RLHF/DPO 小話（その１）

和地 瞭良 / *Akifumi Wachi*

April 21, 2024

RLHF [1]や DPO [2]に関して論文 [Wachi et al., 2024] を最近執筆した。やはり自分で研究すると理解がかなり深まるわけで、巷ではあまり議論されていないことが色々と分かった。なので、すでに公開済みの内容の中から話せる範囲で情報共有して、誰かの役に立てばいいな、と思っている次第である。やる気が持続すればその 4 くらいまでいく予定（多分力尽きる）。

[1] RLHF = Reinforcement Learning from Human Feedback [Ouyang et al., 2022]

[2] DPO = Direct Preference Optimization [Rafailov et al., 2024]

背景： 言語モデルのアライメント

ChatGPT の性能向上に寄与したとされる技術の一つとして、Reinforcement Learning from Human Feedback (RLHF) がある。具体的には、「言語モデルのアライメント」のために用いられ、言語モデルを人間の趣向・好みに合致させるプロセスのことを指す。

RLHF/DPO は結局何を解いているのか？

まず重要なのが、**RLHF や DPO が解いている問題は同じ**、ということを理解 [3]することだろう。これは、DPO の後続研究である、ΨPO/IPO [Azar et al., 2023] や KTO [Ethayarajh et al., 2024] も同様で、以下のような方策最適化問題を解くことを目指す。

$$\max_{\pi_\theta} \quad \mathbb{E}_{x\sim\rho, y\sim\pi_\theta(\cdot|x)}\left[\, r(x,y)\,\right] - \beta \mathbb{D}_{\mathrm{KL}}\left[\, \pi_\theta(y \mid x) \,\|\, \pi_{\mathrm{ref}}(y \mid x)\,\right], \quad (1)$$

[3] （我々の論文含め）大抵の論文はまず、Bradley-Terry モデルが云々から議論がスタートするのだが、RLHF/DPO の本質はそこではない（と思う）ので、その話は一旦あえてスキップ。

この式の意味を丁寧に見ていこう。まず、$x \in \mathcal{X}$ はプロンプト、$y \in \mathcal{Y}$ は言語モデル（Language Model, LM）の出力である。入力 x は何らかの確率分布 ρ からサンプルされるとする。このとき、言語モデルは入力 x に対し出力 y を返す**方策**とみなすことができる。いま、π_{SFT} と π_θ という 2 つの方策が (1) に存在するが、π_θ は θ によってパラメタライズされた学習中の方策、π_{SFT} は元となるリファレンス方策を表す。π_{SFT} は多くの場合、Supervised Finetuning（SFT）が施された言語モデルが用いられる。$r : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ は、入力 x と出力 y のペアに対して、その良し悪しを実数値で返す関数で**報酬関数**と呼ばれる。最後に、$\mathbb{D}_{\mathrm{KL}}$ は Reverse KL divergence [4] であり、(1) を簡潔に書くため、すでに $\mathbb{E}_{x\sim\rho}[\cdot]$ の期待値が取られていることに留意されたい。

[4] Forward KL divergence じゃないのが大事！

最適化問題 (1) の意味

まず、第一項について解説する。

$$\mathbb{E}_{x\sim\rho, y\sim\pi_\theta(\cdot|x)}\left[\, r(x,y)\,\right] \quad (2)$$

この式は、「なんらかの報酬関数 r が仮に与えられたとして、それを最大化するような方策 π_θ を見つけてきてね」 というお気持ちが込められて

いる。つまり、プロンプト x が ρ からサンプルされたのち、方策 π_θ が出力 $y \sim \pi_\theta(\cdot \mid x)$ を生成するわけだが、そのときの (x, y) が「良いペア」になるように π_θ を最適化したい、というわけである。

次に、第二項について。

$$\beta \mathbb{D}_{\mathrm{KL}}[\, \pi_\theta(y \mid x) \,\|\, \pi_{\mathrm{ref}}(y \mid x) \,], \tag{3}$$

逆説的になるが、この項がなかったとすると以下のような問題が生じうる

- RLHF/DPO を用いる際のデータは多くの場合 π_{SFT} によって生成された y を用いて作成される。π_θ が π_{SFT} から乖離すると、データの疎の領域で π_θ が最適化される可能性があり、学習が破綻する。
- π_θ が π_{SFT} から乖離すると、「言語モデル」としての性能が低下し、不自然・意味不明な文章が生成される可能性が高まる。

上記のような理由から、π_θ と π_{SFT} との距離はある程度近く保ちたい、という要求が生じる。そのため、(1) では $\beta \mathbb{D}_{\mathrm{KL}}[\, \pi_\theta(y \mid x) \,\|\, \pi_{\mathrm{ref}}(y \mid x) \,]$ を「引く」という操作をしていることになる。

最適化問題 (1) をどう解くか?: 2 つの主要なアプローチ

RLHF/DPO において、最適化問題 (1) がいかに重要かがおわかりいただけただろうか？RLHF や DPO に限らず、IPO や KTO といった後続研究も大半は (1) を解くことを目指しているのに変わりはない。

このとき、(1) を解くにあたっての問題点は、**報酬関数 r をどのようにして獲得するか？**である。繰り返しになるが、報酬関数 r は、プロンプト x に対して、言語モデル（方策）の出力 y が「どのくらい良いか」かを実数で返す関数である。このとき、２つの主要なアプローチがある。

- **アプローチ１（RLHF）** [5,6,7]
  - reward model r_ψ を学習する
  - (1) の r を r_ψ に置き換える
  - 強化学習を用いて (1) を解く
- **アプローチ２（直接法）** [8]
  - 報酬関数 r とそれに対応する最適方策 π の関係性を解析的に導出
  - 報酬の構造になんらかの仮定をおく
  - 報酬に関する教師あり学習を、方策に関する教師あり学習に変換して解く

DPO 元論文の図が非常に分かり易いので拝借（Figure 1参照）するが、RLHF は報酬モデル（reward model）を学習したのち強化学習を行うが、DPO をはじめとする直接的なアプローチは陽に報酬関数を学習する

[5] 当初は PPO がよく用いられていた [Ouyang et al., 2022] が、最近は REINFORCE でも良いという研究成果も出ている Ahmadian et al. [2024]

[6] PPO = Proximal Policy Optimization Schulman et al. [2017]。深層強化学習アルゴリズムで最も有名なアルゴリズムの一つ。（主観が入るが）まぁまぁ複雑。

[7] REINFORCE Williams [1992]。1992 年誕生の強化学習黎明期のアルゴリズム。めっちゃシンプル。

[8] 個人的な意見としては、「RLHF を使うべき理由が思いつかない」場合は、直接的なアプローチを使うと良いと思う。trl (`https://github.com/huggingface/trl`) 等で実装されており、colab 等で動くサンプルコードも豊富である。

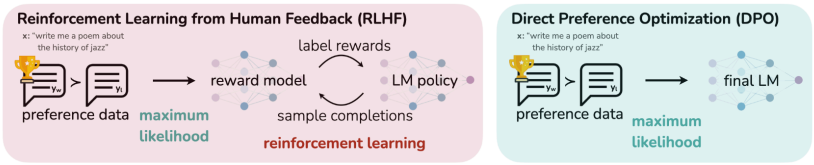

Figure 1: Rafailov et al. [2024] より図を拝借

ことなく一回の maximum likelihood で言語モデルをアライメントしていることになる。

それぞれのアプローチの具体的な中身がなにかは次回解説するとして、直接法の**「報酬関数 r とそれに対応する最適方策 π の関係性を解析的に導出」**について最後に述べる。

直接法: 「あなたの言語モデル、実は報酬モデルですよ」

NeurIPS-23 で受賞した DPO 論文には実は

- *Your Language Model is Secretly a Reward Model*

というサブタイトルがある。この意味深なサブタイトルは言いえて妙であり、「報酬関数 r とそれに対応する最適方策 π の関係性を解析的に導出」に大きく関係する。

あくまで解きたい問題は (1) であり、再度書くと

$$\max_{\pi_\theta} \quad \mathbb{E}_{x\sim\rho, y\sim\pi_\theta(\cdot|x)}\left[r(x,y)\right] - \beta\mathbb{D}_{\mathrm{KL}}\left[\pi_\theta(y\mid x)\,\|\,\pi_{\mathrm{ref}}(y\mid x)\right],$$

である。RLHF では、PPO や RENINFORCE などの強化学習アルゴリズムを使って頑張ってこの問題を解いていたのだが、なんとこの問題の最適解は解析的に解けてしまうのである。以下に定理を示そう。[9]

Theorem 1. 任意の関数 $f : \mathcal{X}\times\mathcal{Y}\to\mathbb{R}$ に対し、

$$\mathbb{E}_{\rho,\pi}\left[f(x,y)\right] - \beta\mathbb{D}_{\mathrm{KL}}\left[\pi(y\mid x)\,\|\,\pi_{\mathrm{ref}}(y\mid x)\right] \tag{4}$$

を最大化する最適方策は、

$$\pi_f^\star(y\mid x) = \frac{1}{Z_f(x;\pi_{\mathrm{ref}})}\pi_{\mathrm{ref}}(y\mid x)\exp\left(\frac{1}{\beta}f(x,y)\right), \tag{5}$$

と書くことができる。ただし、$Z_f(x;\pi_{\mathrm{ref}})$ は

$$Z_f(x;\pi_{\mathrm{ref}})\sum_y \pi_{\mathrm{ref}}(y\mid x)\exp\left(\frac{1}{\beta}f(x,y)\right). \tag{6}$$

のように定義される正規化のための関数ないしは定数である。

証明は最後に付録しているので、興味のある方は数式を追っていただきたい。この式の意味としては、

[9] 我々の論文 Wachi et al. [2024] において、報酬関数を任意の関数 f（e.g., safety function）に拡張するので、Theorem 1 も r ではなく f で記述しておく。力尽きたらごめんなさい。

- 報酬関数が与えられれば、それに対応する最適方策がすでに分かっている
- 方策（言語モデル）が与えられれば、その背後にある報酬モデルもすでに分かっている

ということになり[10]、RLHF のように、「報酬モデルの学習を学習してその後 (1) を最適化」というような面倒なプロセスを踏まなくてよい、ということを意味する。これは学習を簡略化・安定化に大きく寄与しており、実用上も DPO の性能が高いことがわかるにつれ、様々な言語モデルのアライメントで用いられるようになる。そのような理由もあって、DPO は様々な後続手法が登場し、注目されているわけである。

[10] 現実的には、正規化のための Z_f は $\sum_y$ の形になっており、あらゆる出力 y に対して和を計算する必要があるので、計算するのが困難です。しかし、次回？解説しますが、DPO を始めとる直接法はこの Z_f を陽に計算することなく方策を最適化することができます。

今後書くこと（予定）

- 報酬モデルの様々な仮定
  - DPO から IPO, KTO, $\cdots$ へ
- (1) でなぜ Reverse KL divergence が使われているのか？
- (1) と強化学習の関係
- なぜ SACPO は制約付きの方策最適化問題を stepwise に解くことが許されるのか？

References

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting REINFORCE style optimization for learning from human feedback in LLMs. *arXiv preprint arXiv:2402.14740*, 2024.

Mohammad Gheshlaghi Azar, Mark Rowland, Bilal Piot, Daniel Guo, Daniele Calandriello, Michal Valko, and Rémi Munos. A general theoretical paradigm to understand learning from human preferences. *arXiv preprint arXiv:2310.12036*, 2023.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. KTO: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*, 2024.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Akifumi Wachi, Thien Q Tran, Rei Sato, Takumi Tanabe, and Yohei Akimoto. Stepwise alignment for constrained language model policy optimization. *arXiv preprint arXiv:2404.11049*, 2024.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256, 1992.

Proof. 証明は基本的に Rafailov et al. [2024] の Appendix A.1 と同じである。Reverse KL divergence の定義から、以下のような関係 [11] が成り立つ。

[11] 以降、すべて等式であることに注意。

$$
\begin{aligned}
&\max_{\pi} \mathbb{E}_{\rho,\pi}\big[f(x,y)\big] - \beta \mathbb{D}_{\mathrm{KL}}\big[\,\pi(y \mid x) \,\|\, \pi_{\mathrm{ref}}(y \mid x)\,\big] \\
&\quad = \max_{\pi} \mathbb{E}_{\rho,\pi}\left[f(x,y) - \beta \log \frac{\pi(y \mid x)}{\pi_{\mathrm{ref}}(y \mid x)}\right] \\
&\quad = \min_{\pi} \mathbb{E}_{\rho,\pi}\left[\log \frac{\pi(y \mid x)}{\pi_{\mathrm{ref}}(y \mid x)} - \frac{1}{\beta} f(x,y)\right] \\
&\quad = \min_{\pi} \mathbb{E}_{\rho,\pi}\left[\log \frac{\pi(y \mid x)}{\pi_{\mathrm{ref}}(y \mid x) \exp\left(\frac{1}{\beta} f(x,y)\right)}\right] \\
&\quad = \min_{\pi} \mathbb{E}_{\rho,\pi}\left[\log \frac{\pi(y \mid x)}{\frac{1}{Z_f(x;\pi_{\mathrm{ref}})} \pi_{\mathrm{ref}}(y \mid x) \exp\left(\frac{1}{\beta} f(x,y)\right)} - \log Z_f(x;\pi_{\mathrm{ref}})\right],
\end{aligned}
\tag{7}
$$

ただし、$Z_f(x;\pi_{\mathrm{ref}})$ は正規化のための分配関数である。いま、$\pi_f^\star$ を以下のように定義する。

$$
\pi_f^\star(y \mid x) = \frac{1}{Z_f(x;\pi_{\mathrm{ref}})} \pi_{\mathrm{ref}}(y \mid x) \exp\left(\frac{1}{\beta} f(x,y)\right),
$$

すると、(7) は以下のように書くことができる。

$$
\begin{aligned}
&\min_{\pi} \mathbb{E}_{x\sim\rho}\left[\mathbb{E}_{y\sim\pi(y|x)}\left[\log \frac{\pi(y \mid x)}{\pi_f^\star(y \mid x)}\right] - \log Z_f(x;\pi_{\mathrm{ref}})\right] \\
&= \min_{\pi}\Big[\mathbb{D}_{\mathrm{KL}}\big[\,\pi(y \mid x) \,\|\, \pi_f^\star(y \mid x)\,\big] - \mathbb{E}_{x\sim\rho}\left[\log Z_f(x;\pi_{\mathrm{ref}})\right]\Big].
\end{aligned}
$$

$Z_f(x;\pi_{\text{ref}})$ は方策 π に依存しないので、右辺第二項は最適化問題に影響を与えない。したがって、以下の問題を解けばよいことになる。

$$\arg\min_{\pi} \ \mathbb{D}_{\text{KL}}[\,\pi(y \mid x) \,\|\, \pi_f^\star(y \mid x)\,].$$

ギブスの不等式から、KL-divergence は、2つの確率分布が等しいときに 0 をとり最小となるので、すべての $x \in \mathcal{X}$ に対して、

$$\pi(y \mid x) = \pi_f^\star(y \mid x) = \frac{1}{Z_f(x;\pi_{\text{ref}})}\pi_{\text{ref}}(y \mid x)\exp\left(\frac{1}{\beta}f(x,y)\right) \tag{8}$$

となる。すなわち (1) の最適解は、(8) のように書くことができる。 □