PLaMo 2 における事後学習

Post-training in PLaMo 2

野沢 健人 (Kento Nozawa)[1]
[1] 株式会社 Preferred Networks (Preferred Networks, Inc.)
e-mail : nzw0301@preferred.jp

1 はじめに

OpenAI 社の GPT モデル [1] をはじめとする大規模言語モデル（LLM）は，日常会話や文書要約といった伝統的な自然語処理タスクだけでなく，コーディング支援，人工データ生成や推論を伴うテキスト生成など，幅広い用途に対して利用されている．開発されている LLM の多くは英語か中国語が支配的なデータセットを学習に用いており [2]，日本語の占める割合は相対的に低いため，英語や中国語の話者に比べて日本語話者は，LLM の高度な機能を享受しにくい [3]．Preferred Language Model（PLaMo）は，事前学習の段階から日英が支配的なデータセットを用いて，より日本語での言語処理能力の高い LLM として開発されている．最新の PLaMo 2 では，前世代のモデルである PLaMo-1-100B [4] に比べて約 1/3 のパラメータ数しか持たないものの，ベンチマークタスクにおいて PLaMo-1-100B モデルを上回るスコアを達成した．本講演では，PLaMo 2 について，事後学習を重点について紹介する．

2 事後学習

事後学習は，事前学習済みモデルを初期値として指示文に対して適切な応答文が生成できるように学習を行う．PLaMo 2 では，supervised fine-tuning（SFT）と preference learning の 2 段階からなる学習を実施した．以下では採用したアルゴリズムについて概説する．

■Supervised Fine-tuning (SFT)： 事前学習済みモデルでは，大量のテキストが与えられ次のトークンを予測するように学習が行われているため，指示文に対する簡潔な応答は難しい．SFT では，ユーザが実際に入力するような示文に対して，それに対する望ましい応答文を教師データとして学習する．このとき，事前学習の損失関数と異なり，応答分に対してのみ損失関数を計算する．

■Preference Learning： Preference learning では，同じ指示文に対する複数の応答文を特定の指標で順位付けし，順位の高い応答文が低い応答文よりも生成しやすくなるように学習する．例えば，会話を楽しむような指標の場合，指示文が “**こんにちは**” であった場合に，応答文として高圧的に感じられる “**は？**” ではなく，より会話が続きそうな “**こんにちは！今日は何がありましたか？**” を生成しやすくなるように学習する．具体的なアルゴリズムとして，PLaMo 2 では，offline でデータを収集できる direct preference optimization (DPO) [5] を採用した．

■Model Merge： 性能改善を行うために，SFT と DPO のそれぞれの実施後にアンサンブル手法の一種である model soup [6] を行なった．Model soup では，同一のモデルを異なるハイパーパラメータで学習し，得られた異なる重みの平均を取ることで，推論コストを単一モデルと変えることなく性能改善できる．特に SFT では，Cohere [7] によって報告されているデータセットを大きく変えて学習した結果の重み付き平均を取ることで，ベンチマークスコアを改善した．

謝辞　PLaMo 2 は，経済産業省と国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）が日本の生成 AI 基盤モデルの開発力向上を目指して実施している GENIAC 第 2 期の成果を元に作られています．本発表の主な内容は，株式会社 Preferred Networks の Alignment チームによるものです．

参考文献

[1] OpenAI. GPT-4 Technical Report, 2024. URL https://arxiv.org/abs/2303.08774.

[2] DeepSeek-AI. DeepSeek-V3 Technical Report, 2025. URL https://arxiv.org/abs/2412.19437v2.

[3] Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. Language Models are Multilingual Chain-of-Thought Reasoners. In *ICLR*, 2023.

[4] Preferred Elements, Inc. PLaMo-100B: A Ground-Up Language Model Designed for Japanese Proficiency, 2024. URL https://arxiv.org/abs/2410.07563.

[5] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. pages 53728–53741, 2023.

[6] Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model Soups: Averaging Weights of Multiple Fine-tuned Models Improves Accuracy without Increasing Inference Time. In *ICML*, pages 23965–23998, 2022.

[7] Cohere. Command A: An Enterprise-Ready Large Language Model, 2025. URL https://arxiv.org/abs/2504.00698.

バンディットアルゴリズムのランダム化に基づく高速化について

On Speeding up Bandit Algorithms Based on Randomization

本多 淳也 (Junya Honda)[1,2]
[1] 京都大学 (Kyoto University), [2] 理研 AIP (RIKEN AIP)
e-mail : honda@i.kyoto-u.ac.jp

1 概要

バンディット問題は選択肢について事前に知識がない状態から試行錯誤を通じて利益の最大化を目指す代表的な枠組みである．この問題では設定に応じて漸近最適性を達成するための汎用的な枠組みが明らかになりつつあるが，その多くは最適化や積分が関わる複雑な計算が必要となる．本発表ではこれらの計算をランダム化により回避する方法に関する最近の研究を紹介する．

2 はじめに

多腕バンディット問題はスロットマシンをプレイするプレイヤーのモデルで，不確かな知識のもとで意思決定を行う最も基本的なモデルの一つである．この問題ではスロットマシンの K 本のアームが与えられ，プレイヤーは各時刻 $t \in [T] = \{1, 2, \ldots, T\}$ でいずれか 1 つのアーム $I_t \in [K]$ を過去得られた情報に基づいて選択する．時刻 t での各アームの損失ベクトルを $\ell_t = (\ell_{t,1}, \ell_{t,2} \ldots, \ell_{t,K})^\top \in [0,1]^K$ で表し，これは設定した環境に応じて後述の手順に従って生成される．プレイヤーはこれらのうち選択したアーム I_t についての損失 ℓ_{t,I_t} のみが観測可能であり，被った損失の累積和の最小化を目指す．この問題は新薬の治験といった応用から 1930 年頃から考えられており，インターネットの発達に伴いウェブ広告やニュース推薦といった応用から急速に発展したほか，近年では農作物の作付や通信といった応用も考えられている．

プレイヤーが用いるアルゴリズム（方策）の性能は擬リグレットとよばれる指標 $\mathrm{Regret}(T) = \mathbb{E}\left[\sum_{t=1}^T \ell_{t,I_t}\right] - \min_{i\in[K]} \mathbb{E}\left[\sum_{t=1}^T \ell_{t,i}\right]$ によって評価する場合が多い．これは単一アームを引き続けた場合の期待累積損失の最小値と実際の期待累積損失の差を表す．ここで損失を生成する環境については主に確率的環境と敵対的環境の 2 つが主に考えられている．確率的環境では，損失ベクトル ℓ_t は $[0,1]^K$ 上の未知の確率分布 $F^K = (F_1, F_2, \ldots, F_K)$ から独立に生成される場合を考えるのに対して，敵対的環境では損失の確率分布等について一切の仮定をおかず，損失ベクトル ℓ_t はプレイヤーの過去の選択履歴 $\{I_s\}_{s=1}^{t-1}$ に応じて（敵対的に）決定されているものとする．

3 確率的環境における方策

確率的環境における問題インスタンスの難しさの指標として，最適アームとそれ以外のアームの期待損失の差 $\Delta_i = \mu_i - \mu^*$ がよく用いられている．ただし $\mu_i = \mathbb{E}_{\ell_i \sim F_i}[\ell_i]$，$i^* \in \mathrm{argmin}_{i\in[K]} \mu_i$，$\mu^* = \mu_{i^*}$ である．このとき，達成可能なリグレットの下界は $\mathrm{Regret}(T) = \Omega\left(\sum_{i:\Delta_i>0} \frac{\log T}{\Delta_i}\right)$ と表されることが知られており，これは問題依存リグレット下界とよばれる．この下界は UCB 方策やトンプソンサンプリングといった方策で達成できることが知られている．さらに，リグレットのタイトな下界は KL ダイバージェンスの最小化により表されることが知られている [1]．これを達成するための方策はいくつか知られているが，それらは下界に現れる最適化問題を明示的に推定および計算する必要があった．これに対して，トンプソンサンプリングとよばれる方策をディリクレ過程とよばれ

るノンパラメトリック手法と組み合わせることで最適化計算なしに理論限界を達成することが可能であり，本発表ではその手法 [2] について紹介する．

4　敵対的環境における方策

敵対的環境では Follow-the-Regularized-Leader (FTRL) とよばれる方策群を適切な正則化関数と併せて用いることにより $O(\sqrt{KT})$ のリグレットを達成できることが知られており，特に Tsallis エントロピーを正則化関数として用いる方策は敵対的環境での保証だけでなく確率的環境でも $O\big(\sum_{i:\Delta_i>0} \frac{\log T}{\Delta_i}\big)$ のリグレットを達成することも可能であるという強みがあり [3]，このように確率的／敵対的環境の双方に対して最適オーダーの性能をもつことは両環境最適性とよばれる．ただし，FTRL 方策では各アームの選択確率が最適化問題の解として表されており，これを各時刻で明示的に計算する必要がある．これに対して Follow-the-Perturbed-Leader (FTRL) とよばれるランダムノイズを用いた方策は一部の正則化関数に対応した FTRL を最適化計算なしに再現可能であることが知られており，$O(\sqrt{KT})$ のリグレットの達成可能性が 2019 年頃より未解決問題として考えられていた [4]．本発表ではこれを肯定的に解決しつつ両環境最適性の達成可能性を示した研究 [5] について紹介する．

さらに，一般に敵対的環境におけるほとんどの方策では，損失の不偏推定量として逆確率重み付け推定量 $\hat{\ell}_{t,i} = \frac{\ell_{t,i}}{w_{t,i}} \mathbb{1}[I_t = i]$ を用いる．ここで $w_{t,i}$ は時刻 t におけるアーム i の選択確率であり，これは FTPL 方策ではノイズ分布に応じた複雑な積分形で表される．それに対して，逆確率重み $w_{t,i}^{-1}$ をランダムな不偏推定量に置き換える幾何再サンプリングとよばれる手法が知られているが [6]，これは時刻ごとに $O(K^2)$ の計算量が必要であり大きい K では計算コストが大きくなる．これに対して本発表では条件付き幾何再サンプリングという手法を用いることにより計算量を $O(K \log K)$ に抑えつつ性能を大きく改善する研究 [7] を紹介する．

参考文献

[1] A. N. Burnetas and M. N. Katehakis, “Optimal adaptive policies for sequential allocation problems,” *Advances in Applied Mathematics*, vol. 17, no. 2, pp. 122–142, 1996.

[2] C. Riou and J. Honda, “Bandit algorithms based on Thompson sampling for bounded reward distributions,” in *ALT*, 2020, pp. 777–826.

[3] J. Zimmert and Y. Seldin, “Tsallis-INF: An optimal algorithm for stochastic and adversarial bandits,” *JMLR*, vol. 22, no. 28, pp. 1–49, 2021.

[4] B. Kim and A. Tewari, “On the optimality of perturbations in stochastic and adversarial multi-armed bandit problems,” in *NeurIPS*, vol. 32, 2019.

[5] J. Honda, S. Ito, and T. Tsuchiya, “Follow-the-perturbed-leader achieves best-of-both-worlds for bandit problems,” in *ALT*, 2023, pp. 726–754.

[6] G. Neu and G. Bartók, “Importance weighting without importance weights: An efficient algorithm for combinatorial semi-bandits,” *JMLR*, vol. 17, pp. 1–21, 2016.

[7] B. Chen, J. Lee, and J. Honda, “Geometric resampling in nearly linear time for follow-the-perturbed-leader with best-of-both-worlds guarantee in bandit problems,” in *ICML*. 2025, to appear.

