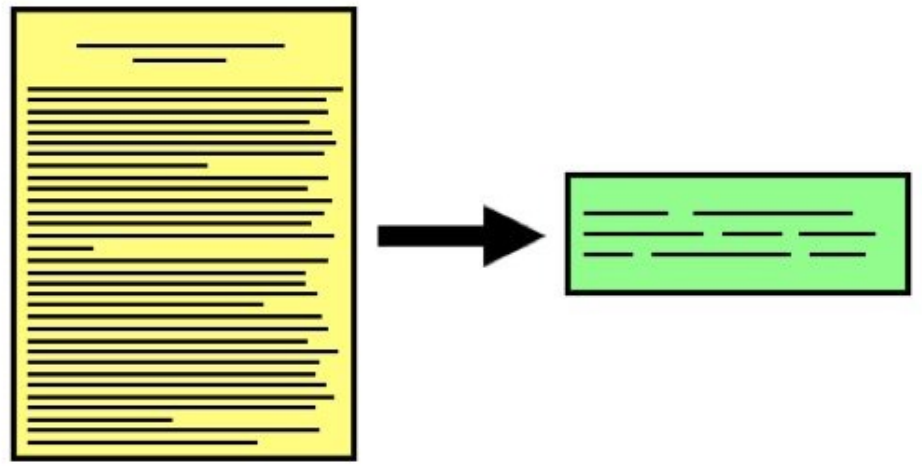






Mingkai Deng* Bowen Tan* Hector Liu Eric P. Xing Zhiting Hu

Text generation tasks have diverse goals



Summarization

Translation

Dialog

Poetry Generation

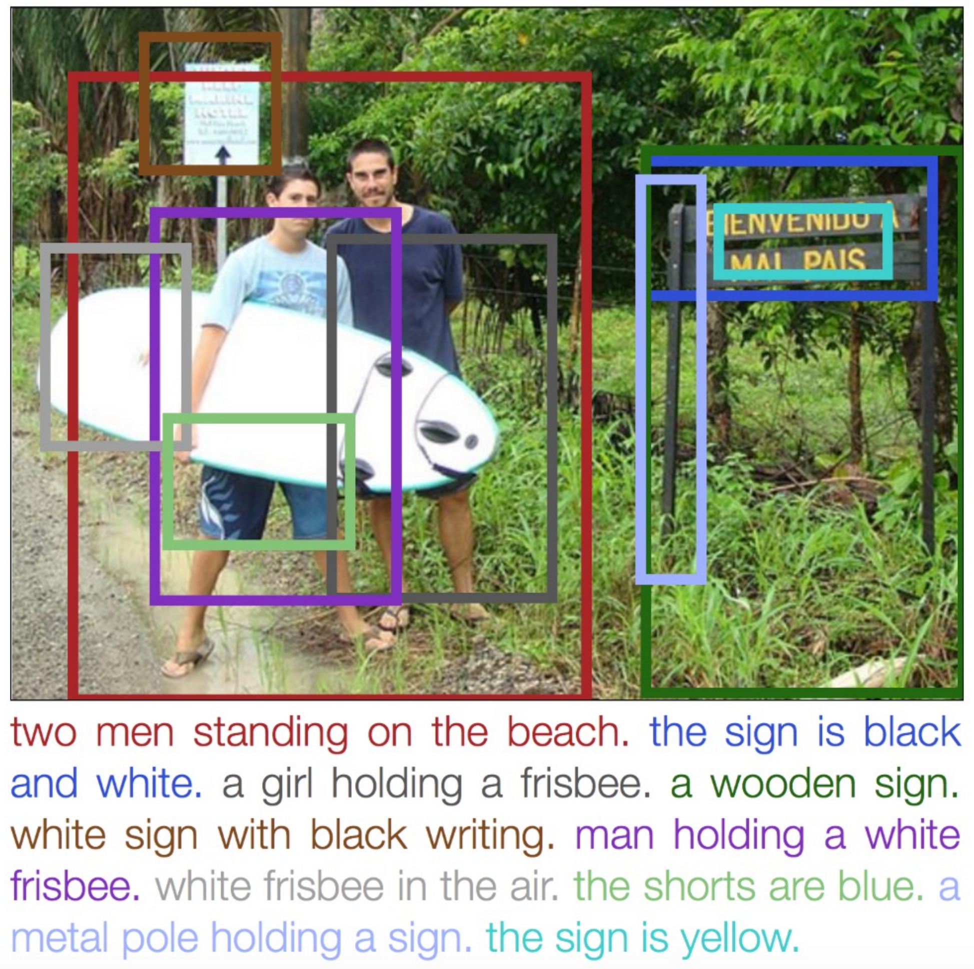
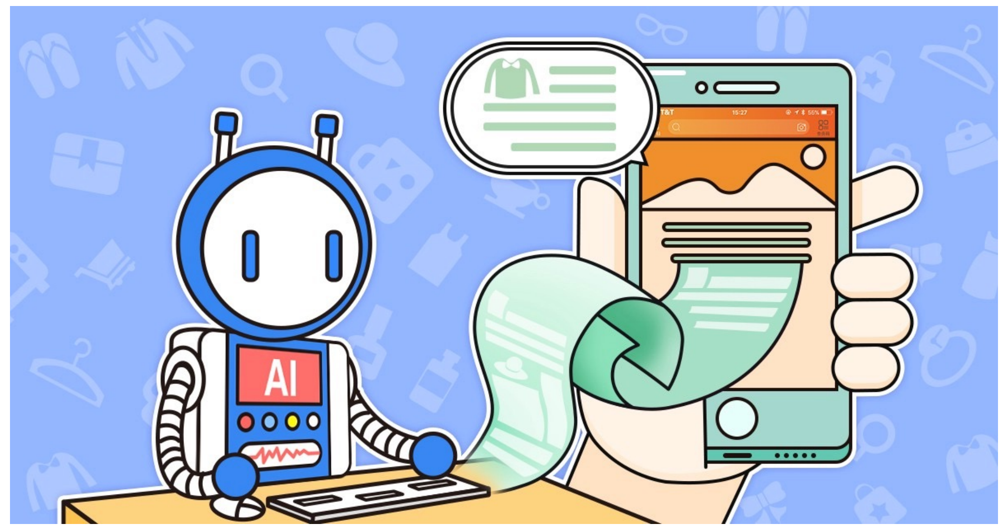
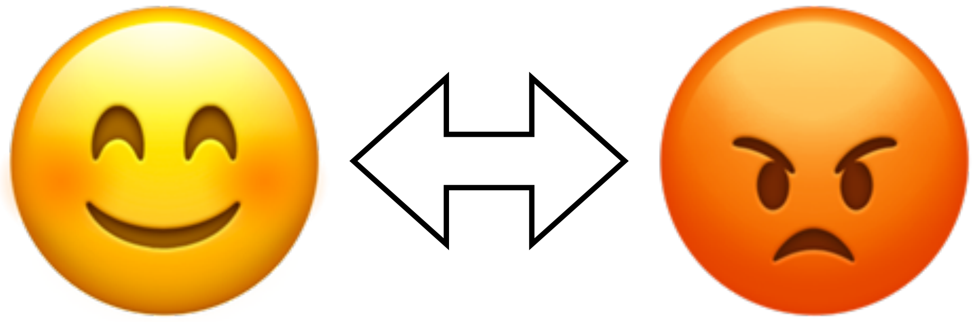


Image Captioning



Story Generation

Data-to-Text



Sentiment Transfer

And the list is growing…

Automatic evaluation is challenging

- Comparing generation with human-written references
  - Expensive to annotate references
  - Incomprehensive evaluation

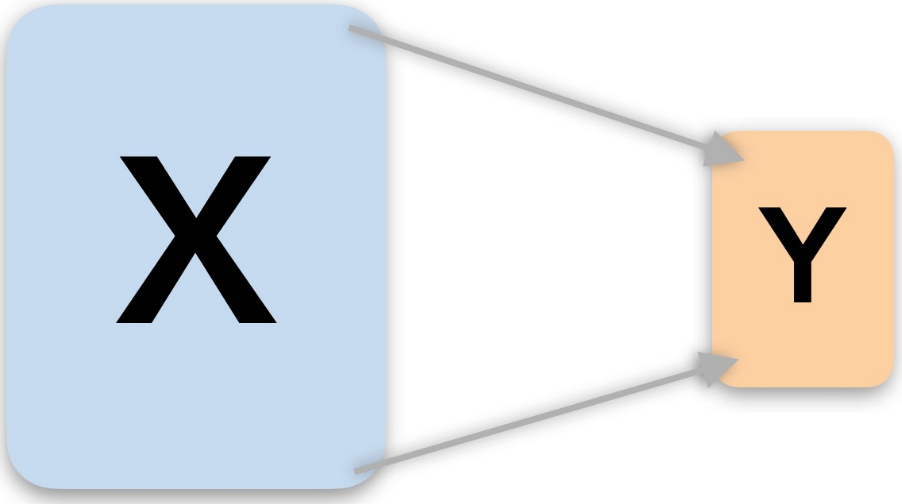
- Different tasks care about different aspects
  - 100s of tasks, 1000s of metrics

Grammaticality Sequence Tagging Interestingness Redundancy
Persona Distinctiveness Direct Assessment Depth Dullness Informativeness
Content Selection Pseudo Reference Hallucination Repetitiveness
Fluency Pointwise Mutual Information Human Score Regression Clarity Novelty
Coverage Knowledge Usage Importance
Linguistic Quality Contradiction Relevance Entailment Classification
Factual Correctness Lexical Matching Perplexity
Semantic Similarity Reference-Free Engagingness
Helpfulness Consistency Factuality Shannon Game
Word Mover Distance Naturalness
Embedding Matching Reference-Based QA Metric Appropriateness
Coherence Faithfulness Knowledge Injection Sensibleness
Automatic Turing Test Diversity

Need a unified theoretical ground across tasks

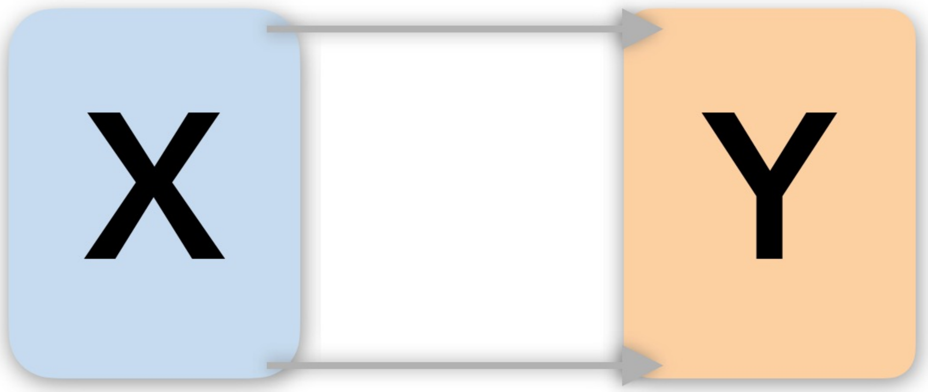
- Categorize tasks based on information change from input (X) to output (Y)

Summarization
Image captioning
Data-to-text
…



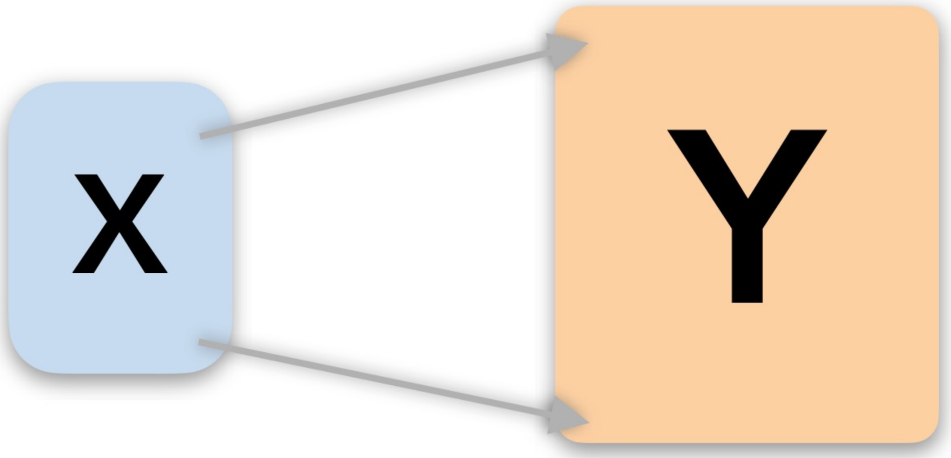
1. Compression (X > Y)

Machine translation
Paraphrasing
Attribute transfer
…

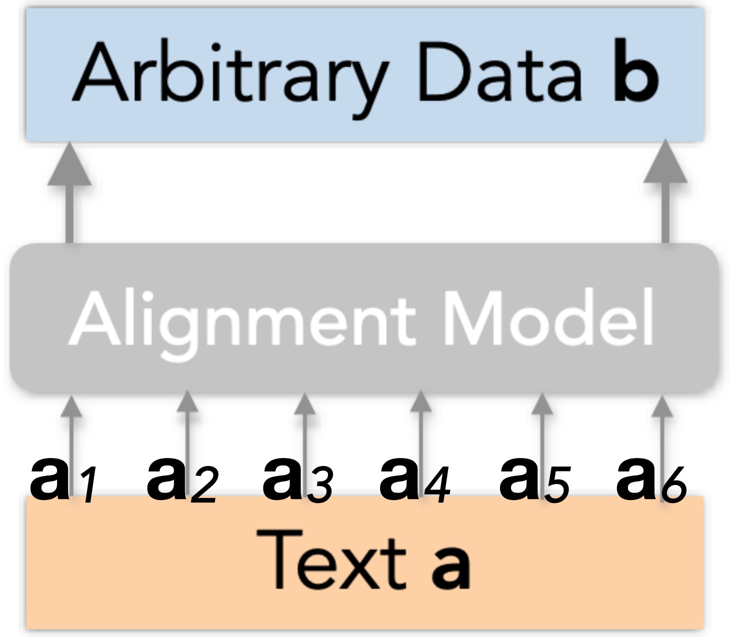


2. Transduction (X = Y)

Dialog
Story generation
…



3. Creation (X < Y)

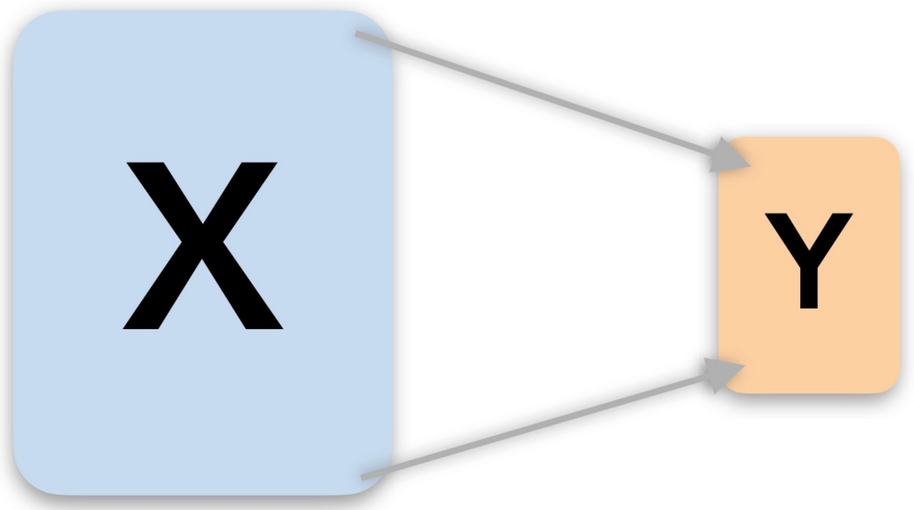


Need a unified theoretical ground across tasks

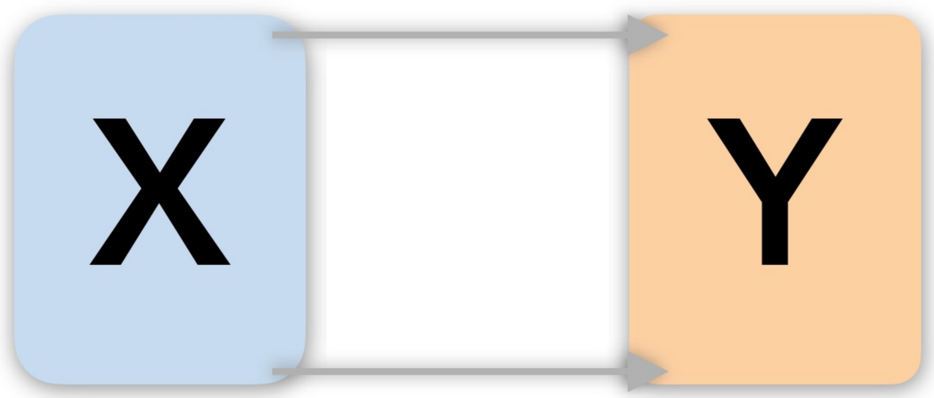
- Categorize tasks based on information change from input (X) to output (Y)
- (Pre-)train an info-alignment model to measure the information change

Example aspects to evaluate:

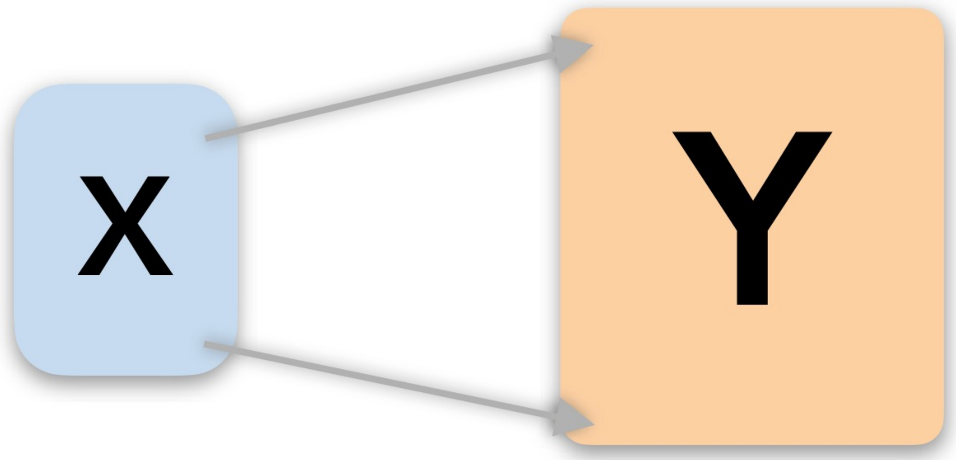
Consistency: *all Y's info must align with X*
Relevance: *all Y's info must align with X's crucial info*



1. Compression (X > Y)

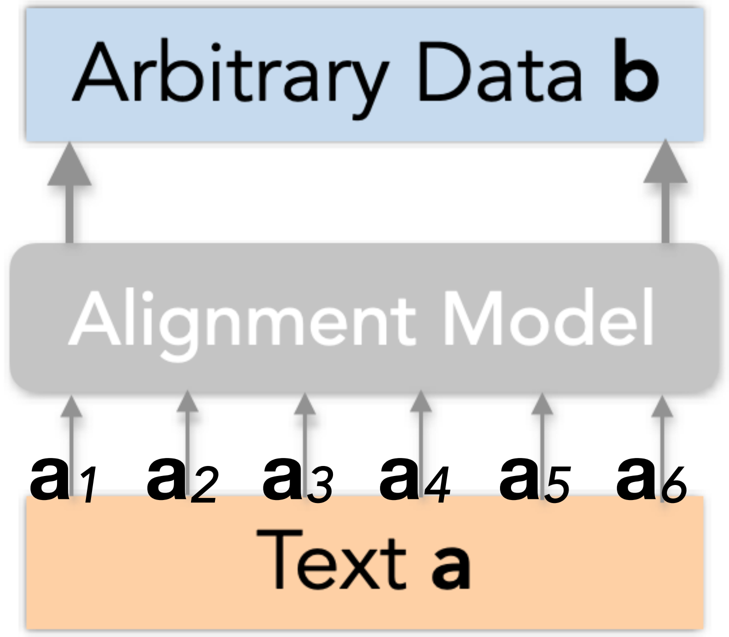


2. Transduction (X = Y)



3. Creation (X < Y)

Need a unified theoretical ground across tasks

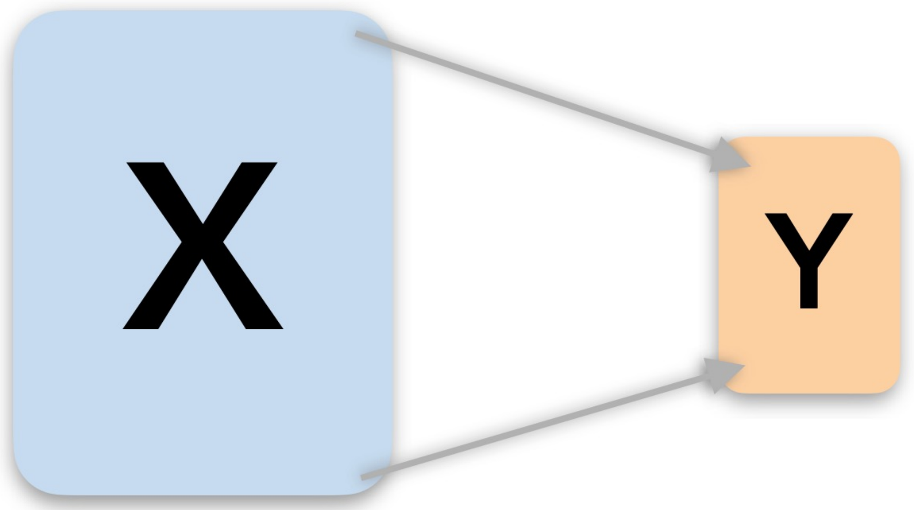


- Categorize tasks based on information change from input (X) to output (Y)
- (Pre-)train an info-alignment model to measure the information change

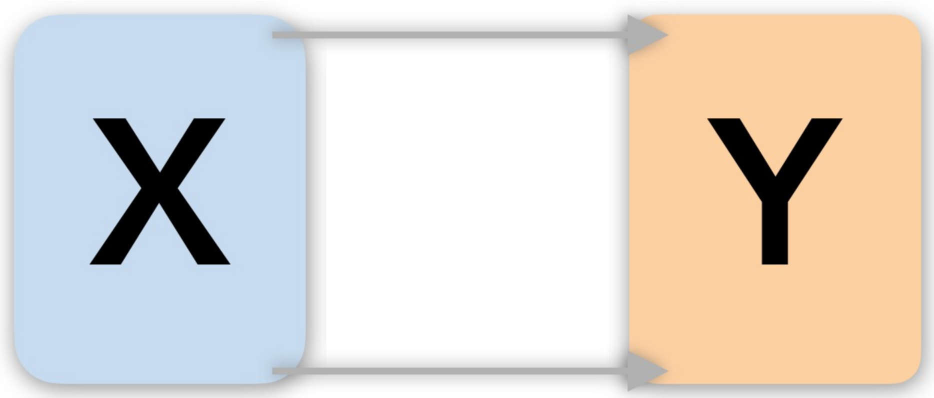
Example aspects to evaluate:

Consistency
Relevance

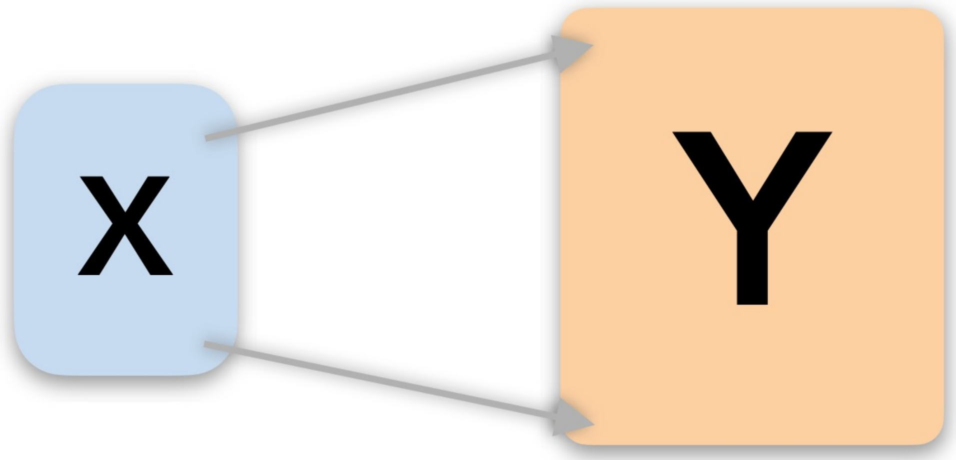
Preservation: *Y and X must align with each other, fully*



1. Compression (X > Y)

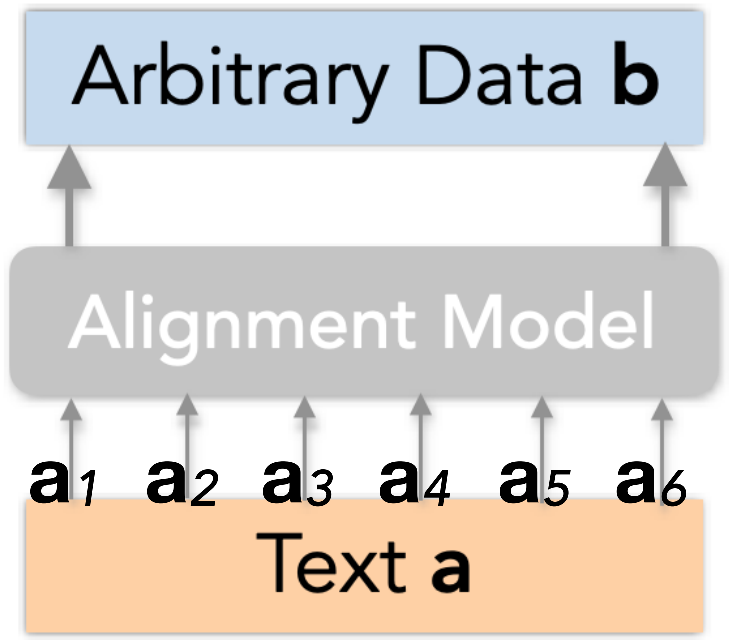


2. Transduction (X = Y)



3. Creation (X < Y)

Need a unified theoretical ground across tasks



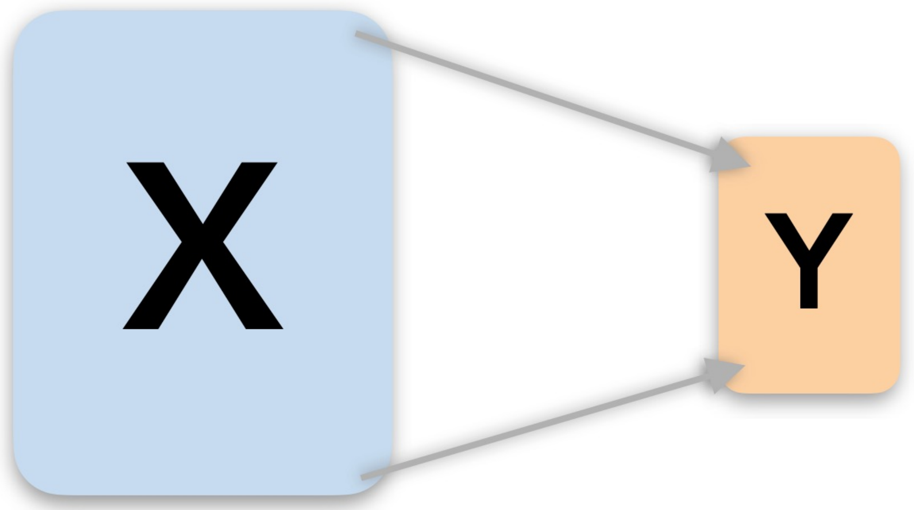
- Categorize tasks based on information change from input (X) to output (Y)
- (Pre-)train an info-alignment model to measure the information change

Example aspects to evaluate:

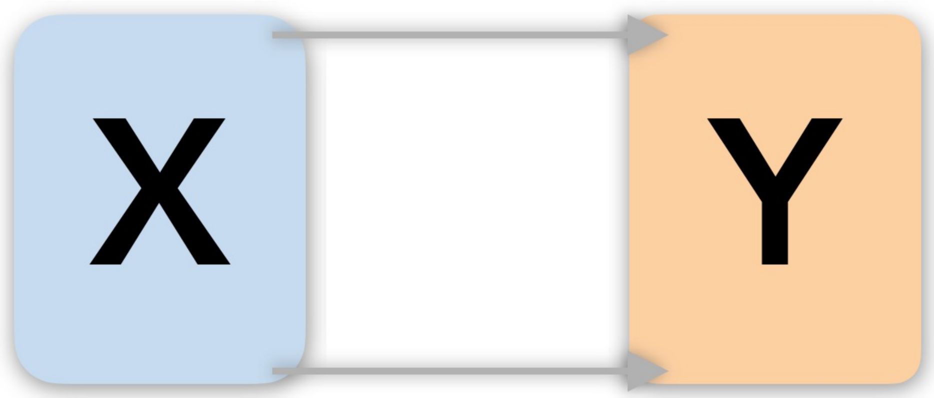
Consistency
Relevance

Preservation

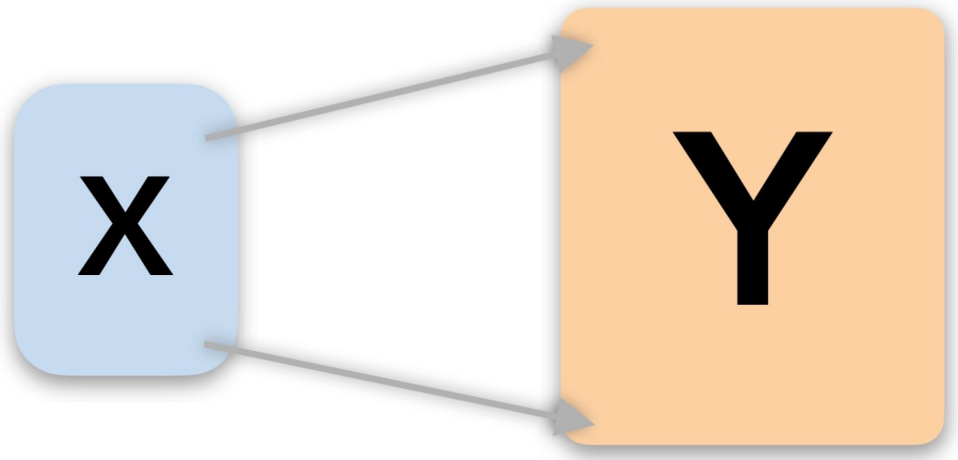
Groundedness: Created info must align with external sources



1. Compression (X > Y)



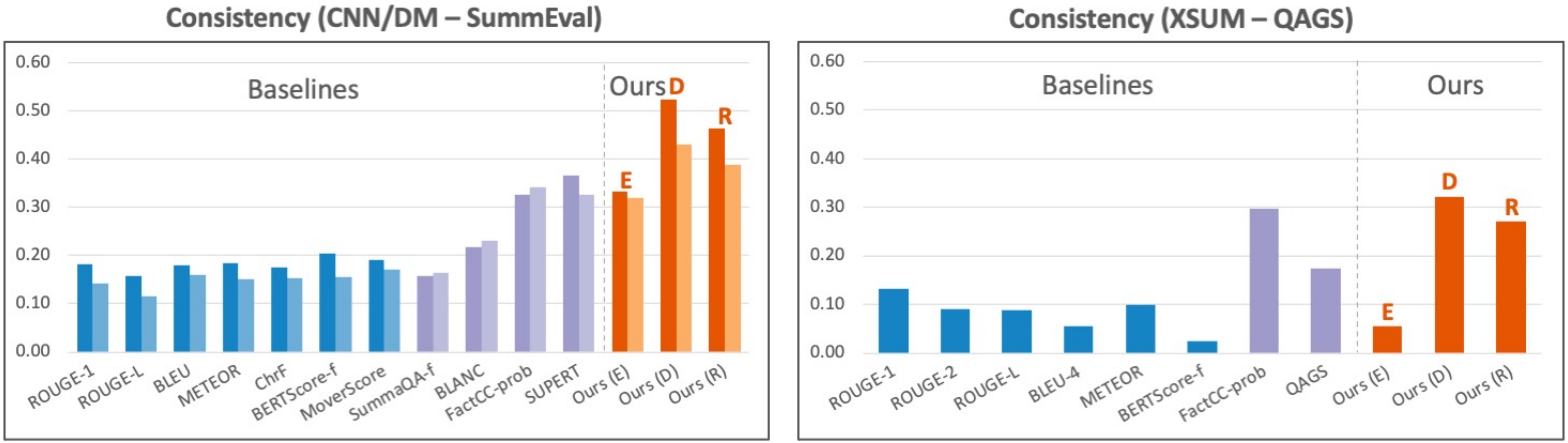
2. Transduction (X = Y)



3. Creation (X < Y)

Uniformly-designed metrics ***vs*** previous *specialized* metrics

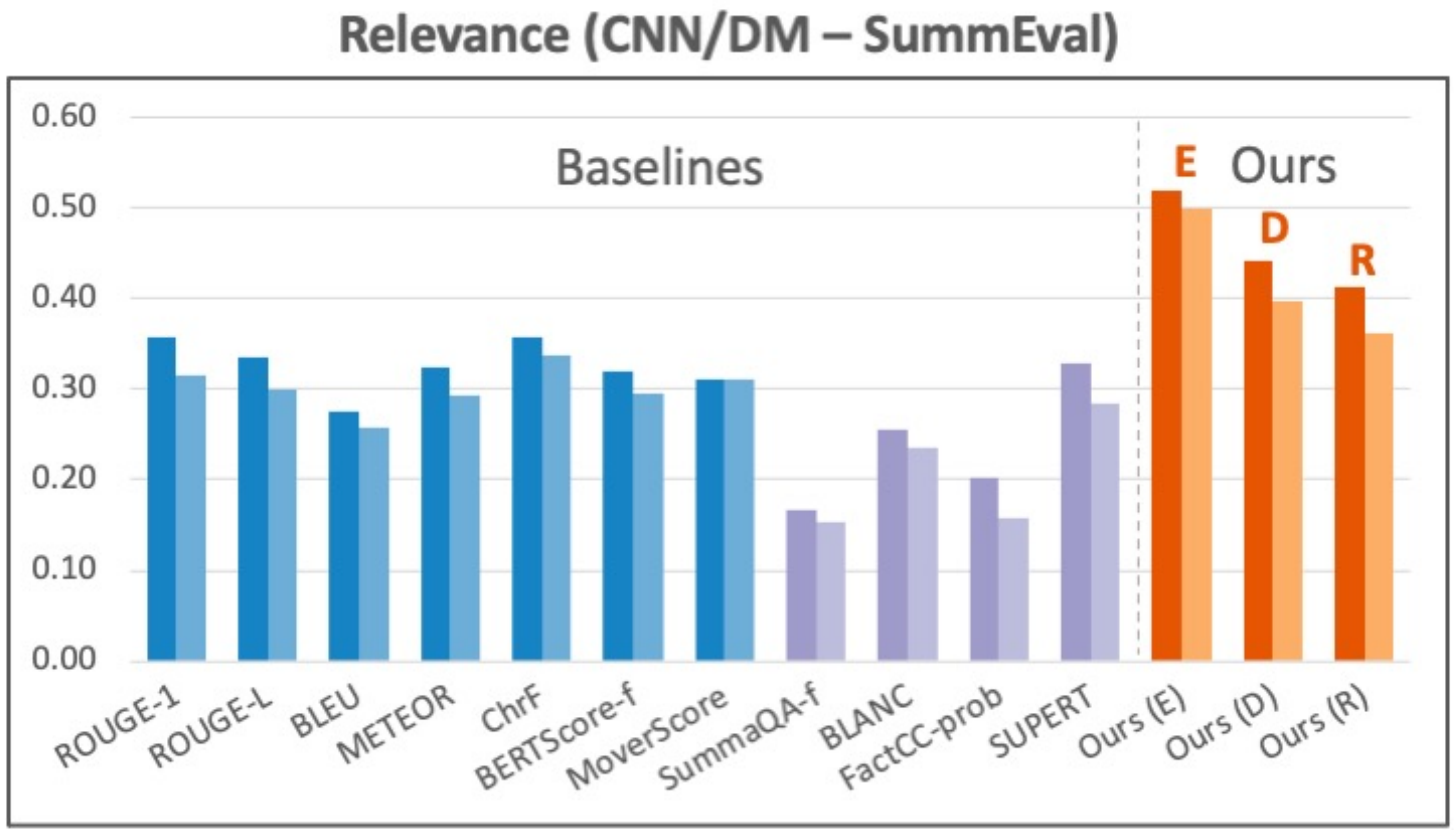
- Summarization: consistency



Human correlation

Uniformly-designed metrics ***vs*** previous *specialized* metrics

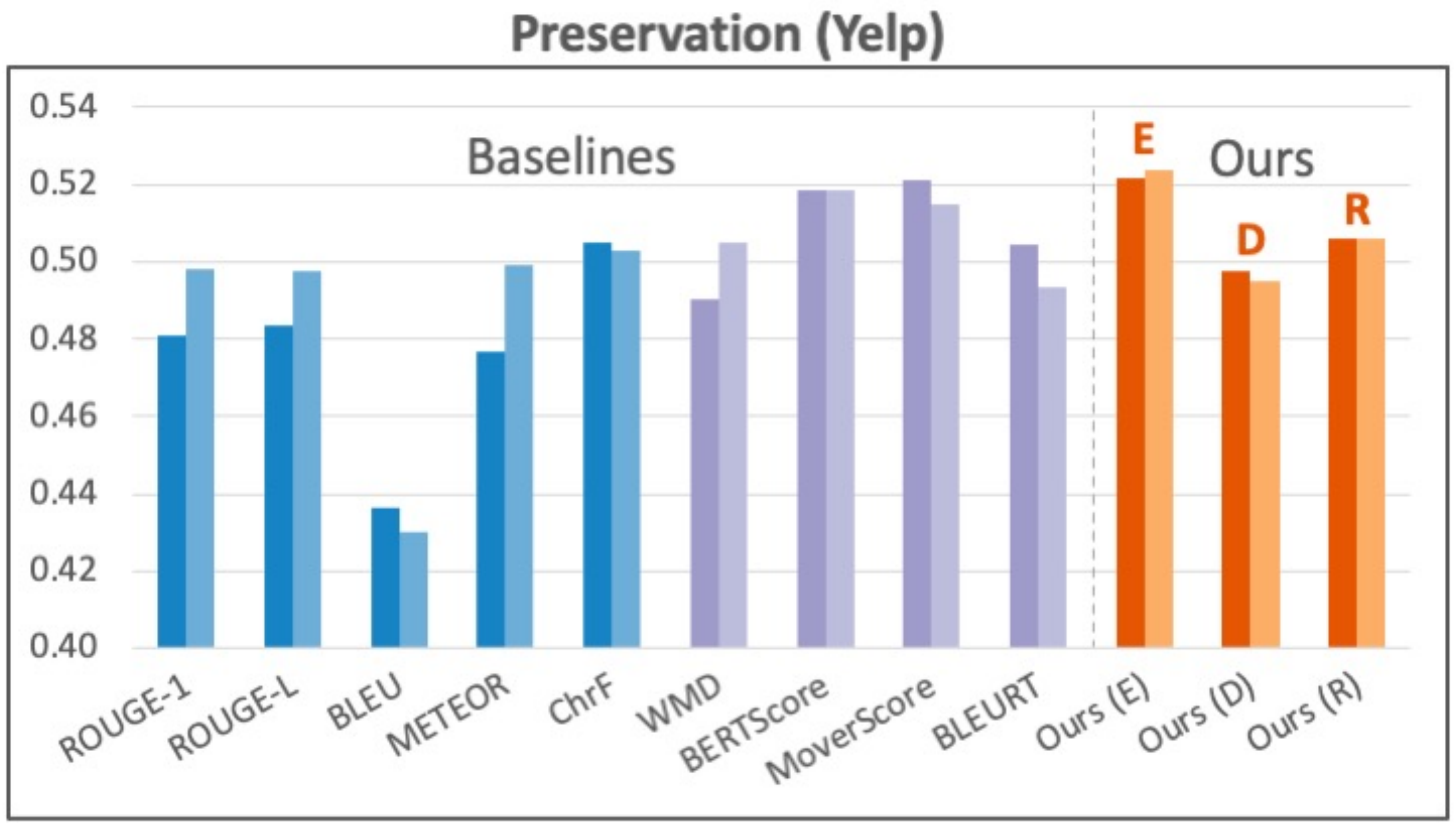
- Summarization: relevance



Human correlation

Uniformly-designed metrics ***vs*** previous *specialized* metrics

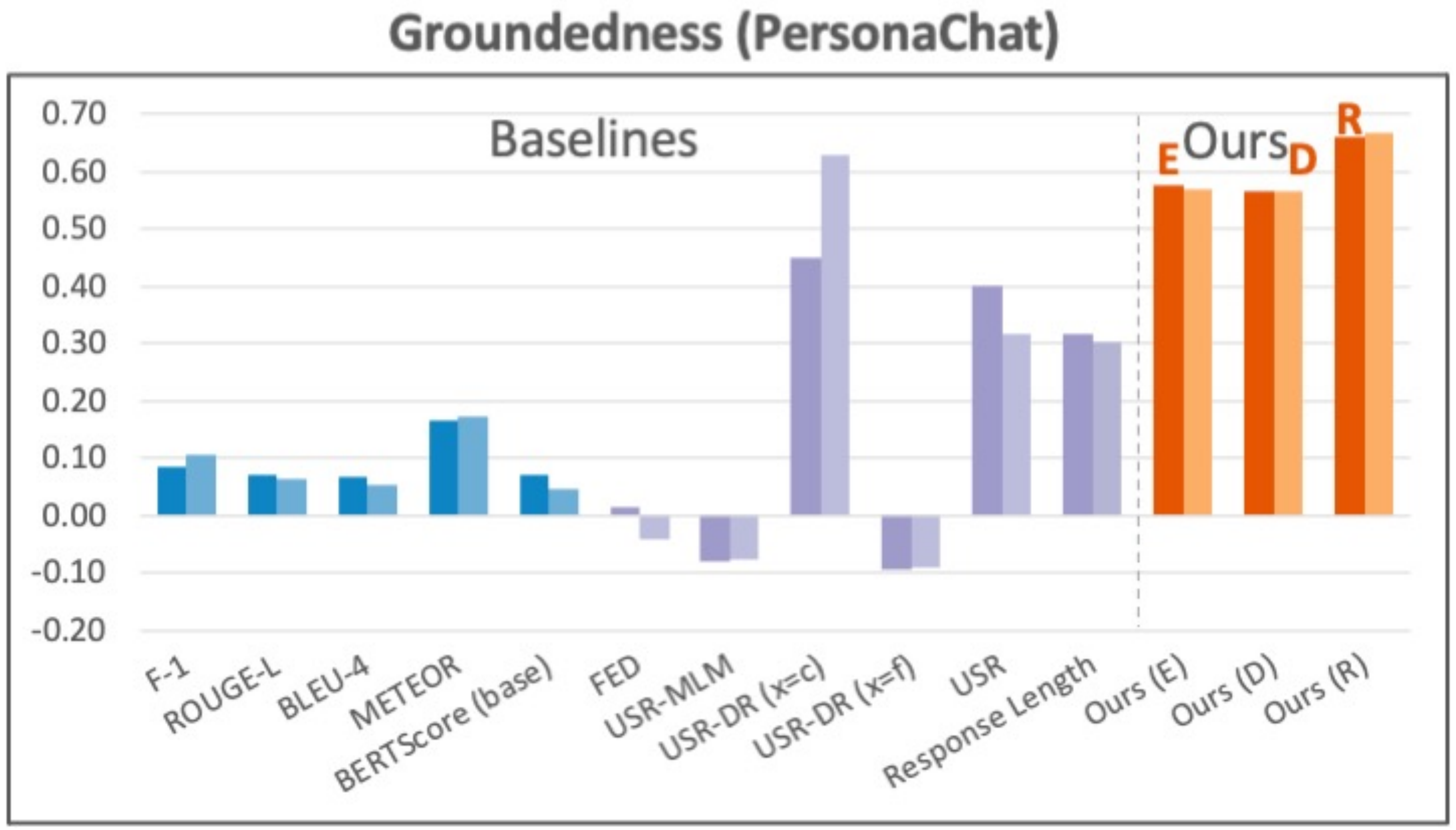
- Attribute transfer: preservation

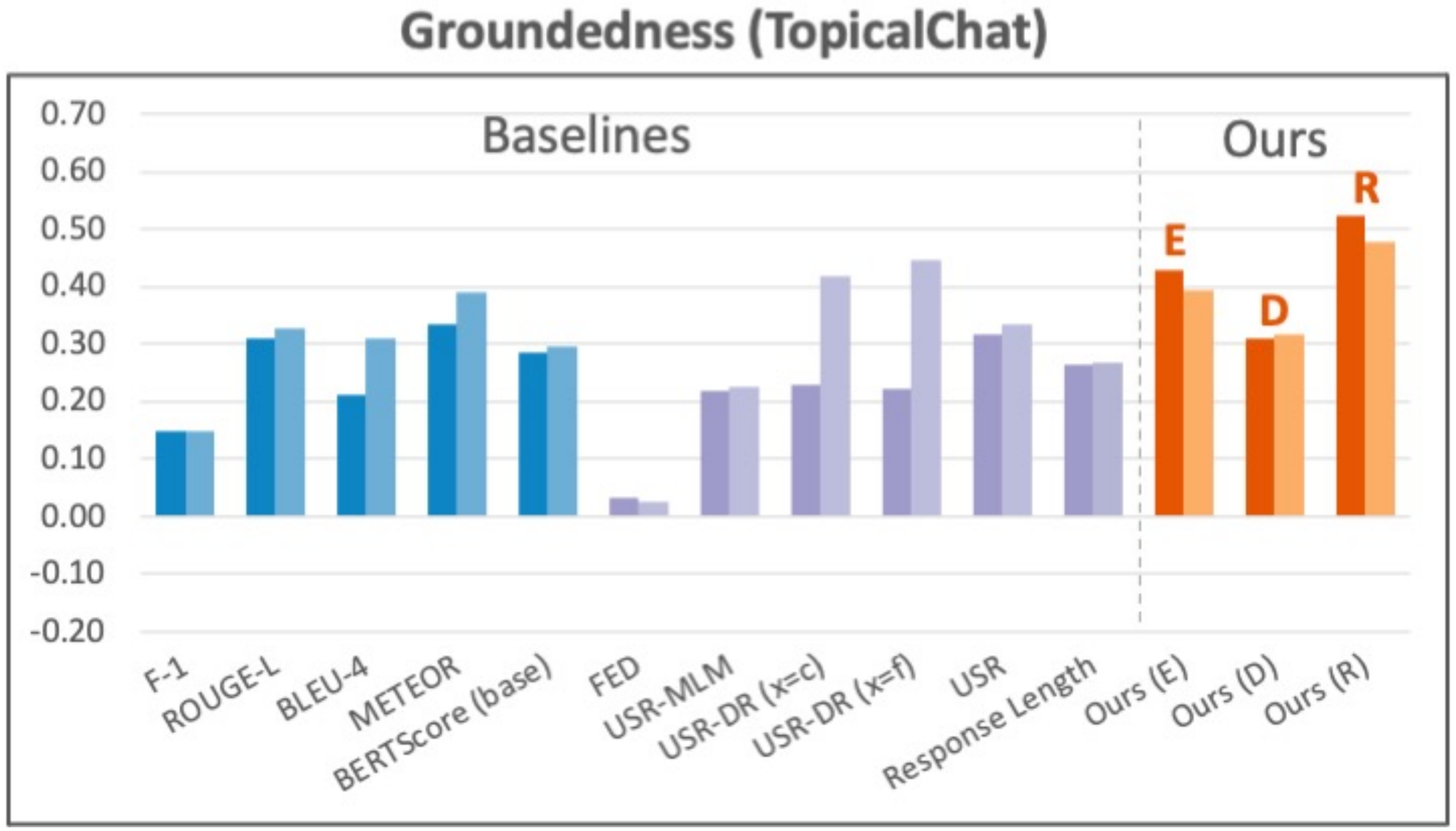


Human correlation

Uniformly-designed metrics ***vs*** previous *specialized* metrics

- Dialog: groundedness

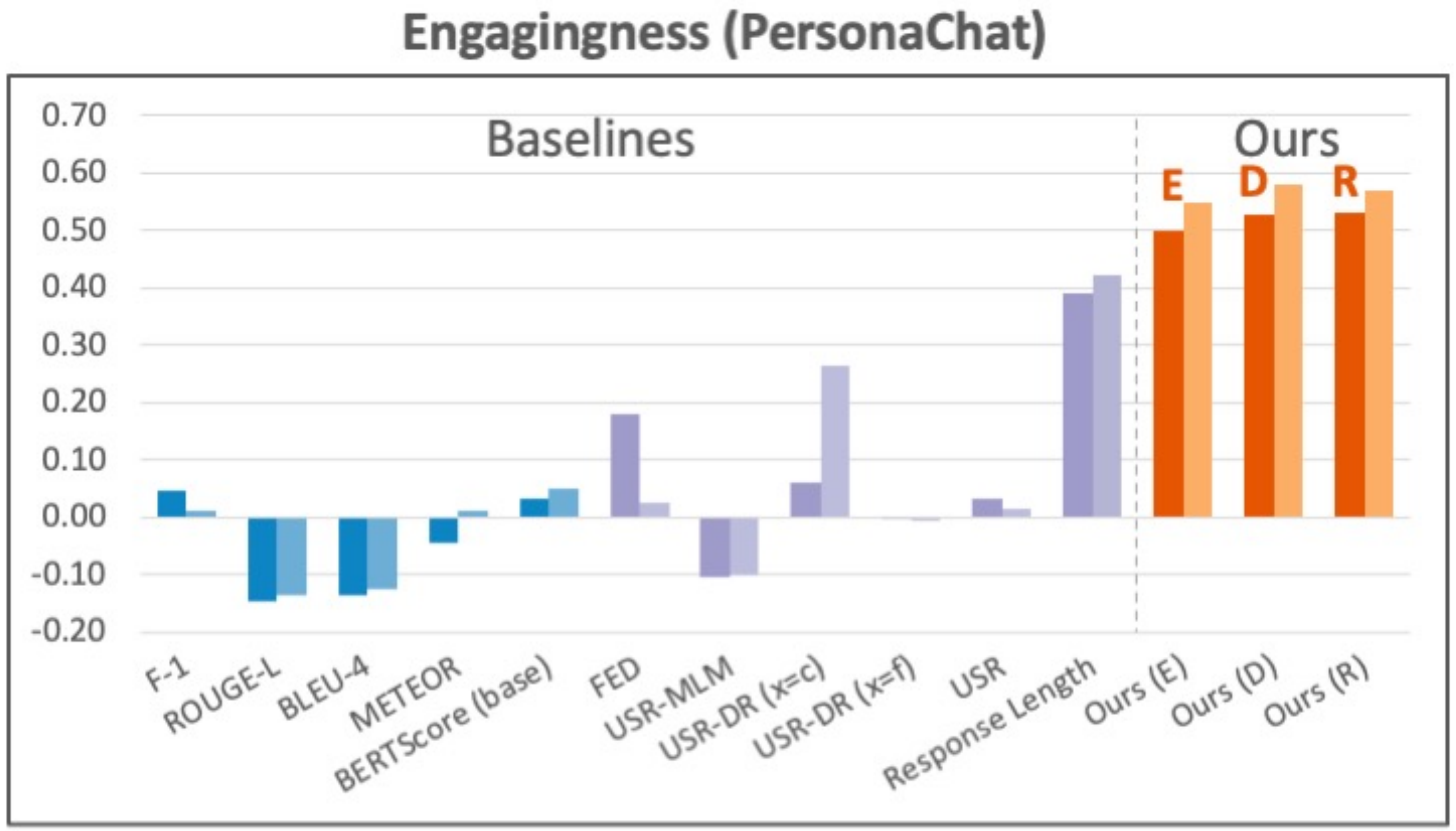


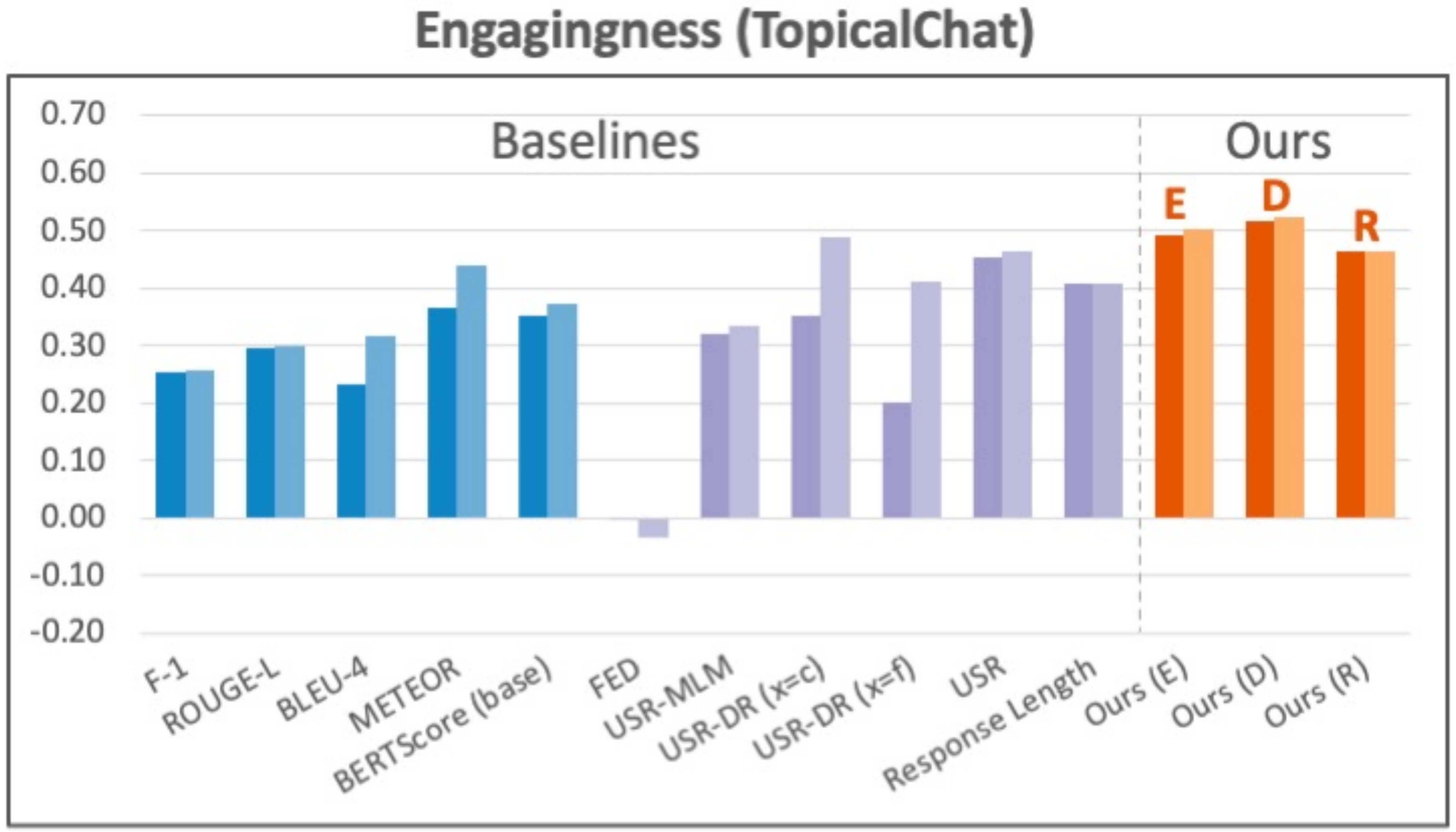


Human correlation

Uniformly-designed metrics ***vs*** previous *specialized* metrics

- Dialog: engagingness

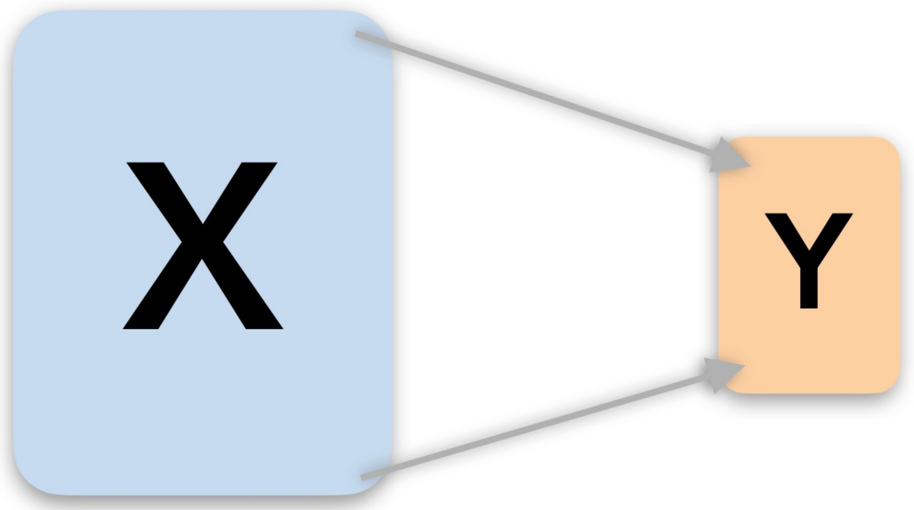




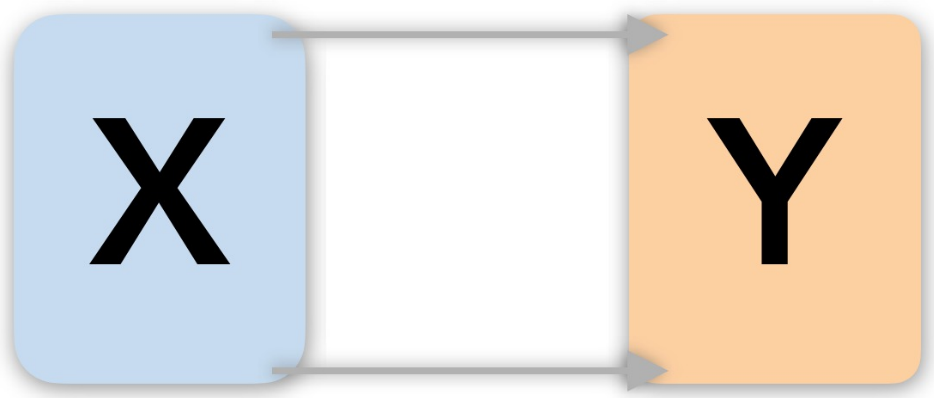
Human correlation

Summary of Unified Text Generation Evaluation

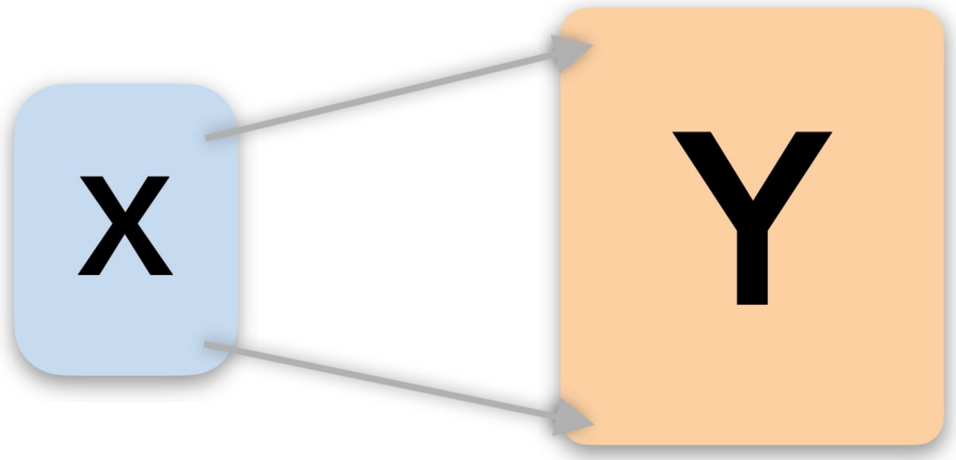
- Information change/alignment characterizes text generation tasks
- (Pre-)trained info-alignment model creates "intermediate representations" for defining desired metrics
- Consistently stronger human correlation compared to specialized metrics



1. Compression (X > Y)



2. Transduction (X = Y)



3. Creation (X < Y)

Thanks !