

ExSeed アウトプット講義

～設計課題～

氏名：KugeKosuke (久家康佑)

企画・目標の設定

ユーザーゴール

- 質問に答えるという仕事を減らしたい
- いつでも状況に関係なく質問したい
- 気軽に聞ける環境が欲しい
- ミスが残った状態をなくしたい

ビジネスゴール

- 3/4 から 3/24 の3週間で完成させる
-

要件定義

機能要件

- 質問の入力欄が存在する
- 質問文を表示する
- 解答を出力し表示する
- 答えられない質問は答えられないと返ってくる
- 実際のチャットのように前にした質問も残す（チャットを閉じたら消える）

非機能要件

- 実際のチャットのような UI
 - 最低限の出力スピード
 - 質問を入力していくと質問候補が出てくる（）
 - アプリ外に漏れない
-

内部設計 ～ユーザー視点での挙動～

機能設計

- 質問の入力欄が存在する

- ・入力欄には「質問は何ですか？」という表示と入力された文字を表示と送信ボタンが存在する。

- ・入力欄をタッチするとキーボードが出現し、入力された文字を入力欄の中に表示する。

- 自分の書いた質問文が表示される
 - ・質問を入力し送信ボタンが押されると質問文がチャット欄に表示される。
 - ・何も入力されず送信ボタンが押された場合何も動作せず質問を受け取る状態のままにする。
- 解答を出力し表示する
 - ・入力された質問の回答をチャット画面に表示する。
- わからない時の対応
 - ・関連するデータが見つからない場合は「担当の人に聞きましょう」と表示する。

画面設計

- 画面のレイアウトとしてチャット画面は大きくし、質問入力画面は最低限の大きさにする
 - 質問入力画面には入力している文字を表示する欄に加えて、入力をリセットするボタンと送信するボタンを入れる。
-

内部設計 ～具体的にどう実装するか～

データ設計

- 外部とつながる API キーは一つのオブジェクトにまとめる。
- チャットデータをリストに入れて直近のデータを出力できるようにする。

モジュール設計

- ユーザーインターフェースに関しての関数
 - ・ **gradio** の **Interface** または **Blocks** を使用する。
 - ・チャットの関数をインターフェースに与えて入力を処理し、指定した内容を出力するようにする。
- ユーザーが入力欄に触れたら反応する関数
 - ・ **gradio** の関数により触れたら質問の入力待ち状態につなげる。
 - ・入力された質問は変数に入れておく。
- ストップワード除去を行う関数
 - ・テキストを渡して除去し終わったテキストを返すようにする。
 - ・NLTK ライブラリを使用する。
(ストップワード除去：文脈にはあまり影響を与えない単語を取り除く→検索効果向上)
- トークン化を行う関数
 - ・テキストを渡してトークン化し終わったテキストを返すようにする。
 - ・NLTK ライブラリを使用する。

- (トークン化：テキストを個々の単位に分割する)
- ベクトルデータベースで類似性検索を行い、その検索結果を返す関数
 - ・ストップワード除去とトークン化を行ったものを検索するものとして指定し、それを渡して検索結果を返すようにする。
 - ・API キーでベクトルデータベースに接続する。
- 質問と検索結果をチャット **GPT** にあたえ、答えを用意する関数
 - ・検索結果と質問をチャット **GPT** に与えて、その回答を返すようにする。
 - ・検索結果が空の時は情報不足なのでチャット **GPT** に渡さずに、回答を「担当の者に聞きましょう」と返すようにする。
 - ・`openai` をインポートして API キーを使用して `openai.Completion.create()` により回答を取得する。
- チャット上に質問と回答を準備するチャット関数
 - ・この関数に質問と回答が渡されたらリストの変数に質問と回答を入れそれらを出力する。
 - ・新しい質問と回答をリストに追加していくようにし、数を指定してその数だけ直近のチャットも出力するようにする。
-

プログラム設計

- 動作全体の流れ
 - (フロントエンド)
 - ユーザーが入力欄に触れる
 - ↓
 - ユーザーが質問を入力し送信ボタンを押す
 - ↓
 - (バックエンド)
 - 質問を変数①に入れる
 - ↓
 - 変数①をストップワード除去の関数に渡し、結果を変数②で受けとる
 - ↓
 - 変数②をトークン化の関数に渡し、結果を変数③で受けとる
 - ↓
 - 変数③をベクトルデータベースの関数に渡し、検索結果を変数④で受けとる
 - ↓
 - 変数①と④を **chatGPT** の関数に渡し、回答を変数⑤で受けとる
 - ↓
 - 変数①と⑤をチャット関数に渡す
 - ↓
 - もらった変数をリストの変数に入れる
 - ↓
 - リスト変数を `list[]` だとすると `list[-8]` を表示し、直近のデータも表示する



(フロントエンド)

ユーザーが質問と回答をチャット上で確認

ベクトルデータベースについて

- 選んだベクトルデータベース : **Pinecone**
 - ・選定した理由

Pinecone、Azure AI Search、ChromaDB はどれも検索機能を持っているが、ベクトルデータに対しての検索機能と正確性は **Pinecone** が優れている。また実装のしやすさの観点で見ても **Pinecone** が優れている。価格とリスクに関しては調べてみたが明確な差が見られなかった。このような理由から今回の会社が利用するなら **Pinecone** が適切だと考えた。