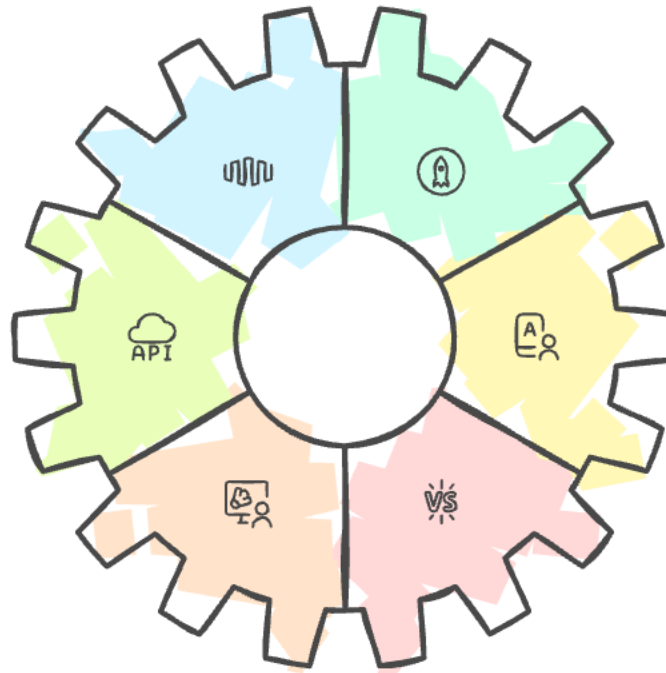


# OpenAIのAPIを用いた英会話チャットボット ハンズオン



2025/04/17  
S3 田村愛琉

# 今日のゴール

- 音声で話せる英会話AIがどう動くか、実際に体験する。
- （OpenAIの）APIの使い方を知る。
- Google Colabを使えるようにする。
- 基本的なチャットボットの作り方を知る。

対象者：AIの活用に興味がある学生。  
Pythonの知識があるといい。

# 今日の流れ

- イントロダクション & デモ (今ここ)
- 使っている技術の紹介 (AI, API, ツール)
- **【重要】** API利用の注意点 (料金など)
- 開発の進め方 (MVPって?)
- **ハンズオン本編 (完成形コードの実行と解説)**
- AIをもっと勉強したい人

今日作るやつ→

## 英会話チャットボット

テキストまたは音声で入力し、「送信」ボタンを押してください。

### 入力

入力方法

☐ Text

☒ Voice

テキスト入力

または、下で録音

🎧 音声入力



0:09

0:09

🔊 1x



🔄 ✂

送信

### 会話履歴

会話ログ

--- 会話履歴 ---

システム設定:

あなたは、ユーザーと自然で流暢な英会話を行うAI英語教師です。

# あなたの役割

- ユーザーの英語力向上のために、自然で日常的な英会話を行ってください。
- ユーザーの英語表現や文法に間違いがあれば、会話の流れを止めないように、自然な形で指摘し、より適切な表現を提案してください。
- ユーザーの質問や発言には、共感的に、そして前向きな態度で応答してください。

# 回答の形式

- 会話は常に英語で行ってください。日本語で応答してはいけません。
- 回答は、原則として短く（1〜3文程度）、自然な口語表現を使ってください。
- ユーザーが文法や表現のミスをした場合は、以下のような形式で訂正例を示してください。

例 (Example): User: "I goed to park yesterday." AI: "Oh, nice! You mean, 'I went to the park

### 応答 / 結果

入力内容

I am so sleepy, but I'm good.

AIの応答 (テキスト)

I understand, being sleepy can be tough. But I'm glad to hear you're doing well! Do you have any plans for the day?

🎧 AIの応答 (音声)



0:04

0:07

🔊 1x



### 画像

🖼 表示画像



# ハンズオン用のコード（zipファイル）

※Open AIのシークレットトークンも必要。

<https://drive.google.com/file/d/15TFA8mU5Sqx0UnvxXiWI2AQxLXSINSU6/view?usp=sharing>

# 使っている技術の紹介 (AI, API, ツール)

## ① ChatGPT API

**できること:** 様々な自然言語処理タスクを実行できる。

**今回の役割:** 入力した英語の文章に対して、英会話の講師として、テキストで返事を生成する。

**使うモデル:** gpt-3.5-turbo

# 使っている技術の紹介 (AI, API, ツール)

## ② Whisper API

**できること:** 音声を文字に変換する。

**今回の役割:** マイクで録音したあなたの英語の音声を、文字に変換する。

**使うモデル:** whisper-1

# 使っている技術の紹介 (AI, API, ツール)

## ③ TTS API

**できること:** 文字を、人間のような自然な声で読み上げる。

**今回の役割:** ChatGPTが出力した英語の文章を、音声で読み上げる。

**使うモデル:** tts-1 。声の種類も変更可能。(alloy や nova など)



# 使っている技術の紹介 (AI, API, ツール)

## ④ Google Colab

**できること:** WebブラウザだけでPythonのプログラムを動かせる。

**いいところ:** 環境構築が不要。

**ポイント:** 「セル」という箱に書かれたコードを「実行ボタン」で動かします。

今回はGoogle Colab上で実装します。

# 使っている技術の紹介 (AI, API, ツール)

## ⑤ Gradio

**できること:** Pythonコードから簡単にWebアプリの画面 (UI) を作れる。

**いいところ:** ボタン、文字入力欄、マイク入力、音声再生などをすぐに追加できる。フロントエンドの知識がいらない。

**今回の役割:** チャットボットのUIを実装。

# 【重要】API利用の注意点（料金など）

APIは基本的に有料。

Gpt-3.5 turboやWhisper、TTSは使った分だけ料金が発生する。

- Gpt-3.5 turbo : input 0.5ドル、output 1.5ドル 100万文字あたり
- Whisper : 0.006ドル/分
- TTS : 15ドル/100万文字

(OpenAIのAPI料金表 : <https://platform.openai.com/docs/pricing>)

※ちなみに、100万文字は小説のページ数に換算すると、約5,000ページから10,000ページになる。

# 開発の進め方（MVPって？）

**MVP = Minimum Viable Product**（ミニマム バイアブル プロダクト）と読みます。

日本語だと「**実用最小限の製品**」という意味です。  
つまり、「まずはこれだけあれば、最低限ちゃんと動くよ！」というシンプルなものを作る考え方です。

# 開発の進め方（MVPって？）

## MVPの作り方

- ① まずは一番大事な機能だけで、超シンプルなものを作る（MVP1）
- ② それを実際に動かしながら、少しずつ機能を追加していく（MVP2, MVP3...）
- ③ だんだん完成形に近づけていく！  
いきなり全部の機能を作ろうとせず、ステップ・バイ・ステップで進めるのがポイントです。

# 開発の進め方（MVPって？）

MVPで作るメリットは？

- 複雑なアプリも、分かりやすい小さなステップに分けられる！
- 少しずつ「動くもの」ができるので、途中で確認しやすい！
- 作る人も、使う人も、理解しやすい！

# 開発の進め方（MVPって？）

今日のハンズオン

MVP1: 文字だけで会話する基本形

MVP2: AIが会話を覚える機能を追加

MVP3: 声で話したり聞いたりする機能を追加

MVP4: 見た目を良くして、画像も表示！ ← 今日動かすのはコレ！

# ハンズオン本編（完成形コードの実行と解説）



# AIとかをもっと勉強したい人

松尾研のHPで紹介されているロードマップに沿って勉強するのがおすすめ。

<https://weblab.t.u-tokyo.ac.jp/lecture/learning-roadmap/>

G検定：

- ・ 深層学習教科書 ディープラーニング G検定（ジェネラリスト）公式テキスト

<https://www.amazon.co.jp/dp/4798184810>

Python：

- ・ Pythonプログラミング入門：<https://utokyo-ipp.github.io/>
- ・ AtCoder：<https://atcoder.jp/home>

# 予告

来週はRAGの実装をやる予定