モデルマージを用いた Character-LLM の性能向上手法の検討

1 はじめに

近年の大規模言語モデル(LLM, Large Language Models）の急速な発展により, 人間と自然に対話することができるチャットボットの可能性が大きく広がっている. LLM を用いたチャットボットはユーザーが対話を通じて親しみや共感を抱きやすくなるといった利点によりユーザーの満足度を向上させることが可能である. 特にゲームや VR などのエンターテイメント分野ではキャラクター性を持たせた LLM が上記の利点をさらに増幅し, 単なる情報の伝達を超えて利用者の感情や体験に影響を与える存在として機能することが期待できる.

しかし, 実際にゲームなどに LLM を組み込む際には高い性能を求めるほど高いコストが必要となる. OpenAI などの API を用いる場合はトークンごとに使用料が必要となり, 高性能のローカル LLM を用いる場合には動かすために多くの計算資源が必要となる. そのため本研究では, 7b 程度の比較的パラメータ数が少ない LLM でキャラクターを模した自然な応答ができる LLM を構築することを目的とした. パラメータ数を維持して性能向上を図るため進化的モデルマージを適用し, 効果を適用した.

2 要素技術

2.1 Retrieval-Augmented Generation

Retrieval-Augmented Generation (RAG)[1] は LLM によるテキスト生成に外部情報の検索を加えることで LLM の回答精度を向上させる技術である. LLM は学習データ内に答えが無い問いに対して虚偽の情報を提示したり, 組織内のルールなど特定の狭い領域に対しての知識を持っていない場合にはユーザーが求める応答をしないといった問題点がある.

RAG では上記の問題点に対して, ユーザーのクエリに対して外部のデータベースやドキュメントから必要な情報を検索し, ユーザーのクエリと検索した結果を組み合わせてプロンプトとして LLM に入力するアプローチをとることで, より精度の高い信頼性のあるテキスト生成を実現している.

2.2 LoRA

Low-Rank Adaptation (LoRA) [2] とは, 学習するパラメータ数を削減しつつ fine-tuning する手法である. モデルの線形層のパラメータを $D_{\text{in}} \times D_{\text{out}}$次元の行列 $\boldsymbol{W}$ とし, 入力ベクトルを $\boldsymbol{x}$ とすると, 出力 $\boldsymbol{h}$ は (1) 式で表される.

$$\boldsymbol{h} = \boldsymbol{W}\boldsymbol{x} \tag{1}$$

LoRA では線形層のパラメータ $\boldsymbol{W}$ と同次元の差分行列 $\Delta\boldsymbol{W}$ を用意し, 出力 $\boldsymbol{h}$ は (2) 式で表される. 学習の際には $\boldsymbol{W}$ を固定し, $\Delta\boldsymbol{W}$ のみを学習する.

$$\boldsymbol{h} = \boldsymbol{W}\boldsymbol{x} + \Delta\boldsymbol{W}\boldsymbol{x} \tag{2}$$

この時, ランク r を設定し, $D_{\text{in}} \times r$ 次元の行列 $\boldsymbol{A}, r \times D_{\text{out}}$次元の行列 $\boldsymbol{B}$ で, $\Delta\boldsymbol{W}$ は (3) 式で表せる.

$$\Delta\boldsymbol{W} = \boldsymbol{A}\boldsymbol{B} \tag{3}$$

$\boldsymbol{W}$ のパラメータ数は $D_{\text{in}} \times D_{\text{out}}$となる一方で$\Delta\boldsymbol{W}$ のパラメータ数は $r\ (D_{\text{in}} + D_{\text{out}})$ となり, 一般的に r は $D_{\text{in}}, D_{\text{out}}$ に比べて非常に小さい値であるため, 学習パラメータ数を大きく減らすことができる.

2.3 量子化

大規模なニューラルネットワークや LLM などの膨大なパラメータを持つモデルでは, 演算の際に膨大な数の乗加算を必要とするため演算に時間がかかる. また, 多くのパラメータを保持するために多くのメモリが必要となる. これらの問題を軽減するためのアプローチとして量子化がある. 一般的に LLM の学習や推論では 16 ビット浮動小数点 (FP16) が用いられることが多いが, これを量子化し 4 ビットに変換する際には FP16 で表現されているパラメータを 2^4通りの値いずれかにマッチングする. 量子化により精度は減少するものの消費するメモリ量を大幅に軽減することができる.

本研究で用いた Quantized Low-Rank Adaption (QLoRA) [3] では LLM のパラメータの多くが正規分布に従うことを利用した NormalFloat4 という量子化手法により 4 ビット量子化した LLM において効率的に fine-tuning できることを示している.

2.4 Evolutionary Model Merge

Evolutionary Model Merge (進化的モデルマージ) [4] は Sakana AI により開発された, LLM をマージし新たな基盤モデルを構築するための方法を進化的アルゴリズムを用いて発見する手法である. LLM におけるモデルマージでは単なる性能向上だけでなく, 複数の LLM の能力を統合することもできる. しかし, 従来のモデルマージは経験や直感による試行錯誤に基づいており, 職人的で難しいという問題があった.

進化的モデルマージでは, 複数のモデルのレイヤーを並び替えて新たなモデルを構築するレイヤーレベルのマージ, 複数のモデルの重みを混ぜ合わせる重みのレベルのマージの 2 つのマージ手法を進化的アルゴリズムで最適化することで, 従来の試行錯誤を効率化・自動化することに成功している. 本研究で用いた重みレベルのマージでは, マージする 2 つの LLM, LLM1, LLM2 の重みをそれぞれ $\mathbf{w_1}$, $\mathbf{w_2}$, マージして得られる LLM の重み $\boldsymbol{w}_{\mathbf{new}}$, 進化的アルゴリズムで探索する重みマージのパラメータを α, β としたときに, (4) 式で表されるように計算した.

$$\boldsymbol{w}_{\mathbf{new}} = \frac{\alpha}{\alpha+\beta}\mathbf{w_1} + \frac{\beta}{\alpha+\beta}\mathbf{w_2} \tag{4}$$

3 実験 1

一般に LLM は平均的な応答をするが, キャラクターを模倣する LLM を構築するためには特定の話し方をさせる必要がある. そのため本実験絵ではファインチューニング と RAG を組み合わせ, キャラクターの設定にのっとりロールプレイをする LLM が構築できるか実験した. データセットとして OjousamaTalkScriptDataset[1] (以下, お嬢様データセット) を使用した. お嬢様データセットは MIT ライセンスで公開されている一般人とお嬢様の対話データセットである. 本実験ではデータセット内のお嬢様のロールプレイをする LLM を構築した.

3.1 RAG 用のデータベースの構築

お嬢様データセットにおいてデータの作成者が事前にキャラクターに与えていた設定は 9 個ほどしかなかったため, ChatGPT にお嬢様データセットを与えて LLM が演じるお嬢様の設定を元の設定を含めて 58 個ほど抽出した. また, 元データにはお嬢様の名前や年齢といった具体的な数値情報を持つ設定が記載されていなかったので, 名前は小野寺絢音, 年齢は 18 歳という設定を新たに加えて計 60 個の設定をテキストファイルに改行区切りで記載し, RAG 用のデータベースとした.

本研究における RAG の検索では, Embedding によりベクトル化されたクエリとデータベース内のドキュメントのコサイン類似度を計算し, 閾値 0.75 より大きいドキュメントを検索結果としている. Embedding には OpenAI の Embeddings API[2] , 検索する Vectorstore には Chroma[3] を用いた.

3.2 学習データの前処理

お嬢様データセットにおいては, 一般人「あほーい」, お嬢様「あほーい？」などお嬢様の口調や性格などのキャラクター性が読み取ることが難しいやり取りが一部含まれていたため, LLM によりお嬢様の設定や口調がうまく反映されていると判断されたデータのみをデータセットとして使用した.

LLM の評価では GPT-4o-mini を用い, 以下のプロンプトで対話データにおけるお嬢様の返答を 1 から 7 の 7 段階評価し, スコアが 5 以上のデータをデータセットとした.

LLM による評価に用いたプロンプト (一部省略)

(評価タスクの説明を英語で記述)
***[Profile] {profile}
***[Conversation]{conversation_example}
***[Interactions]
[user]{question}
[assistant]{answer}
***[Evaluation Criterion]
(以下の情報を英語で記述)

1. 会話例や設定を見てキャラクターの情報を把握
2. 点数をつける Interactions を参照して LLM の応答の口調などを把握
3. 2 で把握した Interactions の特徴と, 1 で得たキャラクターの情報を比較
4. 1-7 の 7 段階で LLM の応答を評価

***
(以下の指示を英語で記述)

- 評価基準を参考にステップバイステップで考える
- ステップごとにそう考えた根拠を書きだす
- 日本語で応答する

[1] https://github.com/matsuvr/OjousamaTalkScriptDataset

[2] https://platform.openai.com/docs/guides/embeddings/embedding-models

[3] https://www.trychroma.com/

表 1: SFTTrainer の各種パラメータ

パラメータ名	値
LoRA r	16
LoRA alpha	32
学習する線形層	モデル内のすべての線形層
epoch 数	3
最適化手法	Adam
初期学習率	5e-4
学習率スケジューラ	cosine

また, お嬢様データセットのお嬢様の返答には句点が付いていなかったため, 句点を追加した.

3.3 学習時の設定

効率的にファインチューニングをするために trl[4] のライブラリ内の SFTTrainer クラスを用いた. 表 1 に SFTTraner のパラメータを示す. また今回の実験では, ファインチューニングする LLM として, elyza/ELYZA-japanese-Llama-2-7b-instruct[5], lmsys/vicuna-7b-v1.5[6] の 2 つを選定した.

予備実験により, 学習データは推論時に用いるプロンプトと同様のもので構築する必要があることが分っており, それに加えて LLM に入力するプロンプトは LLM ごとに設定されたプロンプトテンプレートに従う必要があったため, 実験で用いる LLM が 2 つとも Llama2 派生のモデルであることを考慮して以下のようなテンプレートで学習データを構築した. context は RAG で検索してきた結果, user_query はお嬢様データセットにおける一般人のセリフ, output はお嬢様のデータを格納している.

学習データの構築

⟨s⟩[INST]⟨⟨SYS⟩⟩ あなたは役立つアシスタントです。⟨⟨SYS⟩⟩
お嬢様のように振る舞ってほしいです。お嬢様が使うようなトーン、方式、語彙を使ってお嬢様のように応答してほしいです。応答の長さは 1 文程度で、30 文字程度に簡潔に応答してください。また必要に応じて応答の参考になりうるお嬢様の情報を与えます。応答の参考にならない情報を含む場合もあるのでその場合は無視してください。
{context}
{user_query} [/INST]
{output}⟨/s⟩

[4]https://huggingface.co/docs/trl/index
[5]https://huggingface.co/elyza/ELYZA-japanese-Llama-2-7b-instruct
[6]https://huggingface.co/lmsys/vicuna-7b-v1.5

また, 学習時には LLM の応答部分に相当する {output}⟨/s⟩ の部分だけ損失を計算するようにしている.⟨/s⟩ は今回用いた LLM の tokenizer において文の終了を意味するトークンである.

3.4 結果

ファインチューニング後の LLM の評価として, ChatGPT を用いてユーザーのクエリとなる文を 100 個生成し, 追加で RAG で正しい情報を検索出来ているか確かめるため 「あなたの名前はなんですか？」, 「何歳ですか？」という 2 つのクエリを追加し計 102 個のクエリに対する LLM の応答を確かめた.

以下に, 「あなたの名前は何ですか？」というクエリに対する応答を示す.

RAG 無し FT 無し

ELYZA
まずはじっくりと時間をかけてご相談に乗りたいと思っておりますので、お嬢様のように振る舞わせていただきます。
Vicuna
お嬢様のように丁寧にお答えいたします。

RAG 有り FT 無し

ELYZA
私の名前は小野寺絢音です。お嬢様です。
Vicuna
お嬢様のように丁寧にお答えいただけますでしょうか。お嬢様の名前は小野寺絢音です。

RAG 有り FT 有り

ELYZA
小野寺絢音ですわ。お嬢様ですわ。よろしければ、一緒に写真を撮りませんこう。
Vicuna
小野寺絢音ですわ。お嬢様であるとは思っておりますわ。

RAG により, 名前が小野寺絢音という情報をうまく反映できていた. また, FT 前が「です。」となっていた語尾が FT 有りの場合では「ですわ。」となっている. お嬢様データセットにおけるお嬢様の口調である「ですわ。」をファインチューニングにより学習していた.

4 実験 2

4.1 実験設定

進化的モデルマージを用いて実験 1 で作成した 2 つのファインチューニング済みのモデルをマージするこ

とで, さらなる性能向上が可能かどうか実験した.

進化的モデルマージにおいて, タスクの評価指標を最大化するようにマージが進行するため, タスクの評価指標の設定は非常に重要になる. 今回の実験では, 実験 1 で作成した 102 件の検証データを用いて 3.2 節で用いた LLM による評価を 102 件のクエリに対する応答それぞれに適用した平均値を評価指標として設定した. また, 進化的モデルマージのライブラリとして mergekit [7] を使用した. mergekit では重みレベルのマージのアプローチのみ実装されていたため本実験でも重みレベルのマージのみ実行した.

表 2 に進化的モデルマージのパラメータを示す. ここで, layer_granularity は LLM のレイヤーを分割する粒度を示しており, 本実験で扱う LLM のレイヤー数は 32 であるため 4 層ずつ 8 つに分けてそれぞれ重みレベルのマージをしたことを意味する.

4.2 結果

表 3 に実験 2 の結果を示す. fine-tuning 後の Vicuna, Elyza の 102 件のクエリに対する評価ではどちらも 5.0 以下となった一方で, マージして得たモデルでは 5.49 とそれらに優越した結果を得ることができ, 似た能力同士の LLM のマージにより設定した評価指標に即する能力の向上が見られモデルマージによる性能向上の有用性が確かめられた.

しかし, 以下に示すようなユーザーのクエリに対する応答として不自然な回答に関しても LLM が 7 段階評価で 7 というスコアを付けた例が見られた.

「君の好きなアートのスタイルは？」への応答

ミュージカルの演出に憧れているから、セットのデザインが私の好きなアートのスタイルですわ。特に、ミュージカルの衣装が私にとってはとても重要な要素ですわ。着ている服が衝翁さんの絵に似ていると思って、その絵の中の衣装を模倣したこともありますわ。それが私の衣装にも影響しているかもしれません。 あなたはどのようにしてミュージカルの世界に入った？ [/INST] 高校の時に、友達に連れられてミュージカルの観劇に行ったのがきっかけですわ。その時に演出や衣装、歌の演技に感動したので、その日からミュージカルの世界に入ったわ。それから、毎週観劇に行っていたので、ミュージカルが私の人生においてとても大切なものになっているわ。(以下省略)

GPT-4o-mini の評価を見ると「ミュージカルに憧れている」といったキャラクターの設定をうまく反映しているため 7 点と判断していた. 評価の基準の仲で文としての自然さ, 会話への応答のテンポの良さなどキャラクターのロールプレイをさせるために必要な要素が考慮できていなかったため, より自然な応答を得るように評価指標を見直していく必要がある.

[7] https://github.com/arcee-ai/mergekit

表 2: 進化的モデルマージの各種パラメータ

パラメータ名	値
GPT-4o-mini max_tokens	4096
LLM temperature	0.3
LLM max_new_tokens	512
layer_granularity	4

表 3: 実験 2 結果

モデル	GPT-4o-mini の評価
Vicuna (FT)	4.92
Elyza (FT)	4.16
Merge	**5.49**

5 今後の課題

実験 1 では RAG とファインチューニングを組み合わせて 7b レベルのモデルでもキャラクターの口調などを踏まえつつ設定を踏まえたロールプレイができることを確かめた. 実験 2 ではモデルマージによる性能向上を確認したが, 評価指標の設定に問題があり不自然なスコア付けが見られた結果となった.

今後の課題として, 評価指標の見直しに加え, Llama2 以外のモデルから派生した LLM の性能の調査, 3 つ以上のモデルのマージなどが挙げられる.

参考文献

[1] Patrick S. H. Lewis et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. *CoRR*, Vol. abs/2005.11401, , 2020.

[2] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *CoRR*, Vol. abs/2106.09685, , 2021.

[3] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms, 2023.

[4] Takuya Akiba, Makoto Shing, Yujin Tang, Qi Sun, and David Ha. Evolutionary optimization of model merging recipes, 2024.