Содержание

Первый эксперимент

Первым экспериментом была реализация оригинальной статьи за исключением того факта, что в силу ограниченных ресурсов и времени количество итераций в каждом внутреннем цикле алгоритма T я взяла равной всего 10, а не 100.

I = 2
M = 2
T = 10
beta = 0.1
mu = 0.01
eta = 0.5
optimizer = Adam
lr = 1e-6

Результаты

Результаты получились следующие:

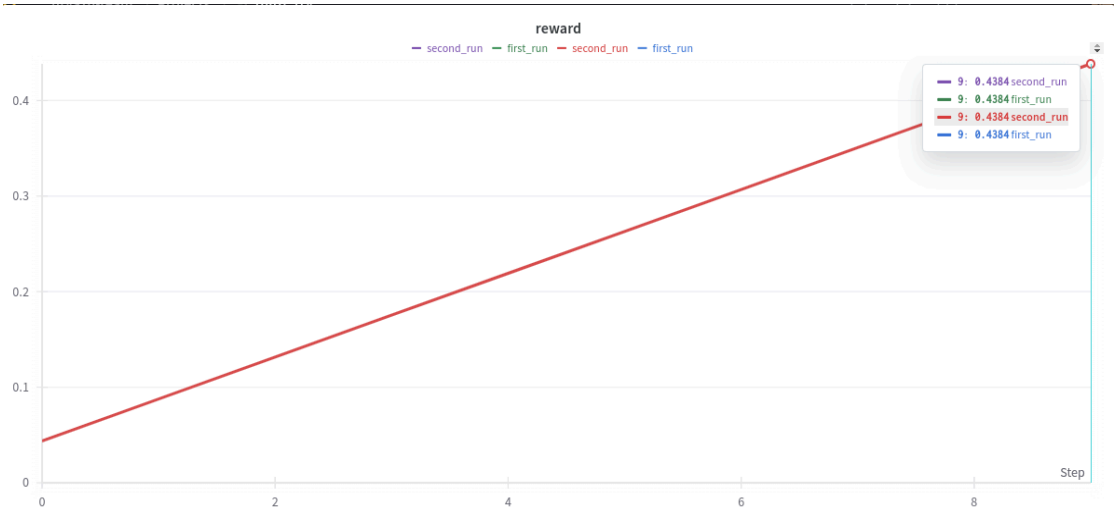
```
Mean SFT reward: 0.0438411
Mean WARP reward: 0.0438411
```

```
Mean KL(sft||warp): 0.00000
Mean KL(warp||sft): 0.00000
```

Мы видим, что модель за всё своё обучение никак не изменилась. Объяснение этому даётся ниже.

Процесс обучения

Что касается обучения, график максимизируемого функционала выглядит так:



Здесь и далее theta1_1 и theta2_1 – это theta для m = 1 и theta для m = 2 из алгоритма при первой итерации (т.е. при i = 1), аналогично theta1_2, theta2_1 – теты при i = 2. Также сразу оговорю, что под первой итерацией я понимаю i = 1, под второй – i = 2; и никакие больше.

Algorithm 1 *WARP* for KL-reward Pareto optimal alignment

Input: Weights θ_{sft} pre-trained and supervised fine-tuned
Reward model r, prompt dataset $\mathcal{X}$, optimizer Opt
I iterations with M RL runs each for T training steps
μ *EMA* update rate, η *LITI* update rate

Define $\theta_{\text{init}} \leftarrow \theta_{\text{sft}}$
for iteration i from 1 to I **do**
  for run m from 1 to M **do** ▷ Run in parallel
    Define $\theta^m, \theta^m_{\text{ema}} \leftarrow \theta_{\text{init}}$
    for step t from 1 to T **do**
      Generate completion $\boldsymbol{y} \sim \pi_{\theta^m}(\cdot \mid \boldsymbol{x})$ for $\boldsymbol{x} \in \mathcal{X}$
      Compute $r_\beta(\boldsymbol{y}) \leftarrow r(\boldsymbol{x}, \boldsymbol{y}) - \beta \log \frac{\pi_{\theta^m}(\boldsymbol{y}|\boldsymbol{x})}{\pi_{\theta^m_{\text{ema}}}(\boldsymbol{y}|\boldsymbol{x})}$ ▷ KL regularized reward
      Update $\theta^m \leftarrow \text{Opt}\big(\theta^m, r_\beta(\boldsymbol{y}) \nabla_\theta [\log \pi_{\theta^m}(\boldsymbol{y} \mid \boldsymbol{x})]\big)$ ▷ Policy gradient
      Update $\theta^m_{\text{ema}} \leftarrow (1-\mu) \cdot \theta^m_{\text{ema}} + \mu \cdot \theta^m$ ▷ Equation (*EMA*): update anchor
    end for
  end for
  Define $\theta^i_{\text{slerp}} \leftarrow \text{slerp}\Big(\theta_{\text{init}}, \{\theta^m\}_{m=1}^M, \lambda = \frac{1}{M}\Big)$ ▷ Equation (*SLERP*): merge M weights
  Update $\theta_{\text{init}} \leftarrow (1-\eta) \cdot \theta_{\text{init}} + \eta \cdot \theta^i_{\text{slerp}}$ ▷ Equation (*LITI*): interpolate towards init
end for

Output: KL-reward Pareto front of weights $\{(1-\eta) \cdot \theta_{\text{sft}} + \eta \cdot \theta^I_{\text{slerp}} \mid 0 \le \eta \le 1\}$

Мы видим, что на первой итерации что первое, что второе обучение происходило абсолютно одинаково (графики функции буквально совпадают). То же самое справедливо и для второй итерации. Скорее всего, это произошло из-за слишком маленького T, т.к. за 10 итераций внутреннего цикла модель не успела начать выучивать что-то специфическое, а значит, оба её обучения прошли одинаково. Также видно, что графики и для i=1, и для i=2 также совпадают. Скорее всего, это из-за минорно изменившейся на первом шаге theta_init и очень маленького lr: веса, которыми мы инициализируем модели при каждом новом i, почти не поменялись, а значит, обучение из той же точки инициализации при таком маленьком lr и при таком маленьком количестве “эпох” не могло привести нас в какую-то кардинально другую точку.

Вывод

Вывод, который был сделан – взят слишком маленький lr.

Второй эксперимент

Конфигурация второго эксперимента была почти такой же, только было решено убрать warmup, чтобы lr был сразу хотя бы 1e-6, а не линейно прогревался к этому значению в течение нескольких шагов.

Результаты

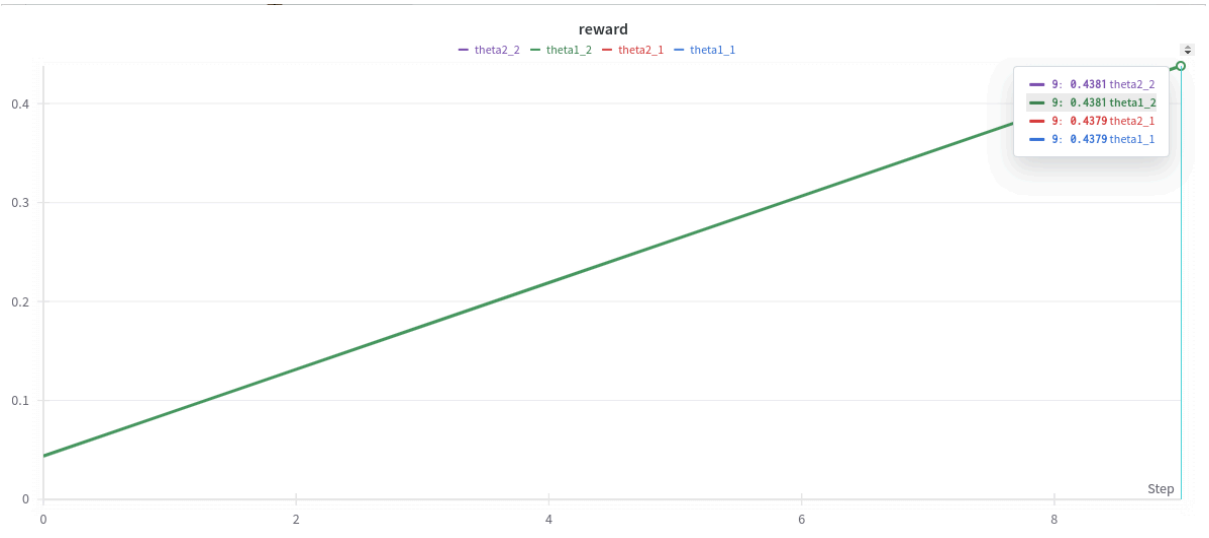
Результаты получились следующими:

```
Mean SFT reward: 0.0438411
Mean WARP reward: 0.0438410

Mean KL(sft||warp): 0.02546
Mean KL(warp||sft): 0.02410
```

Модели несильно отличаются друг от друга, поэтому и награды у них почти одинаковы.

Процесс обучения



Как видим, графики что первой, что второй итерации, опять совпали.

Вывод

Вывод: либо lr всё ещё слишком маленький, либо warmup особо ни на что не влиял. Кажется, что обе причины резонны: аргументы в пользу первой были приведены в первом эксперименте. В пользу второй – warmup используют, чтобы на первых эпохах модель, которая слабо ориентируется в мире, не шагала слишком далеко от своей инициализации. В нашем случае мы берём уже хорошо предобученную модель и файн-тьюним модель на задачу, которая не радикально отличается от того, что она уже умеет делать, так что шагнуть чуть подальше на первых итерациях не должно быть очень страшно.

Третий эксперимент

В третьем эксперименте было решено увеличить lr, причём сразу радикально до 1e-3, а также убрать warmup. В основном, чтобы проверить гипотезу из предыдущего пункта.

Результаты

```
Mean SFT reward: 0.0438411
Mean WARP reward: 0.0438404

Mean KL(sft||warp): 89.02149
Mean KL(warp||sft): 100.29043
```

Модели стали больше отличаться друг от друга, но награда WARP никак не отличалась от награды обычной SFT.

Процесс обучения

Мы видим, что как только мы увеличили lr, обучение модели на двух разных итерациях (т.е. при i = 1 и при i = 2) стало происходить по-разному, потому что на первой итерации мы сильнее изменили theta_init. Однако модель на первой итерации выдавала награды больше, чем на второй, т.е. мы явно начали шагать куда-то не туда в пространстве весов.

Вывод

Был выбран слишком большой lr.

Четвертый эксперимент

В последнем, четвёртом, эксперименте было решено уменьшить lr.

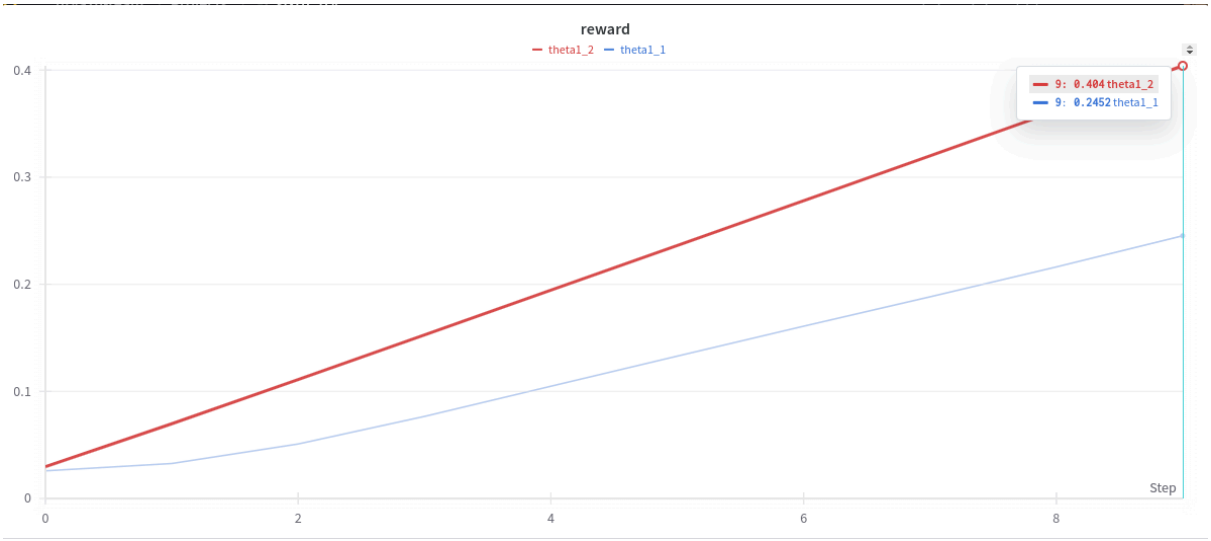
Результаты

```
Mean SFT reward: 0.0438413
Mean WARP reward: 0.0438342

Mean KL(sft||warp): 33.56183
Mean KL(warp||sft): 30.76847
```

Модели стали не так значительно отличаться друг от друга, что адекватно учитывая малое количество эпох, но WARP просела в качестве.

Процесс обучения



Видим, что теперь по мере обучения модель начинает выдавать бОльшие награды, а также на разных итерациях её траектория обучения отличается.

Вывод

Были окончательно исправлены недостатки самого первого эксперимента, однако объективно качество получилось не очень. В качестве дальнейших наработок нужно увеличить T, а также добавить линейный прогрев lr до 1e-4.