

## 進捗報告

### 1 やったこと

- シードを変えてパラメータの変化の確認
- 学習率を揃えて学習

### 2 シードを変えた場合のパラメータの変化の確認

シードを変えて大きくパラメータの変化にずれが発生していれば、解析の意味が無くなってしまうためシードを変えてノルムを再計算した。シードの設定には transformers の `set_seed()` メソッドを用いている。表 1 に学習方法が同一でデータセットが異なる 2 モデル間のシードが異なる場合の L1, L2 ノルムの値を示す。

表 1: 学習方法が同一でデータセットが異なる 2 モデル間の L1 ノルム, L2 ノルム

モデル	L1 ノルム (seed = 42)	L2 ノルム (seed = 42)	L1 ノルム (seed = 100)	L2 ノルム (seed = 100)
$M_{\text{SFT}_{D_0}}, M_{\text{SFT}_{D_1}}$	1073358.0	16.658565739582745	2859111.0	45.28603753158914
$M_{\text{DPO}_{D_0}}, M_{\text{DPO}_{D_1}}$	283154.5	2.7612353904133857	280975.25	2.695570339088263

実験で SFT の初期学習率が  $1e-4$ , DPO は  $1e-5$  と学習率の違いがあるとはいえ、DPO は大きな差が見られなかった一方で、SFT はシードによって大きくノルムの値が変わっている結果となった。

そのため、SFT の初期学習率を DPO と同じ  $1e-5$  にそろえて学習しなおし、異なるシード間での値の差を確かめた。表 2 に結果を示す。

表 2: 学習方法が同一でデータセットが異なる 2 モデル間の L1 ノルム, L2 ノルム (SFT の初期学習率  $1e-5$ )

モデル	L1 ノルム	L2 ノルム
$M_{\text{SFT}_{D_0}}, M_{\text{SFT}_{D_1}}$ (seed = 42)	277361.125	2.20031771804024
$M_{\text{SFT}_{D_0}}, M_{\text{SFT}_{D_1}}$ (seed = 100)	278791.25	2.217655637326153

初期学習率を DPO と同じにすることで SFT においてもシードによる値の大きな変動が無くなった。重みの大小の比較のためにも学習率を揃えた方が都合が良いため、上記の結果をうけ SFT の初期学習率を  $1e-5$  とすることにした。

### 3 学習率を揃えて学習

考察のため、ベースのモデルに加え以下の条件でモデルを学習させる。学習したモデルの重みは huggingface 上で保存している。

$M_{\text{base}}$  ベースモデル (elyza/Llama-3-ELYZA-JP-8B)

$M_{\text{SFT}_{D_0}}$   $M_{\text{base}}$  に対して  $D_0$  を用いて SFT を適用したモデル <sup>1</sup>

$M_{\text{SFT}_{D_1}}$   $M_{\text{base}}$  に対して  $D_1$  を用いて SFT を適用したモデル <sup>2</sup>

<sup>1</sup> [https://huggingface.co/Nisk36/SFT\\_normal\\_lr5](https://huggingface.co/Nisk36/SFT_normal_lr5)

<sup>2</sup> [https://huggingface.co/Nisk36/SFT\\_ojousama\\_lr5](https://huggingface.co/Nisk36/SFT_ojousama_lr5)

$M_{\text{DPO}_{D_0}} M_{\text{base}}$  に対して  $D_0$  を用いて DPO を適用したモデル <sup>3</sup>

$M_{\text{DPO}_{D_1}} M_{\text{base}}$  に対して  $D_1$  を用いて DPO を適用したモデル <sup>4</sup>

$M_{\text{SFT}_{D_2}} M_{\text{base}}$  に対して  $D_2$  を用いて SFT を適用したモデル <sup>5</sup>

$M_{\text{DPO}_{D_0}(M_{\text{SFT}_{D_2}})} M_{\text{SFT}_{D_2}}$  に対して  $D_0$  を chosen として DPO を適用したモデル <sup>6</sup>

$M_{\text{DPO}_{D_1}(M_{\text{SFT}_{D_2}})} M_{\text{SFT}_{D_2}}$  に対して  $D_1$  を chosen として DPO を適用したモデル <sup>7</sup>

$M_{\text{DPO}_{D_0}(M_{\text{SFT}_{D_1}})} M_{\text{SFT}_{D_1}}$  に対して  $D_0$  を chosen として DPO を適用したモデル <sup>8</sup>

$M_{\text{DPO}_{D_0}(M_{\text{SFT}_{D_1}})} M_{\text{SFT}_{D_1}}$  に対して  $D_1$  を chosen として DPO を適用したモデル [https://huggingface.co/Nisk36/DPO\\_ojousamachosen\\_lr5](https://huggingface.co/Nisk36/DPO_ojousamachosen_lr5)

また, elyza/Llama-3-ELYZA-JP-8B と異なるアーキテクチャのモデルでもデータを取りたいと考えた.

試したモデルは以下の通り. Llama 系統とは異なるアーキテクチャであることを重要視しモデルを選んだ.

- tokyotech-llm/Swallow-MS-7b-instruct-v0.1<sup>9</sup>  
Mistral-7B-v0.1 派生のモデル
- stabilityai/japanese-stablelm-instruct-alpha-7b-v2<sup>10</sup>  
アーキテクチャが GPT-NeoX のモデル
- google/gemma-2-9b-it<sup>11</sup>
- google/gemma-2-2b-jpn-it<sup>12</sup>
- Qwen/Qwen2.5-7B-Instruct<sup>13</sup>

上記の中で, google/gemma-2-9b-it は GPU の VRAM で Cuda のエラーとなった.

また tokyotech-llm/Swallow-MS-7b-instruct-v0.1, stabilityai/japanese-stablelm-instruct-alpha-7b-v2, google/gemma-2-9b-it, google/gemma-2-2b-jpn-it は学習したモデルの出力が日本語として成立していない, 出力上限まで出力を続けてしまうといった問題が発生した. 学習に用いたコードは同じのためおそらく元の LLM の性能が結果を左右していると考えられる.

Qwen/Qwen2.5-7B-Instruct は学習後の出力も簡単なテストでは安定していたため, 上記の elyza/Llama-3-ELYZA-JP-8B と同様の条件下で学習した. 表 3, 4, 5 に実験のパラメータを示す.

## 参考文献

<sup>3</sup>[https://huggingface.co/Nisk36/DPO\\_normalchosen\\_lr5](https://huggingface.co/Nisk36/DPO_normalchosen_lr5)

<sup>4</sup>[https://huggingface.co/Nisk36/DPO\\_ojousamachosen\\_lr5](https://huggingface.co/Nisk36/DPO_ojousamachosen_lr5)

<sup>5</sup>[https://huggingface.co/Nisk36/SFT\\_both\\_lr5](https://huggingface.co/Nisk36/SFT_both_lr5)

<sup>6</sup>[https://huggingface.co/Nisk36/DPO\\_normalchosen\\_afterSFTBoth\\_lr5](https://huggingface.co/Nisk36/DPO_normalchosen_afterSFTBoth_lr5)

<sup>7</sup>[https://huggingface.co/Nisk36/DPO\\_ojousamachosen\\_afterSFTBoth\\_lr5](https://huggingface.co/Nisk36/DPO_ojousamachosen_afterSFTBoth_lr5)

<sup>8</sup>[https://huggingface.co/Nisk36/DPO\\_normalchosen\\_afterSFT\\_lr5](https://huggingface.co/Nisk36/DPO_normalchosen_afterSFT_lr5)

<sup>9</sup><https://huggingface.co/tokyotech-llm/Swallow-MS-7b-instruct-v0.1>

<sup>10</sup><https://huggingface.co/stabilityai/japanese-stablelm-instruct-alpha-7b-v2>

<sup>11</sup><https://huggingface.co/google/gemma-2-9b-it>

<sup>12</sup><https://huggingface.co/google/gemma-2-2b-jpn-it>

<sup>13</sup><https://huggingface.co/Qwen/Qwen2.5-7B-Instruct>

表 3: QLoRA パラメータ

パラメータ	値
量子化サイズ	4 ビット
r	8
lora_alpha	128
target_modules	モデル内の線形層全て
lora_dropout	0.05

表 4: SFTTrainer パラメータ

パラメータ	値
epoch 数	3
バッチサイズ	2
最適化手法	Adam
初期学習率	1e-5
学習率スケジューラ	cosine

表 5: DPOTrainer パラメータ

パラメータ	値
epoch 数	3
バッチサイズ	2
最適化手法	Adam
初期学習率	1e-5
学習率スケジューラ	cosine
beta	0.3