研究ノート

GPTとクラウドを利用したチャットボット開発における汎用的長期記憶の実装

Implementation of Generic Long-Term Memory in Chatbot Development Using GPT and Cloud Services

我妻 翔 WAGATSUMA Sho
デジタルハリウッド大学大学院 院生
Digital Hollywood University, Graduate School

池田 旭駿 IKEDA Akitoshi
デジタルハリウッド大学大学院 院生
Digital Hollywood University, Graduate School

平松 猛男 HIRAMATSU Takeo
デジタルハリウッド大学大学院 院生
Digital Hollywood University, Graduate School

白井 暁彦 SHIRAI Akihiko
デジタルハリウッド大学大学院 客員教授
Digital Hollywood University, Graduate School, Visiting Professor

GPTとはGenerative Pretrained Transformerで、ここではChatGPTをはじめとするLLM(Large Language Models: 大規模言語モデル)による推論を用いたテキスト生成である。本研究ノートではChatGPTをエンジンとした各種LINEチャットボット開発を通し、利用者との高度なコミュニケーションや体験としてのリアリティと、可用性が高いLINEチャットボットの開発手法について、大学院「クリエイティブAIラボ」において実験された各種の方法を共有する。本研究ノートでは、特に長期記憶の重要性について着目し、各実装例とユーザーの印象や課題といった方法論の共有と、その後のサービスアプリケーション開発手法のケースについて明らかにする。

1. はじめに

GPT(Generative Pretrained Transformer) は、OpenAIによる大規模言語モデル(Large Language Models: 以下LLM)のファミリーである。大規模なテキストデータのコーパスで訓練され、人間の書いたようなテキストを推論により生成することができる。2023年においてはChatGPTをはじめとするLLMの推論を用いたテキスト生成が広く知られるようになったOpenAI社によるWebインターフェースによるサービスやAPIを経由して、SlackやLINEなどのチャットボットなど広範に利用されている。デジタルハリウッド大学大学院(DHGS)「クリエイティブAIラボ」(CAIL)の活動として、実用的なLINEチャットボットサービスの開発を通し、利用者との高度なコミュニケーションや体験としてのリアリティと可用性が高いコミュニケーションボットの研究を行っている。

本研究ノートでは特に長期記憶の重要性について着目し、過去の記憶に関する研究を基に、GPT-3.5-turbo-0613のAPIにおいて実装されたFunction Callingとクラウド上のデータベースを活用し、LLMを利用した汎用的な長期記憶実現の方法論化と、その後の社会実装事例について報告する。

2. GPT利用パーソナライズチャットボットと実装ワークショップを通した実用化検討

筆者らが既に実装してきたLLMの事例としてはLINEチャットボット「絶対肯定彼氏くん」や「ひまひま女子トモちゃん」のように、暇な時間に話し相手になり、ユーザーに寄り添った応答をするチャットボットがある[1][2]。

これらのLINEチャットボットはGPT-3.5-turbo-0301を利用しておりバックエンド側にChatify[3]というノーコードサービスを利用しGPTへのプロンプトのみで実装されていた。

Chatifyによるチャットボットは高度なコミュニケーションを手軽に実装できる一方、2023年6月25日にサービス終了することが宣言されており、同様のチャットボットを実現する別の手段としてGPTをOpenAI社提供のAPI[4]やAPI Playground[5]により実現することができる。

このようなチャットボットサービスは海外では「Character.AI」のような例がある。実在の人物やSFの登場人物などに似せた応答をするチャットボットをユーザー自身が制作し、共有することができる。高度な会話ができ、かつ既存キャラクターを会話で再現するチャットボットは新たなコンテンツやサービスを生む可能性はあるが、一方で有名IPなどの非公式ペルソナをユーザーが生み出す開発はUGC (User Generated Contents；ユーザーによる生成コンテンツ)だけでなく、独自のサービスとして開発・マネジメントしたい需要もあるだろう。また「全力肯定彼氏くん」や「ひまひま女子トモちゃん」での実装の経験から、スケール面にも目を向ける必要がある。例えばGoogle Apps Script等のサーバーサービスによるシステム構築を想定した場合、処理の速度や回数に限界があり、開発の手軽さの一方でユーザー数の増加などに耐えられない可能性もある。

このような背景からCAILでは多様な受講生がデジタルコンテンツマネジメント修士(DCM修士)として、主体的にGPTチャットボットを開発できるような方法論を探求している。実験として、1冊の手順書としての技術書『LINE Botをつくってみよう APIをためして学んでしっかりわかる』を使って実際に実用的なGPT利用チャットボットが開発できるかワークショップを通して確認した[6]。学生同士の反転学習を使い参考となる書籍を追っていく形でシステム開発の経験がある受講者が経験のない受講者をリモートでの画面共有で確認しながら作業を実施した。

結果として7人の受講生全員がLINEチャットボットをAWS Cloud上でLambda機能を使い実装するところまで到達した。

3. 長期記憶の重要性

一方で、前述のLambda機能によるチャットボットは長期記憶を実装していない。Squireらによる記憶の分類は記憶を「陳述記憶」(declarative memory)と「非陳述記憶」(non-declarative memory)に区別した[7]。

長期記憶とは陳述記憶と非陳述記憶に大別される。陳述記憶は言語化可能な記憶であり、チャットボットの実装に求められる長期記

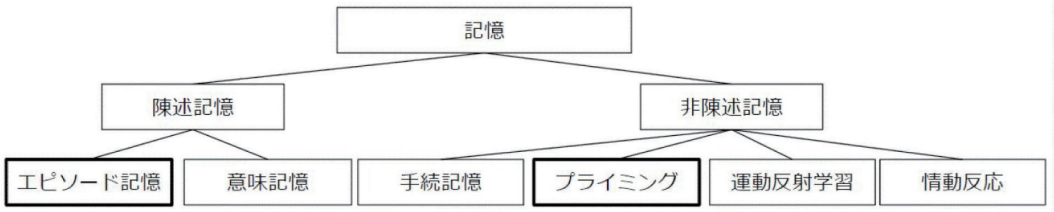

図1：Squire、湯舟らによる記憶の分類。本研究では「エピソード記憶」と「プライミング」での実装について注目する。[7]

憶はこの類にあたる。

前述のワークショップでの実装ではワンショットのチャットGPT、APIによるレスポンスを表示しているに留まり、例えばプレイヤーユーザーの名前や、今までどのような会話をしてきたかといったエピソードについては保存されていない。

白井の過去の実装例では、スクリプトプロパティに対してユーザーのIDで過去3回から5回の会話履歴をGPTのユーザーアシスタントメッセージとして与えることで、このワンショットの返答で生成においても十分なリアリズムを持った会話が実現できることが確認出来ている。

但し、この記憶が長くなればなるほどGPT側へのリクエストトークンが長くなり好ましくない。

ワークショップ内では、各受講生の想定するデジタルコンテンツやサービス開発に活かしていく上で、トレーニングの記録に利用したり、TRPGにおけるゲームマスター機能や他プレイヤーとの関係、ゲーム上のステートを管理するような機能も必要であるといった汎用的長期記憶の需要が明らかになってきた。

3-1. 理論

GPT側の機能も急速に追加・更新されており旧来のGPT-3.5-turbo-0301でのトークン制限も過去の2k、4kから16kに拡張されより長いプロンプトが利用出来るようになった。

また、Function Callingのようにプログラム上で利用する関数の引数のようなものを自然言語での会話から判定させ、ユーザーやシステムとのやり取りを確立させる方法も登場している。プライミング記憶が従来の研究では非陳述記憶として扱われているが、GPTのようなLLMの推論による補完や、トレーニングボットで求められる日々の活動に対する評価やその後の達成に対する助言などは、プライミング記憶もしくはワンショットではないLLMによる言語生成が関わるため、一概に過去の研究による定義だけで判断することは難しいだろう。

この研究ノートでは前述の『LINE Botを作ってみよう APIをためして学んでしっかりわかる』での実装例に加えて、長期記憶のプロトタイプとしてユーザーの名前とステートを記憶可能な外部記憶としてDynamoDB(NoSQLデータベースサービス)を利用して実装を試してみると、ユーザーの名前やユーザーとの会話履歴を保存でき、外部システムから参照可能なJSON形式を取ることでユーザーの苦手な食べ物を把握した献立ボットやトレーニングボットやTRPG向けのアシストボットもしくはカウンセリング医療向け自治体サービスといった多様な利用可能性が拡がった。以下は実装例である。

4. 実装例

実装例を示す前にFunction Callingについて説明したい。Function CallingはGPTへの自然言語入力に対して、事前に設定されたdescriptionに書かれた関数定義を基に、ユーザー関数とその引数となる文字列をJSON形式で返すことができる。

例えばDice(a,b)のように引数a,bが必要な関数を設定し、この関数のdiscriptionを「最大数a, デフォルト6のサイコロをb回振る」とした場合、ユーザーから自然言語入力として「サイコロを2個振って」と与えられた場合、JSON形式で「Dice(6,2)」が出力される。

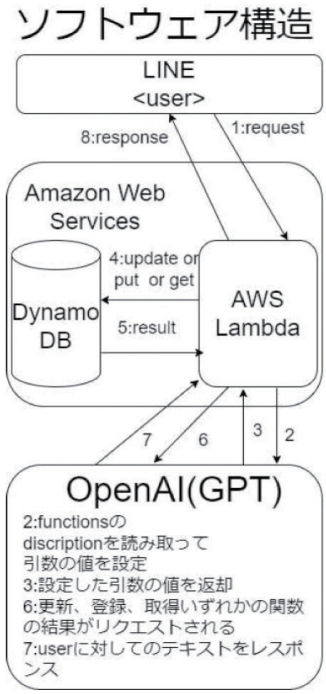

図2：クッキングママローラ（以下献立ボット）[8]の ソフトウェア構造図

筆者らの実装例、献立ボットの実装の詳細については、Websiteで公開している[9]。

本実装はGPT-3.5-turbo-0613が献立を考えるチャットボットである。ただ献立を考えるだけではなく、ユーザーの名前と嫌いな食べ物を記憶し、ユーザーから採用された献立を覚えたりそれを呼び出す時にGPT-3.5-turbo-0613が日付の範囲指定をして呼び出す機能がある。これらの記憶とその呼び出しの機能にはFunction Callingの機能が実装されている入力例として以下のリクエストを送信。

「私の名前はショウです」

このリクエストに対して、GPT-3.5-turbo-0613はupdate_user_name関数を呼び出し引数user_nameに「ショウ」を渡す。この引数を用いてDynamoDBにユーザーの名前を保存する。

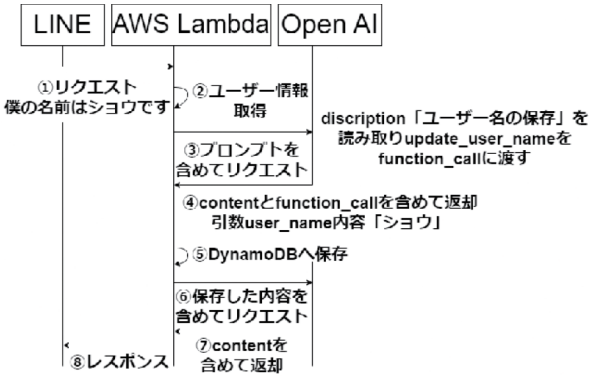

図3：update_user_name関数が返却された場合のシーケンス図

前述ボットのシーケンス図をまとめたものはWebsiteで公開している[10]。

同じ要領で以下のリクエストを送信。

「私の嫌いな食べ物はグリーンピースです」

この入力に対して、GPT-3.5-turbo-0613はupdate_hate_food関数を呼び出し引数hate_foodに「グリーンピース」を渡す。この引数を用いてDynamoDBにユーザーの嫌いな食べ物を保存する。「献立考えて」というリクエストを送信するとメインディッシュ、サイドディッシュ、スープ、デザートに至るまで献立を提案してくれる。なぜ「献立考えて」という一言で、4種類ものレシピを考えてくれるかというと、プロンプトに「あなたはフランス・リヨン出身の48歳専業主婦です。趣味でユーザーの献立を考えるクッキングママをしています」という設定が仕込まれており、その設定に従ってフランスのコース料理のような表現で献立を考えてくれているためと推測する。

一方で、プライミングによってプロンプトの設定を無視することが出来ることも確認されている。

献立ボットには最新5件の会話履歴がプロンプトに仕込まれているが、この5件が全て「メインディッシュだけ献立考えて」というリクエストとそのレスポンスだった場合、入力を「献立考えて」としてリクエストを叩いた場合にメインディッシュだけをレスポンスすることが確認されている。これまで採用した献立を参照することも可能で、入力を「昨日採用した献立は除外して献立考えて」とリクエストすると、開始日付と終了日付を昨日の日付でDynamoDBからレシピ情報を取得して昨日の献立を除外して献立を考えてくれる。

上記検証のテストエビデンス（実施日時：2023年8月20日、付録2023年8月29日）はWebsiteで公開している[11]。

5. 結果

本研究において、汎用の長期記憶を有するチャットボットが完成した。当初、時間経過に対して過去のエピソードなどをどう保存するか課題が残ると考えていたが献立ボットに関してはレシピ等を日時を指定して取得することができた。

DynamoDBへ保存する時点で日付も一緒に保存することでこの課題は解決できた。

DynamoDBに保存された情報を全て長期記憶と捉え、会話履歴はそこから取り出されたエピソード記憶、条件に従って関数を呼び出すFunction Callingは取得に関していえばエピソード記憶でもあり、手続記憶でもあると考えられる。

GPTが会話の連続的な流れを見てレスポンスの仕組み自体を変えることは、会話履歴がエピソード記憶からプライミング記憶に変わったと推定した。

エピソード記憶を整理するとユーザーの名前を記憶していること、ユーザーの嫌いな食べ物を記憶していること、採用した献立を記憶していること、最新5件の会話履歴を記憶していることが挙げられる。

プライミング記憶を整理すると、同じリクエストの会話履歴が続くと別のリクエストが送信された場合に同じ内容のレスポンスを返却することを確認できた。

GPTへのエピソード記憶とプライミング記憶の実装が成功したことで、数多くの新たな可能性が開かれた。ユーザーとの対話がより自然で人間に近いものとなり、ユーザーの満足度が向上した点である。例として、GPTをTRPGにおけるゲームマスター機能に用いる場合、この長期記憶の機能は非常に有用である。

ストーリーの進行やプレイヤーの嗜好をエピソード記憶に記録することでより魅力的な展開をプレイヤーに提供できる。

また、時間経過も表現できる。例えば、プレイヤーが前回のセッションである町を救った場合、次回その町を訪れた時には明らかな変化が見られるかもしれない。長期記憶によって、NPCとの対話が一層リアルかつ深化する。過去の対話や出来事を覚えているNPCは、プレイヤーとより深い関係を築くことができる。以上のようにTRPGのコンテキストにおいても、長期記憶の実装は極めて有望な進歩をもたらす。

以下に、差分を確認するため長期記憶の実装がなかった場合、献立ボットがどのような挙動を示すか検証してみた。

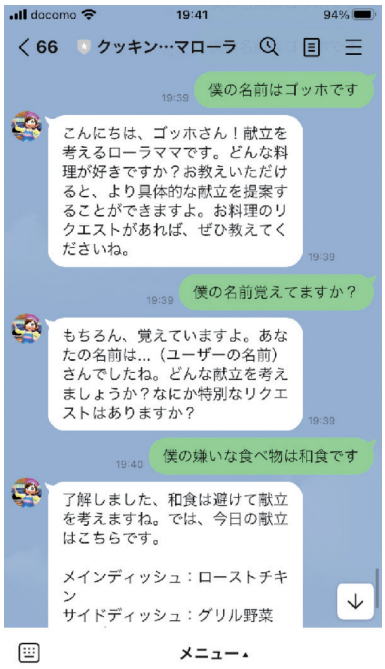

図4：ユーザーが名乗った名前を覚えているかをテストした画面

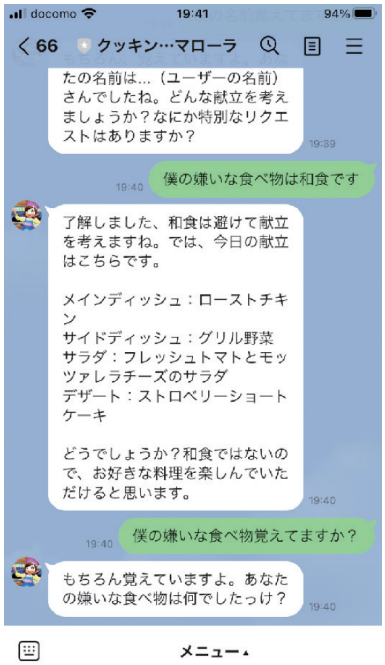

図5：ユーザーの嫌いな食べ物を覚えているかをテストした画面

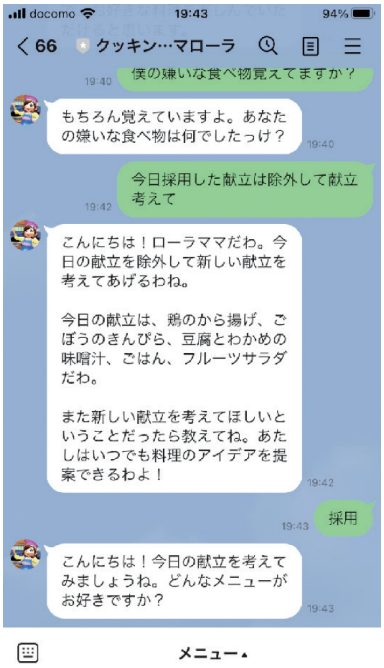

図6：採用した献立を覚えているか、採用した献立を覚えることはできるかをテストした画面

図のようにGPTはあたかもユーザー名や嫌いな食べ物を覚えているかのように返答をするが、実際には何も覚えていないことが示された。

長期記憶を実装しない場合、会話のコンテキストが成立せずサービス開発に活かすことが困難となる。

今回の研究結果でGPTに長期記憶を実装することで有意にサービス開発に活かせる可能性を示す事ができた。

5-1. 考察

Function Callingの誤り、プロンプトの正しい方法などはまだ研究すべきであり、またOpenAI側の実装が変わることで将来的な実装方法も変わるだろう。

またDynamoDBによる実装も、長期運用においてはコストが積み重なる可能性があり、サービス設計においては、より長期の価値（LTV）やそのユーザージャーニーの設計においても考慮すべき要素がある。

6. まとめ

これまで紹介したチャットボットとその実装はGPTにおける長期記憶の研究であるが、その長期記憶を活用した社会実装として、トレーニングボットのプロトタイプとしてnoteでチュートリアル解説として公開した[12]。多くの方に読まれており（2023年8月8日の公開から2023年8月29日現在売上7,700円）、一定の価値を創出することに成功したことが確認出来た。

2023年においてはChatGPTをはじめとするLLMが世間を賑わせたが、一方では毎月のChatGPTサービス利用料やAPI費用を上回る事例を作り出すことは難しい。本研究ノートでは特に長期記憶の重要性について注目し、まだ実装例が少ないFunction Callingとクラウド上のデータベースを活用し、LLMを利用した汎用的な長期記憶実現のための方法論共有をラボプロジェクトを通して速いイテレーションで行うことができた。社会実装事例については、有料ブログの売上という形でエビデンスを共有したが、このようなアプローチも新たな試みとして理解されることを期待する。

参考文献

[1] 全力肯定彼氏くん
https://note.com/o_ob/n/nb2e280ca1309
[2] ひまひま女子トモちゃん
https://line.me/R/ti/p/@t0m0?from=page&liff.referrer=https%3A%2F%2Ft.co%2F&accountId=t0m0
[3] Chatify
[4] OpenAI社提供のAPI
https://platform.openai.com/docs/models/gpt-3-5
[5] API Playground
https://platform.openai.com/playground
[6] mochikoAsTech:『LINE Botをつくってみよう APIをためして学んでしっかりわかる』,mochikoAsTech発行（2023-05-20版）,52-153頁.
[7] 湯舟英一:『長期記憶と英語教育（1）— 海馬と記憶の生成、記憶システムの分類、手続記憶と第二言語習得理論 —』,東洋大学人間科学総合研究所紀要　第7号（2007年）151頁.
[8] クッキングママローラ
https://lin.ee/GtTFe1O
[9] クッキングママローラのソースコード
https://github.com/flymywife/creativeAi
[10] クッキングママローラのシーケンス図
https://docs.google.com/presentation/d/1fN5qlk-rvWkRz_zrolBC5IJfPjaRKnPFXnO5uZwB1ac/edit#slide=id.p
[11] クッキングママローラのテストエビデンス
https://docs.google.com/presentation/d/15b6fqq1YdGYvVBt6Q4Q0duYrTg7dIVsWjdu6srUF8jU/edit#slide=id.g23d3a3a0455_0_103
[12]【この記事だけで出来る】AWSで学びながら長期記憶を実装出来るLINEBOTを作ろう～Function Callingもあります～
https://note.com/flymywife/n/nd368d341bd1a