

ご購入はこちら

池内 奈緒

LLM × Discordで作る 家電操作エージェント

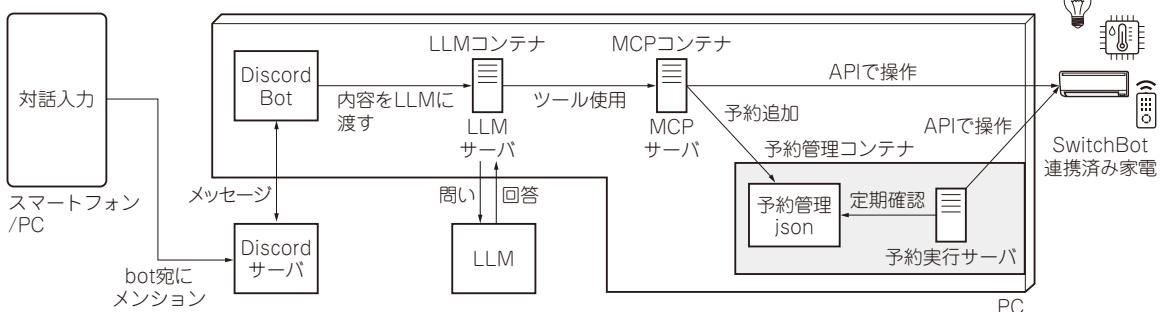


図1 本章で構築する家電操作エージェントのシステム構成

本章では、LLM (Large Language Models) と MCP (Model Context Protocol) サーバを組み合わせることで、SwitchBot^{注1}対応家電を自然言語から操作できるシステムを構築します。対話インターフェースには Discord^{注2}を採用します。LLMがユーザからの指示を受け取り、必要に応じてツールを実行し、家電の制御や予約動作を行う仕組みを実装していきます。

LLMを中心とした家電操作は近年注目されているテーマですが、複数のサーバを連携させる必要があるため、全体像の把握が難しくなりがちです。本章では、Dockerを利用したシステム構成から、各コンポーネントの実装、最終的な動作確認までを段階的に説明し、実際に動作する環境を構築するためのポイントを整理します。

以降で紹介する手順に沿って進めば、ユーザの「電気をつけて」、「明日の朝に暖房をつけて」といった指示をDiscord経由で受け取り、LLMが解釈/判断して

注1: SwitchBot(スイッチボット)は自宅の家電や設備をスマート・ホーム化できるスマート・デバイスのブランド。

注2: Discord(ディスコード)は音声通話、ビデオ通話、テキスト・チャットを組み合わせたコミュニティ作成&交流のためのプラットフォーム。

注3: 溫湿度の取得や赤外線リモコンの登録が可能なデバイス。

注4: Discordのサーバに導入して、さまざまなタスクを自動で実行するプログラムのこと。

家電を操作できるようになります。

システム構成と実装内容

今回のシステム構成は図1の通りです。構築には Dockerを使用し、LLMサーバ、MCPサーバ、予約実行サーバそれぞれでDockerコンテナを立てていきます。

実装する内容は次の通りです。

- SwitchBot連携済みの家電をLLMに操作させる。
今回はハブ^{注3}、シーリング・ライト、エアコン(赤外線リモコン)を使用
 - LLMとの対話はDiscordに作成したBotを通じて行う
 - 即時実行および予約実行が可能
- 図2のようなディレクトリ構成とします。

ステップ1: Discord Botの作成 / 実装

Discord Bot^{注4}(以降、Bot)を作成/実装します。

● 作成手順

Botを作成し、Discord上のサーバに招待します。DiscordアカウントやBotを招待可能なサーバがない場合は事前に作成します。手順は次の通りです。