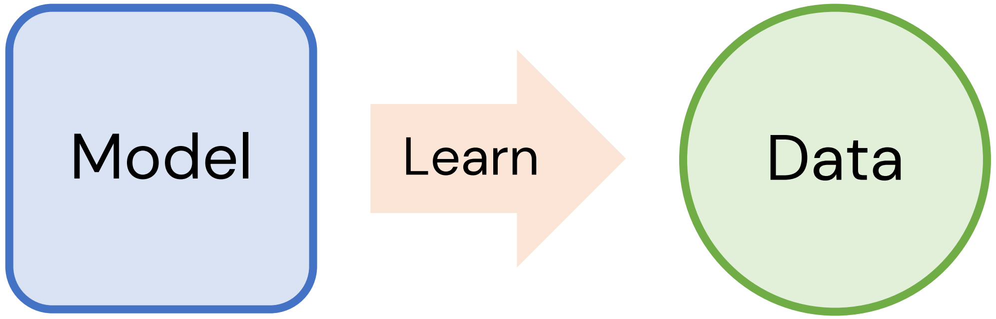
Beyond the Theoretical Limits of Language Modeling: A Distributional Perspective

Haozhe Ji

Tsinghua University

Introduction

- Components of language modeling:



  - **Language data**: $\mathcal{D} = \{\boldsymbol{x}^{(i)}\}_{i=1}^{N}$ drawn from data distribution
  - **Probabilistic Model**: $p_\theta(\boldsymbol{x})$ map data point to probability
  - **Learning objective**: $\mathcal{L}(\theta, \mathcal{D})$ learn model distribution from data
- Choice of model and objective seems not important nowadays. **Really?**

Introduction

- Modern recipe of language modeling:

  Model: Neural language model

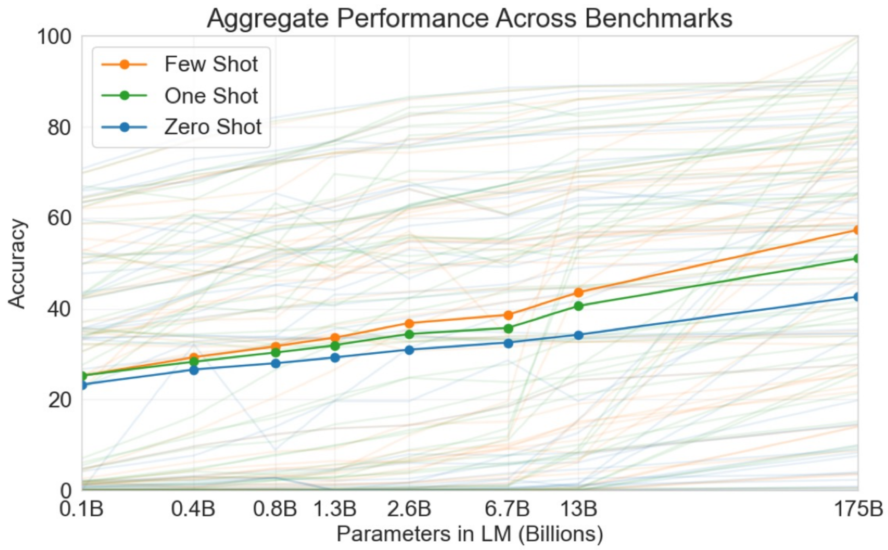
  - Auto-Regressive (AR) model of sequence probability

$$\underbrace{p_\theta(\boldsymbol{x}) = \prod_{t=1}^{T} p_\theta(x_t | x_1, \cdots, x_{<t})}_{\text{Auto-Regressive Modeling}}$$

  Objective: Next token prediction

  - Maximize the likelihood of samples in the dataset

$$\mathcal{L}_{\text{MLE}}(\theta; \mathcal{D}) = \underbrace{\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}}\Big[-\log p_\theta(\boldsymbol{x})\Big]}_{\text{Maximum Likelihood Estimation}}$$



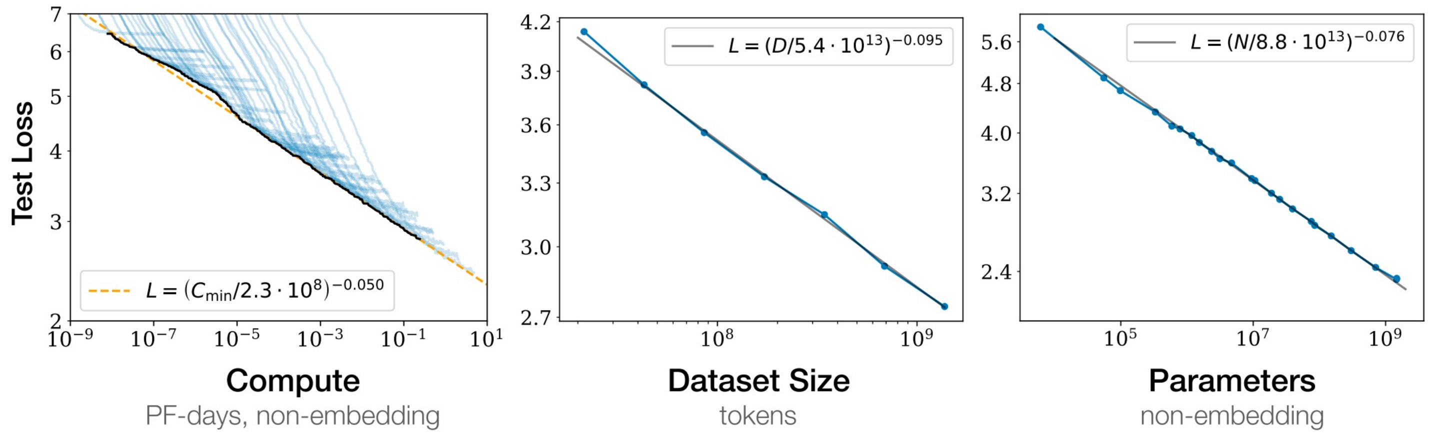
Averaged performance across tasks scales with model sizes

- Language modeling is shown to be the ultimate task towards "intelligence"

Brown, Tom, et al. "Language Models are Few-Shot Learners." *NeurIPS* (2020).

Introduction

- Empirical law for scaling AR language model (LMs) on the MLE loss



$$L(X) \propto X^{\alpha_X}$$

X is one factor from ***{C, D, N}***

MLE loss has a **power-law** relationship with ***C***, ***D***, ***N***

- The power law of scaling one factor depends on the **unbounded value** of the other two factors.
- The return becomes diminished when we **run out of the available human text data** or **cannot afford to increase the model size**!

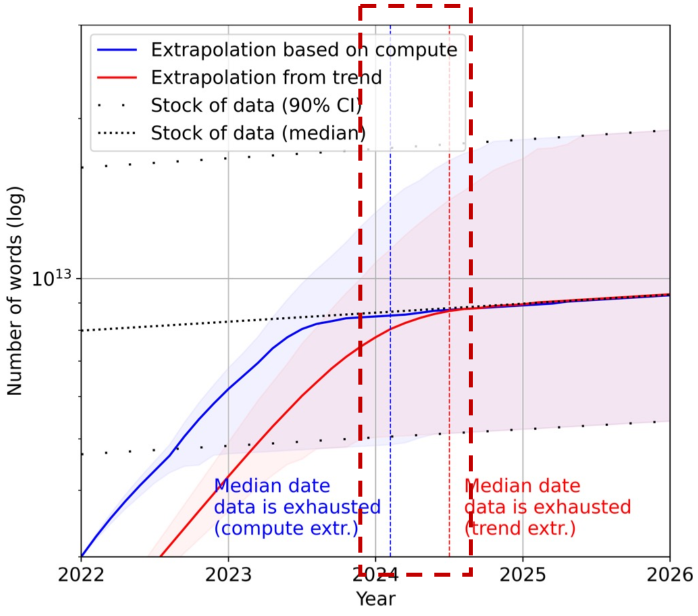
Kaplan, Jared, et al. "Scaling Laws for Neural Language Models." *arXiv preprint* (2020).

Introduction

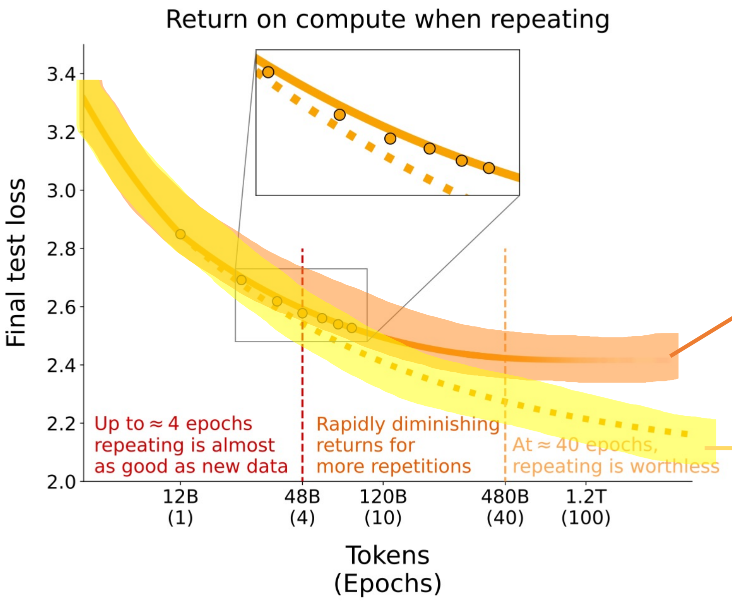
#1 What will happen when we run out of the available human text data?

- Llama3 was trained on 15T tokens, roughly the scale of the quality filtered subsets of Common Crawl, i.e., the high-quality English texts on the Internet.

Data will be "ran out" around 2024 (estimated in 2022)



language data on web



Data-Constrained Scaling law

Data repeating

No data repeating

Muennighoff, Niklas, et al. "Scaling Data-Constrained Language Models." *NeurIPS* (2024).
Villalobos, Pablo, et al. "Will we run out of data? an analysis of the limits of scaling datasets in machine learning." *arXiv preprint* (2022).

Introduction

#1 What will happen when we run out of the available human text data?

- The data spectrum

Quantity ← - - - - - - - - - - → **Quality**

Synthetic data generated by LLM

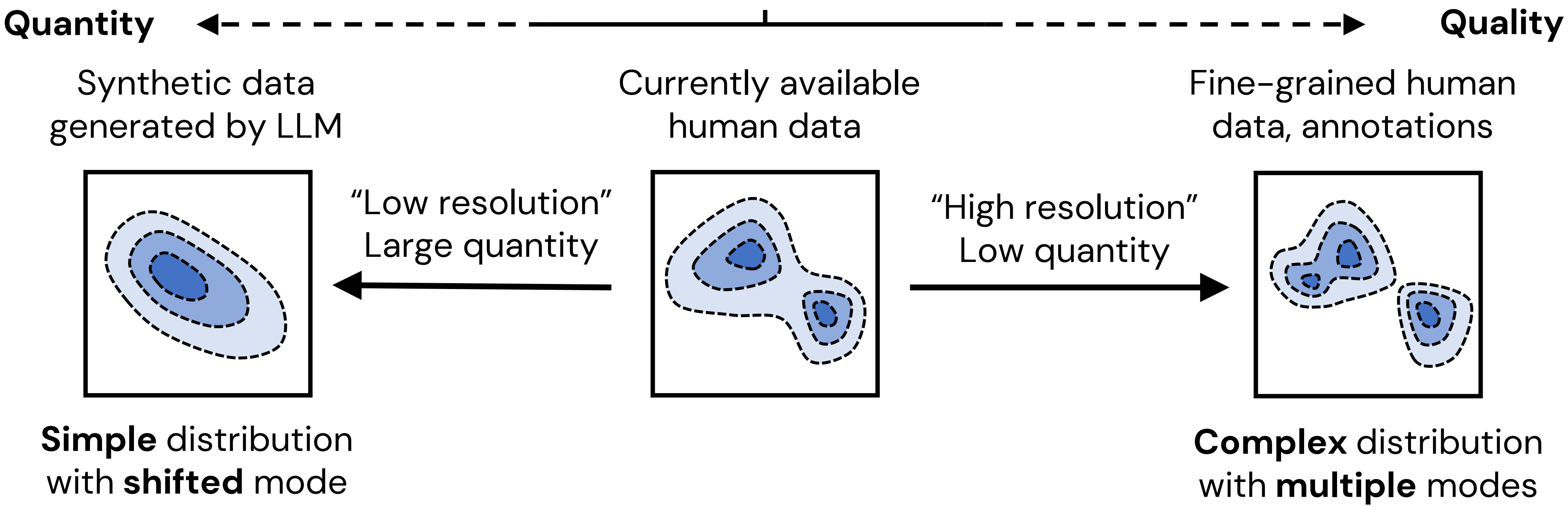
Currently available human data

Fine-grained human data, annotations

Introduction

#1 What will happen when we run out of the available human text data?

- The data spectrum from a **distributional** perspective



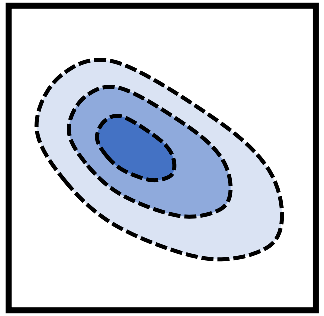
Introduction

#1 What will happen when we run out of the available human text data?

- The data spectrum from a **distributional** perspective

Quantity ← → **Quality**

Synthetic data generated by LLM



Simple distribution with **shifted** mode

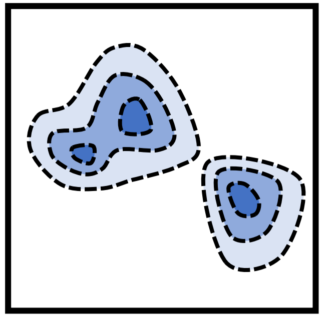
Distillation:
Easy to learn for low-capacity model

Model Collapse:
Cannot persistently improve in long term

Complexity:
Hard to model the entire distribution

***Quality-Aware Objective**:
Selectively capture high-quality modes

Fine-grained human data, annotations



Complex distribution with **multiple** modes

- MLE is **not** aware of quality but coverage (likelihood)!

Shumailov, Ilia, et al. "The Curse of Recursion: Training on Generated Data Makes Models Forget." *arXiv preprint* (2023).

Introduction

#2 What is the parameter complexity of AR LMs to fit the growing data?

- **Theory (Informal)**: AR LMs must be **large enough** to **efficiently compute** the probability of **arbitrary** sequence of length up to ***n*** under the complexity assumption of **P≠NP.**
- **Large parameter**:

$$\left|\theta_n^{\mathrm{AR}}\right| = O(\mathrm{Superpoly}(n))$$

- **Efficient computation**:

$$p_{\theta_n}(\boldsymbol{x}) = \prod_{t=1}^{n} p_{\theta_n}(x_t | x_1, \cdots, x_{t-1})$$

Assumption by AR:
Efficiently predict the **present** based on the **past** in time ***O(poly(n))***

$$p_{\theta_n}(x_t | \boldsymbol{x}_{<t}) = \frac{\sum_{\boldsymbol{x}'_{>t}} p_{\theta_n}(\boldsymbol{x}_{\leq t}, \boldsymbol{x}'_{>t})}{\sum_{\boldsymbol{x}'_{\geq t}} p_{\theta_n}(\boldsymbol{x}_{<t}, \boldsymbol{x}'_{\geq t})}$$

The **present** is predicted by marginalizing out **all possible futures** (Bayesian view)

Lin, Chu-Cheng, et al. "Limitations of Autoregressive Models and Their Alternatives." NAACL (2020).

Introduction

#2 What is the parameter complexity of AR LMs to fit the growing data?

- **Theory (Informal)**: AR LMs must be **large enough** to **efficiently compute** the probability of **arbitrary** sequence of length up to ***n*** under the complexity assumption of **P≠NP.**
- **Large parameter (space)**:

$$\left|\theta_n^{\mathrm{AR}}\right| = O(\mathrm{Superpoly}(n))$$

- **Efficient computation (time)**:

$$p_{\theta_n}(\boldsymbol{x}) = \prod_{t=1}^{n} p_{\theta_n}(x_t | x_1, \cdots, x_{t-1})$$

Assumption by AR:
Efficiently predict the **present** based on the **past** in time ***O(poly(n))***

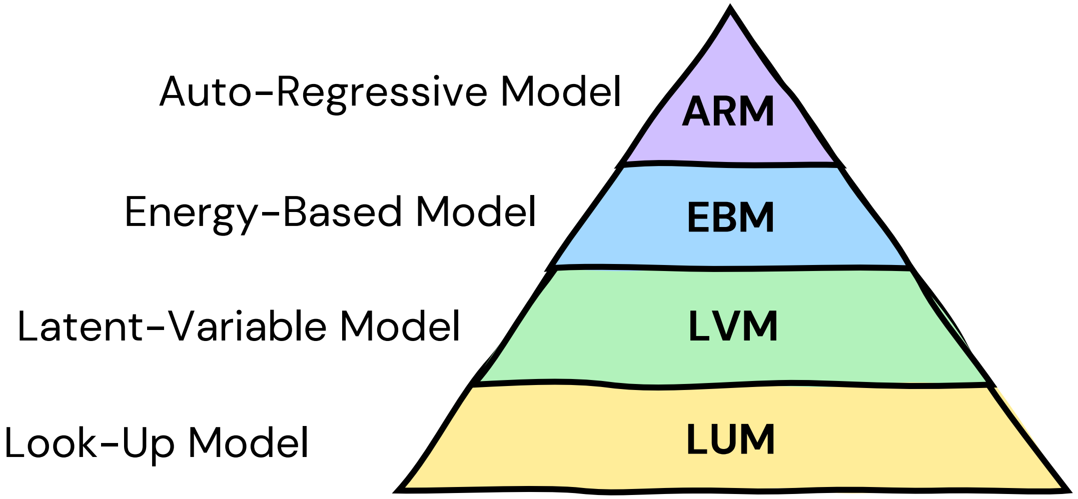
- **Intuition (Space-Time Tradeoff)**: To accurately compute the probability of any sequence, the AR LM must have either **exponential-size computation** or **exponential-size parameters.**

Lin, Chu-Cheng, et al. "Limitations of Autoregressive Models and Their Alternatives." NAACL (2020).

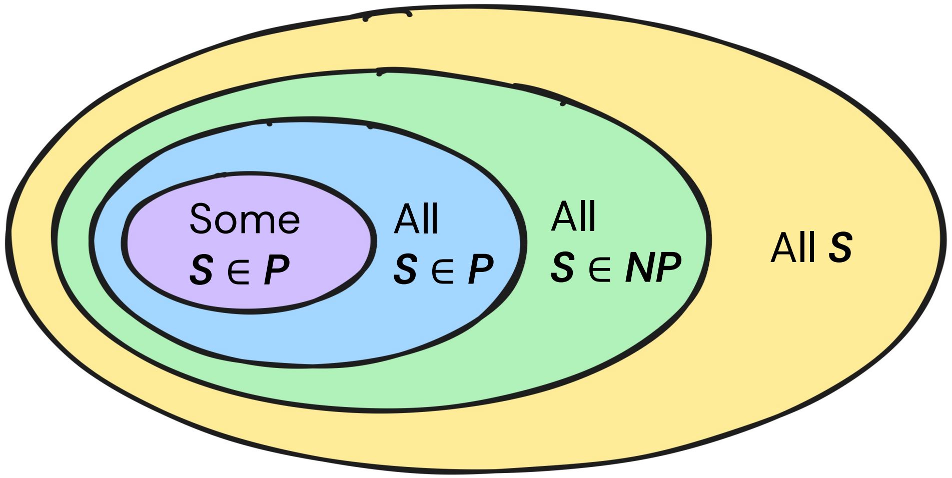
Introduction

#2 What is the parameter complexity of AR LMs to fit the growing data?

- **Corollary**: AR LMs with **compact parameters** grow as ***O(poly(n))*** can only efficiently compute the probability of **a limited subset** of sequences of length up to ***n***.
- Exist more **complex sequence spaces** captured by more **expressive model families**.



Hierarchy of parametric model families



Hierarchy of sequence space
(***S*** is the set of sequences)

Lin, Chu-Cheng, et al. "Limitations of Autoregressive Models and Their Alternatives." NAACL (2020).

Beyond the theoretical limits of language modeling

- **Beyond MLE**: Quality-aware objective
  - **Reverse KL [ICML' 24]**: quality assessed by reward that captures human preference
  - Total variation distance [**ICLR' 23**]: quality assessed by the "optimal classifier" in theory

- **Beyond AR**: Expressive model family
  - **Energy-based model [ICLR' 24]**: Augment AR model with a residual energy model
  - Latent-variable model [**EMNLP' 21**]: Condition AR model with a latent plan
  - Look-up model [**EMNLP' 20**]: Extend AR model with a parallel database look-up

Beyond the theoretical limits of language modeling

- **Beyond MLE: Quality-aware objective**
  - Reverse KL [**ICML' 24**]: quality assessed by reward that captures human preference
  - Total variation distance [**ICLR' 23**]: quality assessed by the "optimal classifier" in theory

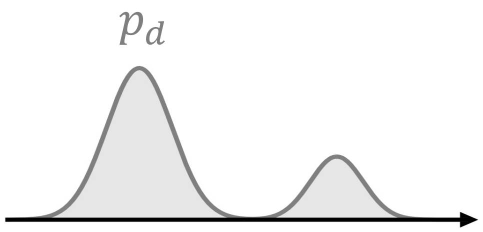
- **Beyond AR**: Expressive model family
  - Energy-based model [**ICLR' 24**]: Augment AR model with a residual energy model
  - Latent-variable model [**EMNLP' 21**]: Condition AR model with a latent plan
  - Look-up model [**EMNLP' 20**]: Extend AR model with a parallel database look-up

MLE for AR LM

- Learning as divergence minimization from a distributional perspective
  - MLE minimizes the **forward-KL (FKL) divergence** from model dist. p_θ to data dist. p_d

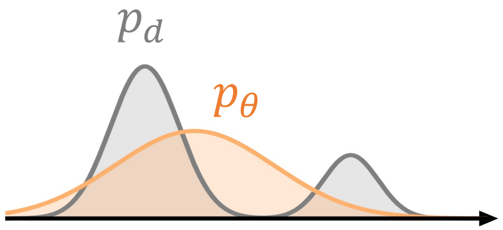
$$\mathbb{E}_{p_d(\boldsymbol{y}|\boldsymbol{x})}\Big[-\log p_\theta(\boldsymbol{y}|\boldsymbol{x})\Big] = \underbrace{\mathbb{D}_{\mathrm{KL}}(p_d \| p_\theta)[\boldsymbol{x}]}_{\text{forward KL}} + \underbrace{H(p_d)[\boldsymbol{x}]}_{\text{entropy}}$$

  - Minimize FKL under **model misspecification**:
    - p_d comes from a more expressive distribution family than p_θ
    - **Example**: p_d is a mixture of Gaussians, p_θ is a single Gaussian



$$\min_\theta \mathbb{E}_{\boldsymbol{y}\sim p_d(\cdot|\boldsymbol{x})}\left[\log \frac{p_d(\boldsymbol{y}|\boldsymbol{x})}{p_\theta(\boldsymbol{y}|\boldsymbol{x})}\right]$$

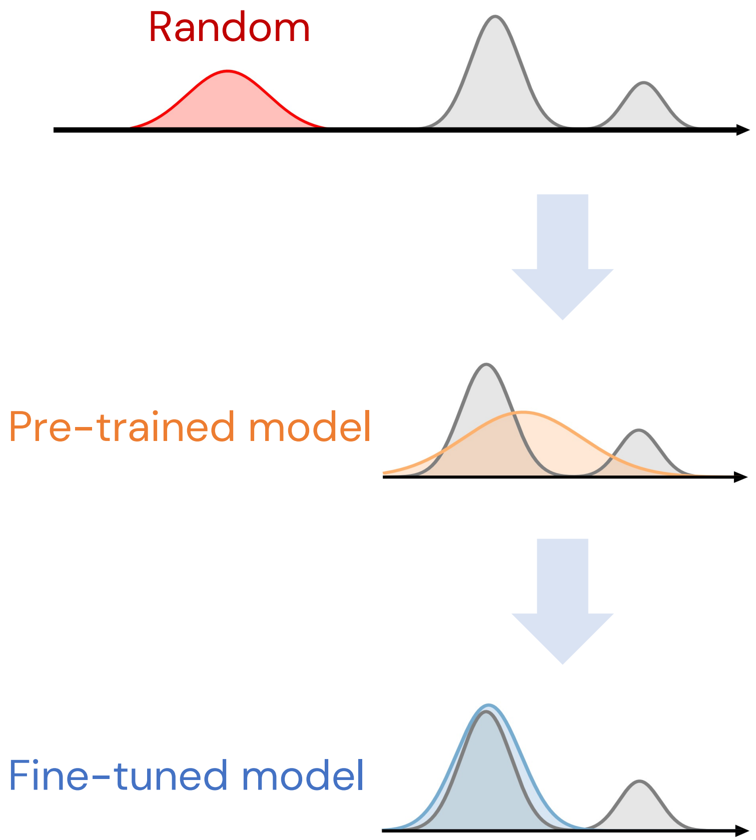
$p_d(y|x) > 0 \rightarrow p_\theta(y|x) > 0$



p_θ cover the support of p_d

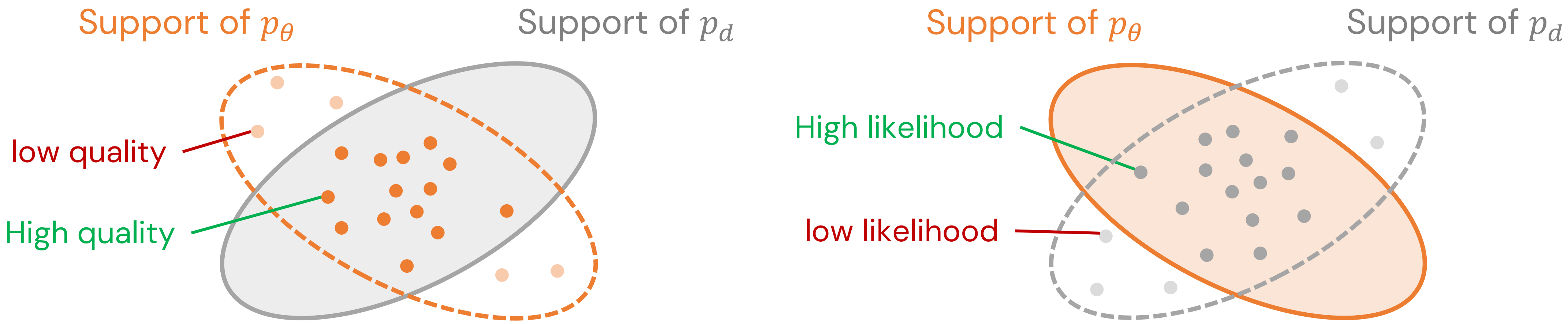
MLE for AR LM

- Is MLE a universal objective for LM training?
  - **Pre-training stage:**
    - Initialization: Random
    - Data: large amount, diverse while noisy
    - Goal: Learn basic knowledge (**coverage**)
  - **Fine-tuning stage:**
    - Initialization: Pre-trained model
    - Data: limited amount, high-quality
    - Goal: Learn fine-grained ability (**quality**)
- **MLE is not desirable when:**
  - Evaluation focuses on quality not coverage
  - Model is mis-specified for the data distribution



Beyond MLE for AR LM

- Forward KL is not informative about the behavior of model on **quality**
- quality vs coverage
  - Quality: Evaluate **samples** generated by model
  - Coverage (likelihood): Evaluate model's **scores** on data samples



- **Challenge of quality-aware objective**: Samples are hard to evaluate than scores!

Beyond the theoretical limits of language modeling

- **Beyond MLE**: Quality-aware objective
  - **Reverse KL [1]**: quality assessed by reward that captures human preference
  - Total variation distance [2]: quality assessed by the "optimal classifier" in theory

- **Beyond AR**: Expressive model family
  - Energy-based model [3]: Augment AR model with a residual energy model
  - Latent-variable model [4]: Condition AR model with a latent plan
  - Look-up model [5]: Extend AR model with a parallel database look-up

Beyond MLE for AR LM

- Controlled assessment of quality by additional human annotation

$x \longrightarrow$ [annotators] $\longrightarrow y$

prompt *response*

$x \quad y_1, \cdots, y_n \longrightarrow$ [annotators] $\longrightarrow y_w > y_l$

prompt *responses* *comparison*

Generative annotation

$$p_d(\boldsymbol{y}|\boldsymbol{x})$$

$$p_\theta(\boldsymbol{y}|\boldsymbol{x})$$

Generative model

Preferential annotation

$$p_d(\boldsymbol{y}_w \succeq \boldsymbol{y}_l \mid \boldsymbol{x})$$

$$r_\phi(\boldsymbol{x}, \boldsymbol{y})$$

Reward model

- Preference data: Fine-grained signal of **quality** to shape the target distribution
- Discrimination vs Generation: EBM can capture more complex distribution than ARM

Ziegler, Daniel M., et al. "Fine-tuning language models from human preferences." *arXiv preprint arXiv:1909.08593* (2019).

LM Alignment

- LM alignment with human preference [**Ouyang et al., 2022**]:
  - Alignment objective (RLHF): KL-regularized reward maximization

$$\mathcal{J}_{\mathrm{lhf}}^{\beta}(\pi_\theta) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}^{\mathrm{pref}}}\Big(\mathbb{E}_{\pi_\theta(\boldsymbol{y}|\boldsymbol{x})}[r_\phi(\boldsymbol{x},\boldsymbol{y})] - \beta\mathbb{D}_{\mathrm{KL}}[\pi_\theta(\boldsymbol{y}|\boldsymbol{x})\|\pi_{\mathrm{sft}}(\boldsymbol{y}|\boldsymbol{x})]\Big)$$

*Reward model (**proxy** human preference)*

reference LM (initialized by MLE)

$$R(\boldsymbol{x},\boldsymbol{y}) = r_\phi(\boldsymbol{x},\boldsymbol{y}) - \beta\log\frac{\pi_\theta(\boldsymbol{y}|\boldsymbol{x})}{\pi_{\mathrm{sft}}(\boldsymbol{y}|\boldsymbol{x})}$$

$$\nabla_\theta\mathcal{J}_{\mathrm{lhf}}^{\beta}(\pi_\theta) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}^{\mathrm{pref}},\boldsymbol{y}\sim\pi_\theta(\boldsymbol{y}|\boldsymbol{x})}\Big[R(\boldsymbol{x},\boldsymbol{y})\nabla_\theta\log\pi_\theta(\boldsymbol{y}|\boldsymbol{x})\Big]$$

*Policy gradient, Actor-Critic, e.g., PPO [**Schulman et al., 2017**]*

*RL has **high variance** in policy gradient estimation*
*RL needs to **sample in training loop***
} ***Inefficiency** of convergence*

Ouyang, Long, et al. "Training language models to follow instructions with human feedback." *Advances in neural information processing systems* 35 (2022)

LM Alignment

- Direct Preference Optimization (DPO) [**Rafailov et al., 2023**]:
  - **Key intuition**: Policy optimization as reward modeling.

$$\mathcal{J}_{\mathrm{lhf}}^{\beta}(\pi_\theta)$$

Alignment objective

KKT condition

Assume ***unlimited*** *model capacity*

$$\pi_\beta^*(\boldsymbol{y}|\boldsymbol{x}) = \pi_{\mathrm{sft}}(\boldsymbol{y}|\boldsymbol{x}) \frac{e^{\frac{1}{\beta} r_\phi(\boldsymbol{x},\boldsymbol{y})}}{Z_\beta(\boldsymbol{x})}$$

Analytic solution of maximizing $\mathcal{J}_{\mathrm{lhf}}^{\beta}(\pi_\theta)$

Simple algebra

$$r_\phi(\boldsymbol{x},\boldsymbol{y}) = \beta \log \frac{\pi_\beta^*(\boldsymbol{y}|\boldsymbol{x})}{\pi_{\mathrm{sft}}(\boldsymbol{y}|\boldsymbol{x})} + \beta \log Z_\beta(\boldsymbol{x})$$

Reward model as a function of π_β^*

BT model

$$\mathcal{L}_{\mathrm{dpo}}(\pi_\theta) = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y}_w,\boldsymbol{y}_l)\sim\mathcal{D}^{\mathrm{pref}}}\Big[-\log\sigma\Big(\beta\log\frac{\pi_\theta(\boldsymbol{y}_w|\boldsymbol{x})}{\pi_{\mathrm{sft}}(\boldsymbol{y}_w|\boldsymbol{x})} - \beta\log\frac{\pi_\theta(\boldsymbol{y}_l|\boldsymbol{x})}{\pi_{\mathrm{sft}}(\boldsymbol{y}_l|\boldsymbol{x})}\Big)\Big]$$

DPO: Optimize the policy using preference loss

Equivalence?

  - L_{dpo} *is* ***not*** *equivalent to* J_{lhf} *considering the expressivity gap between* π_θ *and* π_β^*

Rafailov, Rafael, et al. "Direct preference optimization: Your language model is secretly a reward model." *Advances in Neural Information Processing Systems* 36 (2024)

LM Alignment

- *What does the solution of RLHF look like under this practical constraint?*
  - *KL-regularized RL as probability matching [**Korbak et al., 2021**].*

$$\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}^{\text{pref}}}\left(\mathbb{E}_{\pi_\theta(\boldsymbol{y}|\boldsymbol{x})}[r_\phi(\boldsymbol{x},\boldsymbol{y})] - \beta\mathbb{D}_{\text{KL}}[\pi_\theta(\boldsymbol{y}|\boldsymbol{x})\|\pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})]\right) \quad \overset{\text{equivalent}}{\Longleftrightarrow} \quad \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}^{\text{pref}}}\left[\mathbb{D}_{\text{KL}}(\pi_\theta(\boldsymbol{y}|\boldsymbol{x})\|\pi^*_{\beta_r}(\boldsymbol{y}|\boldsymbol{x}))\right]$$

Maximize reward with KL penalty — *Minimize reverse KL divergence*

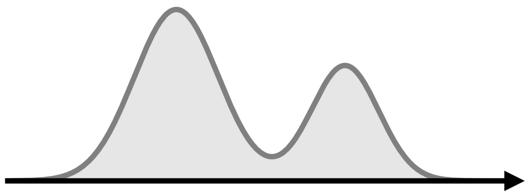
- *The asymmetry of KL divergence:*
  - *Estimate the density of p*

Forward KL

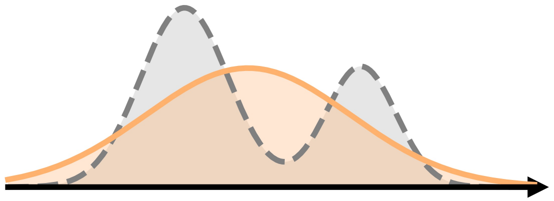
$$\mathbb{D}_{\text{KL}}(p\|\hat{p}) = \mathbb{E}_{x\sim p}\left[\log\frac{p(x)}{\hat{p}(x)}\right]$$

Reverse KL

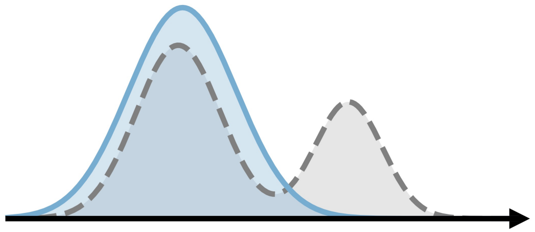
$$\mathbb{D}_{\text{KL}}(\hat{p}\|p) = \mathbb{E}_{x\sim \hat{p}}\left[\log\frac{\hat{p}(x)}{p(x)}\right]$$



Target distribution $p(x)$



Mean-seeking solution



Mode-seeking solution

Korbak, Tomasz, et al. "RL with KL penalties is better viewed as Bayesian inference." *arXiv preprint arXiv:2205.11275* (2022)

Reverse KL for LM Alignment

- *Policy optimization as probability matching under Reverse KL*[**Ji et al., 2023**] (**ICML' 24**):
  - *Without loss of generality, consider the generalized alignment objective:*

$$\mathcal{J}_{\text{lhf}}^{\beta_r}(\pi_\theta^{\beta_\pi}) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}^{\text{pref}}}\left(\mathbb{E}_{\pi_\theta^{\beta_\pi}(\boldsymbol{y}|\boldsymbol{x})}[r_\phi(\boldsymbol{x},\boldsymbol{y})] - \beta_r \mathbb{D}_{\text{KL}}[\pi_\theta^{\beta_\pi}(\boldsymbol{y}|\boldsymbol{x})\|\pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})]\right)$$

  - $\pi_\theta^{\beta_\pi}$ *is the geometric mean of* π_θ *and* π_{sft}

$$\pi_\theta^{\beta_\pi}(\boldsymbol{y}|\boldsymbol{x}) \propto \pi_\theta(\boldsymbol{y}|\boldsymbol{x})^{\beta_\pi}\pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})^{1-\beta_\pi}$$

  - *Decompose the KL regularization*

$$\beta = \beta_r \cdot \beta_\pi$$

regularize reward (β_r) *regularize policy* (β_π)

  - *Analytic solution is also* π_β^*.
  - *Unify the regularization setting of PPO* ($\beta_\pi = 1, \beta_r = \beta$) *and DPO* ($\beta_\pi = \beta, \beta_r = 1$)

Ji, Haozhe, et al. "Towards Efficient Exact Optimization of Language Model Alignment." *ICML* (2024)

Reverse KL for LM Alignment

- *Deriving the probability matching objective of* $\mathcal{J}_{\text{lhf}}^{\beta_r}(\pi_\theta^{\beta_\pi})$

$$\mathbb{D}_{\text{KL}}(\pi_\theta^{\beta_\pi} \| \pi_{\beta_r}^*) = \mathbb{E}_{\pi_\theta^{\beta_\pi}(\boldsymbol{y}|\boldsymbol{x})}\left[\log \frac{\pi_\theta^{\beta_\pi}(\boldsymbol{y}|\boldsymbol{x})}{\pi_{\beta_r}^*(\boldsymbol{y}|\boldsymbol{x})}\right]$$

Importance Sampling (IS)
π_{sft} *as the proposal distribution*

$$\mathbb{D}_{\text{KL}}(\pi_\theta^{\beta_\pi} \| \pi_{\beta_r}^*) = \mathbb{E}_{\pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})}\left[\frac{\pi_\theta^{\beta_\pi}(\boldsymbol{y}|\boldsymbol{x})}{\pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})} \log \frac{\pi_\theta^{\beta_\pi}(\boldsymbol{y}|\boldsymbol{x})}{\pi_{\beta_r}^*(\boldsymbol{y}|\boldsymbol{x})}\right]$$

Define $f_\theta(\boldsymbol{x}, \boldsymbol{y}) = \log \pi_\theta^{\beta_\pi}(\boldsymbol{y}|\boldsymbol{x}) - \log \pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})$
as the log policy ratio

$$\mathbb{D}_{\text{KL}}(\pi_\theta^{\beta_\pi} \| \pi_{\beta_r}^*) = \mathbb{E}_{\pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})}\left[e^{f_\theta(\boldsymbol{x},\boldsymbol{y})} \log \frac{e^{f_\theta(\boldsymbol{x},\boldsymbol{y})}}{\frac{1}{Z_{\beta_r}(\boldsymbol{x})} e^{\frac{r_\phi(\boldsymbol{x},\boldsymbol{y})}{\beta_r}}}\right]$$

Ji, Haozhe, et al. "Towards Efficient Exact Optimization of Language Model Alignment." *ICML* (2024)

Reverse KL for LM Alignment

- *Deriving the probability matching objective of* $\mathcal{J}_{\text{lhf}}^{\beta_r}(\pi_\theta^{\beta_\pi})$

$$\mathbb{D}_{\text{KL}}(\pi_\theta^{\beta_\pi} \| \pi_{\beta_r}^*) = \mathbb{E}_{\pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})}\left[e^{f_\theta(\boldsymbol{x},\boldsymbol{y})} \log \frac{e^{f_\theta(\boldsymbol{x},\boldsymbol{y})}}{\frac{1}{Z_{\beta_r}(\boldsymbol{x})} e^{\frac{r_\phi(\boldsymbol{x},\boldsymbol{y})}{\beta_r}}}\right]$$

  - *The partition function* $Z_{\beta_r}(\boldsymbol{x})$ *is intractable.*
  - *Inspiration from Self-Normalized Importance Sampling (SNIS)*
  - *Sample K i.i.d. continuations* $\boldsymbol{y}_{1:K} = \{\boldsymbol{y}_1, \cdots, \boldsymbol{y}_K\}$ *from* $\pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})$

$$Z_{\beta_r}(\boldsymbol{x}) = \mathbb{E}_{\pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})}[\exp(\frac{r_\phi(\boldsymbol{x},\boldsymbol{y})}{\beta_r})]$$

$$\mathbb{D}_{\text{KL}}(\pi_\theta^{\beta_\pi} \| \pi_{\beta_r}^*) = \lim_{K\to\infty} \sum_{k=1}^{K} \frac{e^{f_\theta(\boldsymbol{x},\boldsymbol{y}_k)}}{\sum_{j=1}^{K} e^{f_\theta(\boldsymbol{x},\boldsymbol{y}_j)}} \log \frac{\frac{e^{f_\theta(\boldsymbol{x},\boldsymbol{y}_k)}}{\sum_{j=1}^{K} e^{f_\theta(\boldsymbol{x},\boldsymbol{y}_j)}}}{\frac{e^{\frac{1}{\beta_r} r_\phi(\boldsymbol{x},\boldsymbol{y}_k)}}{\sum_{j=1}^{K} \frac{1}{\beta_r} e^{r_\phi(\boldsymbol{x},\boldsymbol{y}_j)}}}$$

$p_{f_\theta}(i|\boldsymbol{y}_{1:K}, \boldsymbol{x})$

Distribution of log policy ratio

$p_{r_\phi}(i|\boldsymbol{y}_{1:K}, \boldsymbol{x})$

Distribution of reward model

Ji, Haozhe, et al. "Towards Efficient Exact Optimization of Language Model Alignment." *ICML* (2024)

Reverse KL for LM Alignment

- *Deriving the probability matching objective of* $\mathcal{J}_{\text{lhf}}^{\beta_r}(\pi_\theta^{\beta_\pi})$

$$\mathbb{D}_{\text{KL}}(\pi_\theta^{\beta_\pi} \| \pi_{\beta_r}^*) = \mathbb{E}_{\pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})}\left[e^{f_\theta(\boldsymbol{x},\boldsymbol{y})} \log \frac{e^{f_\theta(\boldsymbol{x},\boldsymbol{y})}}{\frac{1}{Z_{\beta_r}(\boldsymbol{x})} e^{\frac{r_\phi(\boldsymbol{x},\boldsymbol{y})}{\beta_r}}} \right]$$

- *The partition function* $Z_{\beta_r}(\boldsymbol{x})$ *is intractable.*

$$Z_{\beta_r}(\boldsymbol{x}) = \mathbb{E}_{\pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})}[\exp(\frac{r_\phi(\boldsymbol{x},\boldsymbol{y})}{\beta_r})]$$

- *Inspiration from Self-Normalized Importance Sampling (SNIS)*
- *Sample K i.i.d. continuations* $\boldsymbol{y}_{1:K} = \{\boldsymbol{y}_1, \cdots, \boldsymbol{y}_K\}$ *from* $\pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})$

$$\mathbb{D}_{\text{KL}}(\pi_\theta^{\beta_\pi} \| \pi_{\beta_r}^*) = \lim_{K\to\infty} \sum_{k=1}^{K} \frac{e^{f_\theta(\boldsymbol{x},\boldsymbol{y}_k)}}{\sum_{j=1}^{K} e^{f_\theta(\boldsymbol{x},\boldsymbol{y}_j)}} \log \frac{\frac{e^{f_\theta(\boldsymbol{x},\boldsymbol{y}_k)}}{\sum_{j=1}^{K} e^{f_\theta(\boldsymbol{x},\boldsymbol{y}_j)}}}{\frac{e^{\frac{1}{\beta_r} r_\phi(\boldsymbol{x},\boldsymbol{y}_k)}}{\sum_{j=1}^{K} \frac{1}{\beta_r} e^{r_\phi(\boldsymbol{x},\boldsymbol{y}_j)}}}$$

Reverse KL $\mathbb{D}_{\text{KL}}(p_{f_\theta} || p_{r_\phi})$ *of* p_{f_θ} *and* p_{r_ϕ}

Ji, Haozhe, et al. "Towards Efficient Exact Optimization of Language Model Alignment." *ICML* (2024)

Reverse KL for LM Alignment

- *Efficient Exact Optimization (**EXO**) of the alignment objective*
  - ***Learning from the reward model***

$$\mathcal{L}_{\text{exo}}(\pi_\theta) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}^{\text{pref}}}\mathbb{E}_{\pi_{\text{sft}}(\boldsymbol{y}_{1:K}|\boldsymbol{x})}\left[\mathbb{D}_{\text{KL}}\left(p_{f_\theta}(\cdot|\boldsymbol{y}_{1:K},\boldsymbol{x})\|p_{r_\phi}(\cdot|\boldsymbol{y}_{1:K},\boldsymbol{x})\right)\right]$$

- *Where we define:* regularize policy; regularize reward

$$p_{f_\theta}(i|\boldsymbol{y}_{1:K},\boldsymbol{x}) = \frac{e^{\beta_\pi \log\frac{\pi_\theta(\boldsymbol{y}_i|\boldsymbol{x})}{\pi_{\text{sft}}(\boldsymbol{y}_i|\boldsymbol{x})}}}{\sum_{j=1}^{K} e^{\beta_\pi \log\frac{\pi_\theta(\boldsymbol{y}_j|\boldsymbol{x})}{\pi_{\text{sft}}(\boldsymbol{y}_j|\boldsymbol{x})}}} \qquad p_{r_\phi}(i|\boldsymbol{y}_{1:K},\boldsymbol{x}) = \frac{e^{\frac{1}{\beta_r}r_\phi(\boldsymbol{x},\boldsymbol{y}_i)}}{\sum_{j=1}^{K} e^{\frac{1}{\beta_r}r_\phi(\boldsymbol{x},\boldsymbol{y}_j)}}$$

  - ***Learning from the preference data** (K=2)*

$$\mathcal{L}_{\text{exo-pref}}(\pi_\theta) = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y}_w,\boldsymbol{y}_l)\sim\mathcal{D}^{\text{pref}}}\left[\mathbb{D}_{\text{KL}}\left(p_{f_\theta}(\cdot|\boldsymbol{y}_w,\boldsymbol{y}_l,\boldsymbol{x})\|p_{r_h}(\cdot|\boldsymbol{y}_w,\boldsymbol{y}_l,\boldsymbol{x})\right)\right]$$

- *Where the preference probability $p_{r_h}(\cdot\,|\boldsymbol{y}_w,\boldsymbol{y}_l,\boldsymbol{x})$ is a label-smoothed one-hot distribution.*

Ji, Haozhe, et al. "Towards Efficient Exact Optimization of Language Model Alignment." *ICML* (2024)

Reverse KL for LM Alignment

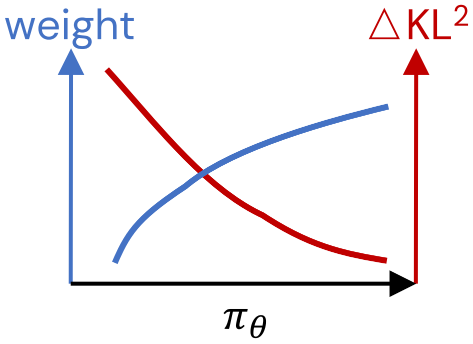
- *Analysis*
  - *Unbiased gradient ($K \to \infty$):*

$$\nabla_\theta \mathcal{L}_{\text{exo}}(\pi_\theta) = \nabla_\theta \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}^{\text{pref}}}\left[\mathbb{D}_{\text{KL}}(\pi_\theta^{\beta_\pi}(\boldsymbol{y}|\boldsymbol{x}) \| \pi^*_{\beta_r}(\boldsymbol{y}|\boldsymbol{x}))\right]$$
$$= -\frac{1}{\beta_r}\nabla_\theta \mathcal{J}_{\text{lhf}}^{\beta_r}(\pi_\theta^{\beta_\pi}).$$

    - *In practice, a finite **K** slightly introduces bias while reduces variance.*
  - *Asymptotic variance comparison:*

$$\text{Var}[\hat{\text{KL}}_{\text{exo}}] = \mathbb{E}_{\boldsymbol{y}\sim\pi_\theta}\left[\frac{w(\boldsymbol{x},\boldsymbol{y})}{\mathbb{E}_{\boldsymbol{y}'\sim\pi_\theta}[w(\boldsymbol{x},\boldsymbol{y}')]}\left(\log\frac{\pi_\theta(\boldsymbol{y}|\boldsymbol{x})}{\pi^*_\beta(\boldsymbol{y}|\boldsymbol{x})} - \text{KL}\right)^2\right] \quad w(\boldsymbol{x},\boldsymbol{y}) = \frac{\pi_\theta(\boldsymbol{y}|\boldsymbol{x})}{\pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})}$$

$$\text{Var}[\hat{\text{KL}}_{\text{ppo}}] = \mathbb{E}_{\boldsymbol{y}\sim\pi_\theta}\left[\left(\log\frac{\pi_\theta(\boldsymbol{y}_i|\boldsymbol{x})}{\pi^*_\beta(\boldsymbol{y}_i|\boldsymbol{x})} - \text{KL}\right)^2\right]$$



approx. negative correlation

Comparison with DPO

- *Generalizing DPO:*
  - *Sample K completions* $\boldsymbol{y}_{1:K} = \{\boldsymbol{y}_1, \cdots, \boldsymbol{y}_K\}$ *from* $\pi_{\text{sft}}(y|x)$
  - *Generalize hard label to soft label*

$$\mathcal{L}_{\text{dpo-rw}}(\pi_\theta) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}^{\text{pref}}}\mathbb{E}_{\pi_{\text{sft}}(\boldsymbol{y}_{1:K}|\boldsymbol{x})}\left[-\sum_{i=1}^{K}\frac{e^{\frac{1}{\beta_r}r_\phi(\boldsymbol{x},\boldsymbol{y}_i)}}{\sum_{j=1}^{K}e^{\frac{1}{\beta_r}r_\phi(\boldsymbol{x},\boldsymbol{y}_j)}}\log\frac{e^{\beta_\pi\log\frac{\pi_\theta(\boldsymbol{y}_i|\boldsymbol{x})}{\pi_{\text{sft}}(\boldsymbol{y}_i|\boldsymbol{x})}}}{\sum_{j=1}^{K}e^{\beta_\pi\log\frac{\pi_\theta(\boldsymbol{y}_j|\boldsymbol{x})}{\pi_{\text{sft}}(\boldsymbol{y}_j|\boldsymbol{x})}}}\right]$$

Forward KL $\mathbb{D}_{\text{KL}}(p_{f_\theta}||p_{r_\phi})$ *of* p_{f_θ} *and* p_{r_ϕ} *(up to a constant)*

  - *The gradient of DPO-rw aligns with the gradient of the forward KL asymptotically for policy with **arbitrary*** θ *when* $K \to \infty$.

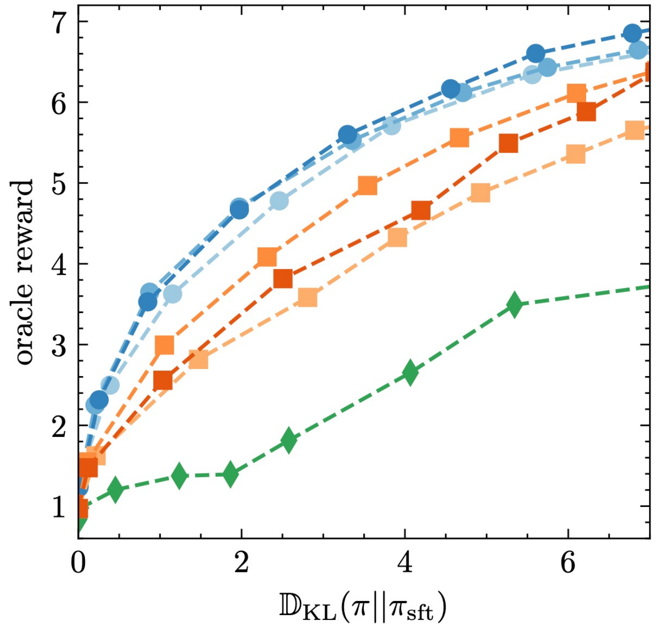
$$\nabla_\theta\mathcal{L}_{\text{dpo-rw}}(\pi_\theta) = \nabla_\theta\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}^{\text{pref}}}\left[\mathbb{D}_{\text{KL}}(\pi^*_{\beta_r}(\boldsymbol{y}|\boldsymbol{x})\|\pi_\theta^{\beta_\pi}(\boldsymbol{y}|\boldsymbol{x}))\right]$$

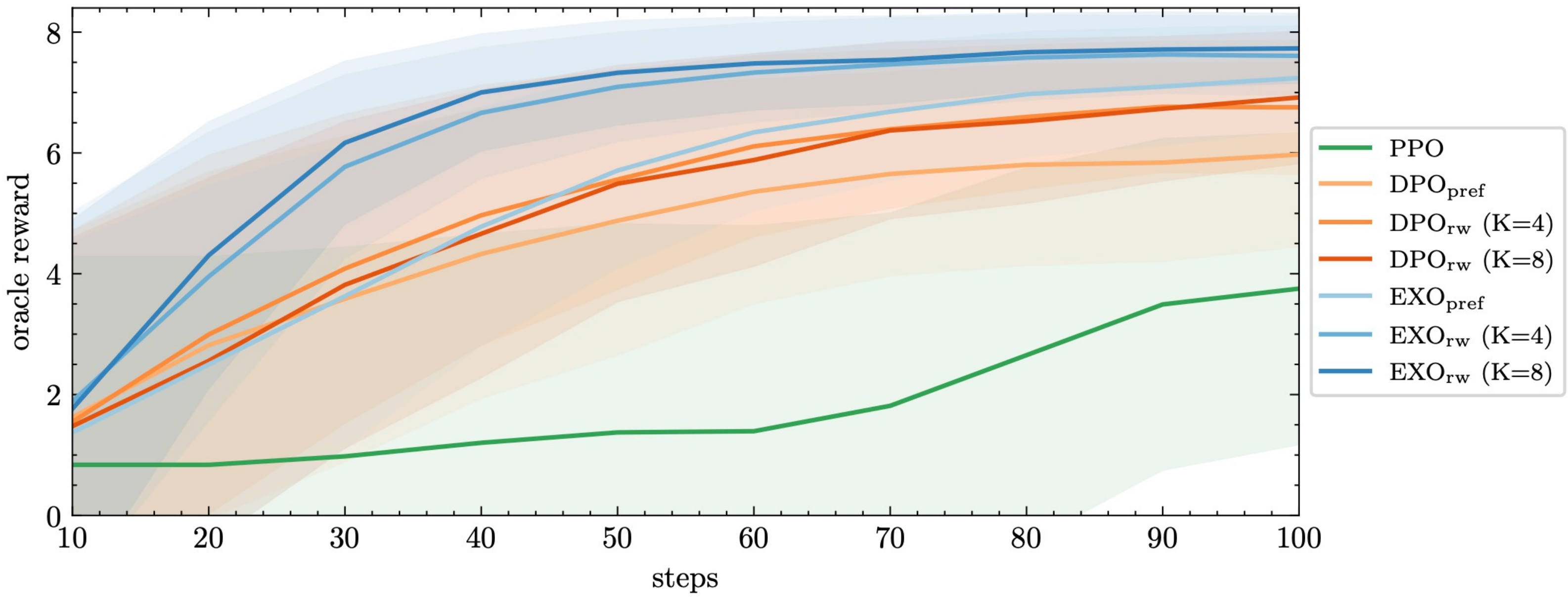
- ***Inexactness****: DPO minimizes the forward KL, while RLHF, e.g., PPO minimizes the reverse KL.*

Ji, Haozhe, et al. "Towards Efficient Exact Optimization of Language Model Alignment." *ICML* (2024)

Experiments

- *Synthetic experiment: Generate IMDB review with positive sentiment*
  - *Oracle reward (Human labeler): Classifier trained on IMDB review classification dataset*





Oracle reward *vs* ***KL*** ***Oracle reward*** *vs* ***Training steps***

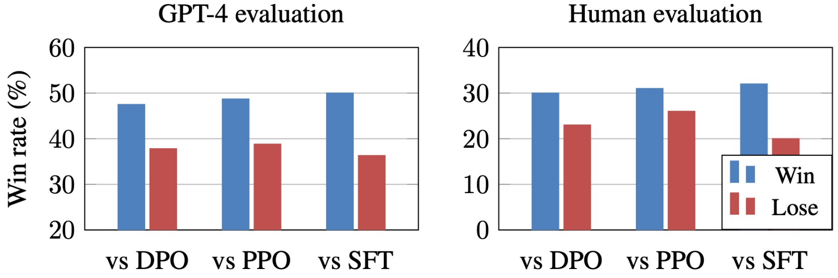
Ji, Haozhe, et al. "Towards Efficient Exact Optimization of Language Model Alignment." *ICML* (2024)

Experiments

- *Alignment on real human preferences:*
  - *Text summarization: TL;DR preference dataset*
  - *Dialogue generation: Anthropic-HH dataset (helpfulness subset)*
  - *Instruction following: Filtered real user query from an online API*

Method	Reward Model (%) vs SFT	Reward Model (%) vs Chosen	GPT-4 (%) vs SFT	GPT-4 (%) vs Chosen
		`w/ Preferences`		
DPO_{pref}	68.3	23.7	57.0	30.5
EXO_{pref}	**92.5**	**60.1**	**83.0**	**55.0**
		`w/ Reward Model`		
Best-of-N	99.3	75.8	83.5	60.0
PPO	93.2	58.3	77.0	52.0
DPO_{rw}	82.7	39.8	70.0	41.0
EXO_{rw}	**97.3**	**76.4**	**88.5**	**64.0**

Method	Reward Model (%) vs SFT	Reward Model (%) vs Chosen	GPT-4 (%) vs SFT	GPT-4 (%) vs Chosen
		`w/ Preferences`		
DPO_{pref}	66.3	65.1	58.0	37.0
EXO_{pref}	**76.4**	**76.7**	**73.0**	**51.0**
		`w/ Reward Model`		
Best-of-N	94.6	98.2	86.0	63.0
PPO	75.0	74.0	66.5	52.0
DPO_{rw}	79.9	81.3	75.5	49.0
EXO_{rw}	**85.6**	**87.2**	**83.5**	**60.0**



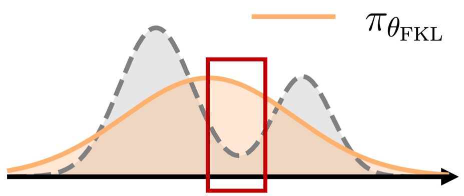
- *Outperforms DPO and PPO in both settings of learning from preferences & reward model.*
- *On par with Best-of-N (N=128) but much more computationally efficient in inference.*
- *Scaling to realistic instruction-following dataset with consistent improvement.*

Experiments

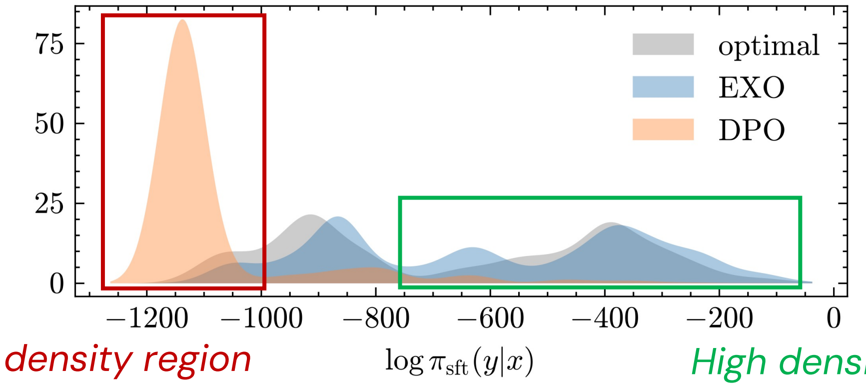
- *Visualization: Compare the density of DPO and EXO with the optimal policy*
  - *Given a test prompt "**This Fox spectacle was a big hit when released in** "*
  - *Estimate the empirical policy distribution of π_θ and π^*_β by SNIS:*

$$\hat{\pi}_\theta(\boldsymbol{y}_i|\boldsymbol{x}) = \frac{M\pi_\theta(\boldsymbol{y}_i|\boldsymbol{x})}{\sum_{j=1}^M \pi_\theta(\boldsymbol{y}_j|\boldsymbol{x})/\pi_{\text{sft}}(\boldsymbol{y}_j|\boldsymbol{x})} \qquad \hat{\pi}^*_\beta(\boldsymbol{y}_i|\boldsymbol{x}) = \frac{M\pi_{\text{sft}}(\boldsymbol{y}_i|\boldsymbol{x})\exp(r(\boldsymbol{x},\boldsymbol{y}_i)/\beta)}{\sum_{j=1}^M \exp(r(\boldsymbol{x},\boldsymbol{y}_j)/\beta)}$$

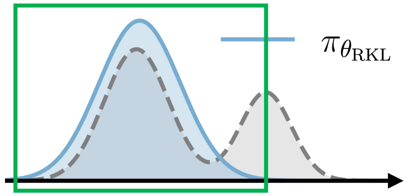
  - *Use Kernel Density Estimation to estimate the density and plot the ratio* $\rho_{\hat{\pi}}(\boldsymbol{y}|\boldsymbol{x}) = \frac{\hat{\pi}(\boldsymbol{y}|\boldsymbol{x})}{\pi_{\text{sft}}(\boldsymbol{y}|\boldsymbol{x})}$



Overestimate the long tail



⇠ Low density region *High density region ⇢*

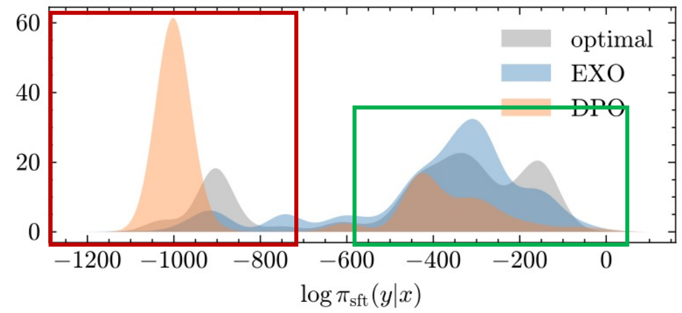


Concentrating on the mode

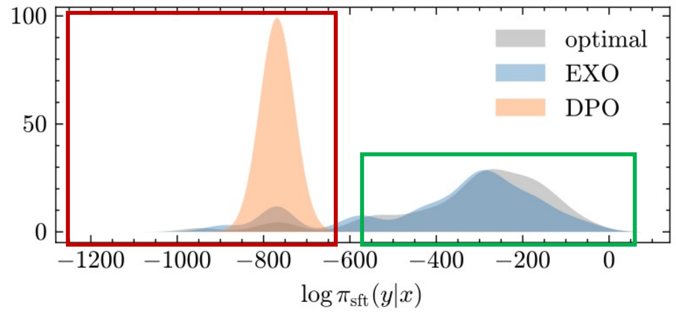
Ji, Haozhe, et al. "Towards Efficient Exact Optimization of Language Model Alignment." *ICML* (2024)

Experiments

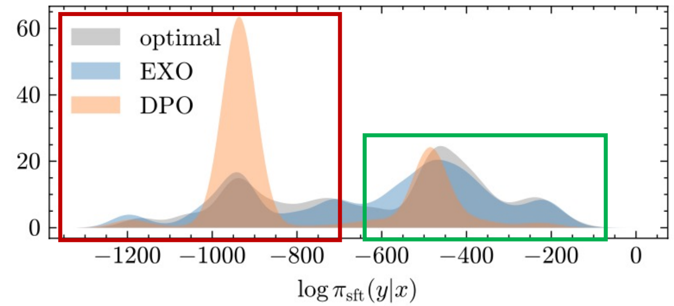
- *More visualization cases: (prevailing phenomenon, no cherry-picking)*



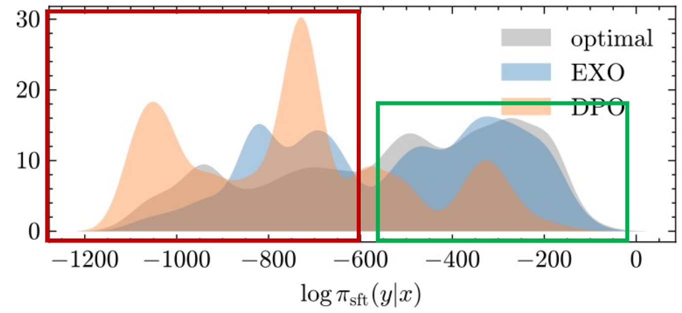
Estimated density ratio of the EXO, DPO and optimal policy given the prompt "*Is this supposed to be serious? I hope not*".



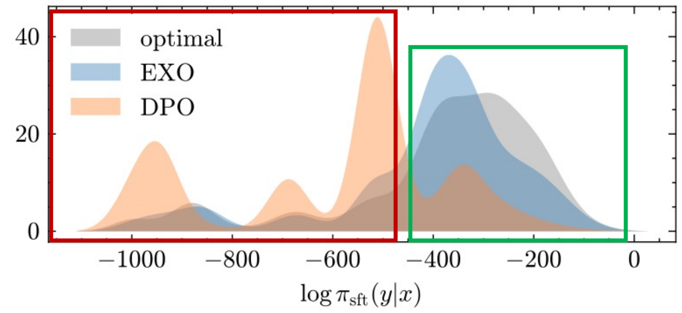
Estimated density ratio of the EXO, DPO and optimal policy given the prompt "*Great book, great movie, great soundtrack. Frank*".



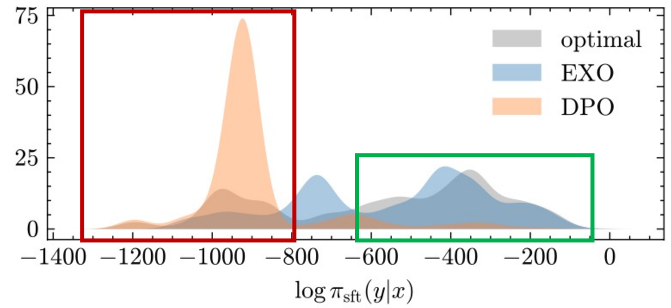
Estimated density ratio of the EXO, DPO and optimal policy given the prompt "*What we have here the standard Disney direct to DVD*".



Estimated density ratio of the EXO, DPO and optimal policy given the prompt "*This is indeed the film that popularized kung*".



Estimated density ratio of the EXO, DPO and optimal policy given the prompt "*This movie is about a group of people who are*".



Estimated density ratio of the EXO, DPO and optimal policy given the prompt "*Once the slow beginning gets underway, the film kicks*".

Ji, Haozhe, et al. "Towards Efficient Exact Optimization of Language Model Alignment." *ICML* (2024)

Beyond the theoretical limits of language modeling

- **Beyond MLE**: Quality-aware objective
  - Reverse KL [**ICML' 24**]: quality assessed by reward that captures human preference
  - **Total variation distance [ICLR' 23]**: quality assessed by the "optimal classifier" in theory
- **Beyond AR**: Expressive model family
  - Energy-based model [**ICLR' 24**]: Augment AR model with a residual energy model
  - Latent-variable model [**EMNLP' 21**]: Condition AR model with a latent plan
  - Look-up model [**EMNLP' 20**]: Extend AR model with a parallel database look-up

Beyond MLE for AR LM

- Total variation distance (TVD): quality assessed by "**optimal classifier**"
  - TVD reflects the "accuracy" of an optimal classifier that try to discriminate true data and model generated data

$$c \sim p(c) = \text{Bernoulli}(\frac{1}{2}) \quad \text{Prior label distribution}$$

$$\boldsymbol{y} \sim p(\boldsymbol{y}|\boldsymbol{x}, c) = \begin{cases} p_d(\boldsymbol{y}|\boldsymbol{x}) & \text{if } c = 1 \quad \text{True data} \\ p_\theta(\boldsymbol{y}|\boldsymbol{x}) & \text{if } c = 0 \quad \text{Model generated data} \end{cases}$$

$$\|p_d - p_\theta\|_{\text{TV}} = 1 - 2 \inf_f \underbrace{\mathbb{P}\Big(f(\boldsymbol{x}, \boldsymbol{y}) \neq c\Big)}_{\text{error rate}} \quad \text{TVD defined by optimal error rate}$$

  - **Intuition**: The closer p_θ and p_d is, the harder for the optimal classifier to discriminate. (The upper-bound of error rate is 50%, i.e., by chance)

Hashimoto, Tatsunori., et al. "Unifying Human and Statistical Evaluation for Natural Language Generation." *ACL* (2019).

TVD for LM Fine-Tuning

- Learning objective for LM based on TVD [**Ji et al., 2023**] (**ICLR'23 Oral**):
  - Measuring the distance in discrete sequence space:

$$\|p_d - p_\theta\|_{\mathrm{TV}} = \frac{1}{2}\sum_{\boldsymbol{y}\in\mathcal{Y}}\Big|p_d(\boldsymbol{y}|\boldsymbol{x}) - p_\theta(\boldsymbol{y}|\boldsymbol{x})\Big| \qquad \text{L1-distance}$$

$$= 1 - \sum_{\boldsymbol{y}\in\mathcal{Y}}\min\Big(p_d(\boldsymbol{y}|\boldsymbol{x}), p_\theta(\boldsymbol{y}|\boldsymbol{x})\Big)$$

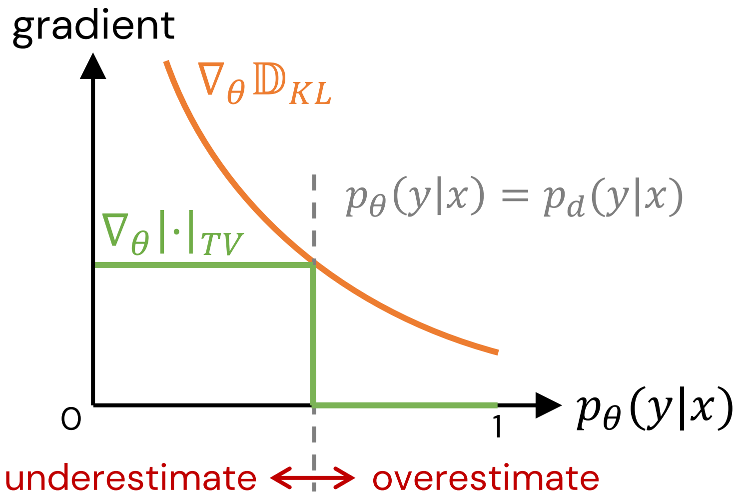
  - Gradient analysis: $y \sim p_d$
    - Gradient of FKL

$$\nabla_\theta \mathbb{D}_{\mathrm{KL}}(p_d\|p_\theta) \approx -\frac{\nabla_\theta p_\theta(\boldsymbol{y}|\boldsymbol{x})}{p_\theta(\boldsymbol{y}|\boldsymbol{x})}$$

Assign **non-zero** p_θ to every data point

    - Gradient of TVD

$$\nabla_\theta\|p_d - p_\theta\|_{\mathrm{TV}} \approx \begin{cases} -\dfrac{\nabla_\theta p_\theta(\boldsymbol{y}|\boldsymbol{x})}{p_d(\boldsymbol{y}|\boldsymbol{x})}, & p_\theta(\boldsymbol{y}|\boldsymbol{x}) < p_d(\boldsymbol{y}|\boldsymbol{x}) \\ 0, & p_\theta(\boldsymbol{y}|\boldsymbol{x}) \geq p_d(\boldsymbol{y}|\boldsymbol{x}) \end{cases}$$



Ji, Haozhe, et al. "Tailoring Language Generation Models under Total Variation Distance." *ICLR* (2023).

TVD for LM Fine-Tuning

- Learning objective for LM based on TVD [**Ji et al., 2023**] (**ICLR'23 Oral**):
  - Measuring the distance in discrete sequence space:

$$\|p_d - p_\theta\|_{\mathrm{TV}} = \frac{1}{2}\sum_{\boldsymbol{y}\in\mathcal{Y}} \left| p_d(\boldsymbol{y}|\boldsymbol{x}) - p_\theta(\boldsymbol{y}|\boldsymbol{x}) \right| \qquad \text{L1-distance}$$

$$= 1 - \sum_{\boldsymbol{y}\in\mathcal{Y}} \min\Big(p_d(\boldsymbol{y}|\boldsymbol{x}), p_\theta(\boldsymbol{y}|\boldsymbol{x})\Big)$$

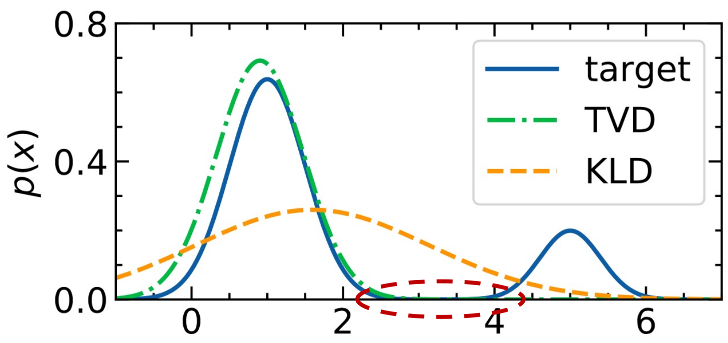
  - Gradient analysis: $y \sim p_d$
    - Gradient of FKL

Learning objective for LM based on TVD [Ji et al., 2023] (ICL
Measuring the distance in discrete sequence space:
Gradient analysis: $y \sim p_d$
Gradient of FKL
Gradient of TVD

Assign **non-zero** p_θ to every data point

    - Gradient of TVD

$$\nabla_\theta \|p_d - p_\theta\|_{\mathrm{TV}} \approx \begin{cases} -\dfrac{\nabla_\theta p_\theta(\boldsymbol{y}|\boldsymbol{x})}{p_d(\boldsymbol{y}|\boldsymbol{x})}, & p_\theta(\boldsymbol{y}|\boldsymbol{x}) < p_d(\boldsymbol{y}|\boldsymbol{x}) \\ 0, & p_\theta(\boldsymbol{y}|\boldsymbol{x}) \geq p_d(\boldsymbol{y}|\boldsymbol{x}) \end{cases}$$



overestimate "data void"

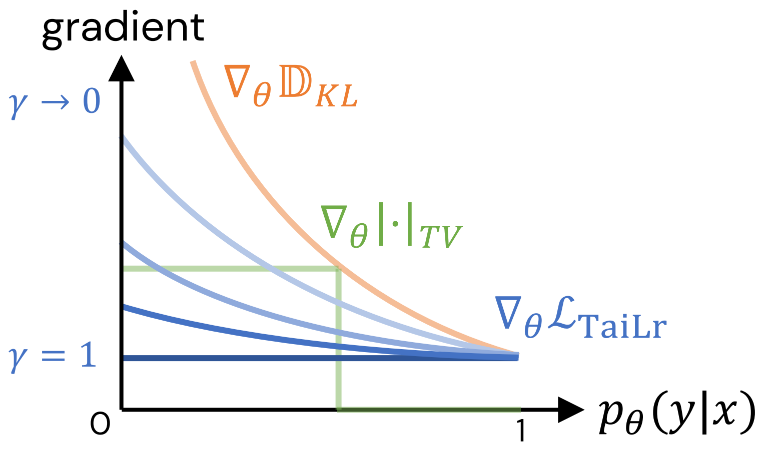
Ji, Haozhe, et al. "Tailoring Language Generation Models under Total Variation Distance." *ICLR* (2023).

TVD for LM Fine-Tuning

- Learning objective for LM based on TVD [**Ji et al., 2023**] (**ICLR'23 Oral**):
  - TaiLr objective

$$\mathcal{L}_{\text{TaiLr}}(w;\theta) = -\underbrace{\left(\frac{p_\theta^{<t}(w)}{\gamma + (1-\gamma)p_\theta^{<t}(w)}\right)}_{\text{stop gradient}} \log p_\theta^{<t}(w)$$

  - γ trade-offs bias and variance: $\gamma = 1$ (unbiased TVD) $\gamma \to 0$ (bias to KLD)



Ji, Haozhe, et al. "Tailoring Language Generation Models under Total Variation Distance." *ICLR* (2023).

Experiments

- **Experiments**: Various text generation tasks

Other MLE variants: Unlikelihood, D2GPo, Loss truncation, GOLD
TVD-based: TaiLr

Method	Dev BLEU	Test BLEU
MLE	35.81‡	34.27‡
Unlikelihood	33.92‡	32.82‡
D2GPo	36.09‡	34.50‡
Loss truncation	35.63†	34.48‡
GOLD	35.74‡	34.68†
TaiLr	**36.44**	**35.05**

One-way Training	Test BLEU
BiBERT (Table 2, Xu et al. 2021)	37.58
BiBERT (Our implementation)	38.01
BiBERT + TaiLr	**39.12**
Dual-directional Training + Fine-Tuning	**Test BLEU**
BiBERT (Table 3, Xu et al. 2021)	38.61
BiBERT (Our implementation)	38.73
BiBERT + TaiLr	**39.23**

Machine translation: Improve over the **2022 SOTA (BiBERT)** on **IWSLT14**

Method	B-1↑	D-4↑	rep-8↓	Mauve↑
MLE	27.85	84.28	10.31†	56.42‡
Unlikelihood	27.88	85.46	10.06	59.35‡
D2GPo	22.73‡	84.10	10.04	53.35‡
Loss truncation	19.49‡	76.51‡	13.41‡	45.35‡
GOLD	25.25‡	46.98‡	28.23‡	15.44‡
TaiLr	**28.62**	**85.56**	**9.73**	**64.64**

Long text generation

Method	R-1	R-2	R-L
MLE	38.24‡	19.12	35.70†
Unlikelihood	37.80‡	18.34‡	34.84‡
D2GPo	38.52†	18.92†	35.64‡
Loss truncation	38.62	19.29	35.85†
GOLD	38.57†	19.27	35.79†
TaiLr	**38.82**	**19.50**	**36.24**

Text summarization

Ji, Haozhe, et al. "Tailoring Language Generation Models under Total Variation Distance." *ICLR* (2023).

Beyond MLE for AR LM

- **Takeaway & Future**:
- The desired learning goal should capture quality, which might not always has a tractable form.
- Effectiveness and efficiency of learning: Bias-variance tradeoff
  - Variance: Sparsity and complexity of data
  - Bias: Inductive bias of estimation method
- **Principle**: Reduce variance with controlled bias

Beyond the theoretical limits of language modeling

- **Beyond MLE**: Quality-aware objective
  - Reverse KL [**ICML' 24**]: quality assessed by reward that captures human preference
  - Total variation distance [**ICLR' 23**]: quality assessed by the "optimal classifier" in theory

- **Beyond AR: Expressive model family**
  - Energy-based model [**ICLR' 24**]: Augment AR model with a residual energy model
  - Latent-variable model [**EMNLP' 21**]: Condition AR model with a latent plan
  - Look-up model [**EMNLP' 20**]: Extend AR model with a parallel database look-up

Beyond Auto-Regressive Model

- Parametric sequence model families [Lin et al., 2020]

Model Family	Compact parameters	Efficient scoring	Efficient sampling	Support of distribution
Auto-Regressive Model (ARM)	✓	✓	✓	Some but **not all** $S \in P$
Energy-Based Model (EBM)	✓	✓	✗	All $S \in P$
Latent-Variable Model (LVM)	✓	✗	✓	All $S \in NP$
Look-Up Model (LUM)	✗	✓	✓	All S

Practical desiderata (Compact parameters, Efficient scoring, Efficient sampling); Expressivity (Support of distribution)

- **Compact parameters**: Parameter complexity grow in $O(poly(n))$
- **Efficient scoring**: Score a sequence in time of $O(poly(n))$
- **Efficient sampling**: Sample a sequence in time of $O(poly(n))$

*n: sequence length

Lin, Chu-Cheng, et al. "Limitations of Autoregressive Models and Their Alternatives." NAACL (2020).

Beyond the theoretical limits of language modeling

- **Beyond MLE**: Quality-aware objective
  - Reverse KL [**ICML' 24**]: quality assessed by reward that captures human preference
  - Total variation distance [**ICLR' 23**]: quality assessed by the "optimal classifier" in theory

- **Beyond AR**: Expressive model family
  - **Energy-based model [ICLR' 24]**: Augment AR model with a residual energy model
  - Latent-variable model [**EMNLP' 21**]: Condition AR model with a latent plan
  - Look-up model [**EMNLP' 20**]: Extend AR model with a parallel database look-up

Energy-Based Model

- **Definition**: Assign low energy to sequence with high probability

$$p(\boldsymbol{y}|\boldsymbol{x}) = \frac{e^{-E_\theta(\boldsymbol{x},\boldsymbol{y})}}{\sum_{\boldsymbol{y}'} e^{-E_\theta(\boldsymbol{x},\boldsymbol{y}')}} = \frac{e^{-E_\theta(\boldsymbol{x},\boldsymbol{y})}}{Z(\boldsymbol{x})}$$

  - Energy function: $E_\theta(\boldsymbol{x}, \boldsymbol{y})$ scores the complete sequence $\boldsymbol{y}$
  - Partition function: $Z(\boldsymbol{x})$ is the normalizing constant which is intractable

- **Advantage**: Conditional probability implicitly marginalizing out the future

$$p(y_t|\boldsymbol{y}_{<t}, \boldsymbol{x}) = \frac{\sum_{\boldsymbol{y}'_{>t}} e^{-E_\theta(\boldsymbol{x},\boldsymbol{y}_{<t},y_t,\boldsymbol{y}'_{>t})}}{\sum_{\boldsymbol{y}'_{\geq t}} e^{-E_\theta(\boldsymbol{x},\boldsymbol{y}_{<t},\boldsymbol{y}'_{\geq t})}} = \frac{Z(\boldsymbol{x}, \boldsymbol{y}_{<t}, y_t)}{Z(\boldsymbol{x}, \boldsymbol{y}_{<t})}$$

  - **Intuition**: EBM shows that **exactly computing** the conditional probability requires considering **all possibilities** in the future. Local normalization is insufficient (AR model)

Lin, Chu-Cheng, et al. "Limitations of Autoregressive Models and Their Alternatives." NAACL (2020).

Energy-Based Model

- **Disadvantage**: MLE, sampling for EBM is expensive due to intractable $Z(\boldsymbol{x})$
- **Noise-Contrastive Estimation (NCE)**: Sampling-free method
  - **Intuition**: Reducing energy **only** on correct data points does not guarantee increasing their probability. Need to "push them down wrong points".
  - Ranking objective:

$$\min_{\theta} \mathbb{E}_{\boldsymbol{y}_+ \sim p_d, \boldsymbol{y}_-^{(1:K)} \sim p_N} \left[-\log \frac{e^{s_\theta(\boldsymbol{x}, \boldsymbol{y}_+)}}{e^{s_\theta(\boldsymbol{x}, \boldsymbol{y}_+)} + \sum_{k=1}^{K} e^{s_\theta(\boldsymbol{x}, \boldsymbol{y}_-^{(k)})}} \right]$$

  - Score function:

$$s_\theta(\boldsymbol{x}, \boldsymbol{y}) = -E_\theta(\boldsymbol{x}, \boldsymbol{y}) - \log p_N(\boldsymbol{y}|\boldsymbol{x})$$

  - It is critical to choose an **appropriate noise distribution** which is useful for fine-grained characterization of the energy landscape.

Gutmann, Michael., et al. "Noise-Contrastive Estimation of Unnormalized Statistical Models with Applications to Natural Image Statistics". *JMLR* (2013)

Energy-Based Model

- **Residual EBM**: Leverage the inductive bias of local normalized AR model

$$p(\boldsymbol{y}|\boldsymbol{x}) = p_\theta(\boldsymbol{y}|\boldsymbol{x}) \frac{\exp[-E_\phi(\boldsymbol{x}, \boldsymbol{y})]}{Z(\boldsymbol{x})}$$

  - NCE improves over the base AR model by setting $p_N = p_\theta$
  - **Facilitate sampling from EBM:**

(1) **Sampling** from AR proposal

$$\{\boldsymbol{y}^{(k)}\}_{k=1}^{K} \sim p_\theta(\boldsymbol{y}|\boldsymbol{x})$$

(2) **Resampling** with energy function

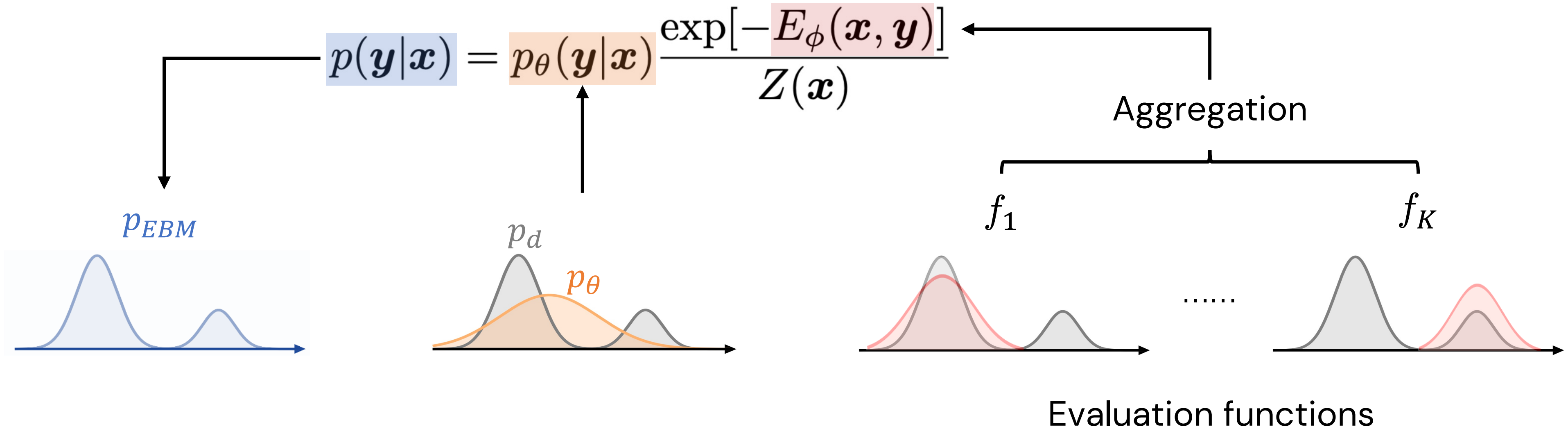
$$\boldsymbol{y} \sim \mathrm{Cat}\Big(\mathrm{softmax}[-E_\theta(\boldsymbol{x}, \boldsymbol{y}^{(k)})]\Big)$$

  - Training a new EBM using NCE every time is **costly** and **restrictive**, considering a large number of available **evaluation metrics**, **reward model**, **classifiers**, etc.
  - Can we leverage those evaluation functions to build EBM?

Bakhtin, Anton., et al. "Residual Energy-Based Models for Text Generation". *JMLR* (2022)

Energy-Based Model

- Build EBM by aggregating evaluation functions [**Ji et al., 2024**] (**ICLR' 24**):



- $\{f_k\}_{k=1}^K$ evaluate different aspect of the distribution
- How to aggregate different evaluation functions?

Ji, Haozhe, et al. "Language Model Decoding as Direct Metrics Optimization." *ICLR* (2024).

Energy-Based Model

- Build EBM by aggregating evaluation functions [**Ji et al., 2024**] (**ICLR' 24**):
  - **Aggregation criteria for unconditional LM decoding:**
    - **Overall quality**: Samples drawn from EBM are "good" on **all** evaluation functions

$$\mathbb{E}_{\boldsymbol{y}\sim p}[f_k(\boldsymbol{y})] = \mathbb{E}_{\boldsymbol{y}\sim p_d}[f_k(\boldsymbol{y})], \forall k \in [1, K]$$

    - **Regularization**: Explore within the support of AR LM distribution:

$$\min_p \mathbb{D}_{\mathrm{KL}}(p \| p_\theta)$$

  - The optimal solution is exactly EBM:

$$p^*(\boldsymbol{y}) \propto p_\theta(\boldsymbol{y}) \exp\left[-\sum_{k=1}^{K} \mu_k^* f_k(\boldsymbol{y})\right]$$

    - Energy function is the **linear combination** of evaluation functions $\{f_k\}_{k=1}^K$
    - ***K*** optimal weights $\{\mu_k^*\}_{k=1}^K$ are automatically determined by solving the constraints.

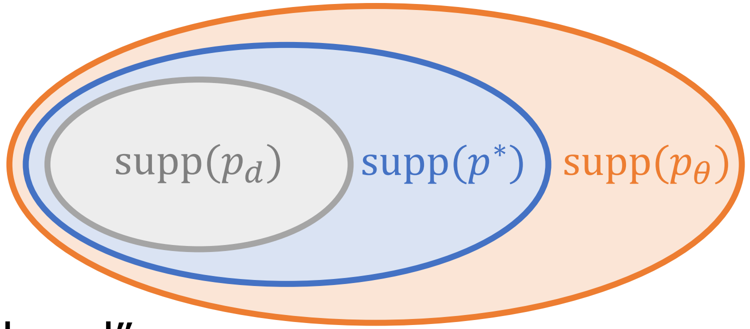
Ji, Haozhe, et al. "Language Model Decoding as Direct Metrics Optimization." *ICLR* (2024).

Energy-Based Model

- Build EBM by aggregating evaluation functions [**Ji et al., 2024**] (**ICLR' 24**):
  - **Theoretical results:** p^* is a better approximation of p_d

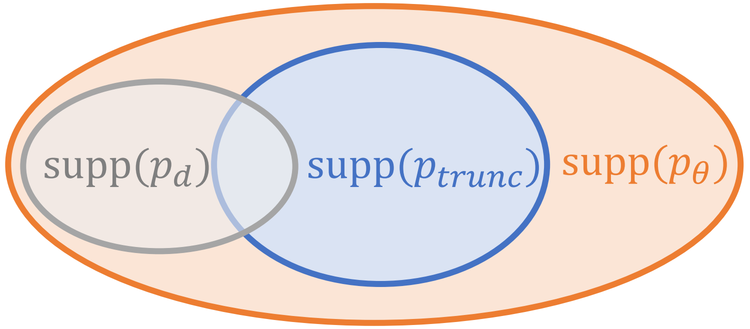
#1 p^* close the **gap of support** to p_d

$$\text{supp}(p_d) \subseteq \text{supp}(p^*) \subseteq \text{supp}(p_\theta)$$



  - Iterating the process effectively approaches p_d
- Heuristic decoding method, e.g., top-k/p truncates p_θ "too hard"

$$\text{supp}(p_d) \nsubseteq \text{supp}(p_{\text{trunc}}) \subseteq \text{supp}(p_\theta)$$



  - Lead to a biased distribution
  - Lose coverage to the complete p_d

Ji, Haozhe, et al. "Language Model Decoding as Direct Metrics Optimization." *ICLR* (2024).

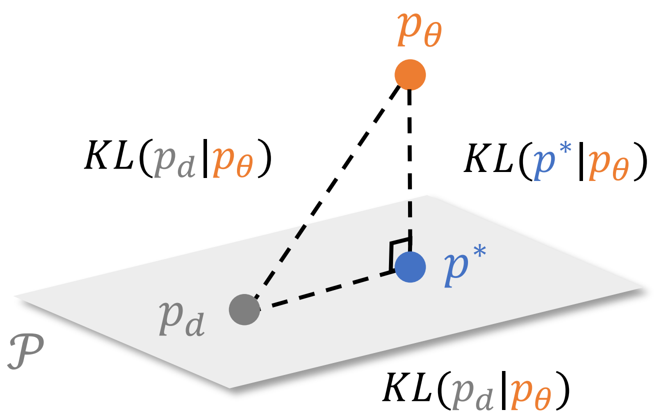
Energy-Based Model

- Build EBM by aggregating evaluation functions [**Ji et al., 2024**] (**ICLR' 24**):
  - **Theoretical results:** p^* is a better approximation of p_d

#2 p^* is guaranteed to improve **perplexity** (2^H) on p_d

$$H(p_d, p^*) = H(p_d, p_\theta) - \underbrace{\mathbb{D}_{\mathrm{KL}}(p^* \| p_\theta)}_{\text{non-negative}}$$

- Pythagorean theorem of KL divergence:



p^* is the **projection** of p_θ on the hyperplane:

$$\mathcal{P} = \{p \mid \mathbb{E}_{\boldsymbol{y}\sim p}[f_k(\boldsymbol{y})] = \mathbb{E}_{\boldsymbol{y}\sim p_d}[f_k(\boldsymbol{y})], \forall k \in [1, K]\}$$

Ji, Haozhe, et al. "Language Model Decoding as Direct Metrics Optimization." *ICLR* (2024).

Experiments

- **Experiments**: Unconditional LM decoding
  - **Evaluation functions**: automatic metrics, e.g., coherence, repetition, diversity, etc.

Truncated Sampling

Contrastive Search

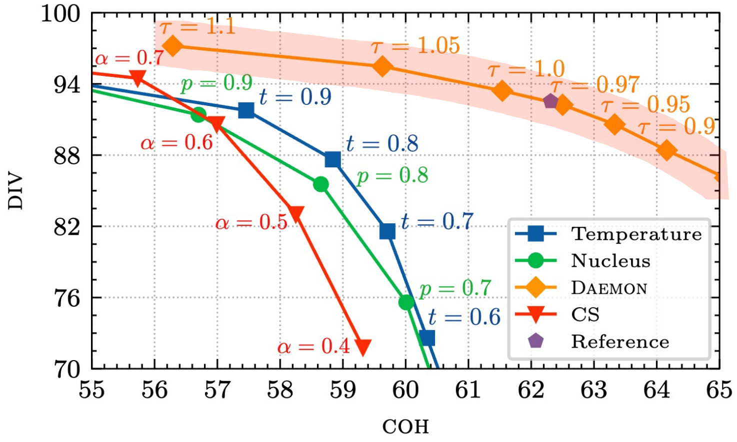
Sample from EBM

	Method	SR-4	TR-32	COH	DIV	e^{ENT}	MAU
		Wikipedia					
	Reference	0.48	21.3	62.3	92.5	23.2	-
GPT-2 XL	Greedy	60.9	65.5	60.2	8.03	2.29	59.7
GPT-2 XL	Top-k	2.11	23.4	60.9	87.8	10.1	77.8
GPT-2 XL	Nucleus	1.19	20.0	57.3	**92.4**	17.3	78.3
GPT-2 XL	Typical	0.81	17.4	54.9	94.5	30.1	78.7
GPT-2 XL	CD	1.31	28.2	68.7	85.9	7.55	77.8
GPT-2 XL	CS	1.78	23.0	56.9	90.6	5.25	83.3
GPT-2 XL	DAEMON	**0.42**	**22.5**	**62.5**	92.2	**22.8**	**88.1**
OPT-6.7B	Greedy	54.8	60.4	62.0	0.12	2.78	64.8
OPT-6.7B	Top-k	2.44	24.1	61.3	86.6	13.9	77.5
OPT-6.7B	Nucleus	2.33	21.9	59.1	88.6	18.9	80.1
OPT-6.7B	Typical	1.06	19.6	57.0	92.9	31.9	77.7
OPT-6.7B	CD	2.90	26.5	68.6	82.3	11.7	78.6
OPT-6.7B	CS	1.13	21.7	57.7	91.8	8.72	83.3
OPT-6.7B	DAEMON	**0.38**	**21.6**	**62.3**	**92.6**	**22.7**	**90.7**

Performance on various metrics

Model	Wikipedia ori	Wikipedia imp	News ori	News imp
GPT-2 XL	23.1	**22.0**	13.9	**13.1**
OPT-6.7B	16.4	**16.2**	10.8	**10.2**

(Tuning-free) Perplexity improvement



coherence-diversity tradeoff

Ji, Haozhe, et al. "Language Model Decoding as Direct Metrics Optimization." *ICLR* (2024).

Experiments

- **Experiments**: Multi-objective alignment
  - **Evaluation functions**: reward models, e.g., helpfulness, harmless, etc.
  - Conditional EBM:

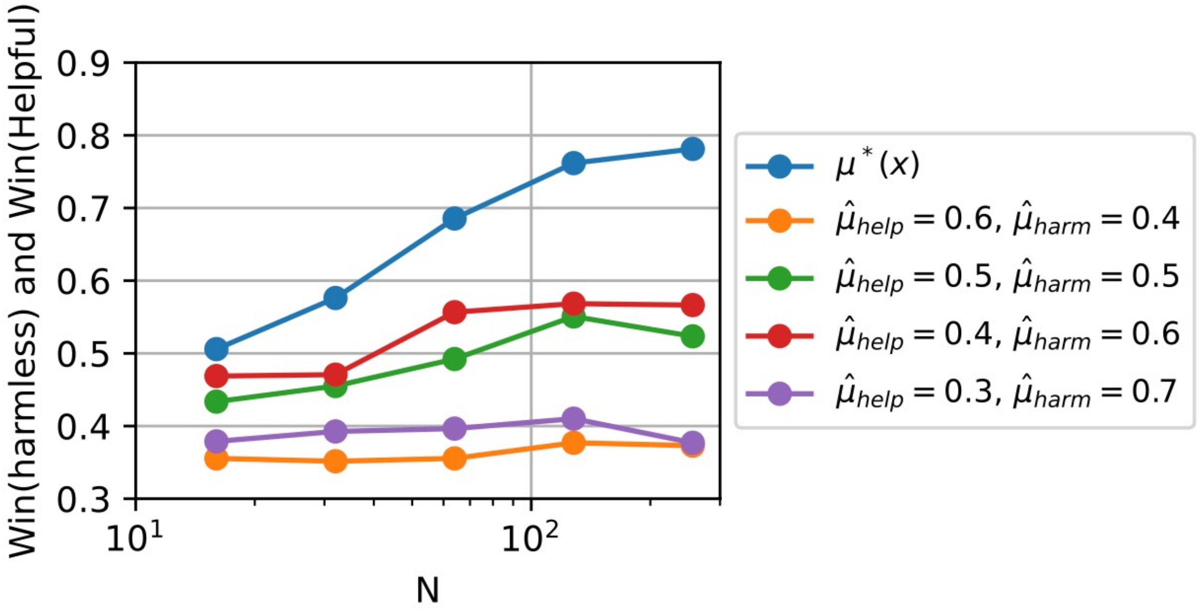
$$p^*(\boldsymbol{y}|\boldsymbol{x}) \propto p_\theta(\boldsymbol{y}|\boldsymbol{x}) \exp\left[-E(\boldsymbol{x},\boldsymbol{y})\right]$$

    - Optimal **instance-level** weight:

$$E(\boldsymbol{x},\boldsymbol{y}) = \sum_{k=1}^{K} \mu_k^*(\boldsymbol{x}) f_k(\boldsymbol{x},\boldsymbol{y})$$

    - Empirical **global** weight:

$$E(\boldsymbol{x},\boldsymbol{y}) = \sum_{k=1}^{K} \hat{\mu}_k f_k(\boldsymbol{x},\boldsymbol{y})$$



Best-of-N experiments on Anthropic-HH

Ji, Haozhe, et al. "Language Model Decoding as Direct Metrics Optimization." *ICLR* (2024).

Energy-Based Model

- **Takeaway & Future**:
- EBM Learning: reward modeling
  - Aggregation: Compositionality of EBM
  - Calibration: Uncertainty-Awareness
- EBM Inference: Acceleration
  - Re-sampling / Rejection sampling
  - MCMC method: Langevin Dynamics
  - Score-guided sampling (learn a score function as in diffusion)
  - Learn tractable AR sampler (lossy due to capacity gap between ARM and EBM)

Beyond the theoretical limits of language modeling

- **Beyond MLE**: Quality-aware objective
  - Reverse KL [**ICML' 24**]: quality assessed by reward that captures human preference
  - Total variation distance [**ICLR' 23**]: quality assessed by the "optimal classifier" in theory

- **Beyond AR**: Expressive model family
  - Energy-based model [**ICLR' 24**]: Augment AR model with a residual energy model
  - **Latent-variable model [EMNLP' 21]**: Condition AR model with a latent plan
  - Look-up model [**EMNLP' 20**]: Extend AR model with a parallel database look-up

Latent-Variable Model

- **Advantage**: Model the unobserved as latent variable increases capacity

$$p(\boldsymbol{y}|\boldsymbol{x}) = \int p_\theta(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z}) p_\theta(\boldsymbol{z}|\boldsymbol{x}) d\boldsymbol{z}$$

  - Theorem [Lin et al., 2020]: Latent-variable AR model has support $\boldsymbol{S} \in \boldsymbol{NP}$
  - Intuition: Marginalizing over the **latent "compression"** $\boldsymbol{z}$ of the future output $\boldsymbol{y}$
- **Disadvantage**: No tractable exact inference of likelihood due to integral over $\boldsymbol{z}$!
- **Variational inference**:

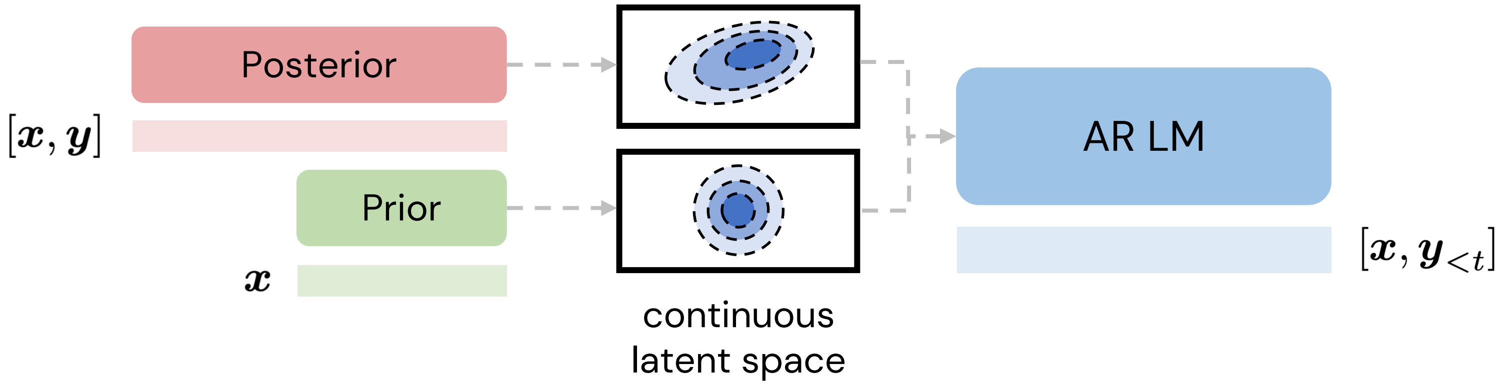
$$p(\boldsymbol{y}|\boldsymbol{x}) = \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})}\left[\frac{p_\theta(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z}) p_\theta(\boldsymbol{z}|\boldsymbol{x})}{q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{y})}\right]$$

  - The inference is "amortized" by first finding a **good approximated posterior** q_ϕ which later facilitates inferring $\boldsymbol{y}$ from $\boldsymbol{z}$.

Lin, Chu-Cheng, et al. "Limitations of Autoregressive Models and Their Alternatives." NAACL (2020).

Latent-Variable Model

- AR model with continuous latent variable [Bowman et al., 2015]:



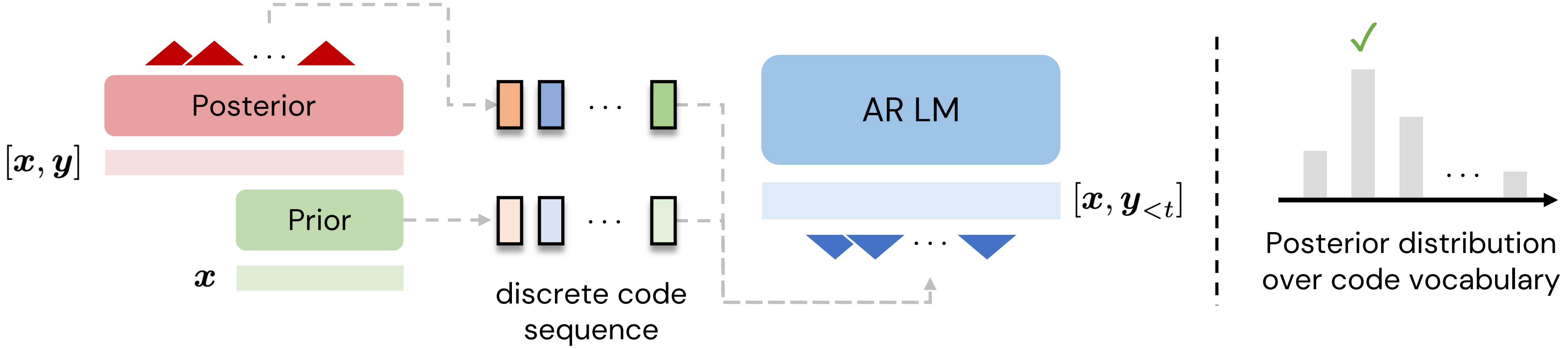
$$\underbrace{-\log p(\boldsymbol{y}|\boldsymbol{x})}_{\text{NLL}} \geq \underbrace{\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{x})}[-\log p_\theta(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z})]}_{\text{negative reconstruction error}} + \underbrace{\mathbb{D}_{\text{KL}}(q_\phi \| p_\theta)[\boldsymbol{x}]}_{\text{posterior-prior gap}}$$

- **Posterior collapse:** Posterior distribution collapses to prior distribution (KL≈0)
- **Losing long-term dependence:** AR generation ignores ***z*** in the long term

Bowman, Samuel., et al. "Generating Sentences From a Continuous Space." *arXiv preprint arXiv:1511.06349* (2015).

Latent-Variable Model

- AR model with structural discrete latent codes [**Ji et al., 2021**] (***EMNLP' 21*** **Oral**):

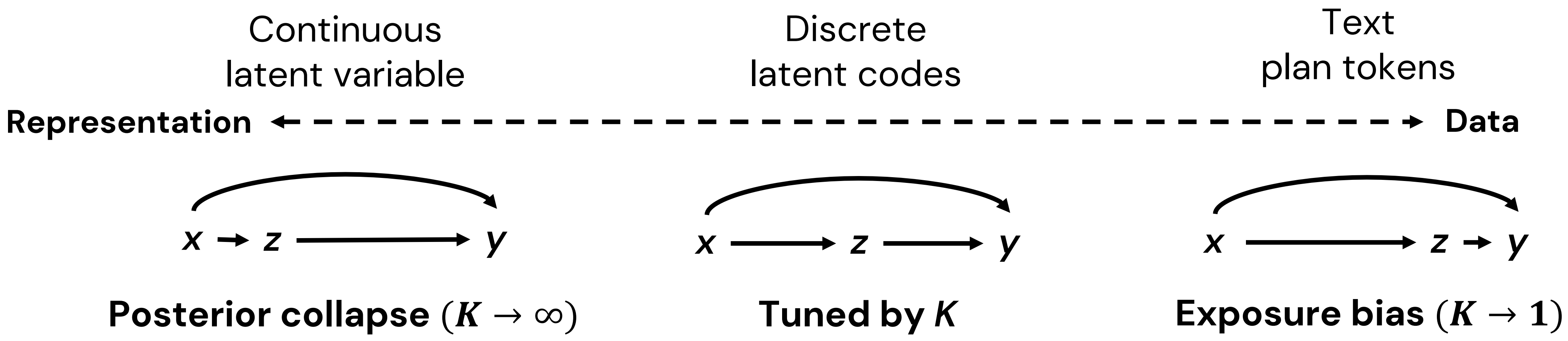


- Discrete code sequence as "latent plan" that captures the long-term structure of y
- **Controlled latent capacity:** # latent codes (L) × # code vocabulary (K)
- **Decoupling ELBO learning** (due to discretization):
  - Obtain code by argmax over posterior distribution
  - Prior AR model learn the code by MLE

Ji, Haozhe., et al., "DiscoDVT: Generating Long Text with Discourse-Aware Discrete Variational Transformer." *EMNLP* (2021).

Latent-Variable Model

- **Takeaway & Future** :
  - A good latent representation control **amortization** of the “bottleneck”



  - Hierarchical latent-variable model: diffusion model
    - Amortize sampling into multiple stages
    - Diffusion for AR LM

Beyond the theoretical limits of language modeling

- **Beyond MLE**: Quality-aware objective
  - Reverse KL [**ICML' 24**]: quality assessed by reward that captures human preference
  - Total variation distance [**ICLR' 23**]: quality assessed by the "optimal classifier" in theory

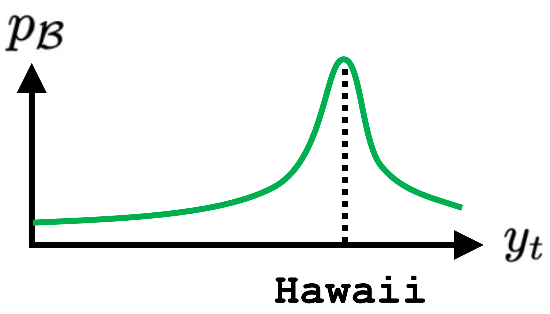
- **Beyond AR**: Expressive model family
  - Energy-based model [**ICLR' 24**]: Augment AR model with a residual energy model
  - Latent-variable model [**EMNLP' 21**]: Condition AR model with a latent plan
  - **Look-up model [EMNLP' 20]**: Extend AR model with a parallel database look-up

Look-Up Model

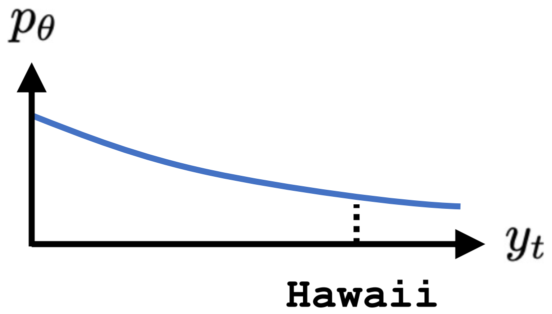
- **Advantage**: Retrieve low-frequency “items” from the distribution long tail
- **Disadvantage**: Naïve look-up model has exploding parameters that stores “all” sequences.
- **Practical look-up model**: Semi-parametric models
  - $\mathcal{B}$: Database, e.g., text documents, knowledge graphs, etc.
  - θ: AR parameters

$$p(y_t|\boldsymbol{x}, \boldsymbol{y}_{<t}) = \lambda(\boldsymbol{x}, \boldsymbol{y}_{<t}) \underbrace{p_{\mathcal{B}}(y_t|\boldsymbol{x}, \boldsymbol{y}_{<t})}_{\text{Database look-up}} + [1 - \lambda(\boldsymbol{x}, \boldsymbol{y}_{<t})] \underbrace{p_{\theta}(y_t|\boldsymbol{x}, \boldsymbol{y}_{<t})}_{\text{AR prediction}}$$

$p(y_t|$`Barack Obama, born in`$)$ =



+



Look-Up Model

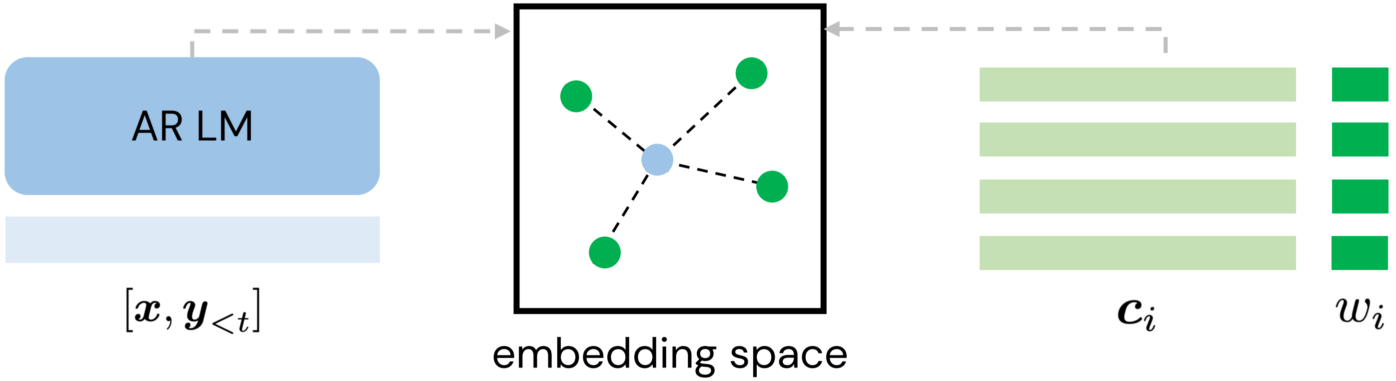
- **Advantage**: Retrieve low-frequency "items" from the distribution long tail
- **Disadvantage**: Naïve look-up model has exploding parameters that stores "all" sequences.
- **Practical look-up model**: Semi-parametric models
  - $\mathcal{B}$: Database, e.g., text documents, knowledge graphs, etc.
  - θ: AR parameters

$$p(y_t|\boldsymbol{x}, \boldsymbol{y}_{<t}) = \lambda(\boldsymbol{x}, \boldsymbol{y}_{<t}) \underbrace{p_{\mathcal{B}}(y_t|\boldsymbol{x}, \boldsymbol{y}_{<t})}_{\text{Database look-up}} + [1 - \lambda(\boldsymbol{x}, \boldsymbol{y}_{<t})] \underbrace{p_{\theta}(y_t|\boldsymbol{x}, \boldsymbol{y}_{<t})}_{\text{AR prediction}}$$

- **Parametric vs Non-parametric**:
  - Parametric AR model is effective at learning local text continuity
  - Non-parametric database is efficient in capturing sparse relationship

Look-Up Model

- Semi-parametric model with text-based $\mathcal{B}$ (kNN-LM) [Khandelwal et al., 2020]:
  - **key-value** from text documents $\mathcal{D}$: $\mathcal{B} = \{(\boldsymbol{c}^i, w^i) | [\boldsymbol{c}^i, w^i] \in \mathcal{D}\}$



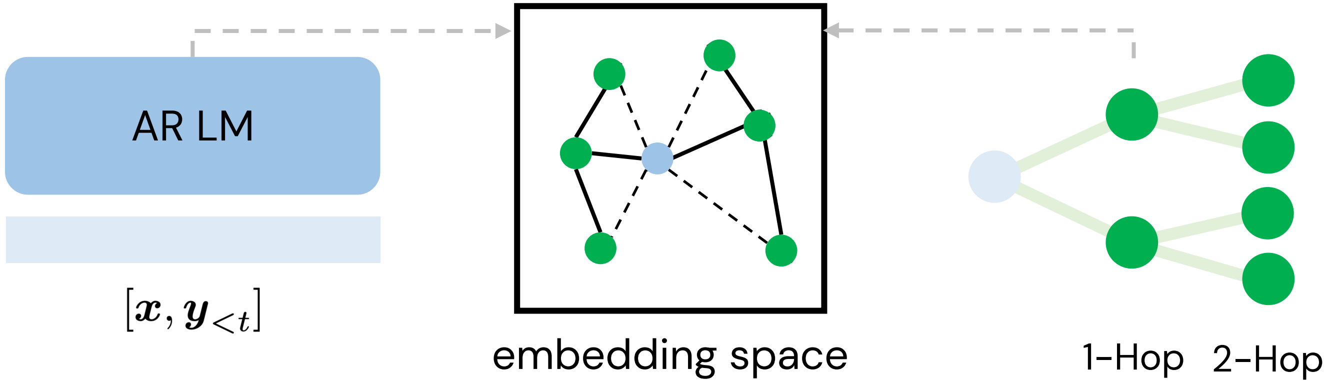
$$p_{\mathcal{B}}(y_t | \boldsymbol{y}_{<t}, \boldsymbol{x}) \propto \sum_{(\boldsymbol{c}^i, w^i)} \mathbb{1}[y_t = w^i] \exp\left(\mathrm{sim}(\boldsymbol{c}^i, [\boldsymbol{x}, \boldsymbol{y}_{<t}])\right)$$

  - Soft matching by context similarity (legacy of text representation learning)
  - The complexity of database grows linearly with the size of training data!

Khandelwal, Urvashi, et al. "Generalization Through Memorization: Nearest Neighbor Language Models." *ICLR* (2020).

Look-Up Model

- Semi-parametric model with graph-based $\mathcal{B}$ [**Ji et al., 2020**] (***EMNLP' 20* Oral**):
  - **Trie** from knowledge graph $\mathcal{G} = (\mathcal{E}, \mathcal{R})$: $\mathcal{B} = \{\tau^i = (\cdots, e^i_j, r^i_{j,j+1}, e^i_{j+1}, \cdots) | e^i_j, e^i_{j+1} \in \mathcal{E}, r^i_{j,j+1} \in \mathcal{R}\}$



$$p_{\mathcal{B}}(y_t|\boldsymbol{y}_{<t}, \boldsymbol{x}) \propto \exp\left(\sum_{\boldsymbol{\tau}^i}\sum_{j=1}^{H} \mathbb{1}[y_t = \tau^i_j]\mathrm{sim}(\tau^i_{<j}, [\boldsymbol{x}, \boldsymbol{y}_{<t}])\right)$$

  - **Gain of structure**:
    - Accumulate and reuse evidence along the branch of the tree
    - The complexity of tree grows linearly with the context length ($\ll$ #docs)
  - Build graph from documents to increase connectivity (followed by future works)

Ji, Haozhe, et al. "Language Generation with Multi-Hop Reasoning on Commonsense Knowledge Graph." *EMNLP* (2020).

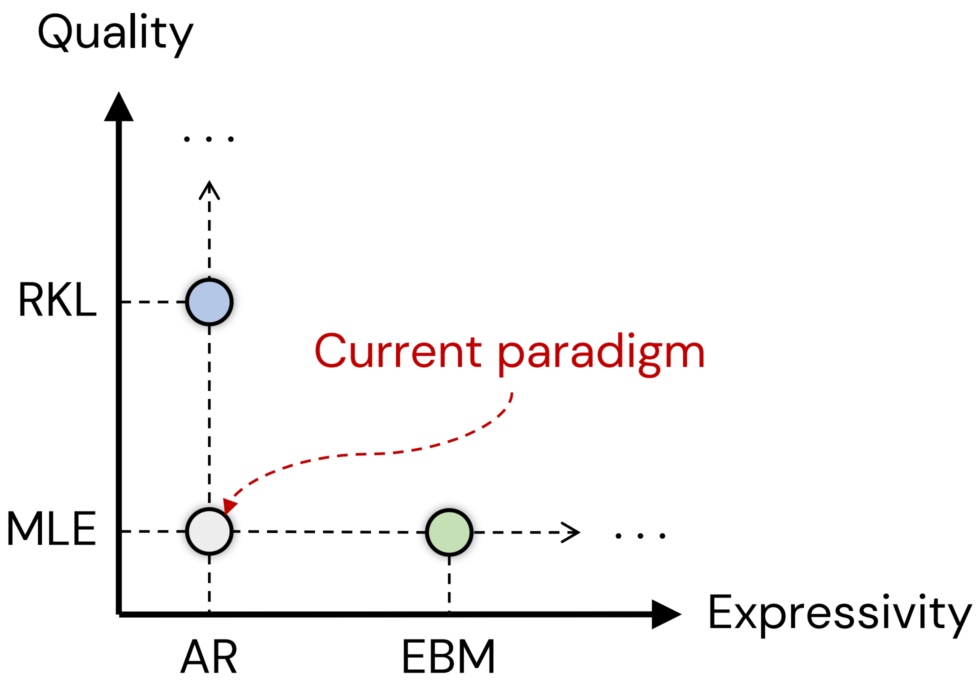
Look-Up Model

- **Takeaway & Future** :
- Look-up at decoding phase:
  - Semi-parametric model: Merging look-up probability with LM probability
  - Induce noise, need dynamic balancing the intensity
- Look-up at encoding phase:
  - Retrieve-Augmented Generation (RAG): LM performing implicit look-up
  - High fluency with hallucination

Conclusion & Future

- Push the boundary of language modeling in a **principled** and **scalable** way:
- **#1** Learn from Data in high quality
  - Fine-grained annotations:

    Generative → **Preferential** → **Process** → ?
  - **Solution**: Quality-aware objective
    - **Key**: quality evaluation
- **#2** Increase model expressivity
  - Data growing slows down
    - Need to increase data utilization
  - **Solution**: Expressive model families
    - **Key**: Scaling up upon AR model



Thanks for Attention!

Q & A

Homepage: https://haozheji.github.io

Email: jihaozhe@gmail.com