RLHF/DPO 小話（その 4）

和地 瞭良 / *Akifumi Wachi*

April 30, 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]

これまでのコラム（その 1 - 3）

- `https://akifumi-wachi-4.github.io/website/jp.html`

前回のおさらい

本コラムで扱ってきた手法のほとんど（RLHF, DPO, KTO など）は、

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

という Reverse KL divergence $\mathbb{D}_{\mathrm{KL}}$ を用いた定式化に基づいており、この問題の解析解は

$$\pi_r^\star(y \mid x) = \frac{1}{Z_r(x;\pi_{\mathrm{ref}})}\pi_{\mathrm{ref}}(y \mid x)\exp\left(\frac{1}{\beta}r(x,y)\right), \tag{2}$$

という形で書ける、と述べてきた。

一方で、$\mathbb{D}_{\mathrm{KL}}$ を任意の f-divergence $\mathbb{D}_f$ で置き換えることも可能で、その場合は解析解が

$$\pi_r^\star(y \mid x) = \frac{1}{\widehat{Z}_r(x;\pi_{\mathrm{ref}})}\pi_{\mathrm{ref}}(y \mid x)(f')^{-1}\left(\frac{1}{\beta}r(x,y)\right), \tag{3}$$

という (2) の一般型となる。例えば、Forward KL divergence の場合は、

$$\pi_r^\star(y \mid x) \propto -\pi_{\mathrm{ref}}(y \mid x)\left(\frac{1}{\beta}r(x,y)\right)^{-1} \tag{4}$$

を得る。

Reverse KL divergence の大きすぎるメリットとは

いま、ある指標 r に関してすでにアライメント済みの方策を、また別の指標 g でアライメントすることを考えよう。Reverse KL divergence を用いているとき、指標 r に関してアライメント済みの方策は、

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

を満たす[3]。いまこの $\pi_r^\star$ を更に指標に g に関してアライメントするとき、いま我々が解いている最適化問題は

$$\max_{\pi_\theta} \quad \mathbb{E}_{x\sim\rho, y\sim\pi_\theta(\cdot|x)}\left[\, g(x,y)\,\right] - \beta_g \mathbb{D}_{\mathrm{KL}}\left[\,\pi_\theta(y\mid x)\,\|\,\pi_r^\star(y\mid x)\,\right], \tag{6}$$

となり、その最適解 $\pi^\sharp$ は

$$\pi^\sharp(y\mid x) = \frac{1}{Z^\sharp(x)}\pi_r^\star(y\mid x)\exp\left(\frac{1}{\beta_g}g(x,y)\right), \tag{7}$$

なる。ただし、KL ペナルティのためのパラメータは、より一般的なケースを考えるため、$\beta \to \beta_g$ とした。もちろん $\beta_g = \beta$ でも構わない。ただし、$Z^\sharp$ は正規化のための分配関数で、

$$Z^\sharp(x) := \sum_y \pi_r^\star(y\mid x)\exp\left(\frac{1}{\beta_g}g(x,y)\right). \tag{8}$$

である。$\pi^\sharp$ を直感的に説明すると、以下のようになる。

- **$\pi_r^\star$ をパラメータ β_g で関数 g に関してアライメントしたときの最適解は (6) のような形で書けますよ**

いま、$\pi_r^\star$ の定義式 (2) を代入してみよう。すると、

$$\begin{aligned}
\pi^\sharp(y\mid x) &= \frac{1}{Z^\sharp(x)}\pi_r^\star(y\mid x)\exp\left(\frac{1}{\beta_g}g(x,y)\right) \\
&= \frac{1}{Z^\sharp(x)}\underbrace{\frac{1}{Z_r(x;\pi_{\mathrm{ref}})}\pi_{\mathrm{ref}}(y\mid x)\exp\left(\frac{1}{\beta}r(x,y)\right)}_{\pi_r^\star(y|x)}\exp\left(\frac{1}{\beta_g}g(x,y)\right) \\
&= \frac{1}{Z^\sharp(x)}\frac{1}{Z_r(x;\pi_{\mathrm{ref}})}\pi_{\mathrm{ref}}(y\mid x)\exp\left(\frac{1}{\beta}r(x,y)+\frac{1}{\beta_g}g(x,y)\right)
\end{aligned}$$

再び、この式の意味を直感的に考えてみると、

- **$\pi_r^\star$ をパラメータ β_g で関数 g に関してアライメントしたときの最適解は、パラメータ β で π_{ref} を関数 $r + \frac{\beta_g}{\beta}g$ に関してアライメントしたときの最適解と一致する。**

となる。つまり、アライメント済みの方策を更にアライメントする、という操作は、ないぶでは多目的最適化を解いていることになるわけである。

しかし、このような結果が得られるのは、exp が任意の実数 $\mathsf{x},\mathsf{y}\in\mathbb{R}$ に対し、

$$\exp(\mathsf{x}+\mathsf{y}) = \exp(\mathsf{x})\exp(\mathsf{y}) \tag{9}$$

を満たすからで、exp が登場するのは Reverse KL divergence を使っているからである。

Forward KL divergence のときはどうであろうか？最適解の形が (4) になるわけだが、意味がないとは言わないまでも、

$$\left(\frac{1}{\beta}r(x,y)\right)^{-1}\left(\frac{1}{\beta_g}g(x,y)\right)^{-1} \tag{10}$$

の積の形がいまいち何をしたいのが判然としないだろう。

[3] 厳密には、（特に RLHF のときに）アルゴリズムの最適化誤差等により上式を満たしているとは限らないが、最適化が上手になされたとして方策が (5) のような形状をしていると仮定して話を進める。

言語モデルのアライメントアルゴリズムに必要な性質とは

言語モデルは、これから最も社会に浸透する人工知能の一つになるだろう。そうすると、「有用なことを人間らしく喋る」というだけではダメで、社会的な責任を果たす必要が出てくるわけである。どの程度の厳しい安全性基準を設けるかは、単純な技術の問題ではないので一旦おいておくとしても、

- 有害発言 (暴言・差別発言) を頻繁に吐いてしまう
- 敵対的なユーザーに対してあまりに脆弱
- プライバシーを漏洩してしまう

等といった言語モデルはやはり許されないわけである。

多目的最適化・制約付き最適化

現状のアライメントアルゴリズムは、単一の指標である報酬関数 r に基づいてアライメントしているわけであるが、「より有用な出力」と「より安全な出力」を同じものとして扱っているわけで、アルゴリズム選択・データセット構築双方の面で柔軟性が低いと言わざるを得ない。

そうすると、報酬関数とは別の関数 g を用意して、**多目的最適化・制約付き最適化**問題として解いたほうが見通しが良いことが多い。

いま、リファレンス方策 π_{ref} を関数

$$h(x,y) := r(x,y) + \lambda g(x,y) \tag{11}$$

に関してアライメントすることを考えよう。$\lambda \in \mathbb{R}$ は、r と g の影響度を調整するパラメータである。制約付き最適化についても適切な仮定を置くことにより、11に帰着するので、詳細が気になる方はWachi et al. [2024] をご覧頂きたい。

このコラムで何度も登場しているが、このときの最適解は、

$$\pi_h^\star(y \mid x) = \frac{1}{Z_h(x;\pi_{\text{ref}})}\pi_{\text{ref}}(y \mid x)\exp\left(\frac{1}{\beta}r(x,y) + \frac{\lambda}{\beta}g(x,y)\right), \tag{12}$$

となる。exp の性質から、

$$\pi_h^\star(y \mid x) = \frac{1}{Z_h(x;\pi_{\text{ref}})}\pi_{\text{ref}}(y \mid x)\exp\left(\frac{1}{\beta}r(x,y)\right)\exp\left(\frac{\lambda}{\beta}g(x,y)\right),$$

が成り立つこと、$\pi_r^\star$ が (2) のように表せることから、

$$\begin{aligned}\pi_h^\star(y \mid x) &= \frac{Z_r(x;\pi_{\text{ref}})}{Z_h(x;\pi_{\text{ref}})}\underbrace{\frac{1}{Z_r(x;\pi_{\text{ref}})}\pi_{\text{ref}}(y \mid x)\exp\left(\frac{1}{\beta}r(x,y)\right)}_{\pi_r^\star(y|x)}\exp\left(\frac{\lambda}{\beta}g(x,y)\right)\\ &= \frac{Z_r(x;\pi_{\text{ref}})}{Z_h(x;\pi_{\text{ref}})}\pi_r^\star(y \mid x)\exp\left(\frac{\lambda}{\beta}g(x,y)\right)\end{aligned}$$

を得る。

段階的に最適化すれば *OK*

前章で最後に登場した式、$Y(x) := \frac{Z_h(x;\pi_{\text{ref}})}{Z_r(x;\pi_{\text{ref}})}$ とおけば、

$$\pi_h^\star(y \mid x) = \frac{1}{Y(x)} \pi_r^\star(y \mid x) \exp\left(\frac{\lambda}{\beta} g(x,y)\right)$$

のように表せる。簡単な式変形から、

$$g(x,y) = \frac{\beta}{\lambda} \log \frac{\pi_h^\star(y \mid x)}{\pi_r^\star(y \mid x)} + \frac{\beta}{\lambda} \log Y(x). \tag{13}$$

となる。$Y(x)$ は今まで登場した分配関数同様出力 y に依存しない関数ゆえ、

$$\begin{aligned} &g^\star(x,y_w) - g^\star(x,y_l) \\ &= \frac{\beta}{\lambda^\star} \log \frac{\pi^\star(y_w \mid x)}{\pi_{r^\star}^\star(y_w \mid x)} + \cancel{\frac{\beta}{\lambda^\star} \log Y(x)} - \frac{\beta}{\lambda^\star} \log \frac{\pi^\star(y_l \mid x)}{\pi_{r^\star}^\star(y_l \mid x)} - \cancel{\frac{\beta}{\lambda^\star} \log Y(x)}. \end{aligned}$$

が成立する。たとえば DPO を用いて再びアライメントする際には、

$$\begin{aligned} &\mathcal{L}_{\text{DPO}}(\pi_\theta, \pi_r^\star, \beta/\lambda) \\ &\quad = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}} \left[\log \sigma \left(\frac{\beta}{\lambda} \log \frac{\pi_\theta(y_w \mid x)}{\pi_r^\star(y_w \mid x)} - \frac{\beta}{\lambda} \log \frac{\pi_\theta(y_l \mid x)}{\pi_r^\star(y_l \mid x)}\right)\right]. \end{aligned}$$

というロス関数を最小化すればよいことになるのである。

(13) は、最適化問題 (1) から導かれたものであり、報酬関数の構造等には一切仮定をおいていない。したがって、Reverse KL divergence を用いた定式化ベースの手法（e.g., DPO, IPO, KTO）であれば同様の議論が成り立つ。唯一の違いは、ロス関数の定義において、

$$\mathcal{L}(\pi_\theta, \pi_{\text{ref}}, \beta) \to \mathcal{L}(\pi_\theta, \pi_r^\star, \beta/\lambda)$$

というように、リファレンス方策 と KL penalty のパラメータ が微妙に変わっているだけである。

段階的に最適化することのメリットとは

段階的に最適化することの妥当性が示されると何が嬉しいか？

- 各アライメントで別のアルゴリズム（e.g., DPO と KTO）が使える
- 各アライメントで別形式のデータセットが使える。報酬に関しては (DPO で使われるような）Preference data、安全性に関しては （KTO で使われるような） Unpaied data、など。
- 既存のアライメント済みモデルを使うことができる
- デプロイ後になにか問題が発覚した際（安全性に問題が見つかったなど）に、最初からアライメントし直す必要がない。

これらの恩恵は、Reverse KL divergence を用いて定式化しているからこそ得られていたわけで、

- 段階的なアライメント = 多目的/制約付きアライメント

という人間にも理解しやすく、安心して行うことのできる操作になっていたわけである。

余談：*RLHF/DPO* でバイアスは助長される

その 3 で焦らしていた伏線を回収しておこう。結局は上の式変形とほぼ同じことをするのですでに分かっている読者も多いと思うが。

いま、ラベル付けする人間（A さんと呼ぶ）にバイアスがあるとしよう。この人は、以下のような報酬関数 $\hat{r}$ に基づいてラベル付けしているとする

$$\hat{r}(x,y) = r^\star(x,y) + b(x,y)$$

$r^\star$ は聖人君主のような人がもつ潜在的な報酬関数で一切のバイアスや間違いのないものであるとする。b はバイアスを表す関数で、簡単のため $b(x,y) \geq 0, \forall (x,y)$ とする。つまり、A さんは、聖人君主と比較して、$b(x,y)$ 分のバイアスを潜在的にもって学習データを作成している、ということになる。

いま、A さんが作成したデータで、RLHF なり DPO を行ったとしよう。すると、RLHF/DPO は、そのデータを完全に信用して方策を最適化するので、それらが目指す最適解は、

$$\begin{aligned}\pi^\star_{\hat{r}}(y \mid x) &= \frac{1}{Z_{\hat{r}}(x;\pi_{\text{ref}})}\pi_{\text{ref}}(y \mid x)\exp\left(\frac{1}{\beta}\hat{r}(x,y)\right) \\ &= \frac{1}{Z_{\hat{r}}(x;\pi_{\text{ref}})}\pi_{\text{ref}}(y \mid x)\exp\left(\frac{1}{\beta}r^\star(x,y) + b(x,y)\right) \qquad (14)\end{aligned}$$

である。聖人君主が集めたデータで方策を最適化した場合、RLHF/DPO が目指す最適解が

$$\pi^\star_{r^\star}(y \mid x) = \frac{1}{Z_{r^\star}(x;\pi_{\text{ref}})}\pi_{\text{ref}}(y \mid x)\exp\left(\frac{1}{\beta}r^\star(x,y)\right) \qquad (15)$$

であったことを考えると、(14) と (15) の間には、

$$\pi^\star_{\hat{r}}(y \mid x) \propto \pi^\star_{r^\star}(y \mid x)\exp\left(\frac{1}{\beta}b(x,y)\right)$$

という関係が成り立つことになる。つまり、ラベルに $b(x,y)$ というバイアスがあると、結果的に得られる方策の確率分布は、$\exp\left(\frac{1}{\beta}b(x,y)\right)$ の影響を受けてしまうということになるのである。

おわりに

その 4 をもって、RLHF/DPO 小話は以上となります。最後までお読みいただきありがとうございました。

References

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.

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.