

DSC291: Advanced Statistical Natural Language Processing

Text Generation

Zhiting Hu

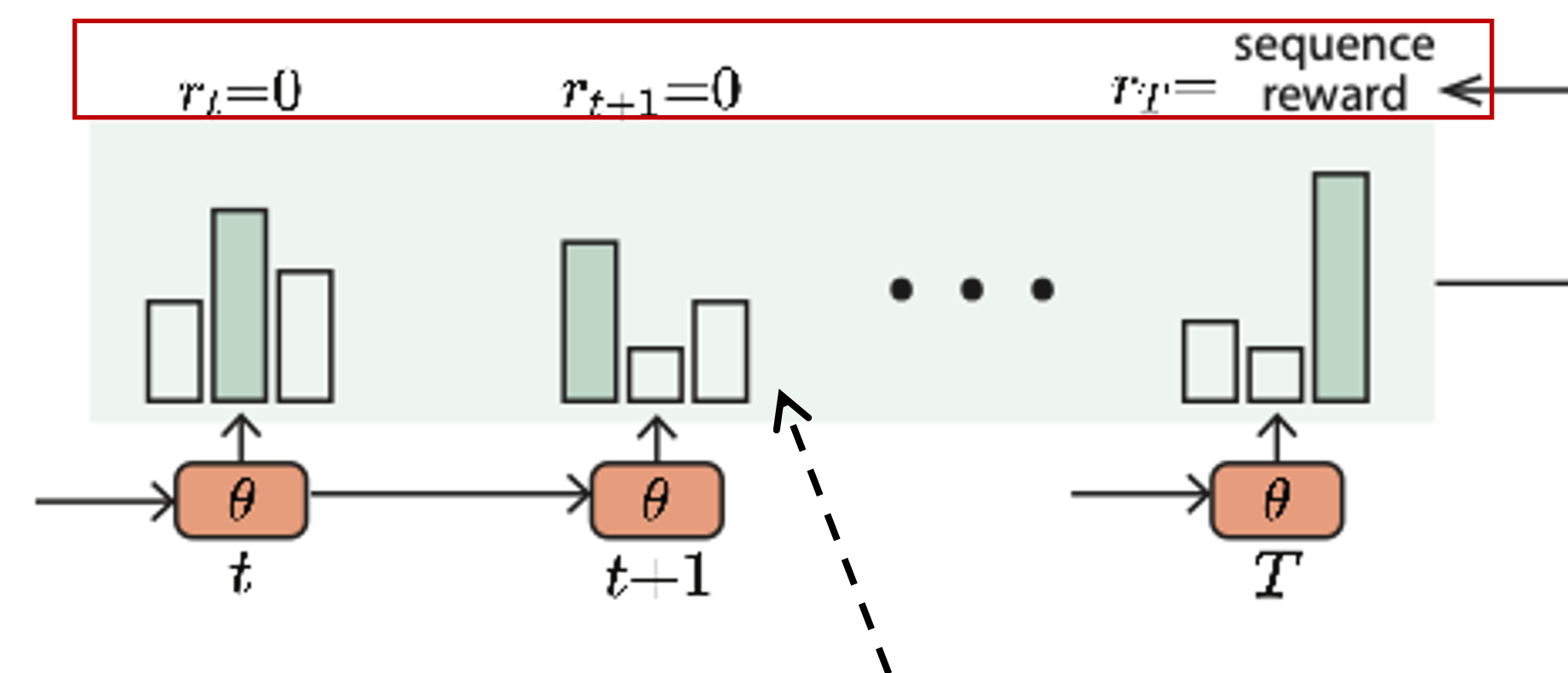
Lecture 15, May 17, 2022

UC San Diego

HALICIOĞLU DATA SCIENCE INSTITUTE

Recap: RL for Text Generation

- (Autoregressive) text generation model:



Sentence $\mathbf{y} = (y_0, \dots, y_T)$ $\pi_{\theta}(y_t | \mathbf{y}_{<t}) = \frac{\exp f_{\theta}(y_t | \mathbf{y}_{<t})}{\sum_{y'} \exp f_{\theta}(y' | \mathbf{y}_{<t})}$ logits

In RL terms:

trajectory, τ

action, a_t

state, s_t

policy $\pi_{\theta}(a_t | s_t)$

- Reward $r_t = r(\mathbf{s}_t, a_t)$
 - Often **sparse**: $r_t = 0$ for $t < T$

Recap: RL for Text Generation: REINFORCE

Given a dataset of input output pairs, $\mathcal{D} \equiv \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)*})\}_{i=1}^N$

learn a conditional distribution $p_{\theta}(\mathbf{y} | \mathbf{x})$ that minimizes

expected loss:

$$\mathcal{L}_{\text{RL}}(\boldsymbol{\theta}) = \sum_{(\mathbf{x}, \mathbf{y}^*) \in \mathcal{D}} - \sum_{\mathbf{y} \in \mathcal{Y}} p_{\theta}(\mathbf{y} | \mathbf{x}) r(\mathbf{y}, \mathbf{y}^*)$$

*Sample from the
model distribution*

Recap: RL for Text Generation: REINFORCE

Given a dataset of input output pairs, $\mathcal{D} \equiv \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)*})\}_{i=1}^N$

learn a conditional distribution $p_{\theta}(\mathbf{y} | \mathbf{x})$ that minimizes

expected loss:

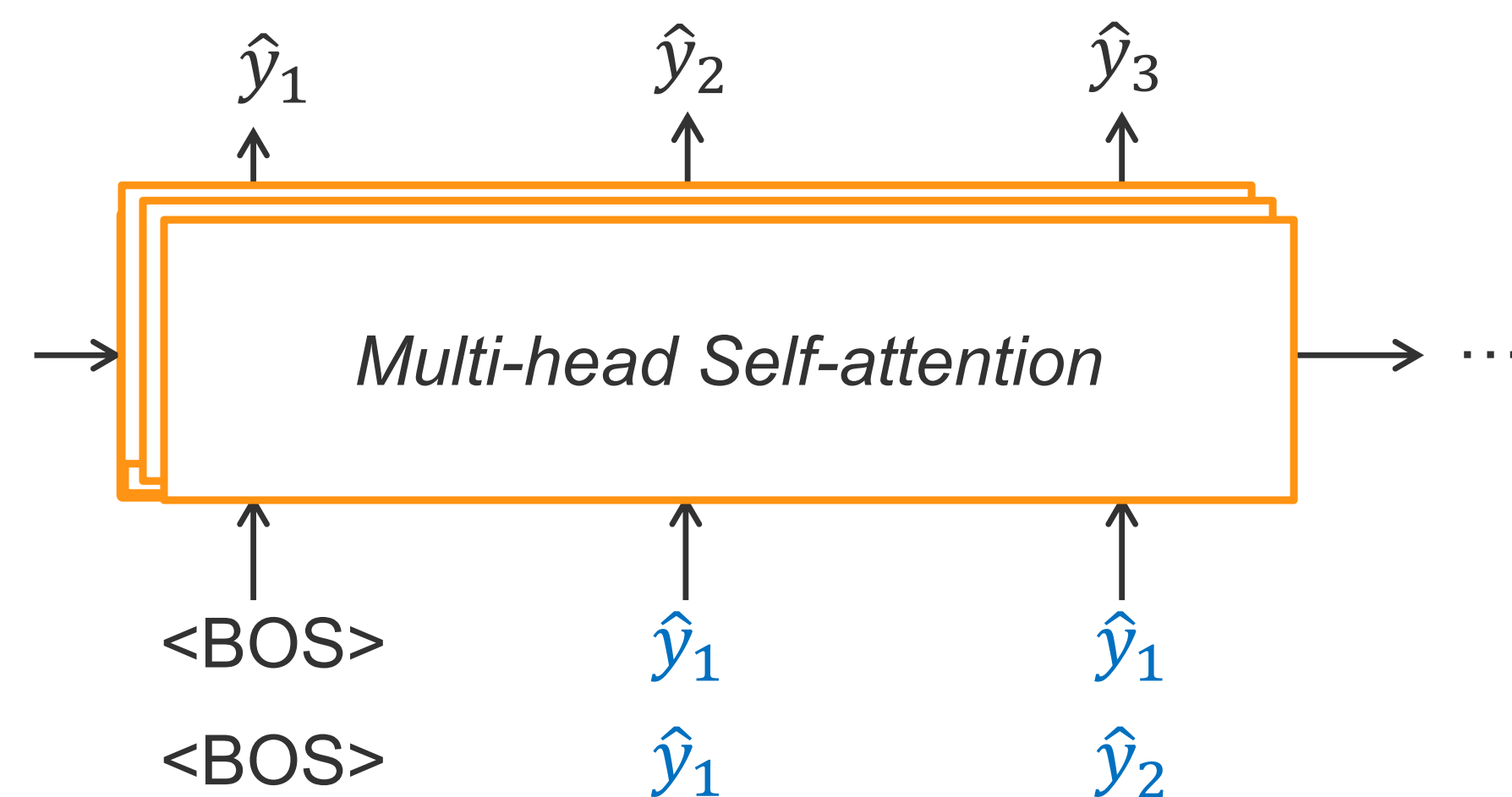
$$\mathcal{L}_{\text{RL}}(\boldsymbol{\theta}) = \sum_{(\mathbf{x}, \mathbf{y}^*) \in \mathcal{D}} - \sum_{\mathbf{y} \in \mathcal{Y}} p_{\theta}(\mathbf{y} | \mathbf{x}) r(\mathbf{y}, \mathbf{y}^*)$$

Sample from the
model distribution

No exposure bias

Training:

Evaluation:



Recap: RL for Text Generation: REINFORCE

Given a dataset of input output pairs, $\mathcal{D} \equiv \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)*})\}_{i=1}^N$

learn a conditional distribution $p_{\theta}(\mathbf{y} | \mathbf{x})$ that minimizes

expected loss:

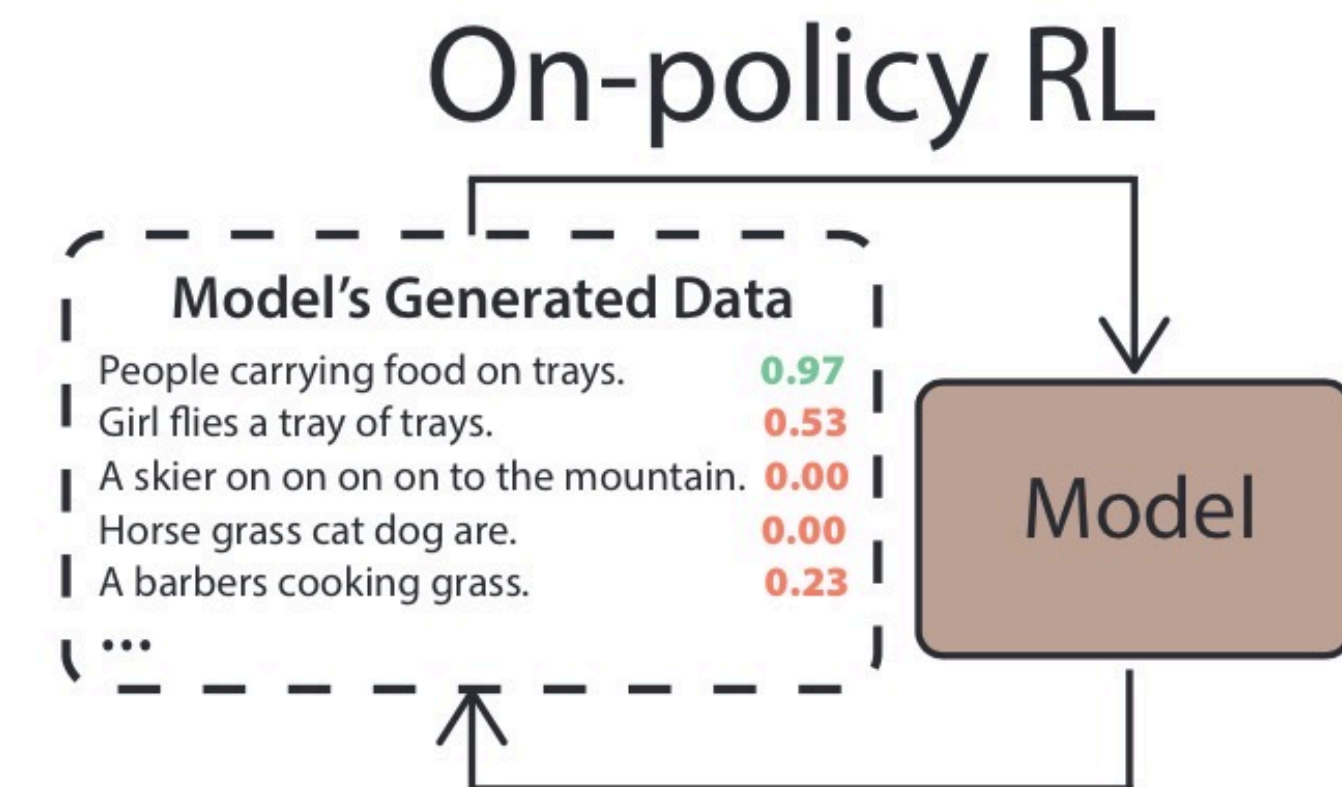
$$\mathcal{L}_{\text{RL}}(\theta) = \sum_{(\mathbf{x}, \mathbf{y}^*) \in \mathcal{D}} - \sum_{\mathbf{y} \in \mathcal{Y}} p_{\theta}(\mathbf{y} | \mathbf{x}) r(\mathbf{y}, \mathbf{y}^*)$$

On-policy RL: generate text samples from the current policy p_{θ} itself

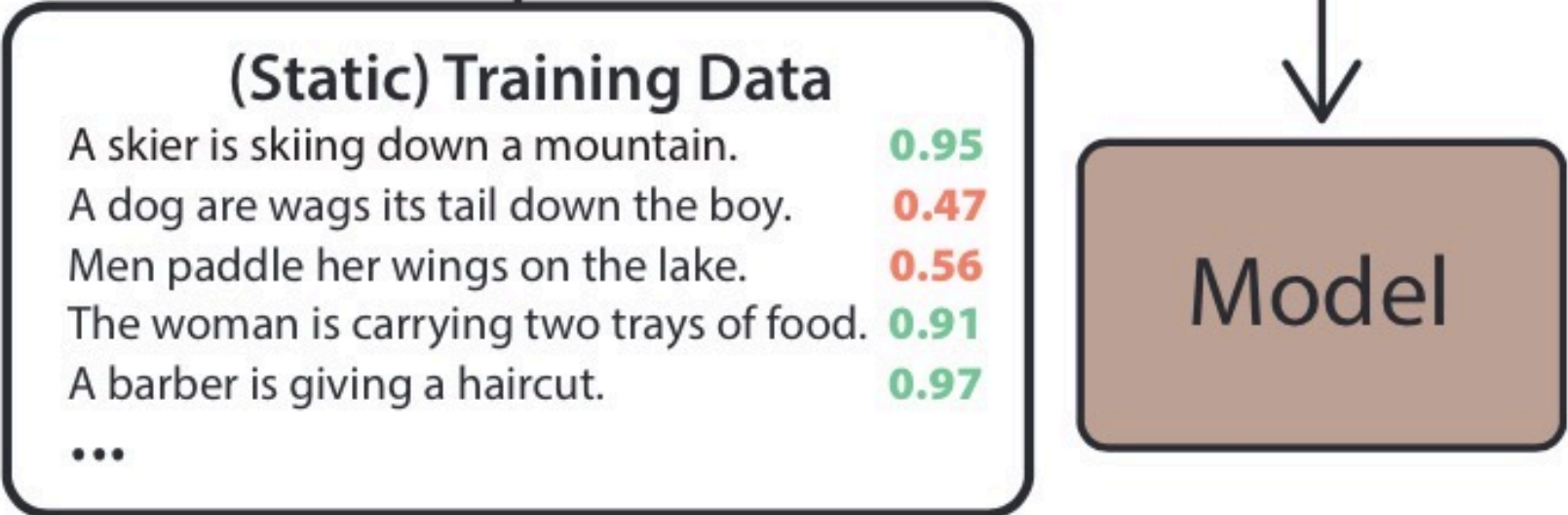
- On-policy exploration to maximize the reward directly



Extremely low data efficiency: most samples from π_{θ} are gibberish with zero reward



Off-policy RL



Recap: RL for Text Generation

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

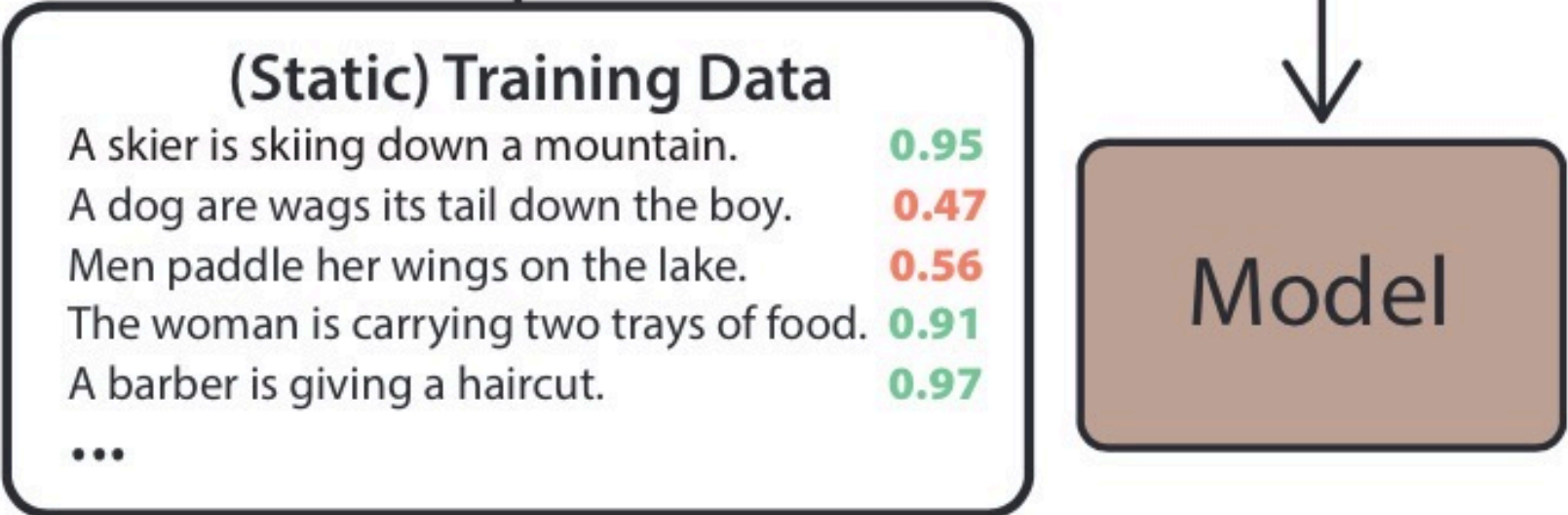
target Q-network

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

Arbitrary policy, e.g., training data

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

Off-policy RL



Recap: RL for Text Generation

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

Slow updates: gradient involves only Q_θ -value of one action a_t (vs 10^6 vocab size)

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

Arbitrary policy, e.g., training data

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

Recap: RL for Text Generation

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

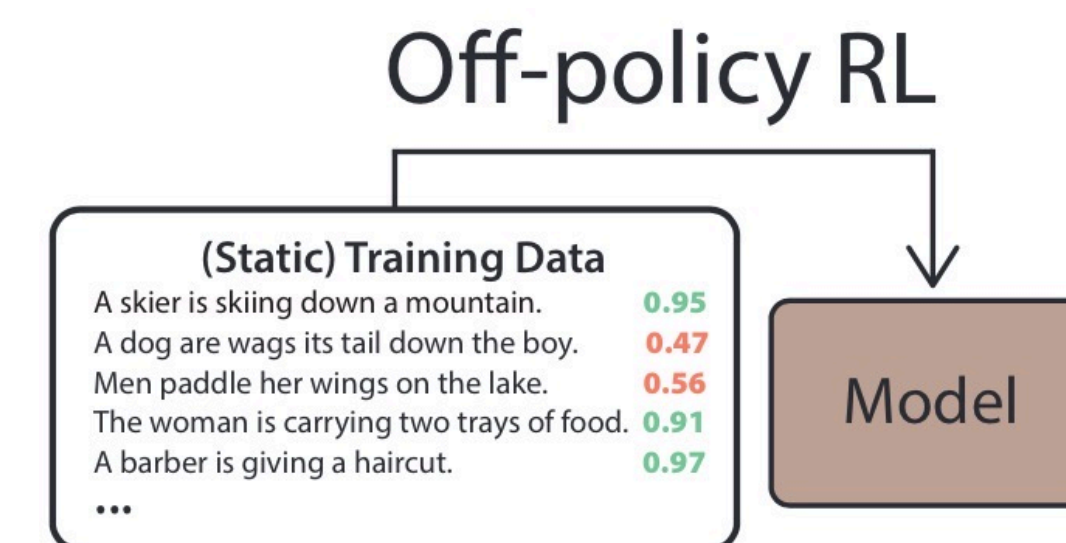
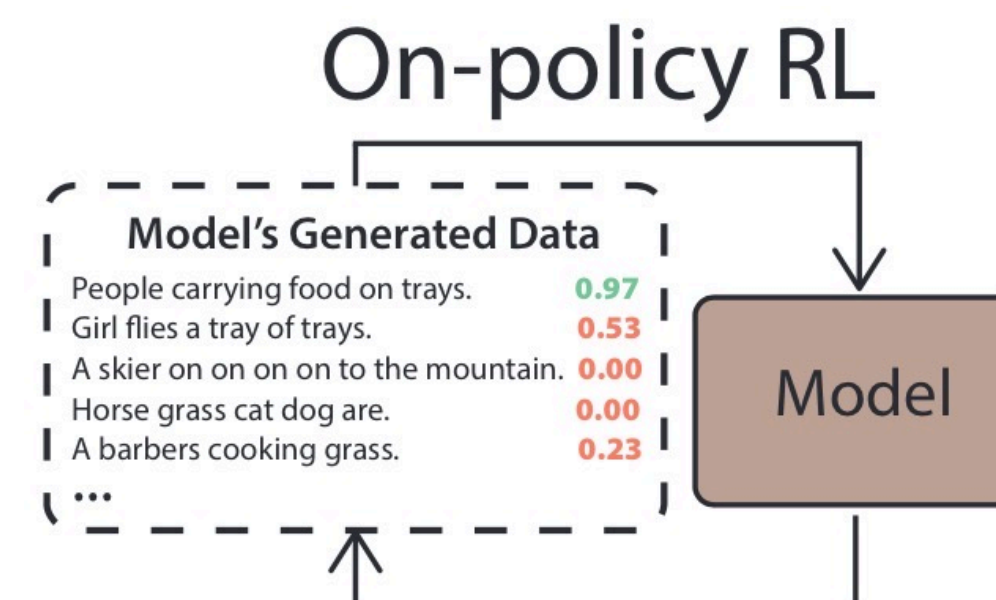
👹 Extremely low data efficiency

- Off-policy RL, e.g., *Q-learning*

👹 Unstable training due to bootstrapping & sparse reward

👹 Slow updates due to large action space

👹 Sensitive to training data quality; lacks on-policy exploration



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

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

SQL

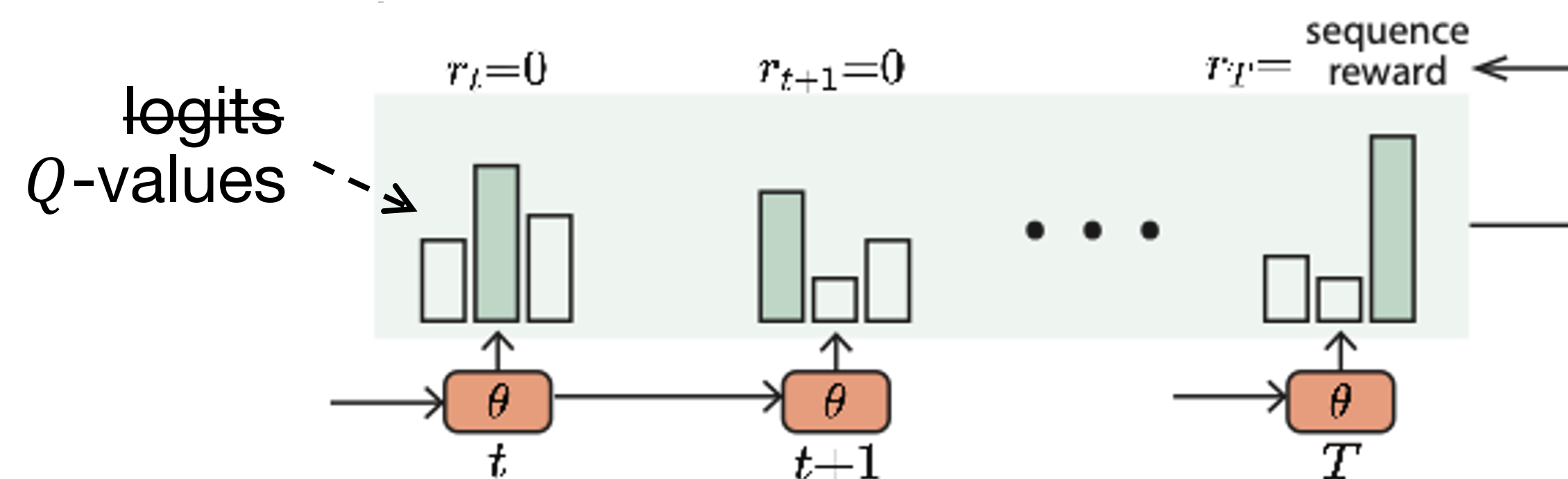
- Goal: entropy regularized

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

- Induced policy

$$\pi_{\theta^*}(a_t | \mathbf{s}_t) = \frac{\exp Q_{\theta^*}(a_t | \mathbf{s}_t)}{\sum_a \exp Q_{\theta^*}(a | \mathbf{s}_t)}$$

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

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

- Training objective:

- Based on temporal consistency

 Unstable training / slow updates

SQL

- Goal: entropy regularized

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

- Induced policy

$$\pi_{\theta^*}(a_t | \mathbf{s}_t) = \frac{\exp Q_{\theta^*}(a_t | \mathbf{s}_t)}{\sum_a \exp Q_{\theta^*}(a | \mathbf{s}_t)}$$

- Training objective:

- Based on **path consistency**

 Stable / efficient

Efficient Training via Path Consistency

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

$$\pi^*(a | \mathbf{s}) = \frac{\exp Q^*(\mathbf{s}, a)}{\sum_{a'} \exp Q^*(\mathbf{s}, a')}$$

- (Single-step) path consistency

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

- Objective

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

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



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

Efficient Training via Path Consistency

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

$$\pi^*(a | \mathbf{s}) = \frac{\exp Q^*(\mathbf{s}, a)}{\sum_{a'} \exp Q^*(\mathbf{s}, a')}$$

- (Single-step) path consistency

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

- Objective

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



Fast updates: gradient involves Q_{θ} values of *all* tokens in the vocab

- (Multi-step) path consistency

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



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

- Objective

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

Efficient Training via Path Consistency

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

$$\pi^*(a | \mathbf{s}) = \frac{\exp Q^*(\mathbf{s}, a)}{\sum_{a'} \exp Q^*(\mathbf{s}, a')}$$

- (Single-step) path consistency

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

- Objective

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



Fast updates: gradient involves Q_{θ} values of all tokens in the vocab

Arbitrary policy:

- Training data (if available) → off-policy updates
- Current policy → on-policy updates
- We combine both for the best of the two



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

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

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

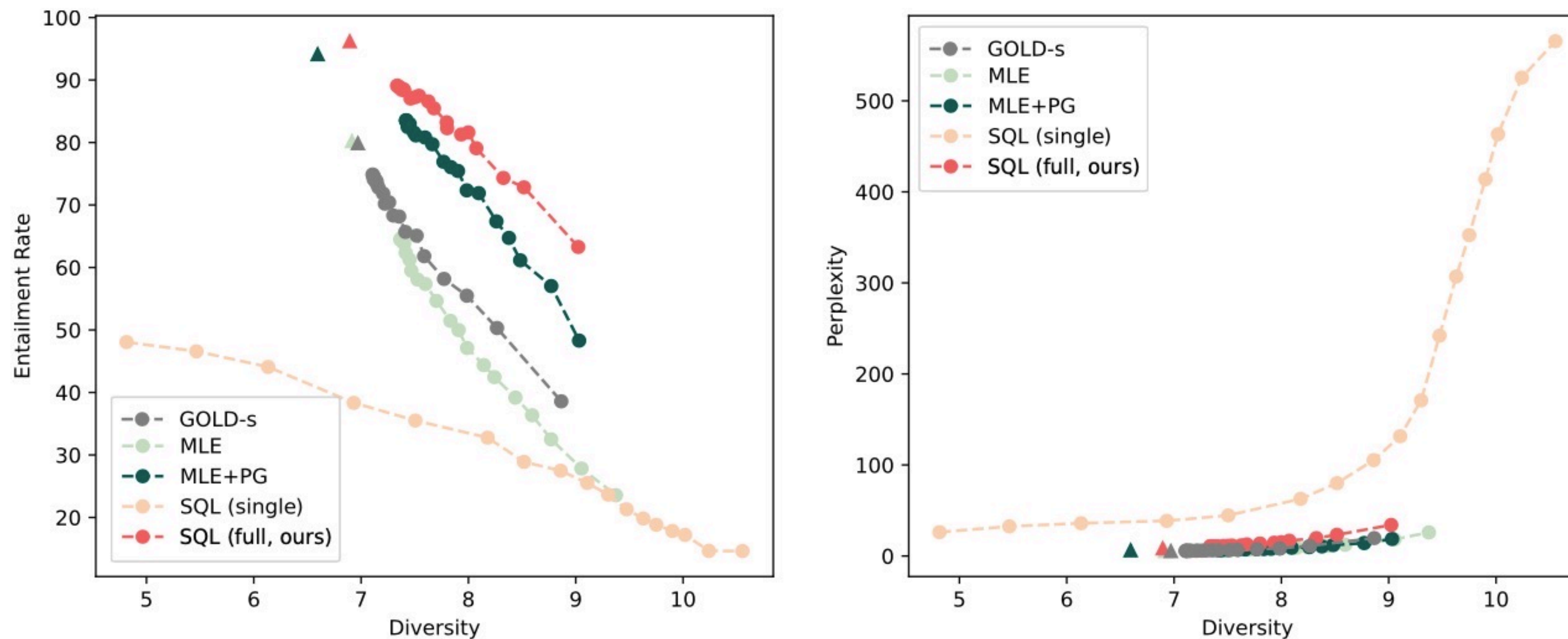
Application (I): Learning from Noisy (Negative) Text

- 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% (\approx **negative** samples)
- Rewards:
 - Entailment classifier
 - Pretrained LM for perplexity
 - BLEU w.r.t input premises (which effectively prevents trivial generations)

Whiteboard

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

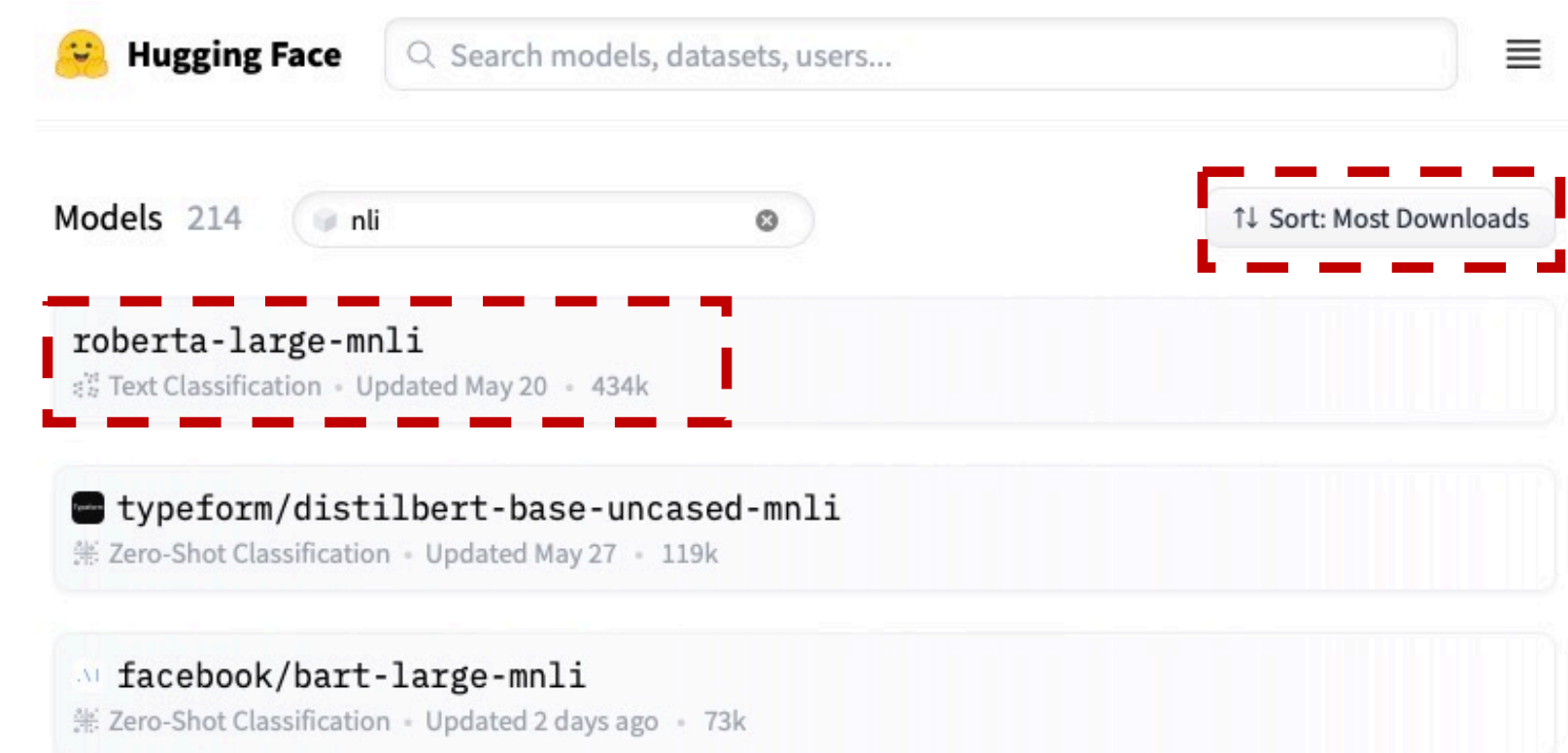
- MLE and pure off-policy RL (GOLD-s) do not work ← rely heavy on data quality
- SQL (full) > MLE+PG (PG alone does not work)
- SQL (single-step only) does not work: the multi-step SQL objective is crucial



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



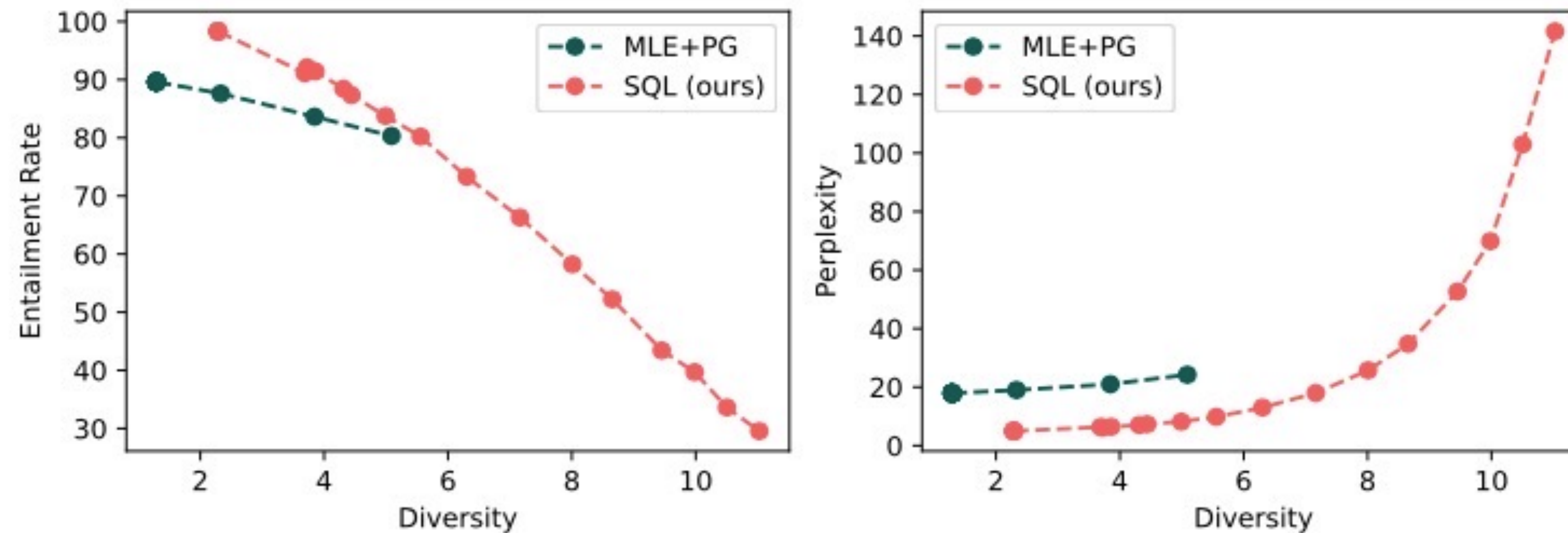
Previous adversarial algorithms are not applicable here:

- only attack for specific premise
- not readable

Whiteboard

Application (II): Universal Adversarial Attacks

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

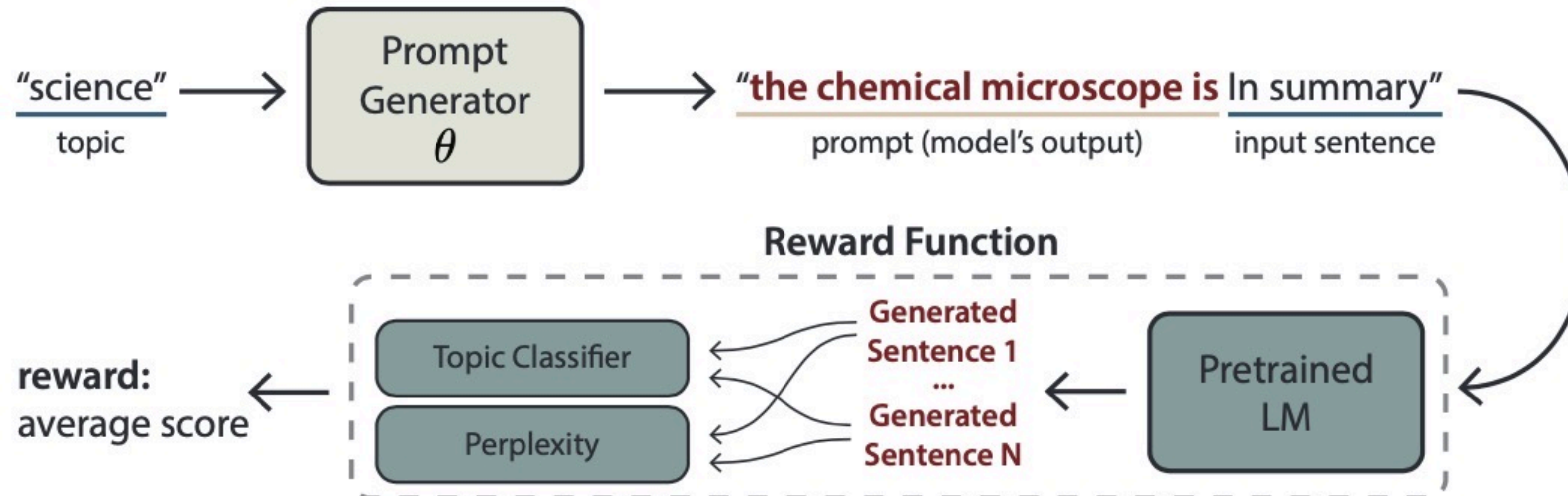


Model	Generation	Rate
MLE+PG	it 's .	90.48
SQL (ours)	the person saint-pierre-et-saint-paul is saint-pierre-et-saint-paul .	97.40

Samples of highest attack rate

Application (III): Prompt Generation for Controlling LMs

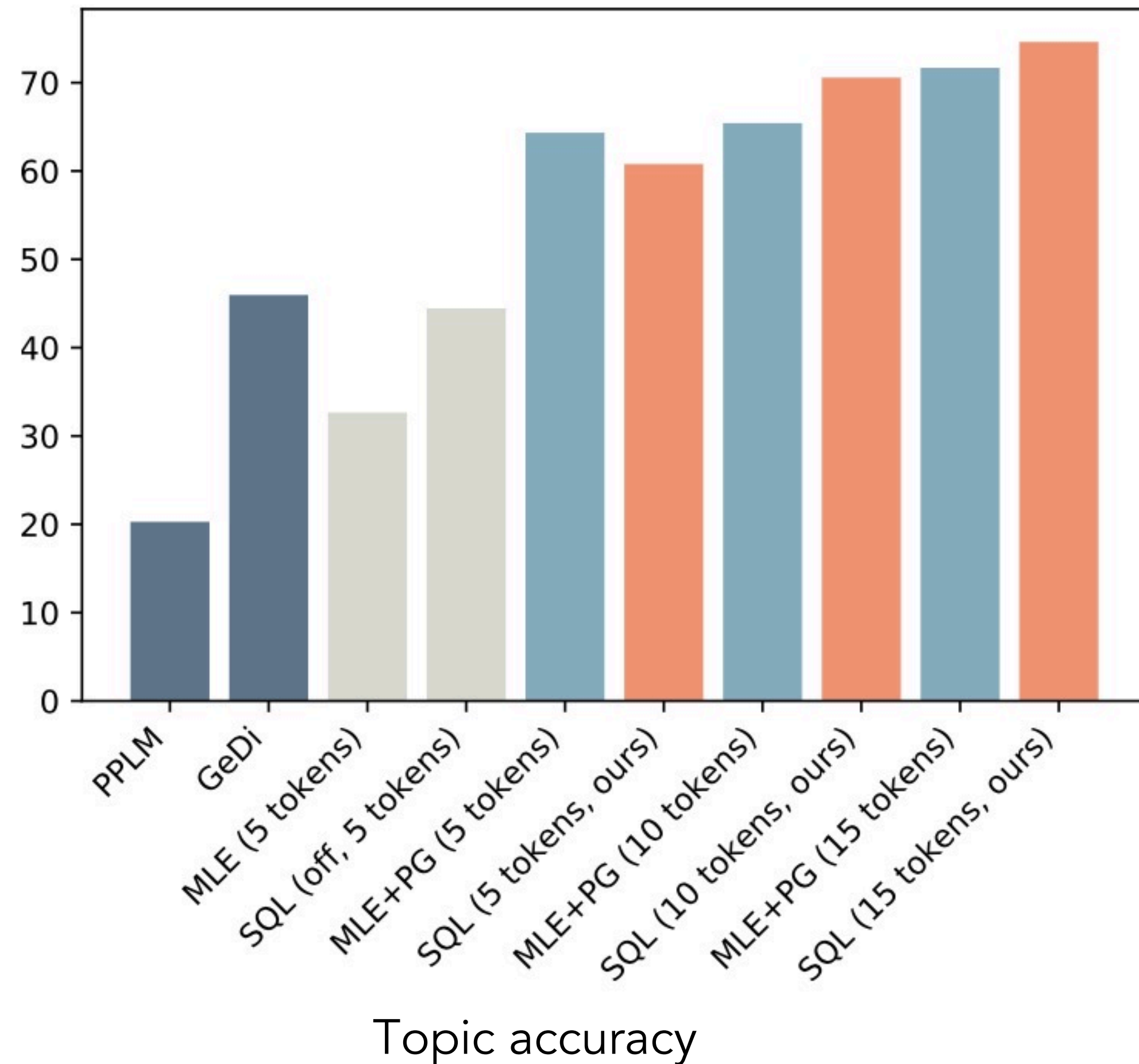
- Generate prompts to steer pretrained LM to produce topic-specific sentences



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 best accuracy-fluency trade-off
- Prompt control by **SQL, MLE+PG** > PPLM, GeDi
 - and much faster at inference!
- **SQL (off-policy only)** > MLE



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

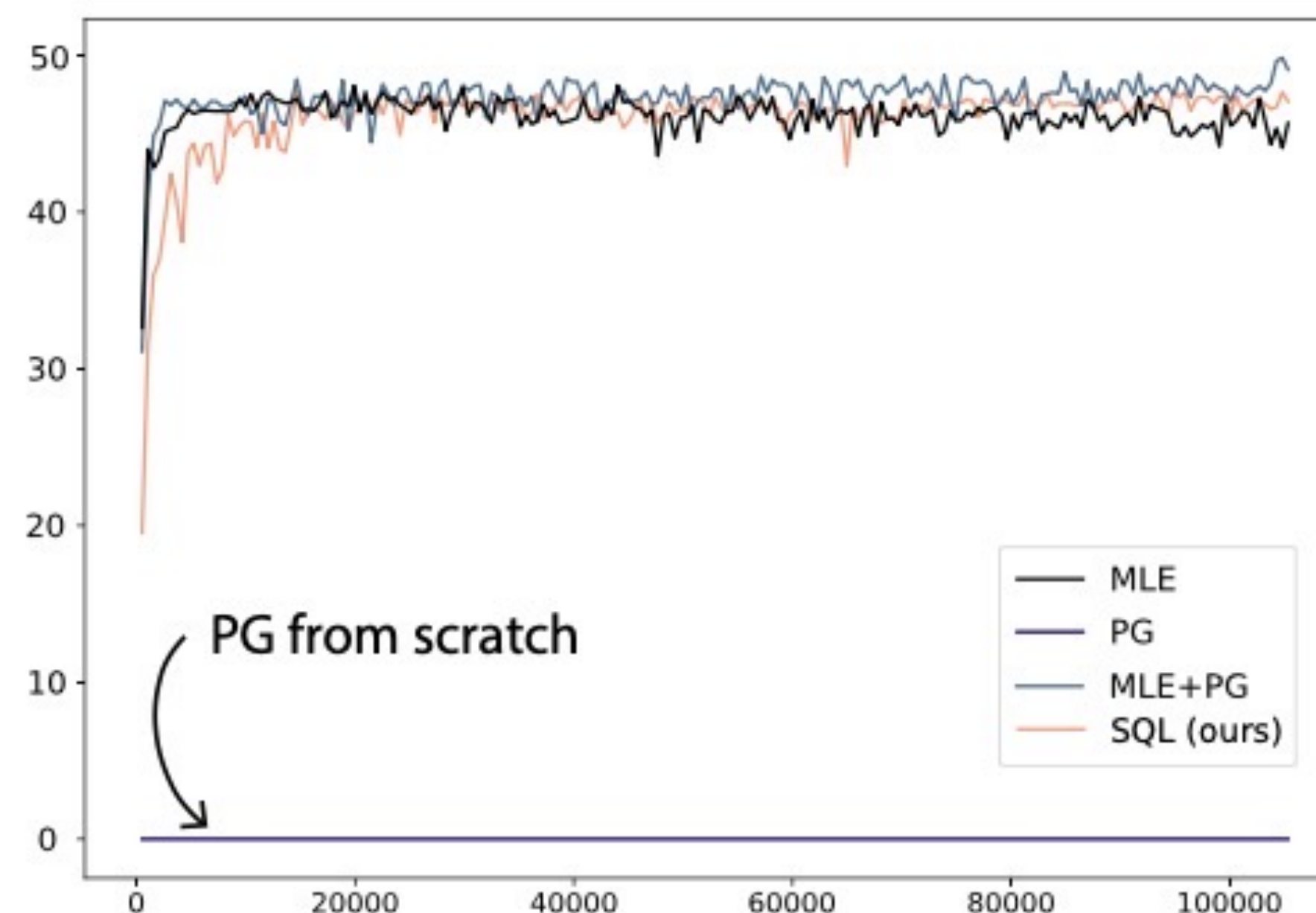
Time cost for generating one sentence

Promising results on standard supervised tasks

- **SQL** from scratch is competitive with **MLE** in terms of performance and stability
 - Results on E2E dataset
 - **PG** from scratch fails

Model	MLE	PG	MLE+PG	SQL (ours)
val	45.67	0.00	49.08	47.04
test	41.75	0.00	42.26	41.70

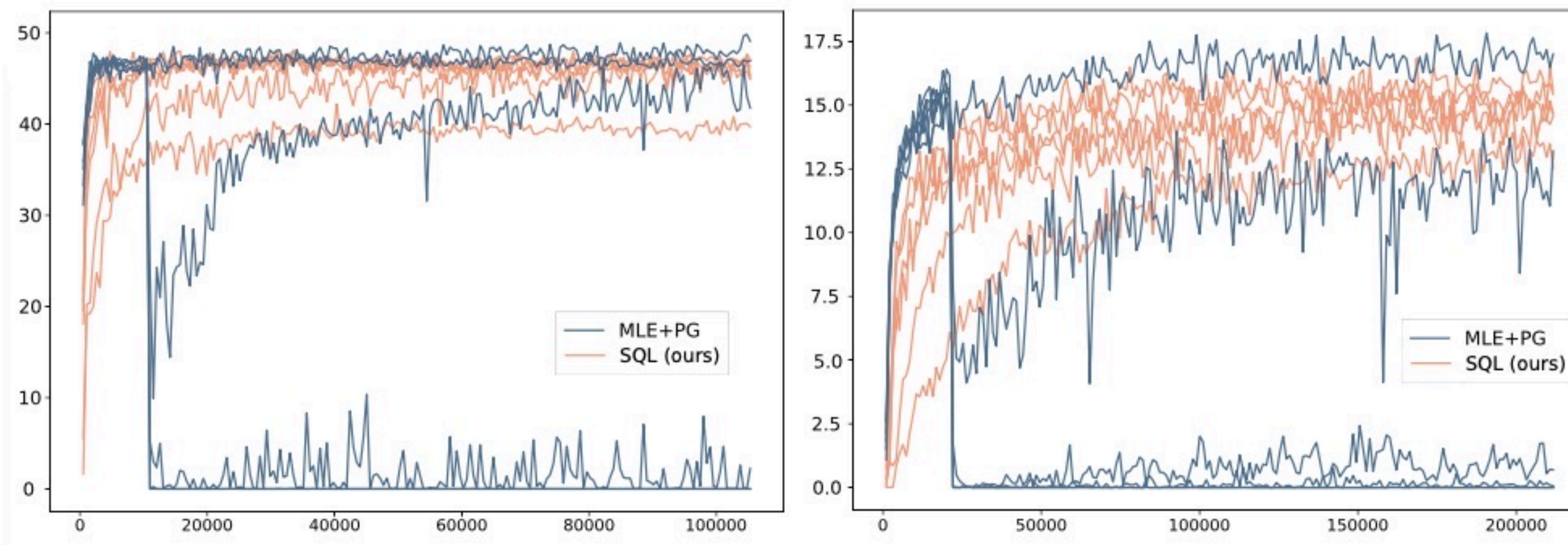
BLEU scores



Training curves

Promising results on standard supervised tasks

- **SQL** from scratch is competitive with **MLE** in terms of performance and stability
 - Results on E2E dataset
 - **PG** from scratch fails
- **SQL** is less sensitive to hyperparameters than **MLE+PG**



Training curves of different reward scales

Key Takeaways

- On-policy RL, e.g., *REINFORCE*, *Policy Gradient (PG)*
 - 👹 Extremely low data efficiency
- Off-policy RL, e.g., *Q-learning*
 - 👹 Unstable training; slow updates; sensitive to training data quality
- SQL
 - Objectives based on path consistency
 - 😊 Combines the best of on-/off-policy
 - 😊 More stable training from scratch given sparse reward
 - 😊 Faster updates given large action space
- Enormous new opportunities for integrating more advanced RL for text generation!

Two Central Goals

- Generating human-like, grammatical, and readable text
 - I.e., generating **natural** language
- Generating text that contains desired information inferred from inputs
 - Machine translation
 - Source sentence --> target sentence w/ the same meaning
 - Data description
 - Table --> data report describing the table
 - Attribute control
 - Sentiment: positive --> "I like this restaurant"
 - Conversation control
 - Control conversation strategy and topic

Two Central Goals

- Generating human-like, grammatical, and readable text
 - Exposure bias, criteria mismatch: reinforcement learning (next lecture)
- Generating text that contains desired information inferred from inputs
 - Machine translation
 - Source sentence --> target sentence w/ the same meaning
 - Data description
 - Table --> data report describing the table
 - Attribute control
 - Sentiment: positive --> "I like this restaurant"
 - Modify sentiment from positive to negative
 - Conversation control
 - Control conversation strategy and topic

Two Central Goals

- Generating human-like, grammatical, and readable text
 - Exposure bias, criteria mismatch: reinforcement learning (next lecture)
- Generating text that contains desired information inferred from inputs

#supervision data

- Machine translation
 - Source sentence --> target sentence w/ the same meaning -----> 10s of millions
- Data description
 - Table --> data report describing the table -----> 10s of 1000s
- Attribute control
 - Sentiment: positive --> "I like this restaurant" -----> 10s of 1000s
 - Modify sentiment from positive to negative -----> 0
- Conversation control
 - Control conversation strategy and topic -----> 0

Two Central Goals

Controlled generation in unsupervised settings

- Generating human-like, grammatical, and readable text
 - Exposure bias, criteria mismatch: reinforcement learning (next lecture)

- Generating text that contains desired information inferred from inputs

#supervision data

- Machine translation
 - Source sentence --> target sentence w/ the same meaning -----> 10s of millions
- Data description
 - Table --> data report describing the table -----> 10s of 1000s
- Attribute control
 - Sentiment: positive --> "I like this restaurant" -----> 10s of 1000s
 - Modify sentiment from positive to negative -----> 0
- Conversation control
 - Control conversation strategy and topic -----> 0

Unsupervised Controlled Generation of Text

- Sentence-level control
 - Text attribute transfer (style transfer)
 - Text content manipulation
- Conversation-level control
 - Target-guided open-domain conversation

Unsupervised Controlled Generation of Text

- Sentence-level control
 - Text attribute transfer (style transfer)
 - Text content manipulation
- Conversation-level control
 - Target-guided open-domain conversation

Text Attribute Transfer

- Modify a given sentence to
 - Have desired attribute values
 - While keeping all other aspects unchanged
- Attribute: sentiment, tense, voice, gender, ...

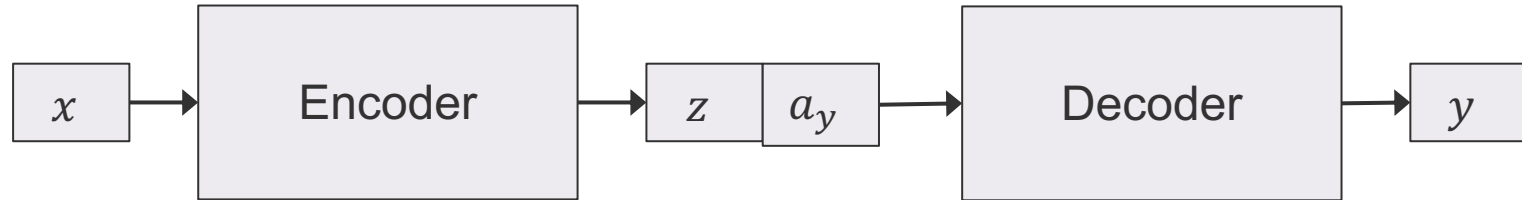
- E.g., transfer sentiment from **negative** to **positive**:
 - “It was super **dry** and had a **weird** taste to the entire slice .”
 - “It was super **fresh** and had a **delicious** taste to the entire slice .”
- Applications:
 - Personalized article writing, emotional conversation systems, ...

Text Attribute Transfer

- Original sentence \mathbf{x} , original attribute \mathbf{a}_x
- Target sentence \mathbf{y} , target attribute \mathbf{a}_y
- Task: $(\mathbf{x}, \mathbf{a}_y) \rightarrow \mathbf{y}$
 - \mathbf{y} has the desired attribute \mathbf{a}_y
 - \mathbf{y} keeps all attribute-independent properties of \mathbf{x}
- Usually, only have pairs of $(\mathbf{x}, \mathbf{a}_x)$, but no $((\mathbf{x}, \mathbf{a}_x), (\mathbf{y}, \mathbf{a}_y))$ for training
 - E.g., two sets of sentences: one with positive sentiment, the other with negative

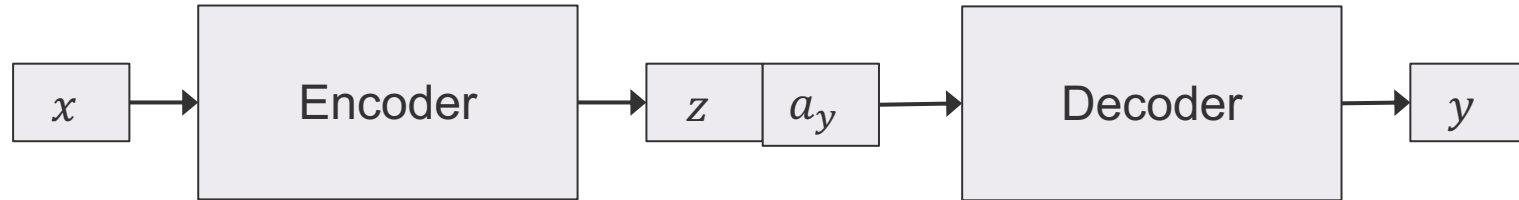
Text Attribute Transfer: Solution

- Task: $(x, a_y) \rightarrow y$
 - y has the desired attribute a_y
 - y keeps all attribute-independent properties of x
- Model $p_\theta(y|x, a_y)$

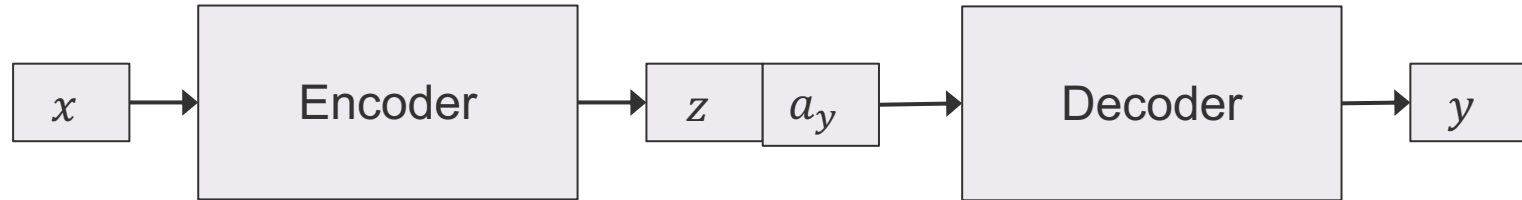


Text Attribute Transfer: Solution

- Task: $(x, a_y) \rightarrow y$
 - y has the desired attribute a_y
 - y keeps all attribute-independent properties of x
- Model $p_\theta(y|x, a_y)$
- Key intuition for learning:
 - Decompose the task into competitive sub-objectives
 - Use direct supervision for each of the sub-objectives

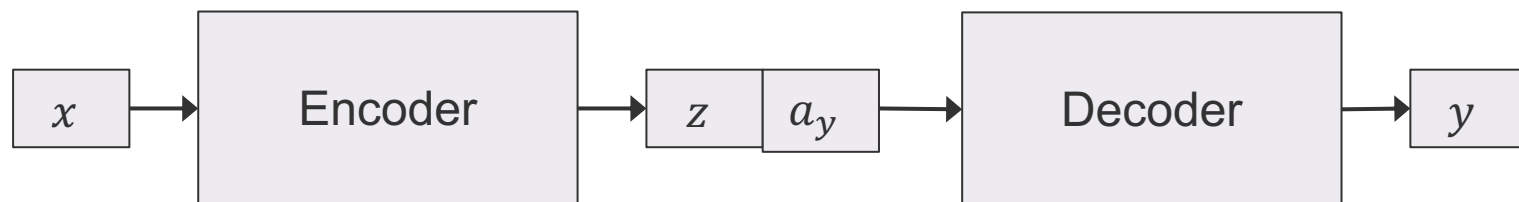


Text Attribute Transfer: Solution



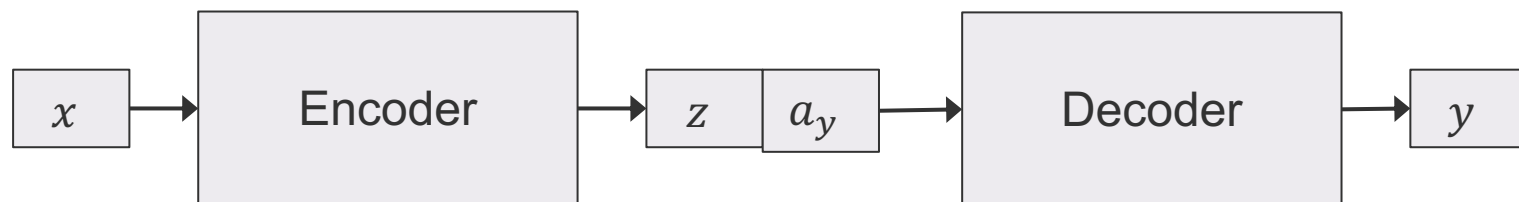
- Task: $(x, a_y) \rightarrow y$
 - y has the desired attribute a_y
 - y keeps all attribute-independent properties of x
- Model $p_\theta(y|x, a_y)$
- Key intuition for learning:
 - Decompose the task into competitive sub-objectives
 - Use direct supervision for each of the sub-objectives
- Auto-encoding loss: $(x, a_x) \rightarrow x$

Text Attribute Transfer: Solution

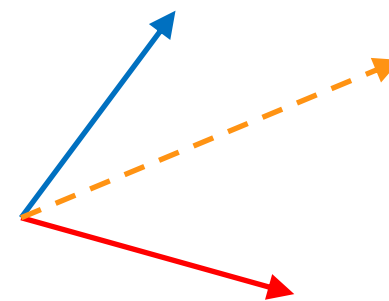


- Task: $(x, a_y) \rightarrow y$
 - y has the desired attribute a_y
 - y keeps all attribute-independent properties of x
- Model $p_\theta(y|x, a_y)$
- Key intuition for learning:
 - Decompose the task into competitive sub-objectives
 - Use direct supervision for each of the sub-objectives
- Auto-encoding loss: $(x, a_x) \rightarrow x$
- Classification loss: $\hat{y} \sim p_\theta(y|x, a_y), f(\hat{y}) \rightarrow a_y$
 - where f is a pre-trained attribute classifier

Text Attribute Transfer: Solution



- Task: $(x, a_y) \rightarrow y$
 - y has the desired attribute a_y
 - y keeps all attribute-independent properties of x
- Model $p_\theta(y|x, a_y)$
- Key intuition for learning:
 - Decompose the task into competitive sub-objectives
 - Use direct supervision for each of the sub-objectives
- Auto-encoding loss: $(x, a_x) \rightarrow x$
- Classification loss: $\hat{y} \sim p_\theta(y|x, a_y), f(\hat{y}) \rightarrow a_y$
 - where f is a pre-trained attribute classifier
- The above two losses are competitive; minimize jointly to avoid collapse



Text Attribute Transfer: Results & Improvement

- Performance on sentiment:
 - Accuracy: 92%
 - BLEU against input sentence: 54

Text Attribute Transfer: Results & Improvement

- Performance on sentiment:
 - Accuracy: 92%
 - BLEU against input sentence: 54
- Problem:
 - Language quality is often not good
 - LM perplexity: 239.8

Original: if i could give them a zero star review i would !

Output: if i **lite** give them a **sweetheart** star review i would !

Original: uncle george is very friendly to each guest

Output: uncle george is very **lackluster** to each guest

Original: the food is fresh and the environment is good

Output: the food is **atrocious** and the environment is **atrocious**

Text Attribute Transfer: Results & Improvement

- Performance on sentiment:
 - Accuracy: 92%
 - BLEU against input sentence: 54
- Problem:
 - Language quality is often not good
 - LM perplexity: 239.8
- Improvement:
 - Use an LM as a direct supervision!
 - $\hat{\mathbf{y}} \sim p_{\theta}(\mathbf{y}|\mathbf{x}, \mathbf{a}_y)$, $\max_{\theta} \text{LM}(\hat{\mathbf{y}})$
 - Accuracy: 91%
 - BLEU against input sentence: 57
 - LM perplexity: 60.9

Original: if i could give them a zero star review i would !

Output: if i **lite** give them a **sweetheart** star review i would !

Original: uncle george is very friendly to each guest

Output: uncle george is very **lackluster** to each guest

Original: the food is fresh and the environment is good

Output: the food is **atrocious** and the environment is **atrocious**

Text Attribute Transfer: Results & Improvement

- Performance on sentiment:
 - Accuracy: 92%
 - BLEU against input sentence: 54
- Problem:
 - Language quality is often not good
 - LM perplexity: 239.8
- Improvement:
 - Use an LM as a direct supervision!
 - $\hat{y} \sim p_{\theta}(y|x, a_y), \max_{\theta} \text{LM}(\hat{y})$
 - Accuracy: 91%
 - BLEU against input sentence: 57
 - LM perplexity: 60.9

Original: if i could give them a zero star review i would !

Output: if i **like** give them a **sweetheart** star review i would !

+ LM: if i can give them a great star review i would !

Original: uncle george is very friendly to each guest

Output: uncle george is very **lackluster** to each guest

+ LM: uncle george is very rude to each guest

Original: the food is fresh and the environment is good

Output: the food is **atrocious** and the environment is **atrocious**

+ LM: the food is bland and the environment is bad .

Unsupervised Controlled Generation of Text

- Sentence-level control
 - Text attribute transfer (style transfer)
 - Text content manipulation
- Conversation-level control
 - Target-guided open-domain conversation

Key idea:

- Decompose the task into **competitive** sub-objectives
- Use **direct supervision** for each of the sub-objectives

Unsupervised Controlled Generation of Text

- Sentence-level control
 - Text attribute transfer (style transfer)
 - Text content manipulation
- Conversation-level control
 - Target-guided open-domain conversation

Key idea:

- Decompose the task into **competitive** sub-objectives
- Use **direct supervision** for each of the sub-objectives

Text Content Manipulation

- Generate a sentence to describe content in a given data record

Data Record

Name	Food	Area	Price	Near
Loch Fyne	Italian	Riverside	£20-25	Strada

Text Content Manipulation

- Generate a sentence to describe content in a given data record
- But language is rich with variation -- there are diverse possible ways of saying the same content (writing style):
 - word choice, expressions, transitions, tones, ...

Data Record

Name	Food	Area	Price	Near
Loch Fyne	Italian	Riverside	£20-25	Strada

Text Content Manipulation

- Generate a sentence to describe content in a given data record
- But language is rich with variation -- there are diverse possible ways of saying the same content (writing style):
 - word choice, expressions, transitions, tones, ...
- We want to control the **writing style**: use the writing style of a reference sentence

Data Record

Name	Food	Area	Price	Near
Loch Fyne	Italian	Riverside	£20-25	Strada

Text Content Manipulation

- Generate a sentence to describe content in a given data record

Data Record

Name	Food	Area	Price	Near
Loch Fyne	Italian	Riverside	£20-25	Strada

Exemplar 1

Zizzi is a pub providing fine French dining but with an expensive price, located near Cocum in the city center.

Generation 1

Loch Fyne provides fine Italian dining with a £20-25 price, located near Strada at the riverside.

Exemplar 2

Located near the Blue Spice, there is a highly-rated place, the Mill, as a choice that frugally priced.

Generation 2

Located near Strada by the river, there is a place with Italian foods, Loch Fyne, as a choice that priced £20-25.

Exemplar 3

With a family-friendly atmosphere and a 5-star rating, Aromi is a pub in the city center.

Generation 3

With Italian foods and a moderate price range, Loch Fyne is near Strada at the riverside.

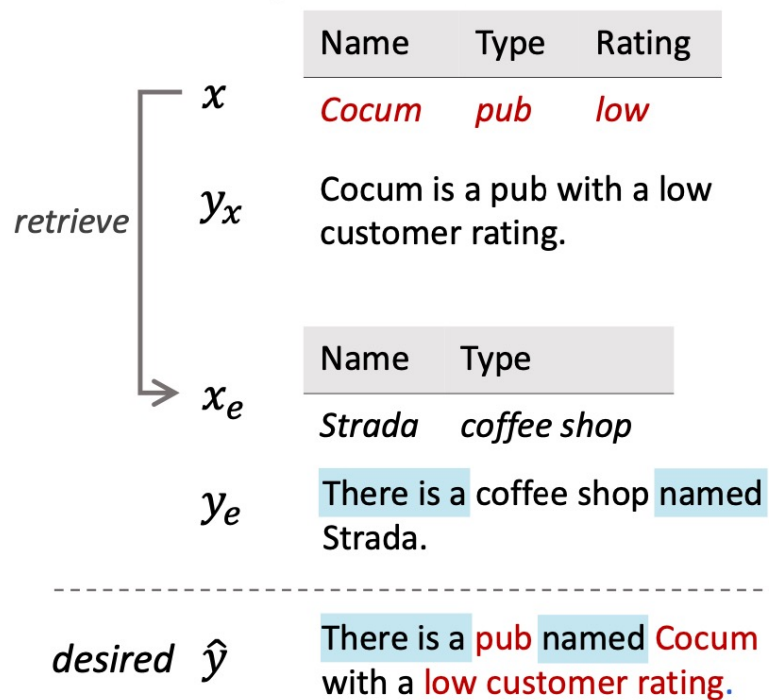
Text Content Manipulation

- Generate a sentence to describe content in a given data record

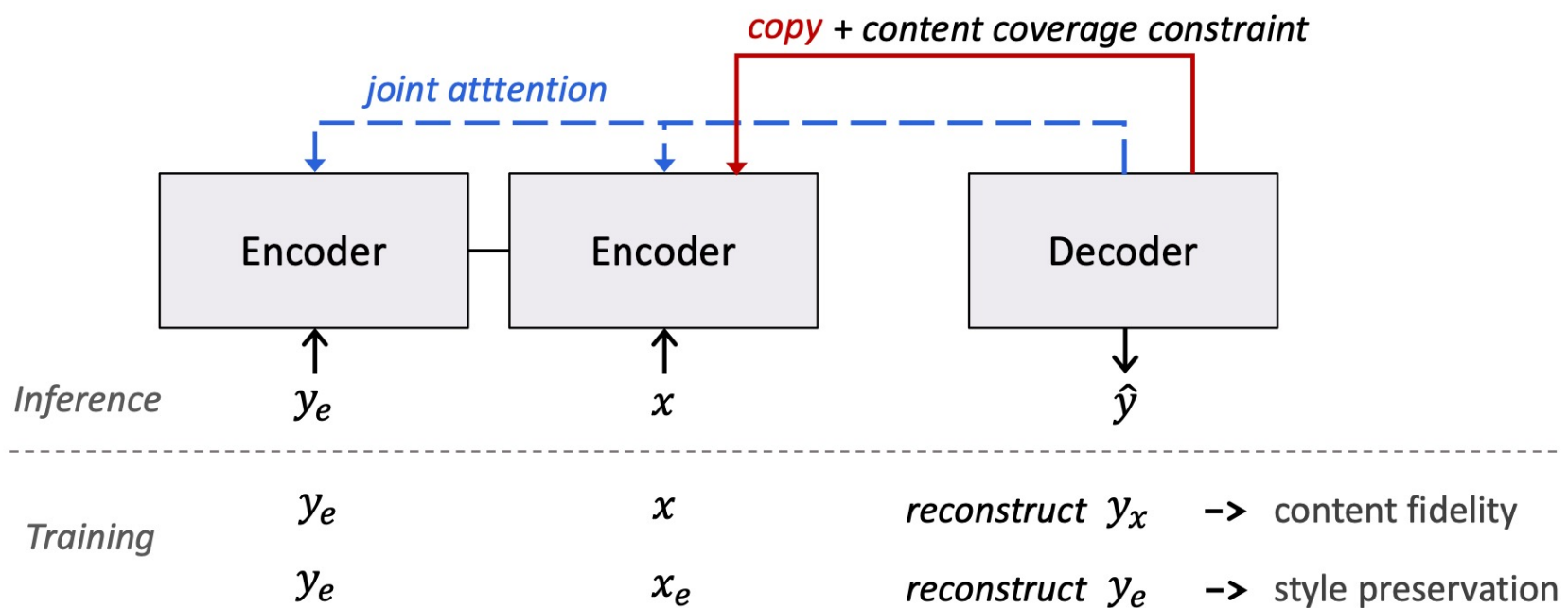
Content Record	PLAYER LeBron_James	PT 32	RB 4	AS 7	PLAYER Kyrie_Irving	PT 20
Reference Sentence	Jrue_Holiday led the way with 26 points and 6 assists , while Goran_Dragic scored 23 points and pulled down 8 rebounds .					
Output	LeBron_James led the way with 32 points , 7 assists and 4 rebounds , while Kyrie_Irving scored 20 points .					

Method

Record and exemplar:



Model:



Results

Content Record	Name	EatType	Food	PriceRange	CustomRating	FamilyFriendly
	Cocum	coffee shop	Italian	£20-25	high	family friendly
Exemplar 1	Looking for French food near Zizzi? Come try Strada, which has a 3-star customer rating and priced lowly.					
Slot filling	Looking for Italian [...] food near Zizzi? Come try [...] Cocum, which has a high customer rating and priced £20-25.					
AdvST	For Italian [...] place near Zizzi? Come try [...] Cocum, which has a high customer rating with priced £20-25.					
Ours	Looking for an Italian coffee shop? Come try family-friendly Cocum, which has a high customer rating and priced £20-25.					
Exemplar 2	Along the riverside near Cafe Rouge, there is a Japanese food place called The Golden Curry. It has an average customer rating since it is not a family-friendly environment.					
Slot-filling	Along the riverside near Cafe Rouge [...], there is a Italian food [...] place called Cocum. It has an high customer rating since it is not a family-friendly environment.					
AdvST	Along the riverside near the Ranch [...], there is a Italian food [...] place called Cocum. It has [...] high customer rating since it is not a family-friendly environment.					
Ours	Priced £20-25, there is an Italian food coffee shop called Cocum. It has a high customer rating since it is a family-friendly environment.					

Results

		Restaurant Recommendations			NBA Reports		
Method		Content		Style	Content		Style
		% Incl.-new	% Excl.-old	m-BLEU	Precision	Recall	m-BLEU
Reference	AttnCopy-S2S	78.88 \pm 2.08	99.71 \pm 0.06	13.95 \pm 0.52	81.62 \pm 3.25	75.65 \pm 7.42	45.5 \pm 0.71
	Slot-filling	61.23	66.2	100	56.69	71.34	100
Baselines	MAST	36.28 \pm 0.25	37.06 \pm 0.16	91.76\pm0.28	23.06 \pm 3.90	27.37 \pm 3.88	95.43\pm2.71
	AdvST	51.64 \pm 4.45	57.06 \pm 4.44	76.02 \pm 5.27	67.37 \pm 0.66	66.79 \pm 1.43	64.67 \pm 4.81
Ours	Transformer w/o Coverage	60.03 \pm 2.16	74.65 \pm 2.69	77.81 \pm 3.83	62.58 \pm 2.88	70.22 \pm 3.58	81.75 \pm 2.32
	+ Coverage	61.84 \pm 1.31	81.14 \pm 2.73	80.29 \pm 0.35	67.74 \pm 0.79	74.35\pm1.22	81.97 \pm 2.87
	LSTM w/o Coverage	60.83 \pm 1.29	81.45 \pm 1.10	78.91 \pm 1.05	68.74 \pm 3.07	69.35 \pm 3.30	79.88 \pm 2.44
	+ Coverage	65.02\pm4.16	82.53\pm0.70	82.92 \pm 3.18	69.54\pm1.16	73.27 \pm 1.18	80.66 \pm 1.89

Unsupervised Controlled Generation of Text

- Sentence-level control
 - Text attribute transfer (style transfer)
 - Text content manipulation
- Conversation-level control
 - Target-guided open-domain conversation

Key idea:

- Decompose the task into **competitive** sub-objectives
- Use **direct supervision** for each of the sub-objectives

Unsupervised Controlled Generation of Text

- Sentence-level control
 - Text attribute transfer (style transfer)
 - Text content manipulation
- Conversation-level control
 - Target-guided open-domain conversation

Key idea:

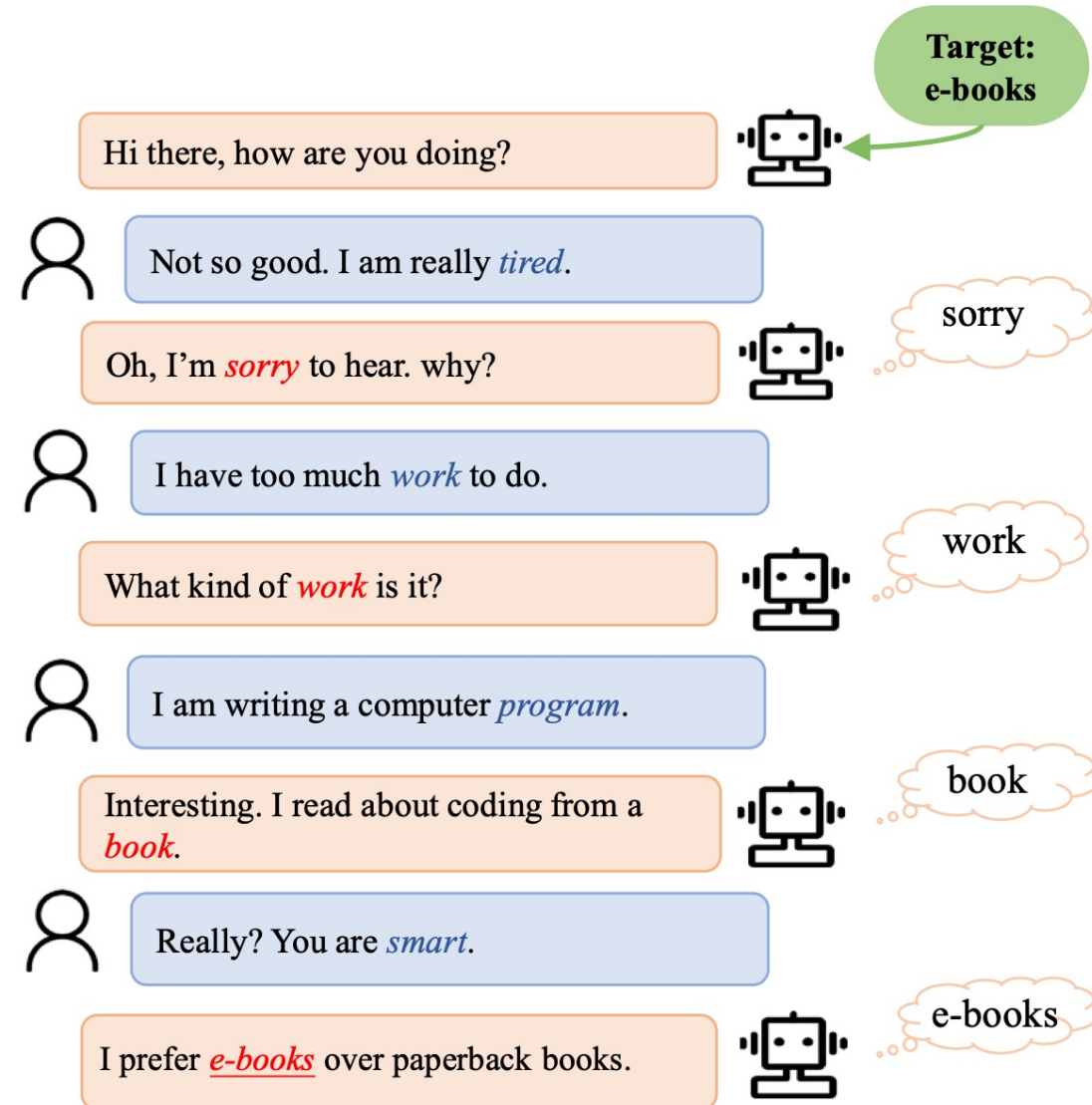
- Decompose the task into **competitive** sub-objectives
- Use **direct supervision** for each of the sub-objectives

Target-guided Open-domain Conversation

- Task-oriented dialog:
 - Address a specific task, e.g., booking a flight
 - Close domain
- Open-domain chit-chat:
 - Improve user engagement
 - Random conversation, hard to control
- Target-guided conversation:
 - Open-domain conversation
 - Controlled conversation strategy to reach a *desired topic* in the end of conversation
 - Applications:
 - Bridges task-oriented dialog and open-domain chit-chat
 - Conversational recommender system, education, psychotherapy

Target-guided Open-domain Conversation

- Two goals:
 - Starting from any topic, reach a desired topic in the end of conversation
 - Natural conversation: smooth transition



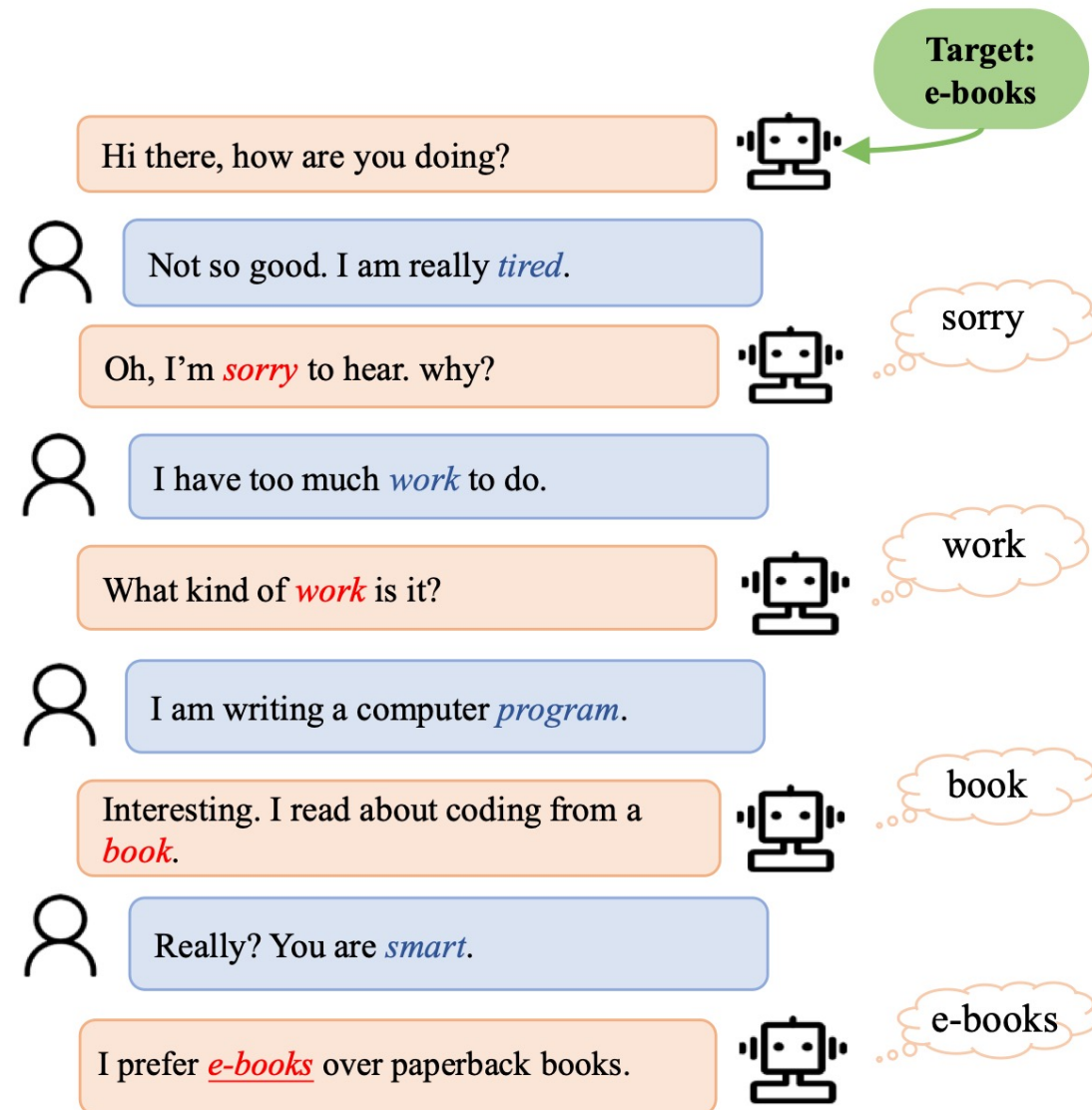
Target-guided Open-domain Conversation

- Two goals:
 - Starting from any topic, reach a desired topic in the end of conversation
 - Natural conversation: smooth transition

Challenge: No supervised data for the task

Solution: Use competitive sub-objectives and partial supervision

- **Natural conversation:** rich chit-chat data to learn smooth **single-turn** transition
- **Reaching desired target:** rule-based **multi-turn** planning



Method

Target: dance

Conversation History

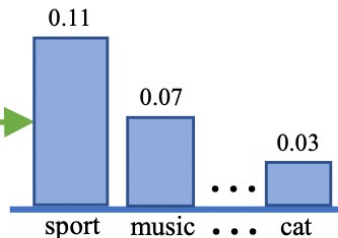
I play **basketball**, do you play?

Yes, I also *like* basketball.

Do you *like* rap **music**? I listen to a lot of rap **music**.

Turn-level Keyword Transition

Keyword Predictor



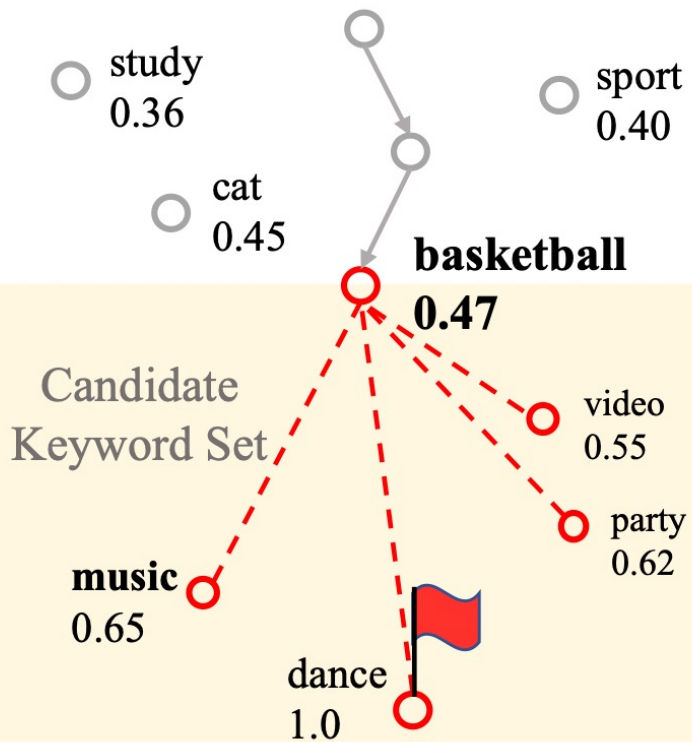
Keyword Selection

music

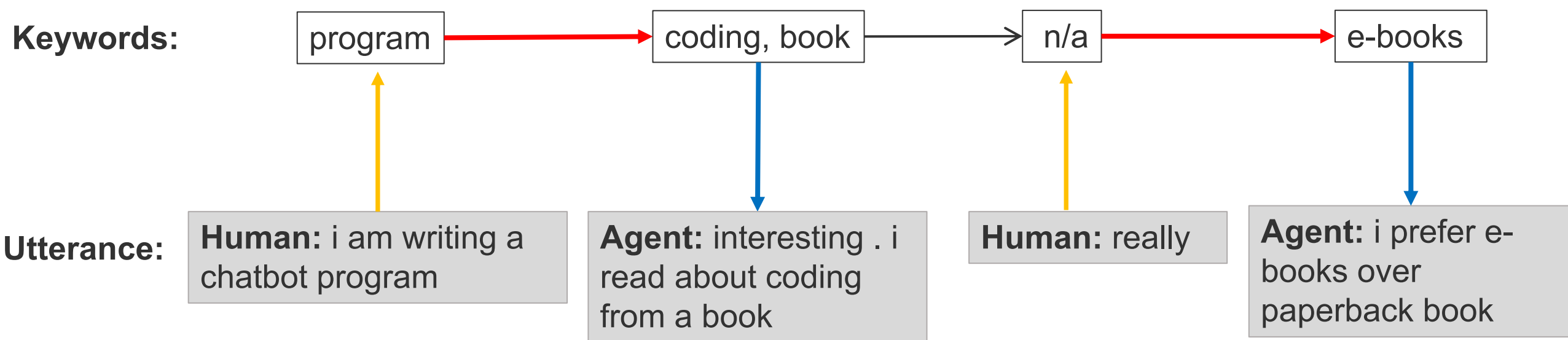
Response Retrieval

Keyword Augmented Response Retrieval

Discourse-level Target-Guided Strategy

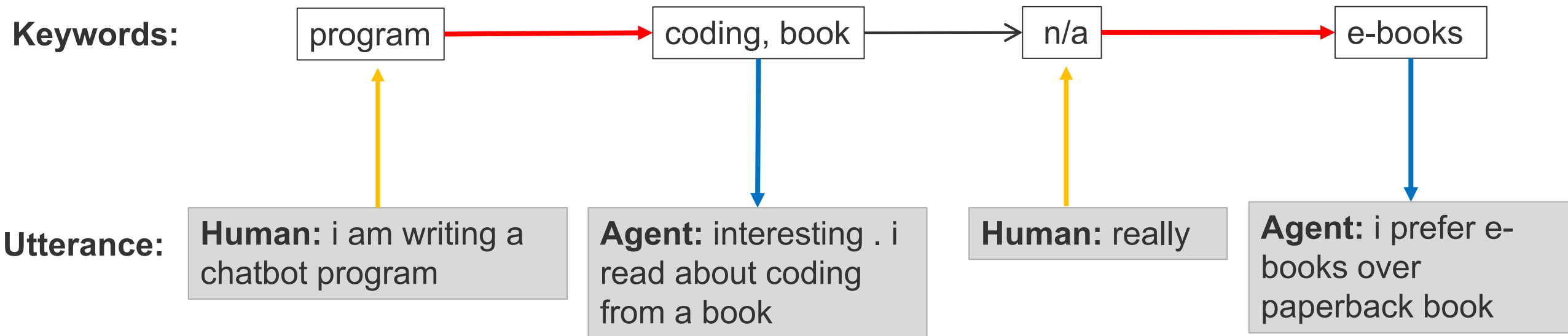


Method





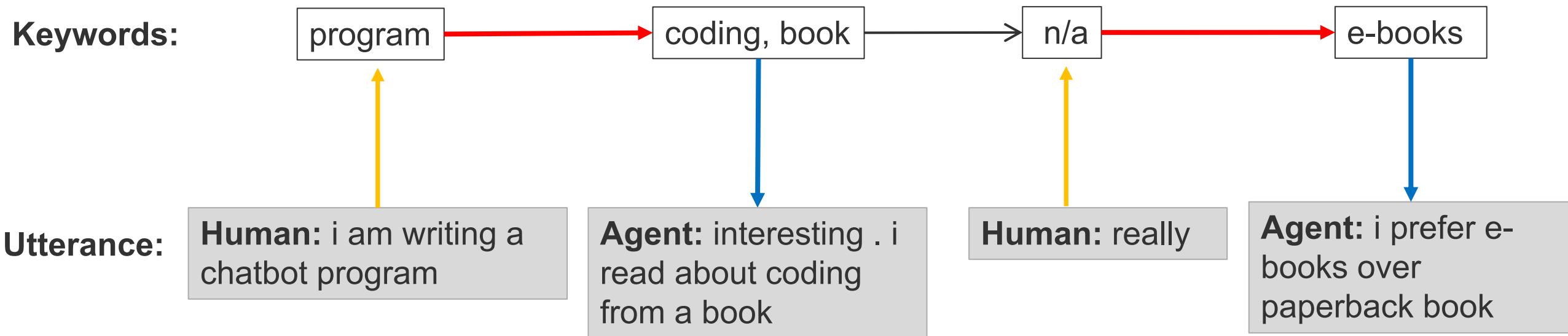
Method

- → keyword extraction






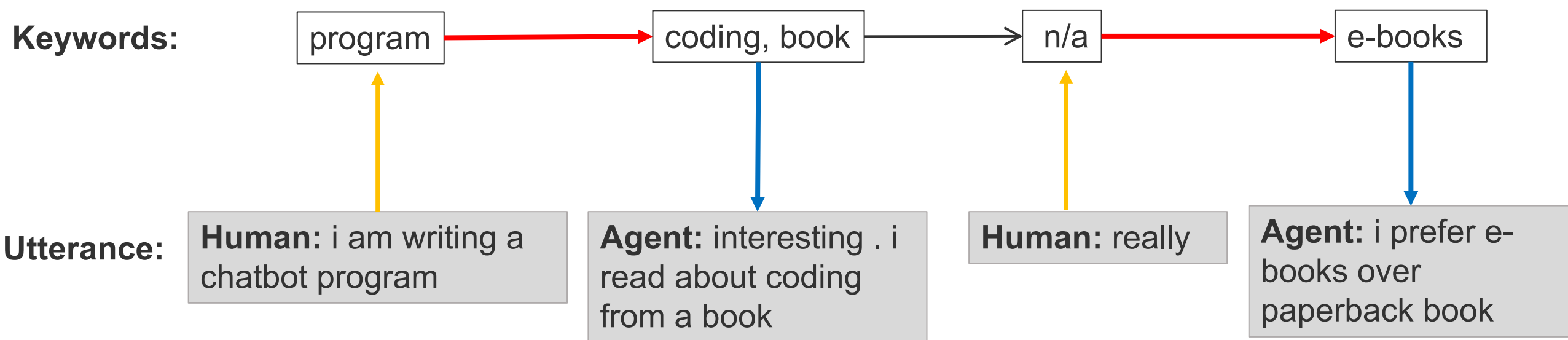
Method

-  keyword extraction
-  keyword conditional response retrieval



Method

-  keyword extraction
-  keyword conditional response retrieval
-  learned kernel-based topic transition + target-guided rule:
 - (1) transit to keywords that are close in the word embedding space – smooth
 - (2) next keywords must get closer to the target keyword – target-guided
 - Tune relative weight of (1) and (2) to control aggressiveness



Example Results

Target: dance

Agent : hi how is it going with you

Human : fine . I'm playing sports

Agent : i play basketball . u play ?

Human : yeah i also like basketball .

Agent : you like rap music ? i listen to a lot of rap music

Human : no i like classic music

Agent : its a taylor swift song

Human : taylor swift is a great singer

Agent : sing to me , what are you going to sing

Human : i know nothing about singing

Agent : oh i think i can dance lol

Example Results – Failure Case

Target: listen

Agent : hi ! how are you ? i am very exciting about my new job !

Human : fine . what is your new job ?

Agent : that sounds exciting .

Human : i like read books

Agent : what kind of books do you read ?

Human : i read comics

Agent : i do not have tv

Human : watch anime is fun

Agent : what position d you play ?

Human : i play a lot of sports

Agent : oh really ? what kind of music . i listen to classical

Unsupervised Controlled Generation of Text

- Sentence-level control
 - Text attribute transfer (style transfer)
 - Text content manipulation
- Conversation-level control
 - Target-guided open-domain conversation

Key idea:

- Decompose the task into **competitive** sub-objectives
- Use **direct supervision** for each of the sub-objectives

Key Takeaways: Two Central Goals

- Generating human-like, grammatical, and readable text
 - Exposure bias, criteria mismatch: reinforcement learning (next lecture)
- Generating text that contains desired information inferred from inputs
 - Machine translation
 - Source sentence --> target sentence w/ the same meaning
 - Data description
 - Table --> data report describing the table
 - Attribute control
 - Sentiment: positive --> "I like this restaurant"
 - Conversation control
 - Control conversation strategy and topic

Questions?