Estimating the Probabilities of Rare Outputs in Language Models

Gabriel Wu, Jacob Hilton

Alignment Research Center (ARC)

March 23, 2025

Motivation

- Want to improve the worst-case behavior of a model
- Standard solution: adversarial training
  - Find input on which model does bad, then train on it
- Alternate proposal: low probability estimation
  - Estimate the probability mass of inputs that cause model to do bad

Problem Statement: Low Probability Estimation

Given:

- A language model $M : \mathcal{V}^* \to \mathcal{V}$
- An input distribution $\mathcal{D}$ over $\mathcal{V}^*$
- A "target token" t

Output an estimate for:

$$\Pr_{x \sim \mathcal{D}}[M(x) = t]$$

Naive Sampling

- Draw n inputs from $\mathcal{D}$, and count how many of them satisfy $M(x) = t$.
- Bad at detecting probabilities below $1/n$.

Importance Sampling

- Let p be the original input distribution. Define a new distribution q that "upweights" regions associated with the target behavior.
- We sample a bunch from q, then output our empirical estimate of:

$$\mathbb{E}_{x\sim p}[\mathbb{1}(M(x)=t)] = \mathbb{E}_{x\sim q}\left[\frac{p(x)}{q(x)}\mathbb{1}(M(x)=t)\right]$$

Metropolis Hastings Importance Sampling

Define:

$$q(x) \propto p(x) \cdot \exp\left(\frac{M_t(x)}{T}\right)$$

But this is hard to sample from.

Metropolis Hastings Importance Sampling

Define:

$$q(x) \propto p(x) \cdot \exp\left(\frac{M_t(x)}{T}\right)$$

But this is hard to sample from.
Solution:

- Define a random walk (in input space $\mathcal{V}^k$) that has q as its stationary distribution
- For the walk's proposal function, re-sample a random token according to a Boltzmann distribution defined by current gradient of $M_t(x)$

Once the walk mixes, this lets us get samples from q.

Activation Extrapolation

- Importance Sampling methods are analogous to what is already done in standard adversarial training.
  - They require finding explicit examples of inputs that trigger the behavior.

Activation Extrapolation

- Importance Sampling methods are analogous to what is already done in standard adversarial training.
  - They require finding explicit examples of inputs that trigger the behavior.
- In contrast, activation extrapolation describes a *non-constructive* class of methods.

Activation Extrapolation

- Importance Sampling methods are analogous to what is already done in standard adversarial training.
  - They require finding explicit examples of inputs that trigger the behavior.
- In contrast, activation extrapolation describes a *non-constructive* class of methods.
- Roughly: fit some idealized distribution to an internal activation of the model, then calculate the probability of the target behavior under this distributional assumption.

Activation Extrapolation

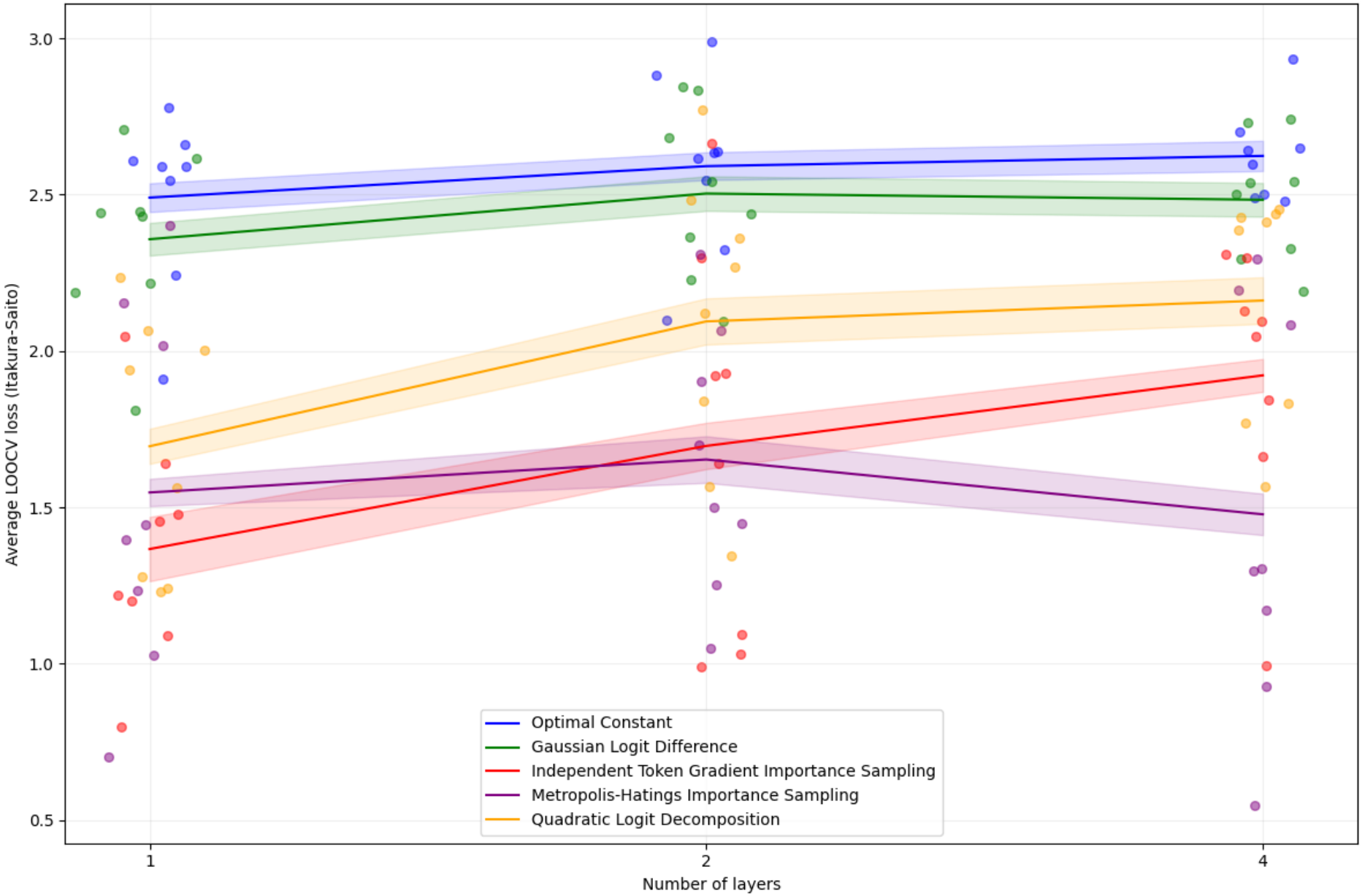
- Importance Sampling methods are analogous to what is already done in standard adversarial training.
  - They require finding explicit examples of inputs that trigger the behavior.
- In contrast, activation extrapolation describes a *non-constructive* class of methods.
- Roughly: fit some idealized distribution to an internal activation of the model, then calculate the probability of the target behavior under this distributional assumption.
  - This second step can sometimes be done more efficiently than end-to-end sampling (e.g. if there's an analytic formula)

Quadratic Logit Decomposition

- Collect n samples of the logit vector $\mathbf{v}(x) \in \mathbb{R}^{|\mathcal{V}|}$.
- Choose an "important" direction $\mathbf{d} \in \mathbb{R}^{|\mathcal{V}|}$
- Decompose each sample $\mathbf{v}^{(i)}$ into $\mathbf{a}^{(i)} + \mathbf{b}^{(i)}$ where $\mathbf{a}^{(i)} \,||\, \mathbf{d}$ and $\mathbf{b}^{(i)} \perp \mathbf{d}$.
- Output:

$$\frac{1}{n^2} \left| \left\{ (i,j) \in [n]^2 \;\middle|\; \mathbf{a}^{(i)} + \mathbf{b}^{(j)} \text{ has highest logit at } t \right\} \right|$$

Results


3.0
2.5
2.0
1.5
1.0
0.5
Average LOOCV loss (Itakura-Saito)
1
2
4
Number of layers
Optimal Constant
Gaussian Logit Difference
Independent Token Gradient Importance Sampling
Metropolis-Hatings Importance Sampling
Quadratic Logit Decomposition

Thank you!

- Paper:

  `arxiv.org/abs/2410.13211`
- Empirical blog post:

  `www.alignment.org/blog/low-probability-estimation-in-language-models/`
- Theoretical/philosophical blog post:

  `www.alignment.org/blog/estimating-tail-risk-in-neural-networks/`