



5600 XP

Azure Database for PostgreSQL を使用して AI アプリを構築する

2 時間 26 分 • ラーニング パス • 7 モジュール

中級 開発者 Azure Database for PostgreSQL Azure AI サービス

このラーニング パスでは、Azure Database for PostgreSQL - フレキシブル サーバー用の Azure AI 拡張機能によって提供される Azure AI と Azure Machine Learning Services の統合を使用して、AI を利用したアプリを構築する方法について説明します。

- 本講義の内容はMicrosoft Learnの教材に準拠しています

<https://learn.microsoft.com/ja-jp/training/parts/build-ai-apps-azure-database-postgresql/>

コースの構成

モジュール1	Azure Database for PostgreSQLで生成AIの利用を開始する
モジュール2	Azure Database for PostgreSQL でセマンティック検索を有効にする
モジュール3	Azure Database for PostgreSQL を使用して文章を要約する
モジュール4	Azure Database for PostgreSQL を使用して感情分析とオピニオンマイニングを実行する
モジュール5	Azure AI Language と Azure Database for PostgreSQL を使用して分析情報を抽出する
モジュール6	Azure AI 翻訳と Azure Database for PostgreSQL を使用してテキストを翻訳する
モジュール7	Azure Database for PostgreSQL からの推論に Azure Machine Learning を使用する

モジュール1では、全体の概要を説明する。



Azure AI Services リソース



Azure OpenAI Service

モデル

→ デプロイ
deployname



Azure AI Language



Azure Machine Learning ワークスペース

モデル



仮想マシン

デプロイ

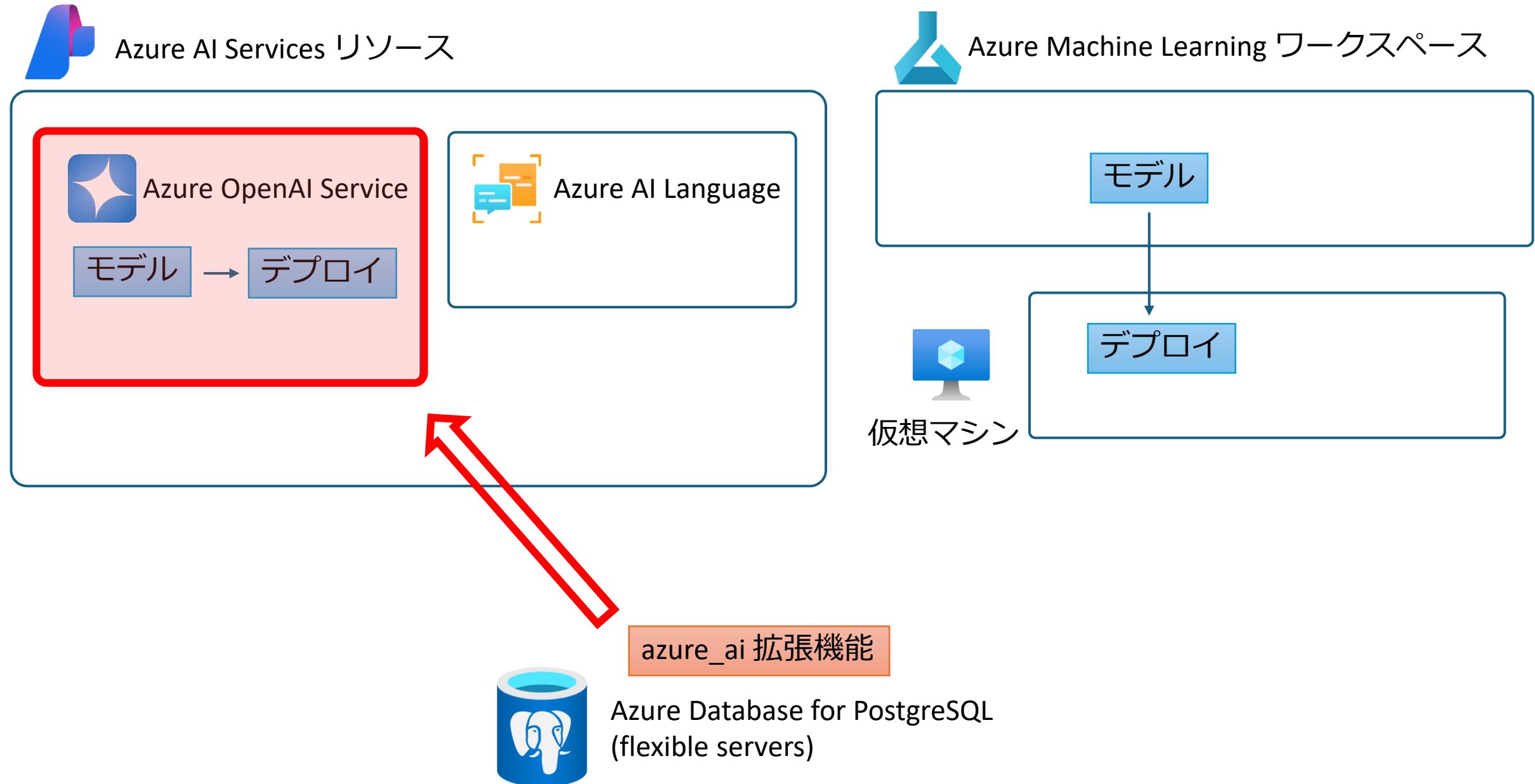
deployname



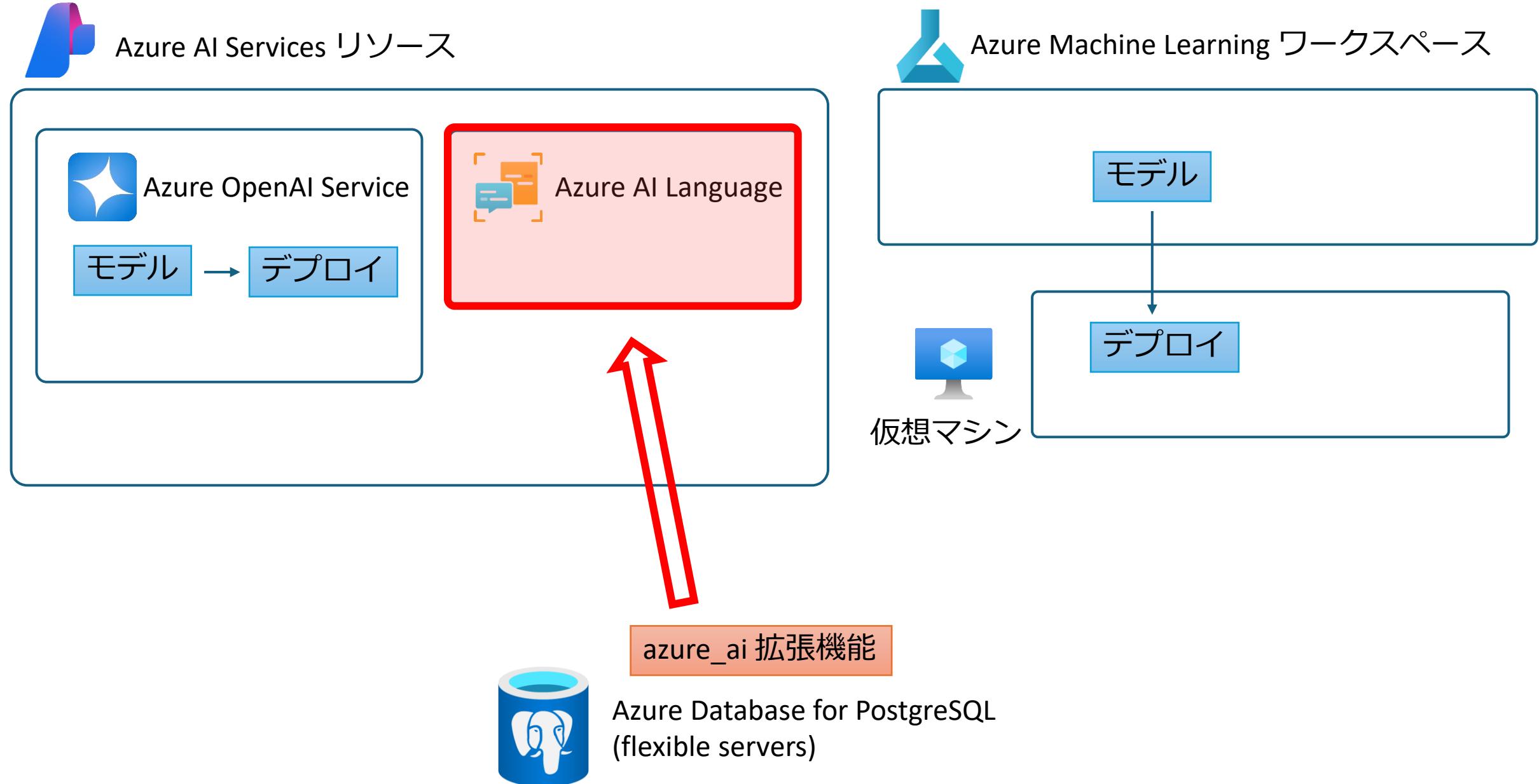
azure_ai 拡張機能

Azure Database for PostgreSQL
(flexible servers)

モジュール2では、 Azure OpenAI Service（生成AI）の利用について説明する。



モジュール3～6では、 Azure AI Language（言語）の利用について説明する。



モジュール7では、Azure Machine Learning（機械学習）の利用について説明する。



Azure AI Services リソース



Azure OpenAI Service

モデル

→ デプロイ



Azure AI Language



Azure Machine Learning ワークスペース

モデル



仮想マシン

デプロイ



Azure Database for PostgreSQL
(flexible servers)



azure_ai 拡張機能

コースの構成

モジュール1	Azure Database for PostgreSQLで生成AIの利用を開始する
モジュール2	Azure Database for PostgreSQL でセマンティック検索を有効にする
モジュール3	Azure Database for PostgreSQL を使用して文章を要約する
モジュール4	Azure Database for PostgreSQL を使用して感情分析とオピニオンマイニングを実行する
モジュール5	Azure AI Language と Azure Database for PostgreSQL を使用して分析情報を抽出する
モジュール6	Azure AI 翻訳と Azure Database for PostgreSQL を使用してテキストを翻訳する
モジュール7	Azure Database for PostgreSQL からの推論に Azure Machine Learning を使用する

モジュール1



Azure Database for PostgreSQL で生成 AI の使用を開始する

1000 XP

32 分 • モジュール • 9 ユニット

このモジュールでは、Azure Database for PostgreSQL - フレキシブル サーバーと Azure AI 拡張機能を効果的に使用して、強力な生成 AI アプリケーションを開発する方法について説明します。

開始 >

モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

Azure Database for PostgreSQLとは？

- Azureのデータベースサービスの一種
- Azure上でPostgreSQLデータベースを簡単にセットアップし、運用できる
- マネージド型サービス
 - インストールやバックアップといった多くの操作が自動化される
 - サーバー、ストレージ、ネットワークの設計や運用は不要
- 「azure_ai拡張機能」（後述）などを利用できる
- コスト: 23.4 USD/月（約3,500円/月）～※
 - 使用していない間はインスタンスを停止させることでさらに節約可

※ 最も低い性能の B1ms を選択した場合。1 USD = 145円として計算

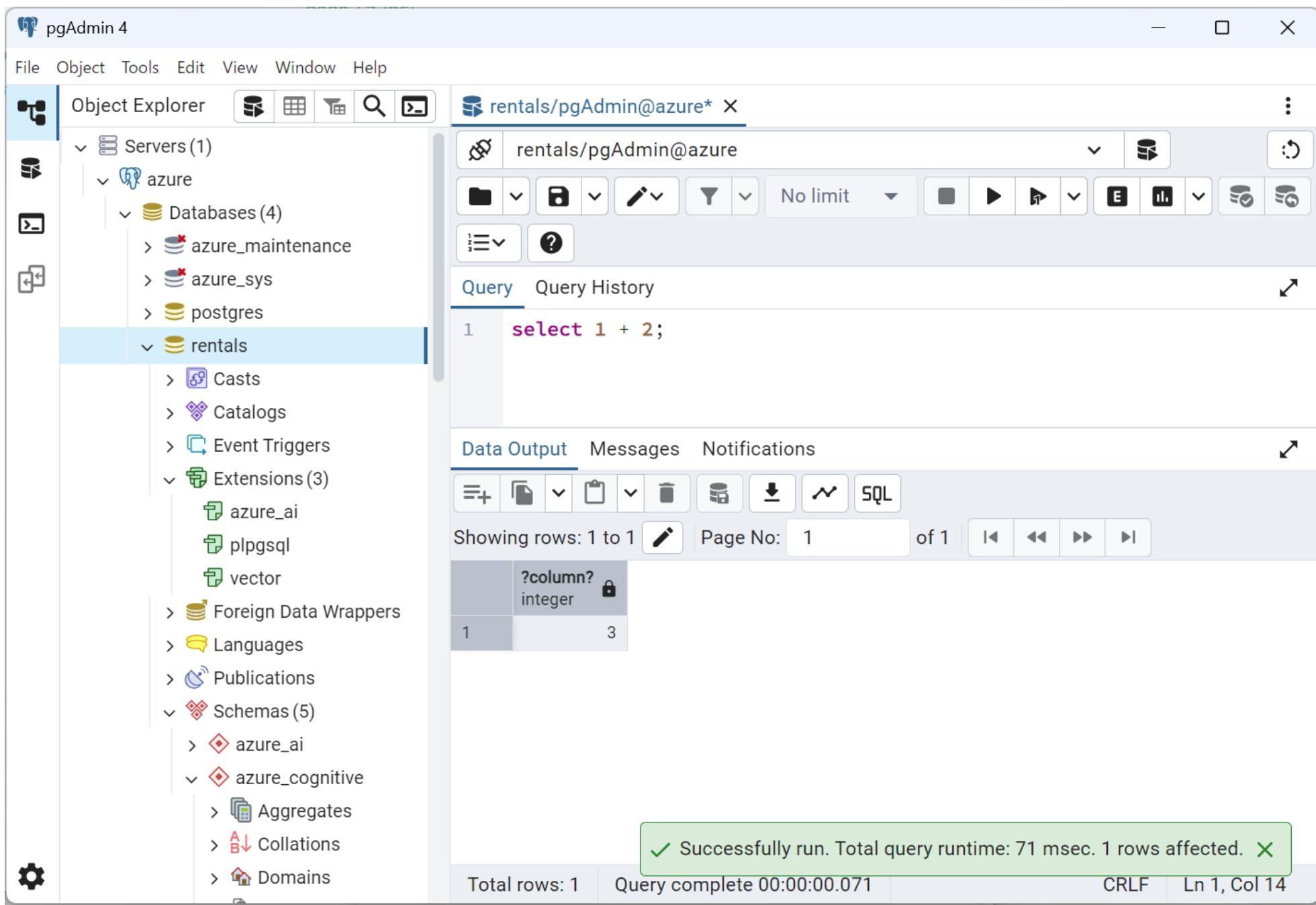
モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

Azure Database for PostgreSQLの管理ツール

- ・「**pgAdmin 4**」ツールや「**psql**」コマンドなどの標準的な PostgreSQL 管理ツールを使用して Azure Database for PostgreSQL に接続し、データベースの作成・テーブルの作成・データの操作などを実行できる。
- ・ **pgAdmin 4** のダウンロード
 - ・ <https://www.pgadmin.org/download/>
- ・ **psql** コマンドのダウンロード
 - ・ <https://www.postgresql.org/download/>

「pgAdmin 4」ツールの利用例。データベースをGUIで簡単に操作できる。



「psql」コマンドの利用例。環境変数（またはコマンドライン引数）で接続先サーバーやユーザー名を指定する。SQLをすばやく実行できる。

```
$ export PGHOST=pgsv1.postgres.database.azure.com
export PGUSER=pgAdmin
export PGPASSWORD='[REDACTED]'
export PGPORT=5432
export PGDATABASE=postgres

$ psql
psql (17.4, server 16.8)
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_256_GCM_SHA384, compression: off, ALPN: none)
Type "help" for help.

postgres=> select 1 + 2;
?column?
-----
      3
(1 row)

postgres=> █
```

モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

Azure portalなどを使用して、 Azure Database for PostgreSQLのリソースを作成する



Azure portal, ARMテンプレート、Bicepなど



Azure Database for PostgreSQL
(flexible servers)

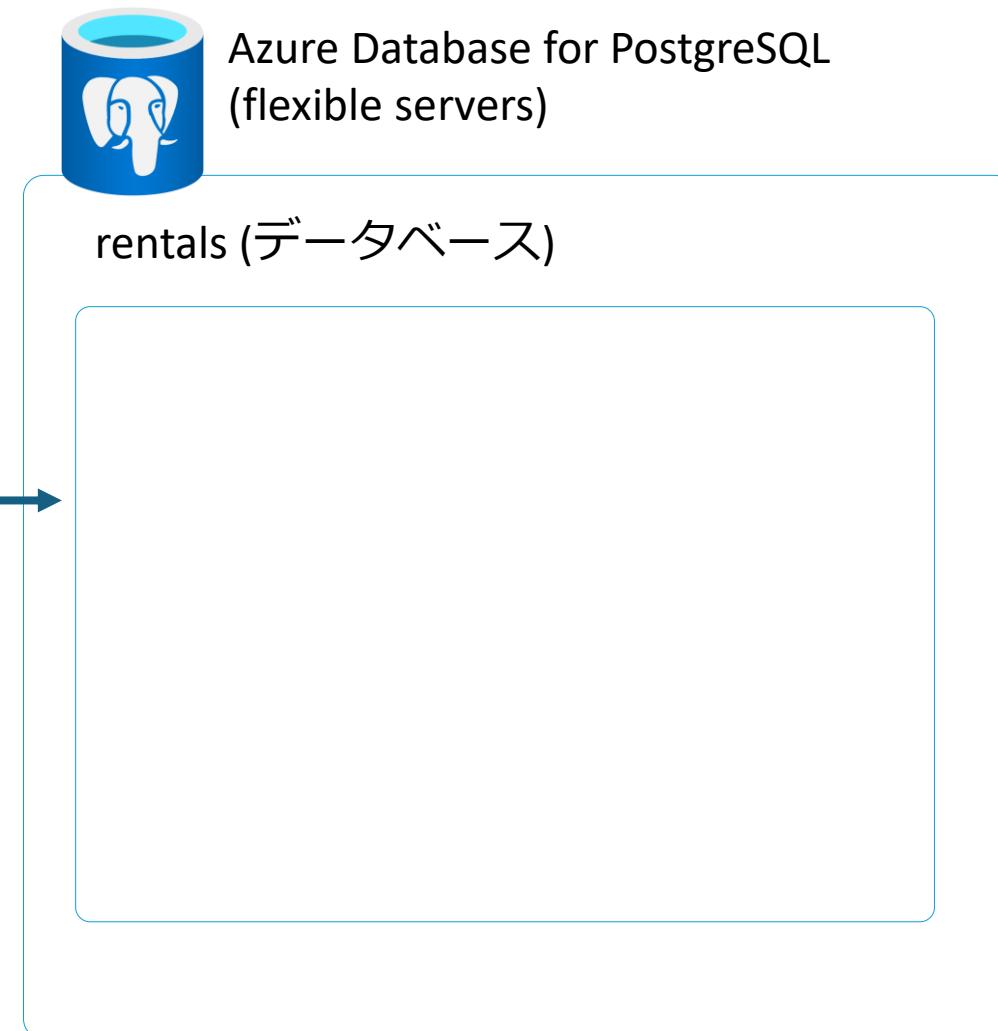


pgAdmin4やpsqlを使用してPostgreSQLに接続し、データベースやテーブルを作成する



CREATE DATABASE

pgAdmin4



pgAdmin4やpsqlを使用してPostgreSQLに接続し、データベースやテーブルを作成する



CREATE TABLE



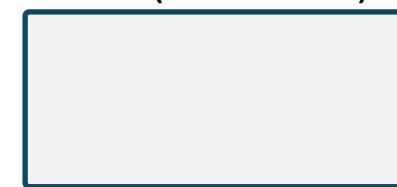
Azure Database for PostgreSQL
(flexible servers)

rentals (データベース)

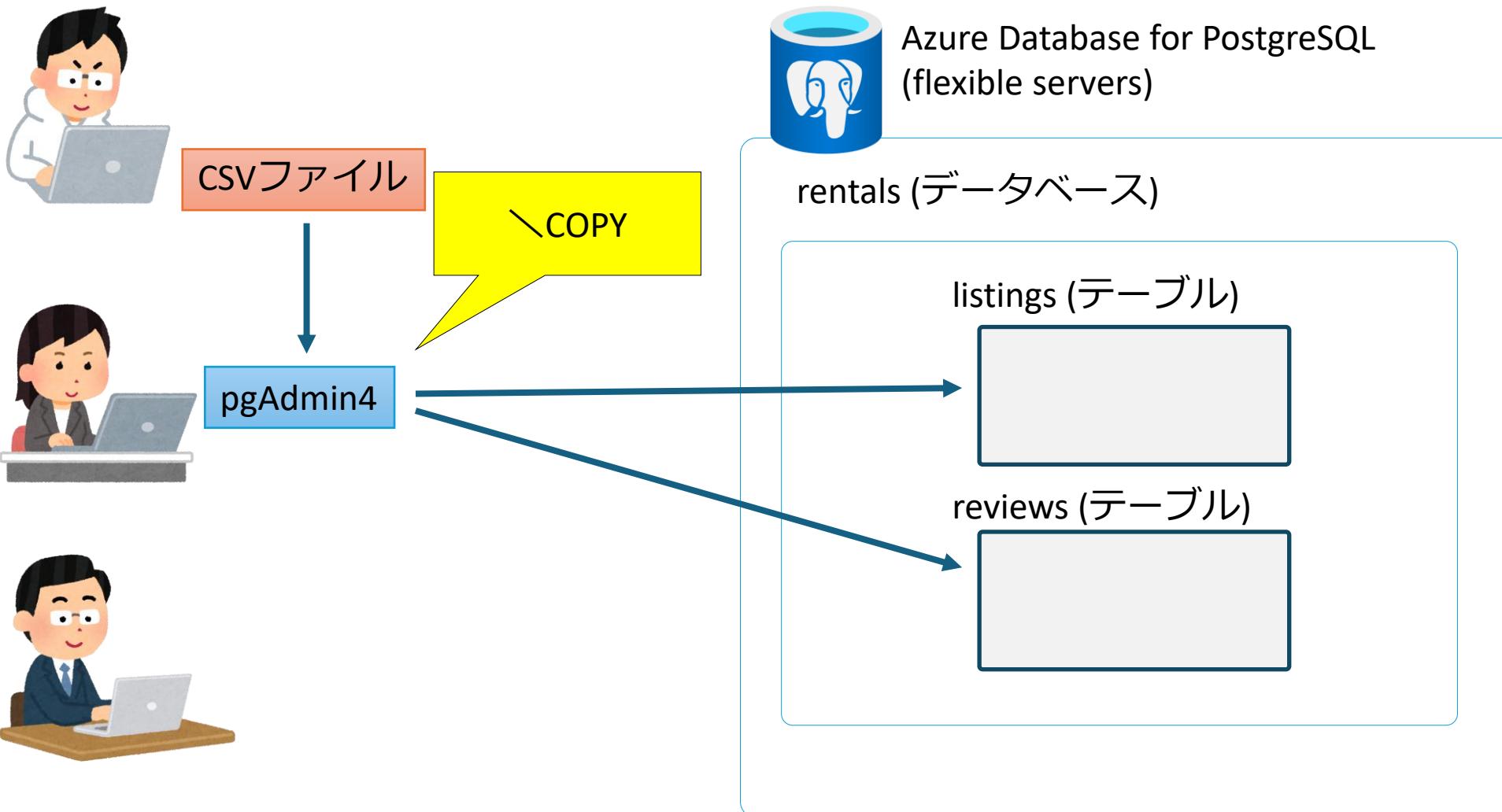
listings (テーブル)



reviews (テーブル)



pgAdmin4やpsqlの「\COPY」コマンドを使用して、csvファイルのデータをテーブルに追加できる



(参考) 「psycopg」パッケージを使用して、PythonプログラムからPostgreSQLにアクセスできる



Pythonプログラム

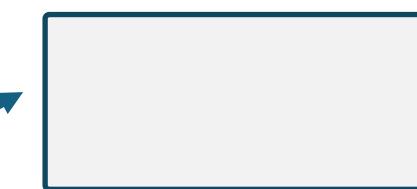
psycopg



Azure Database for PostgreSQL
(flexible servers)

rentals (データベース)

listings (テーブル)



reviews (テーブル)



モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

Azure portalでのリソースの作成

The screenshot shows the Microsoft Azure portal's search interface. The search bar at the top contains the text "postgres". Below the search bar, there are two tabs: "すべて" (All) and "サービス (6)" (Services). A red arrow points from the search bar to the "サービス (6)" tab. To the right of the tabs, there is a button labeled "増やす (5)" (Add 5). The main area is titled "サービス" (Services) and lists five items:

- PostgreSQL サーバー - Azure Arc
- PostgresSQL サーバー - Azure Arc
- Azure Cosmos DB for PostgreSQL Cluster
- Azure Database for PostgreSQL** (This item is highlighted with a red box and has a red arrow pointing to it from the bottom right)

Below the services, there is a section titled "Marketplace" with a "もっと見る" (View more) link.

[すべてのサービス >](#)

Azure Database for PostgreSQL



Contoso (MngEnvMCAP959990.onmicrosoft.com)

[+ 作成](#)

ビューの管理



更新



CSV にエクスポート



クエリを開く



タグ

[任意のフィールドのフィルター...](#)サブスクリプション 次の値と等しい **すべて**

リソース グループ



Azure Database for PostgreSQL シングル サーバーは廃止パスにあり、Azure portal での作成エクスペリエンスを作成できます。

0 件中 0 ~ 0 件のレコードを表示しています。

名前 ↑↓

リソースの種類 ↑↓



サーバー名、リージョン、サーバーのスペックなどを指定する

サブスクリプション * ⓘ

リソース グループ * ⓘ (新規) postgresqlrg ⓘ

[新規作成](#)

サーバーの詳細

場所の選択や、コンピューティングとストレージのリソースの構成など、このサーバーに必要な設定を入力します。

サーバー名 * ⓘ pgsv1 ⓘ

リージョン * ⓘ Japan East ⓘ

PostgreSQL バージョン * ⓘ 16 ⓘ

ワーカロードの種類 ⓘ 開発 運用

見積総額 USD 23.40/月

コンピューティングとストレージ ⓘ

バースト可能 B1ms ⓘ
1 個の仮想コア、2 GiB RAM、32 GiB ストレージ、P4 (120 IOPS)
Geo 冗長性 : Disabled
[サーバーの構成](#)

管理者のユーザー名とパスワードを指定する

認証

この PostgreSQL サーバーへのアクセスをサポートする認証方法を選択してください。PostgreSQL パスワード認証を有効にすると、PostgreSQL。

内に格納されているユーザー名とパスワードで認証できます。Microsoft Entra認証を有効にすると、Microsoft Entra IDに格納されているアカウントにマップされる PostgreSQL でユーザー名を作成できます。Microsoft Entra IDに対して認証されたユーザーまたはアプリケーションは、対応する期限付きパスワードとして PostgreSQL に提示されたトークンを取得できます。[詳細情報](#)

認証方法

PostgreSQL の認証のみ

Microsoft Entra 認証のみ

PostgreSQL と Microsoft Entra 認証

管理者のログイン * ⓘ

pgAdmin



パスワード * ⓘ



パスワードの確認 *



開発環境（クライアントPC）から接続ができるように、クライアントのIPアドレスを許可リストに追加する

基本 ネットワーク セキュリティ タグ 確認および作成

サーバーのネットワーク アクセスとセキュリティを構成します。

ネットワーク接続

パブリック IP アドレスを指定するか、プライベート エンドポイントを作成するか、または選択した仮想ネットワーク内から、ご使用のサーバーに接続できます。

接続方法 ① パブリック アクセス (許可された IP アドレス) とプライベート エンドポイント プライベート アクセス (VNET 統合)

パブリック アクセス

パブリック IP アドレスを使用したインターネット経由でのこのリソースへのパブリック アクセスを許可する ①

ファイアウォール規則

以下で指定した IP アドレスからのインバウンド接続は、このサーバーのポート 5432 に対して許可されます。 [詳細情報](#)

Azure 内の任意の Azure サービスにこのサーバーへのパブリック アクセスを許可する ①

+ 現在のクライアント IP アドレスを追加する () + 0.0.0.0 から 255.255.255.255 を追加する

ファイアウォール規則名 開始 IP アドレス 終了 IP アドレス

ClientIPAddress_ 

ファイアウォール規則名	開始 IP アドレス	終了 IP アドレス
ClientIPAddress_		

モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

Azure AI Servicesとは？

- ・旧名称「Azure Cognitive Services」
- ・マイクロソフトが開発した独自の機械学習モデルに基づく、すぐに使えるAI機能を提供するサービス
- ・**言語**、音声、視覚などのAI機能を提供
 - ・「azure_ai拡張機能」からは**言語**の機能のみ利用可
- ・2015/4～（当時「Project Oxford」と呼ばれていた）

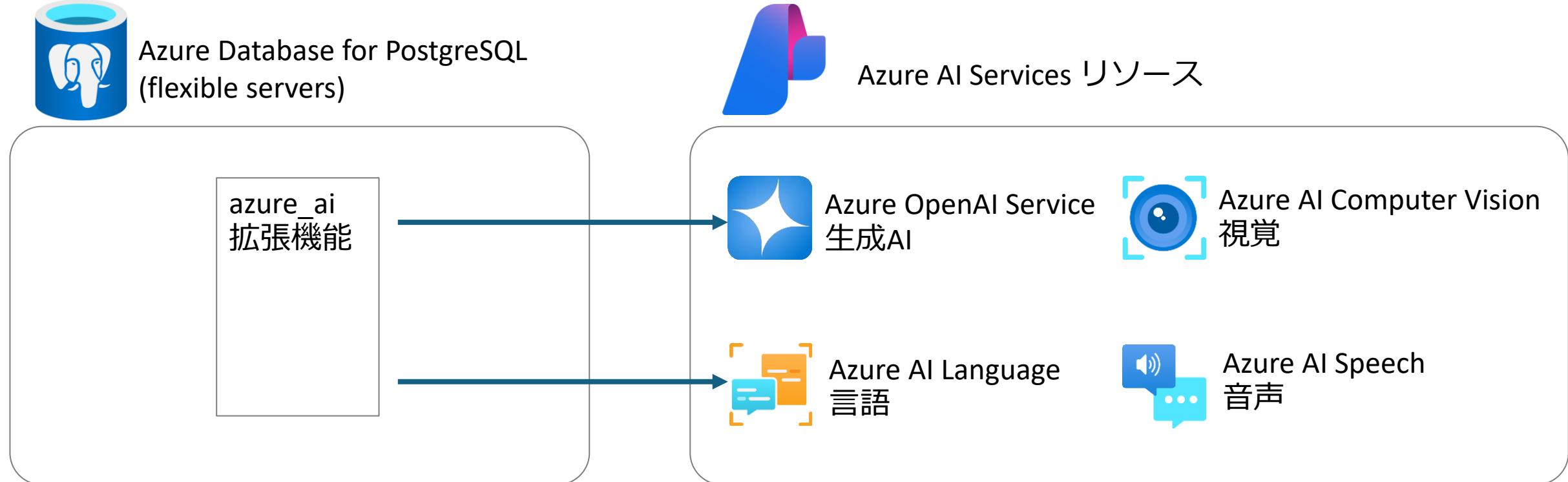
種類	名称	概要
言語	Azure AI Language	自然言語処理（文章中に含まれる個人情報の認識、翻訳など）
音声	Azure AI Speech	音声認識・音声合成など
視覚	Azure AI Computer Vision	画像処理（画像の中に映っている物体の認識など）

Azure AI Servicesとは？

- (2023/1～) Azure AI Services に、 Azure OpenAI Service (生成AI) も含まれるようになった。
 - 「azure_ai拡張機能」からは**埋め込み（ベクトルの生成）** 機能のみ利用可

種類	名称	概要
言語	Azure AI Language	自然言語処理（文章中に含まれる個人情報の認識、翻訳など）
音声	Azure AI Speech	音声認識・音声合成など
視覚	Azure AI Computer Vision	画像処理（画像の中に映っている物体の認識など）
生成AI	Azure OpenAI Service	コンテンツの生成

「Azure AI Services リソース」を作成することで、複数のAIサービスが利用可能となる。
azure_ai 拡張機能からは、生成AIの機能の一部、言語の機能の一部を利用できる。



なお、Azure AI Services リソースを作っただけでは料金は発生しない。AI機能を利用すると、利用回数（言語の場合）やトークン数（生成AIの場合）に比例した料金がかかる。

トークン

- ・生成AIモデルがテキストを処理する際の処理単位・料金単位
- ・入力されたテキストはトークンに分解される
- ・出力では、あるトークンの次に出現する確率が最も高いトークンを予測しながら、新しいテキストを生成する
 - ・"日本の首都は ?" : ? の部分に出現する確率が高いトークンは「東京」
- ・入力トークンと出力トークンの量に比例した料金がかかる
- ・<https://platform.openai.com/tokenizer> で、トークンを確認できる

こんにちは、私の名前は山田です。
Hi, my name is Hiromichi.



Tokens	Characters
18	39
こんにちは、私の名前は山田です。	
Hi, my name is Yamada.	

モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

Azure portalを使用して、Azure AI Servicesを作成する

The screenshot shows the Microsoft Azure portal interface. The title bar reads "Microsoft Azure". The search bar contains the Japanese text "リソース、サービス、ドキュメントの検索 (G+ /)". Below the search bar, the breadcrumb navigation shows "すべてのサービス > Azure AI services". The main title "Azure AI services | Azure AI services" is displayed, along with the "Azure AI services" logo. The left sidebar lists several services under "Azure AI services": "Azure AI services" (selected and highlighted with a red box), "Azure OpenAI", "AI Search", "Computer Vision", "Face API", "Custom Vision", and "音声サービス". A red arrow points from the "Azure AI services" link in the sidebar to the "作成" (Create) button in the top navigation bar, which is also highlighted with a red box. The top navigation bar also includes other buttons for "削除されたリソースの管理" (Deleted resources management), "ビューの管理" (View management), "更新" (Update), and "CSV にエクスポート" (Export to CSV). A search bar in the top right is labeled "任意のフィールドのフィルター..." (Filter by any field...). Filter options include "サブスクリプション 次の値と等しい すべて" (Subscription equals all) and "種類 次の値と" (Type equals all). The main content area displays a single record: "1 件中 1 ~ 1 件のレコードを表示しています。" (1 record displayed). The record details are: "名前" (Name) with value "ai-hub5324523343671299464", "サブタイプ" (Subtype) with value "AI Services", and "場所" (Location) with value "Sweden".



ホーム > Azure AI services | Azure AI services >

Create Azure AI services



Basics Network Tags 確認と作成

Get access to Azure OpenAI, Speech, Vision, Language, and Translator Azure AI services with a single API key. Quickly connect services together to achieve more insights into your content and easily integrate with other services.

[Learn more](#)

Project Details

サブスクリプション * ⓘ

▼

リソース グループ * ⓘ

▼

[新規作成](#)

Instance Details

Region ⓘ

▼

Name * ⓘ

✓

Pricing tier * ⓘ

▼

< 前へ

次へ

確認と作成

フィードバックの送信

モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

Azure OpenAI Serviceとは？

- OpenAI社が開発した「生成AIモデル」を利用するサービス
- 2023/1～一般提供開始
- 生成AIモデルの例
 - GPT モデル (gpt-4oなど) : テキストコンテンツの生成
 - DALL-E モデル (dall-e 3など) : 画像の生成
 - **Embedding モデル** (text-embedding-3-largeなど) : **ベクトルの生成**
- 「**azure_ai 拡張機能**」からは、**ベクトルの生成のみ利用可**

モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

モデルのデプロイ

- Azure OpenAI Servicesのモデル（text-embedding-3-largeなど）を使用するには、モデルの**デプロイ**が必要
- **デプロイ**とは、基本的には、モデルを仮想マシンに配置して、外部からそのモデルを利用できるように準備する作業を指す
- ただし、Azure OpenAI Servicesでは、モデルのデプロイに際し、**デプロイ先の仮想マシンの指定は不要**である

サービスによるデプロイ方法の違い

- Azure OpenAI Servicesでのモデルのデプロイ
 - **デプロイ先の仮想マシンの指定は不要**
 - **デプロイを行つただけでは料金はかからない**
 - デプロイを通じて利用したモデルの利用量（入出力されたトークンの量）に比例した料金が発生
- Azure Machine Learningでのモデルのデプロイ
 - **デプロイ先の仮想マシンが必要**
 - ユーザーがスペックや台数を指定して仮想マシンを作成する指示を行う
 - 仮想マシンに対して料金が発生
 - 詳しくはモジュール7で解説
- Azure AI Services（言語・音声・視覚など）
 - 「カスタムモデル」を利用する場合などを除き、基本的にデプロイ作業自体が不要

Azure OpenAI Serviceのモデルのデプロイ

- (Azure portalではなく) 「Azure AI Foundry」 という画面で、モデルのデプロイを行う。
 - Azure AI Foundry: <https://ai.azure.com/>
- 手順
 - 「モデルカタログ」で利用したいモデルを探してクリック
 - 「デプロイ」をクリック
 - オプションで「デプロイ名」を指定
 - デフォルトではモデル名と同じデプロイ名が付与される



ホーム > Azure AI services | Azure AI services >



Azure AI services



...



検索



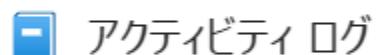
Go to Azure AI Foundry portal



削除



概要



アクティビティ ログ



アクセス制御 (IAM)



タグ



問題の診断と解決



リソース ビジュアライザー



リソース管理



キーとエンドポイント

^ 基本

リソース グループ ([移動](#))API Kind
AI Services

状態

価格レベル
Standard

アクティブ

場所

エンドポイント
[エンドポイントを表示するには、](#)サブスクリプション ([移動](#))キーの管理
[キーを管理するにはここをクリック](#)

サブスクリプション ID

カスタム AI ソリューションを構築するための適切なモデルを見つける

? ヘルプ

ホーム

概要

モデルカタログ

プレイグラウンド ^

チャット

アシスタント
プレビュー

言語 プレビュー

リアルタ

イムオー プレビュー
ディオ

イメージ

入力候補

ツール ^

業界

機能

推論タスク

タスクの微調整

ライセンス

embedding

モデル 5

text-embedding-3-large
埋め込みtext-embedding-ada-002
埋め込みtext-embedding-3-small
埋め込みCohere-embed-v3-multiling...
埋め込みCohere-embed-v3-english
埋め込み

< 前へ

次へ >



text-embedding-3-large

▷ デプロイ

ホーム

概要

モデル カタログ

プレイグラウンド ^

チャット

アシスタント
プレビュー

言語
プレビュー

リアルタ
イム オー
ディオ
プレビュー

イメージ

入力候補

詳細 既存のデプロイ ライセンス

Text-embedding-3 series models are the latest and most capable embedding model. The text-embedding-3 models offer better average multi-language retrieval performance with the MIRACL benchmark while still maintaining performance for English tasks with the MTEB benchmark.

モデル バージョン

リージョンの可用性に関する詳細情報 ▾

Sweden Central



モデル text-embedding-3-large をデプロイする

デプロイ名*

text-embedding-3-large

デプロイの種類

グローバル標準

Global Standard: 最も高い制限が適用された API 呼び出しあたりの支払。グローバル展開の種類
[の詳細をご覧ください。](#)

データはリソースの Azure 地域の外部でグローバルに処理される可能性がありますが、データストレージは AI リソースの Azure 地域に残ります。データ所在地 [の詳細をご覧ください。](#)

▼ デプロイの詳細

 カスタマイズ

モデル バージョン

1

AI リソース

ai-hub5324523343671299464

容量

150K 1 分あたりのトークン数 (TPM)

リソースの場所

Sweden Central

コンテンツの安全性

DefaultV2

バージョン更新ポリシー

モデルのバージョンは自動的にアップグレードされませ

デプロイ

取り消し

モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
 - azure_ai 拡張機能でできること
 - azure_ai 拡張機能を使うための準備
 - azure_ai 拡張機能を利用するためには必要な情報
 - azure_ai 拡張機能の「スキーマ」と「関数」
 - azure_ai 拡張機能のメリット
 - azure_ai 拡張機能ではできないこと
 - azure_ai 拡張機能の注意点
 - azure_ai 拡張機能の使い所
 - まとめ

「azure_ai 拡張機能」とは？

- Azure Database for PostgreSQL 内から、Azure AI Services、Azure OpenAI Services、Azure Machine Learningの機能を呼び出すための拡張機能
- Azure Database for PostgreSQL でのみ利用可能
- 参考: PostgreSQL自体には他にもさまざまな拡張機能がある。
 - 例: 「PostGIS」: 緯度経度などの地理情報を扱う
 - 例: 「pgvector (vector)」: ベクトルデータを扱う
 - 「azure_ai拡張機能」は多数の拡張機能の中の一つ。

モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

「azure_ai 拡張機能」でどのようなことができるか？

- ・セマンティック検索（モジュール2で解説）
- ・文章の要約（モジュール3で解説）
- ・文章の感情分析（モジュール4で解説）
- ・文章からの個人情報等の抽出（モジュール5で解説）
- ・文章の翻訳（モジュール6で解説）
- ・機械学習モデルを利用した推論※（モジュール7で解説）

※推論(inference): 構築済みのモデルにデータを送信して予測値を得ること。例えば「住所や物件条件から家賃を予測する」など。

モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

「azure_ai 拡張機能」を使うための準備

- ・サーバー（Azureリソース）レベル
 - Azure portalで、Azure Database for PostgreSQL リソースを表示
 - サーバーパラメーター「azure.extension」の「AZURE_AI」をオンにし、保存
- ・データベースレベル
 - pgAdminやpsqlでサーバーに接続
 - 「CREATE DATABASE」で「データベース」を作成し、接続
 - 「CREATE EXTENSION azure_ai」を実行（拡張機能の「作成」）
 - キー、エンドポイント、リージョンなどを設定

サーバーレベルの設定: Azure Database for PostgreSQLのサーバーパラメーターの設定を行う。

「azure.extensions」の「AZURE_AI」にチェックを入れ、保存する。

これで、このサーバーで「azure_ai拡張機能」の作成（CREATE EXTENSION）が許可される。

「azure.extensions」はDynamic パラメーターなので、設定すると即座にサーバーに反映される。

The screenshot shows the Azure portal interface for managing PostgreSQL server parameters. The left sidebar navigation bar is visible, with the 'Setting' section highlighted. The main content area displays the 'pgsv1 | サーバー パラメーター' (PostgreSQL Flexible Server Parameters) page. The 'azure.extensions' parameter is selected in the search bar. The parameter table shows the 'azURE_AI' value is set to 'Dynamic'. A red box highlights the 'azURE_AI' checkbox, which is checked. Red arrows point from the 'azURE_AI' label in the table to the checked checkbox and from the checked checkbox back to the 'azURE_AI' label. The 'azURE_AI' checkbox is also highlighted with a red border.

パラメーター名	値	パラメーターの型
azURE_AI	AZURE_AI	Dynamic
ADDRESS_STANDARDIZER	<input type="checkbox"/>	
ADDRESS_STANDARDIZER_DAT...	<input type="checkbox"/>	
AGE	<input type="checkbox"/>	
AMCHECK	<input type="checkbox"/>	
ANON	<input type="checkbox"/>	
AZURE_AI	<input checked="" type="checkbox"/>	
AZURE_STORAGE	<input type="checkbox"/>	
BLOOM	<input type="checkbox"/>	
BTREE_GIN	<input type="checkbox"/>	

データベースレベルの設定: まずデータベースを作成する

Query Query History

1 CREATE DATABASE rentals;

Data Output Messages Notifications

CREATE DATABASE

Query returned successfully in 555 msec.

そのデータベースに接続して、CREATE EXTENSION azure_ai を実行

1

CREATE EXTENSION azure_ai;

Data Output Messages Notifications

CREATE EXTENSION

Query returned successfully in 225 msec.

モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

azure_ai拡張機能を利用するためには必要な情報

- ・キー
 - ・リソースにアクセスする際の、認証のための情報
 - ・接続元のクライアントが、自分が正しいクライアントであることを証明するための情報
 - ・**機密情報であるため、漏洩させないように注意！**
- ・エンドポイント
 - ・接続先のリソースのアドレス（URL）
- ・リージョン名
 - ・リソースがデプロイされたリージョンの名前
- ・デプロイ名
 - ・モデルのデプロイ名

azure_ai拡張機能を利用するためには必要な情報

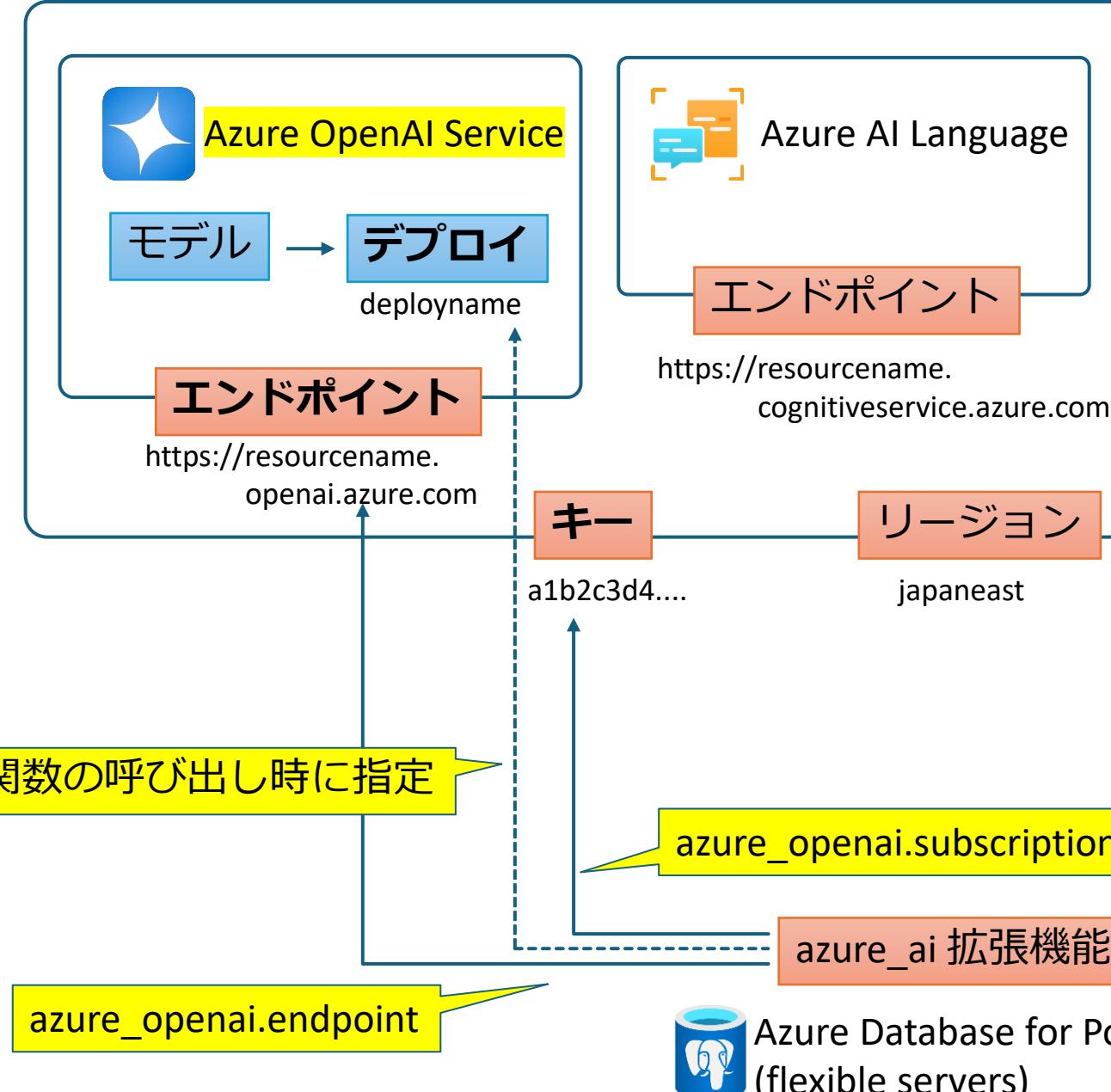
- Azure OpenAI Service の機能にアクセスする場合
 - キー
 - エンドポイント
 - デプロイ名
- Azure AI Language の機能にアクセスする場合
 - キー
 - エンドポイント
 - リージョン名
- Azure Machine Learning の機能にアクセスする場合
 - キー
 - エンドポイント
 - デプロイ名

azure_ai拡張機能を利用するためには必要な情報

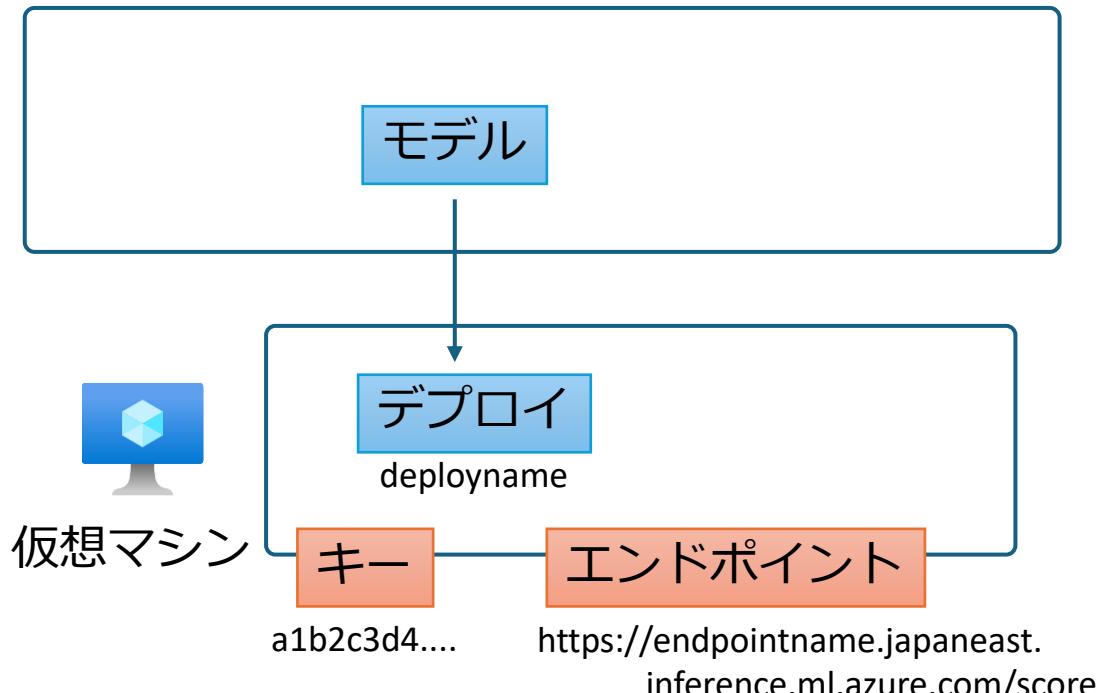
- Azure OpenAI Service の機能にアクセスする場合
 - キー
 - エンドポイント
 - デプロイ名
- Azure AI Language の機能にアクセスする場合
 - キー
 - エンドポイント
 - リージョン名
- Azure Machine Learning の機能にアクセスする場合
 - キー
 - エンドポイント
 - デプロイ名



Azure AI Services リソース



Azure Machine Learning ワークスペース



リソースのキーとエンドポイントを調べる

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+/)

ホーム > Azure AI services | Azure AI services >

Azure AI services 検索 Go to Azure AI Foundry portal 削除

概要 アクティビティ ログ アクセス制御 (IAM) タグ 問題の診断と解決 リソース ビジュアライザー リソース管理 キーとエンドポイント 暗号化 価格レベル ネットワーク

リソース グループ ([移動](#)) : [azureaifoundryrg](#)
状態 : アクティブ
場所 : Sweden Central
サブスクリプション ([移動](#)) : [REDACTED]
サブスクリプション ID : [REDACTED]
タグ ([編集](#)) : [タグの追加](#)

Develop

Keys and endpoint

You will need your key, endpoint and service region to call the service. Select the 'Keys' tab to view them.

Azure OpenAI Serviceを使用するためのキーとエンドポイントをコピー

⌚ キー 1 の再生成 ⌚ キー 2 の再生成

注意 These keys are used to access your Azure AI services API. Do not share your keys. Store them securely—for example, using Azure Key Vault. We also recommend regenerating these keys regularly. Only one key is necessary to make an API call. When regenerating the first key, you can use the second key for continued access to the service.

キーの表示

キー 1
.....
[Copy]

キー 2
.....
[Copy]

場所/地域 ⓘ
swedencentral
[Copy]

OpenAI Speech Content Safety Computer Vision ...

情報 Use the below endpoints to call into Azure OpenAI endpoints. [Learn more](#)

Language APIs https://openai.azure.com/
[Copy]

Dall-e APIs https://openai.azure.com/
[Copy]

Whisper APIs https://openai.azure.com/
[Copy]

azure_ai.set_setting(設定名, 値) を使用して、キーとエンドポイントを設定。

-- Azure OpenAI Serviceのキー

SELECT azure_ai.set_setting('azure_openai.subscription_key', '

-- Azure OpenAI Serviceのエンドポイント

SELECT azure_ai.set_setting('azure_openai.endpoint', '

キー

エンドポイント

デプロイ名は拡張機能の関数の呼び出しの際に指定。

SELECT azure_openai.create_embeddings('text-embedding-3-large', 'こんにちは')

デプロイ名

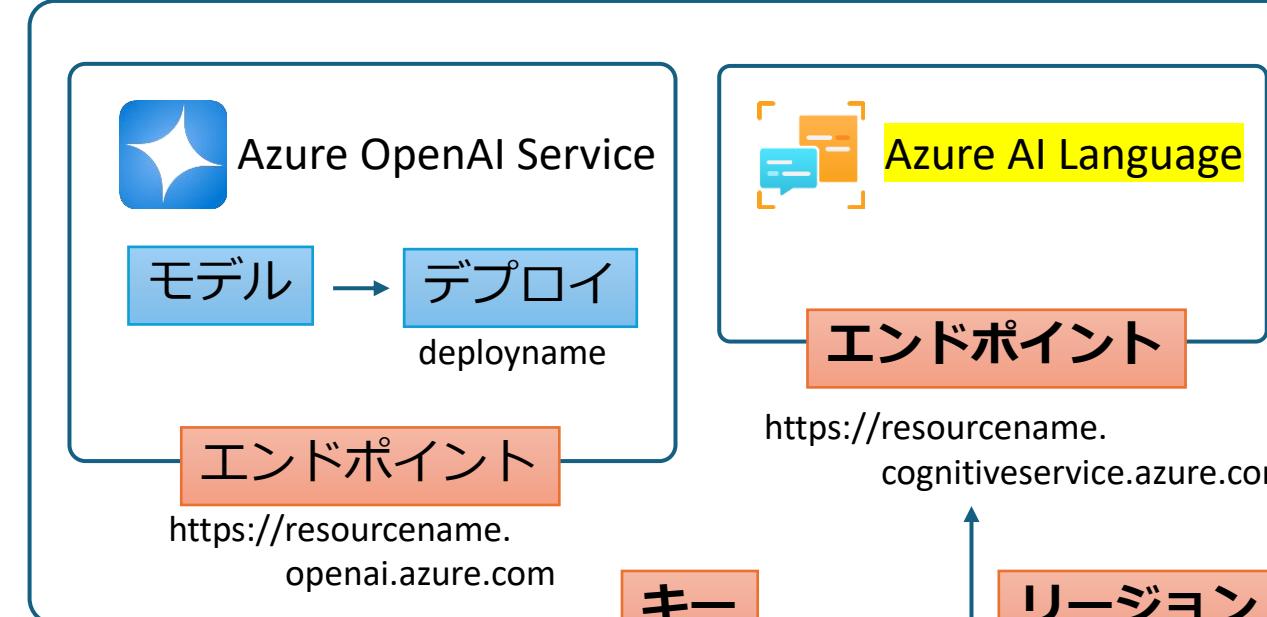
※ 「デプロイ」の詳細についてはモジュール2で解説。

azure_ai拡張機能を利用するためには必要な情報

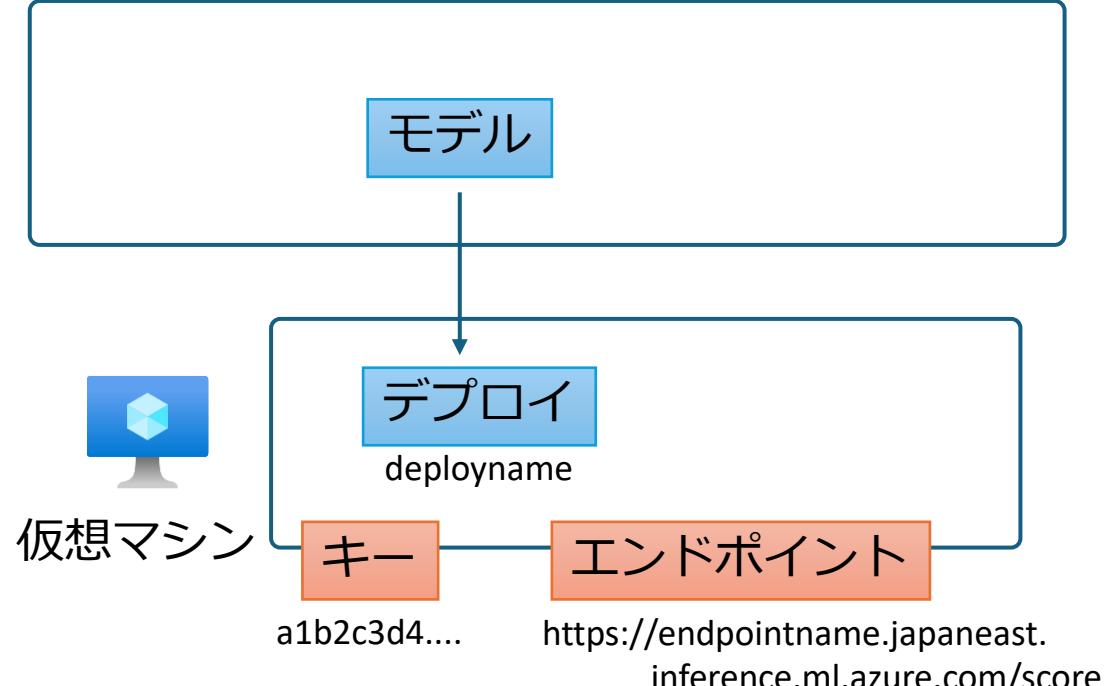
- Azure OpenAI Service の機能にアクセスする場合
 - キー
 - エンドポイント
 - デプロイ名
- Azure AI Language の機能にアクセスする場合
 - キー
 - エンドポイント
 - リージョン名
- Azure Machine Learning の機能にアクセスする場合
 - キー
 - エンドポイント
 - デプロイ名



Azure AI Services リソース



Azure Machine Learning ワークスペース



Azure Database for PostgreSQL
(flexible servers)

リソースのキーとエンドポイントを調べる

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+/)

ホーム > Azure AI services | Azure AI services >

Azure AI services 検索 Go to Azure AI Foundry portal 削除

概要 アクティビティ ログ アクセス制御 (IAM) タグ 問題の診断と解決 リソース ビジュアライザー リソース管理 キーとエンドポイント 暗号化 価格レベル ネットワーク

リソース グループ ([移動](#)) : [azureaifoundryrg](#)
状態 : アクティブ
場所 : Sweden Central
サブスクリプション ([移動](#)) : [REDACTED]
サブスクリプション ID : [REDACTED]
タグ ([編集](#)) : [タグの追加](#)

Develop

Keys and endpoint

You will need your key, endpoint and service region to call the service. Select the 'Keys' tab to view them.

「Language」へ移動

The screenshot shows the Azure portal interface for configuring an endpoint. At the top, there are two placeholder fields labeled 'ホーク' (Hawk) with icons for cloning or deleting. Below them is a '場所/地域' (Region) field set to 'swedencentral'. Under the 'OpenAI' tab, there is a note: 'Use the below endpoints to call into Azure OpenAI endpoints.' followed by a 'Learn more' link. It lists three API types with their respective URLs:

API Type	Endpoint URL
Language APIs	https://openai.azure.com
Dall-e APIs	https://openai.azure.com
Whisper APIs	https://openai.azure.com

A red arrow points from the 'Language' tab in the dropdown menu to the 'Language' endpoint URL. The dropdown menu also includes other services: Speech, Content Safety, Computer Vision, Content Understanding, Translator, and Document Intelligence.

Azure AI Languageを使用するためのエンドポイントをコピー

⌚ キー 1 の再生成 ⌚ キー 2 の再生成

重要 These keys are used to access your Azure AI services API. Do not share your keys. Store them securely—for example, using Azure Key Vault. We also recommend regenerating these keys regularly. Only one key is necessary to make an API call. When regenerating the first key, you can use the second key for continued access to the service.

キーの表示

キー 1
.....


キー 2
.....


場所/地域 (i)
swedencentral


Content Safety Computer Vision Content Understanding **Language** ...

重要 Use the below endpoint to call into the language APIs.

Language <https://cognitiveservices.azure.com/>


azure_ai.set_setting(設定名, 値) を使用して、キーとエンドポイント、リージョンを設定。

```
-- Azure AI Languageのキー  
SELECT azure_ai.set_setting('azure_cognitive.subscription_key', 'XXXXXXXXXX');  
-- Azure AI Languageのエンドポイント  
SELECT azure_ai.set_setting('azure_cognitive.endpoint', 'XXXXXXXXXX');  
-- Azure AI Languageのリージョン  
SELECT azure_ai.set_setting('azure_cognitive.region', 'XXXXXXXXXX');'');
```

キー

エンドポイント

リージョン

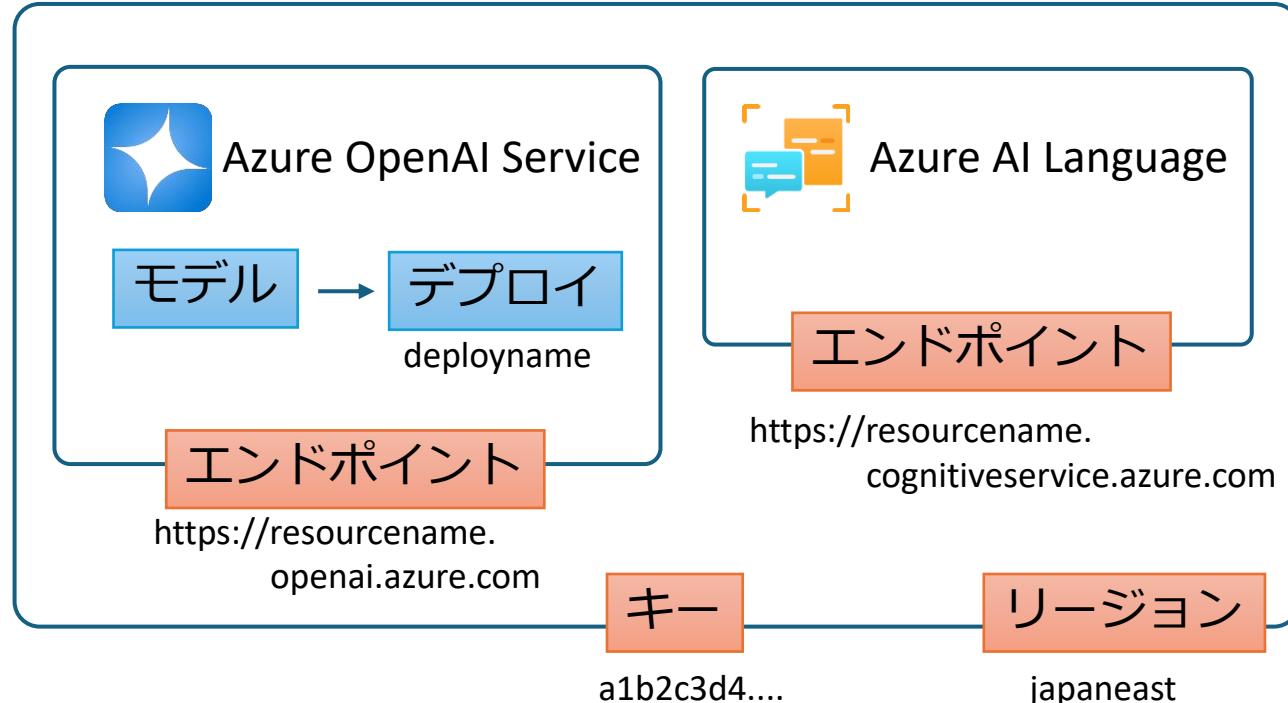
azure_ai拡張機能を利用するためには必要な情報

- Azure OpenAI Service の機能にアクセスする場合
 - キー
 - エンドポイント
 - デプロイ名
- Azure AI Language の機能にアクセスする場合
 - キー
 - エンドポイント
 - リージョン名
- Azure Machine Learning の機能にアクセスする場合
 - キー
 - エンドポイント
 - デプロイ名

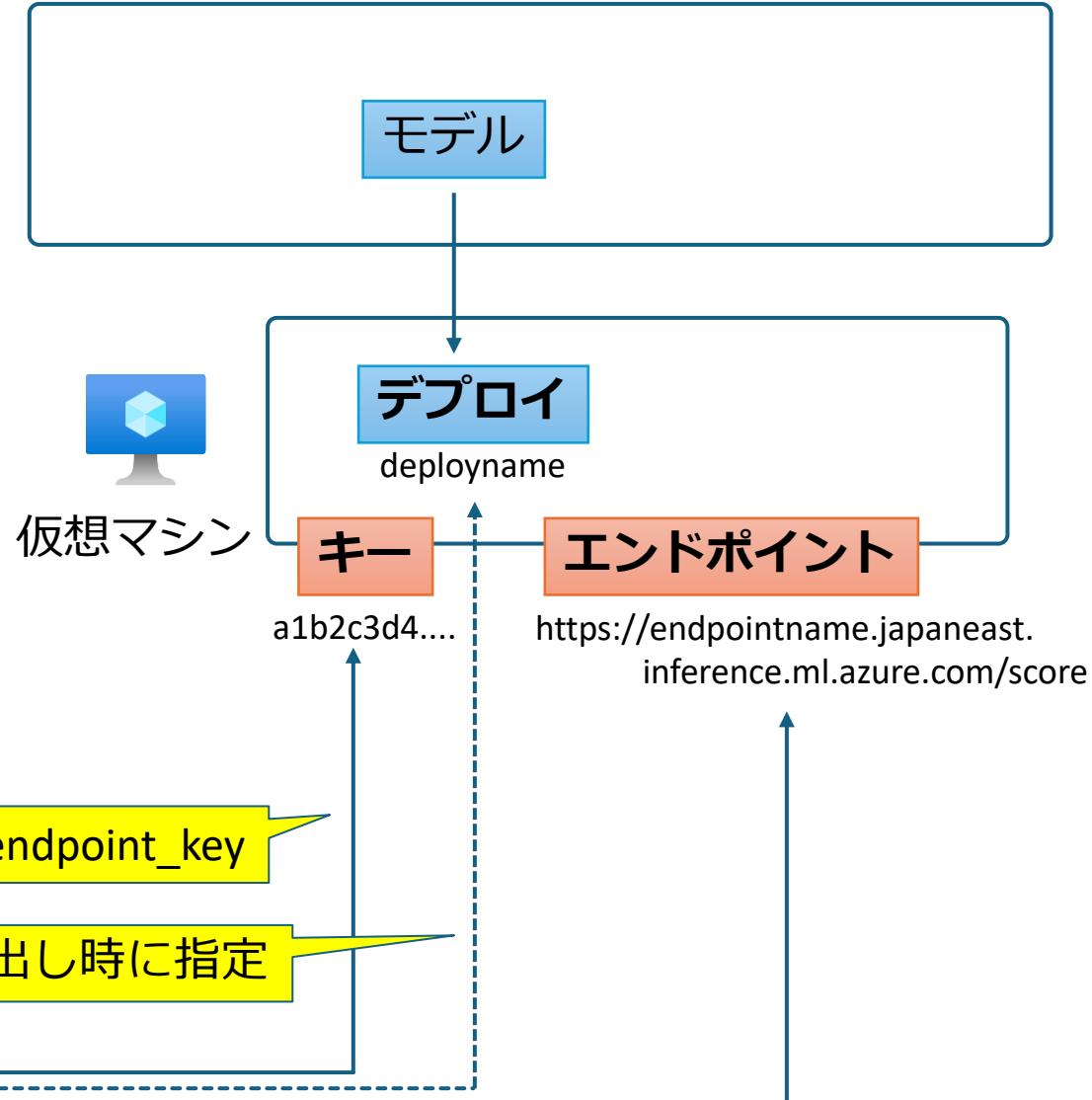
Azure Machine Learningについて
詳しくはモジュール7で解説。



Azure AI Services リソース



Azure Machine Learning ワークスペース



Azure Database for PostgreSQL
(flexible servers)

azure_ml.scoring_endpoint

`azure_ai.set_setting(設定名, 値)` を使用して、キーとエンドポイントを設定。
デプロイ名は関数の呼び出しの際に指定。

エンドポイント

```
SELECT azure_ai.set_setting('azure_ml.scoring_endpoint', 'https://i...jp...a...')
```

```
SELECT azure_ai.set_setting('azure_ml.endpoint_key', '...')
```

キー

```
23  
24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42  
  SELECT azure_ml.inference({  
    "input_data": {  
      "columns": [  
        "host_is_superhost",  
        "host_has_profile_pic",  
        "host_identity_verified",  
        "neighbourhood_group_cleansed",  
        "zipcode",  
        "property_type",  
        "room_type",  
        "accommodates",  
        "bathrooms",  
        "bedrooms",  
        "beds"  
      ],  
      "index": [0],  
      "data": [[0, 0, 0, "Central Area", "98122", "House", "Entire home/apt", 4, 1.5, 3, 3]  
    }  
  }, deployment_name=>'rentallistings-1')
```

デプロイ名

モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

スキーマと関数

- ・「azure_ai 拡張機能」を作成（CREATE EXTENSION）すると、「azure_ai」「azure_openai」「azure_cognitive」「azure_ml」の4つのスキーマが作成される
- ・スキーマに関数が含まれる
 - ・`azure_openai.create_embeddings` など
- ・関数を呼び出すことでAI機能を実行できる
 - ・`SELECT azure_openai.create_embeddings('デプロイ名', 'テキスト')` など

azure_ai 拡張機能の関数



Azure Database for PostgreSQL
(flexible servers)

rentals (データベース)

azure_ai 拡張機能

azure_ai.get_setting

azure_ai.set_setting

azure_openai.create_embeddings

azure_cognitive.summarize_extractive

azure_cognitive.summarize_abstractive

azure_cognitive.analyze_sentiment

azure_cognitive.extract_key_phrases

azure_cognitive.recognize_entities

azure_cognitive.recognize_pii_entities

azure_cognitive.translate

azure_ml.inference



Azure OpenAI Service

埋め込みモデル
text-embedding-3-large



Azure AI Language

抽出要約
extractive summarization

抽象要約
abstractive summarization

感情分析
sentiment analysis

キーフレーズ抽出
key phrase extraction

エンティティ認識
named entity recognition

個人識別情報認識
pii entity recognition



Azure Machine Learning

カスタム機械学習モデル

azure_openai スキーマに含まれる create_embeddings 関数の実行例

```
47    SELECT azure_openai.create_embeddings('text-embedding-3-large', 'こんにちは')
```

Data Output Messages Notifications



Showing rows: 1

	create_embeddings
real	
1	{0.0053383727,-0.0070507475,-0.018627813,0.018415974,0.017568612,0.033131804,-0.020845074,0.038328953,-0.00171}

※この関数について詳しくはモジュール2で解説。

モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

「azure_ai 拡張機能」のメリット

- SQLだけで簡単にAI機能を試せる・活用できる
 - AI機能を活用するシステムをすばやく実装できる（PoCなど）
 - データエンジニア・データアナリストがAI機能を活用しやすい
- クライアント側のロジックが不要
 - たとえばDBからデータを取り出す→データにAI処理を実行→結果をDBに格納、といったクライアント側のロジックが不要になる
 - トリガーを使用すれば、テーブルへのデータ追加・更新時に、そのデータに対するAI処理を同時に実行できる
 - 既存のシステムを改修せずDB側でAI機能を追加できる
 - DBに接続する複数のシステムで、AI機能を共通化できる
- エラー時のリトライ処理が組み込まれている
 - AI処理が「再試行可能なエラー」（レート制限など）で失敗した際、Nミリ秒後に再試行する・再試行をN回繰り返す、といった指定が可能
 - デフォルト設定は再試行なし

「azure_ai 拡張機能」のメリット

- SQLだけで簡単にAI機能を試せる・活用できる
 - AI機能を活用するシステムをすばやく実装できる（PoCなど）
 - データエンジニア・データアナリストがAI機能を活用しやすい
- クライアント側のロジックが不要
 - たとえばDBからデータを取り出す→データにAI処理を実行→結果をDBに格納、といったクライアント側のロジックが不要になる
 - トリガーを使用すれば、テーブルへのデータ追加・更新時に、そのデータに対するAI処理を同時に実行できる
 - 既存のシステムを改修せずDB側でAI機能を追加できる
 - DBに接続する複数のシステムで、AI機能を共通化できる
- エラー時のリトライ処理が組み込まれている
 - AI処理が「再試行可能なエラー」（レート制限など）で失敗した際、Nミリ秒後に再試行する・再試行をN回繰り返す、といった指定が可能
 - デフォルト設定は再試行なし

登場人物



Pythonが得意

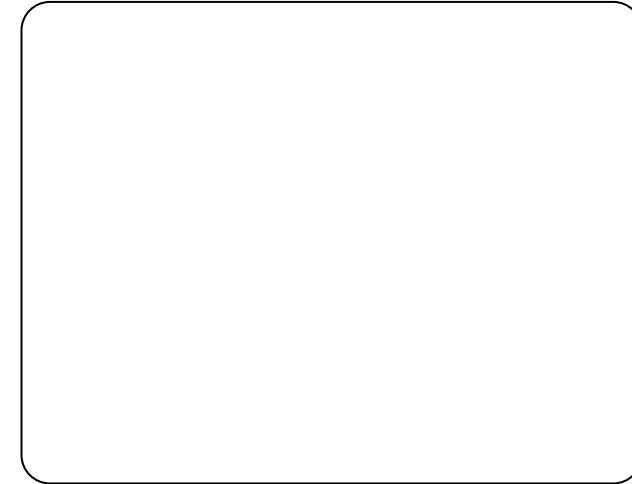


SQLが得意

例: データベースを利用するシステムがあるとする。 「text_ja」 には日本語のテキストが記録されている。



Azure 仮想マシン



Azure Database for PostgreSQL
(flexible servers)

id	text_ja
1	こんにちは
2	お元気ですか

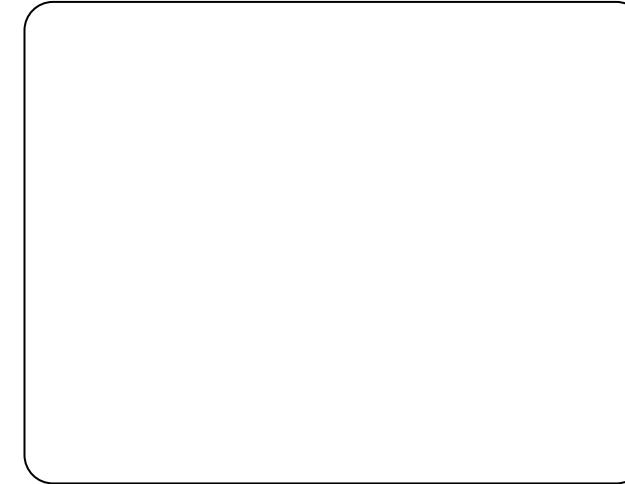


Azure AI Language

データベースに新しい列「text_en」が追加された。「text_ja」列の値を英語に翻訳して書き込みたい。



Azure 仮想マシン



Azure Database for PostgreSQL
(flexible servers)

id	text_ja	text_en
1	こんにちは	
2	お元気ですか	

翻訳
translation



Azure AI Language



たとえばPythonなどでプログラムを開発し、Azure仮想マシンにデプロイする。



Visual Studio Code



Azure 仮想マシン



Python



Azure Database for PostgreSQL
(flexible servers)

id	text_ja	text_en
1	こんにちは	
2	お元気ですか	

翻訳
translation



Azure AI Language

プログラムはまず「text_ja」列からテキストを取り出す。



Azure 仮想マシン



Python

"こんにちは"



Azure Database for PostgreSQL
(flexible servers)

id	text_ja	text_en
1	こんにちは	
2	お元気ですか	

翻訳
translation



Azure AI Language

プログラムは次にそのテキストをAzure AI Languageに送信して翻訳文を得る



Azure 仮想マシン



Python

"こんにちは"
"Hello"



Azure AI Language



Azure Database for PostgreSQL
(flexible servers)

id	text_ja	text_en
1	こんにちは	
2	お元気ですか	

プログラムは得られた翻訳文をtext_en列に書き込む（UPDATEする）。この処理をループで繰り返す。



Azure 仮想マシン



Python

"こんにちは"
"Hello"



Azure Database for PostgreSQL
(flexible servers)

id	text_ja	text_en
1	こんにちは	Hello
2	お元気ですか	



Azure AI Language

azure_ai 拡張機能を使用する場合は、プログラムを開発する必要はない。まず azure_ai 拡張機能を追加する。



`CREATE EXTENSION azure_ai;`



Azure Database for PostgreSQL
(flexible servers)

azure_ai 拡張機能

id	text_ja	text_en
1	こんにちは	
2	お元気ですか	

翻訳
translation



Azure AI Language

次に、azure_ai に含まれる「translate」関数を使用して、UPDATE文一発ですべての行の翻訳を行える。



```
UPDATE table1
SET text_en = (
    SELECT value
    FROM jsonb_each_text(
        (azure_cognitive.translate(text_ja, 'en'))->0
    )
    WHERE key = 'text'
);
```



Azure Database for PostgreSQL
(flexible servers)

azure_ai 拡張機能

id	text_ja	text_en
1	こんにちは	Hello
2	お元気ですか	How are you

翻訳
translation



Azure AI Language

azure_ai 拡張機能のメリット

- Pythonなどのコードを開発する必要がない
- SQLの実行だけで、AIを使った処理を実現できる
- 手間がかからない

SQLでやれる仕事なら
SQLでやったほうが
手っ取り早そう！



azure_ai 拡張機能 (SQL) のデメリット

- 細かい制御が難しい可能性

- たとえばデータが何千万件ある場合、SQLのUPDATE文で一気にそれを処理しようとすると、Azure AI Languageの呼び出しが短時間に大量に発生する
- Azure AI Languageの翻訳機能のレート制限（1時間あたり4000万文字まで）を超えるとエラーが起きてしまう
- この場合「レート制限が発生したら1時間待ち、続きから処理を再開する」あるいは「複数のAzure AI Languageリソースを作成し翻訳処理を分散させる」といったロジックを実装する必要がある。

そういう制御はたぶん
Pythonのほうが得意だね！



モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

「azure_ai 拡張機能」ではできないこと

- Azure AI Services
 - 視覚 (Computer Vision) の機能の利用 (画像の説明文の作成など)
 - 音声 (Speech) の機能の利用 (読み上げ、音声認識など)
- Azure OpenAI Service
 - GPT を使用したテキストコンテンツの生成など
 - DALL-E を使用した画像コンテンツの生成など
- Azure Machine Learning
 - モデルのトレーニングやデプロイなど
- このように、「azure_ai 拡張機能」から Azure のすべての AI 機能が利用できるわけがないことに注意！

(参考) 「azure_ai 拡張機能」に含まれていない機能は、比較的簡単に自作することもできる。たとえばPython3を使用して関数を定義し、関数の中でGPTのChat Completion APIを呼び出すことで、質問の回答を得ることができる。

```
CREATE EXTENSION plpython3u;

CREATE OR REPLACE FUNCTION call_gpt_azure(prompt TEXT, api_key TEXT, endpoint TEXT, deployment TEXT, api_version TEXT)
RETURNS TEXT
AS $$
import requests, json
u = f'{endpoint}/openai/deployments/{deployment}/chat/completions?api-version={api_version}'
h = {'Content-Type':'application/json','api-key':api_key}
d = {'messages':[{'role':'user','content':prompt}]}
r = requests.post(u, headers=h, json=d)
return r.json()['choices'][0]['message']['content']
$$ LANGUAGE plpython3u;
```

```
SELECT call_gpt_azure(
    'AzureのAI機能について教えて',
    'your-azure-api-key',
    'https://your-resource-name.openai.azure.com',
    'your-deployment-name',
    '2024-03-01'
);
```

モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

「azure_ai 拡張機能」採用時の検討事項

- ・ベンダーロックイン/移行の困難さ
 - ・「azure_ai 拡張機能」は Azure Database for PostgreSQLでしか動かない
- ・トランザクションに時間がかかる ※実装方法にもよるが...
 - ・AI処理には通常、1件あたり数百ミリ秒～数秒程度の待ち時間が必要
 - ・データのINSERT/UPDATE時にトリガーで自動的にAI処理を呼び出すことも可能だが、そうするとINSERT/UPDATEにとても時間がかかるようになる
- ・制御が難しい ※チームメンバーのスキルにもよるが...
 - ・レート制限を意識した処理が難しい
 - ・柔軟なログ記録処理が実装しにくい
- ・責任問題 ※チームメンバーの意識改革が必要かも...



AI処理、SQLができるの
で、DBチーム側で
実装してください～

えー、できますけど、もし
AI処理でトラブルが起きたら
DBチームの責任ですか？



モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

azure_ai 拡張機能の使い所

- ・対象システムがAzure Database for PostgreSQLを使用する前提である
- ・必要なAI機能が「azure_ai拡張機能」によって提供されている
- ・アプリ側ではなく、DB側でAI処理を実装すべき、適切な理由がある
 - ・実装がよりシンプルになる
 - ・トータルで工数の削減が可能である

モジュール1 Azure Database for PostgreSQL で生成AIの利用を開始する

- Azure Database for PostgreSQLとは？
- 管理ツール
- Azure Database for PostgreSQLの利用例
- Azure portalでのリソース作成
- Azure AI Servicesとは？
- Azure portalでのリソース作成
- Azure OpenAI Servicesとは？
- モデルのデプロイ
- azure_ai 拡張機能とは？
- azure_ai 拡張機能でできること
- azure_ai 拡張機能を使うための準備
- azure_ai 拡張機能を利用するためには必要な情報
- azure_ai 拡張機能の「スキーマ」と「関数」
- azure_ai 拡張機能のメリット
- azure_ai 拡張機能ではできないこと
- azure_ai 拡張機能の注意点
- azure_ai 拡張機能の使い所
- まとめ

モジュール1のまとめ(1/2)

Azure Database for PostgreSQL	Azure上でPostgreSQLデータベースを簡単にセットアップし、運用できる、マネージド型のサービス。「pgAdmin4」や「psqlコマンド」など、PostgreSQLの標準的な管理ツールが利用できる。
Azure AI Services	マイクロソフトが開発した独自の機械学習モデルに基づく、すぐに使えるAI機能を提供するサービス。言語、音声、視覚などのAI機能を提供。
Azure OpenAI Service	OpenAI社が開発した生成AIモデル（GPT、DALL-Eなど）をAzure上で利用できるサービス。
Azure AI Servicesリソース	このリソースを一つ作成することで、Azure AI Servicesの機能（言語、音声、視覚など）と、Azure OpenAI Serviceの機能（生成AIの利用）が可能。
モデルのデプロイ	Azure OpenAI Serviceで生成AIモデルを使用するには、モデルをデプロイして利用可能にする必要がある。Azure OpenAI Serviceのモデルのデプロイでは、仮想マシンは特に使用されない。

モジュール1のまとめ(2/2)

azure_ai 拡張機能	Azure Database for PostgreSQL 内から、Azure AI Services、Azure OpenAI Services、Azure Machine Learningの機能を呼び出すための拡張機能。Azure Database for PostgreSQLでのみ利用できる。
azure_ai 拡張機能を利用するための設定	サーバー（リソース）レベルの設定（拡張機能の利用の許可）と、データベースレベルの設定（拡張機能の作成、キー・エンドポイント等の設定）が必要。また関数の呼び出し時にデプロイ名を指定する必要がある場合がある。
スキーマと関数	azure_ai 拡張機能には「azure_ai」「azure_openai」「azure_cognitive」「azure_ml」の4つのスキーマが含まれ、スキーマには関数が含まれる。関数を呼び出すことでAI機能を実行できる。
azure_ai 拡張機能の注意点	azure_ai 拡張機能はAzure Database for PostgreSQLでしか利用できない。AI機能の実行には数百ミリ秒以上の時間がかかる。AI機能の呼び出しがDBレイヤーに加わることで、保守性が低下する可能性がある。
azure_ai 拡張機能の使い所	Azure Database for PostgreSQLを採用すること・必要な機能がazure_ai 拡張機能によって提供されていることが前提。DBレイヤーにAI機能を組み込むことで設計・実装・テストがシンプルになる・共通化が可能・トータルで開発コストを節約できる、といったことが見込める場合に採用するのがおすすめ。

モジュール 2



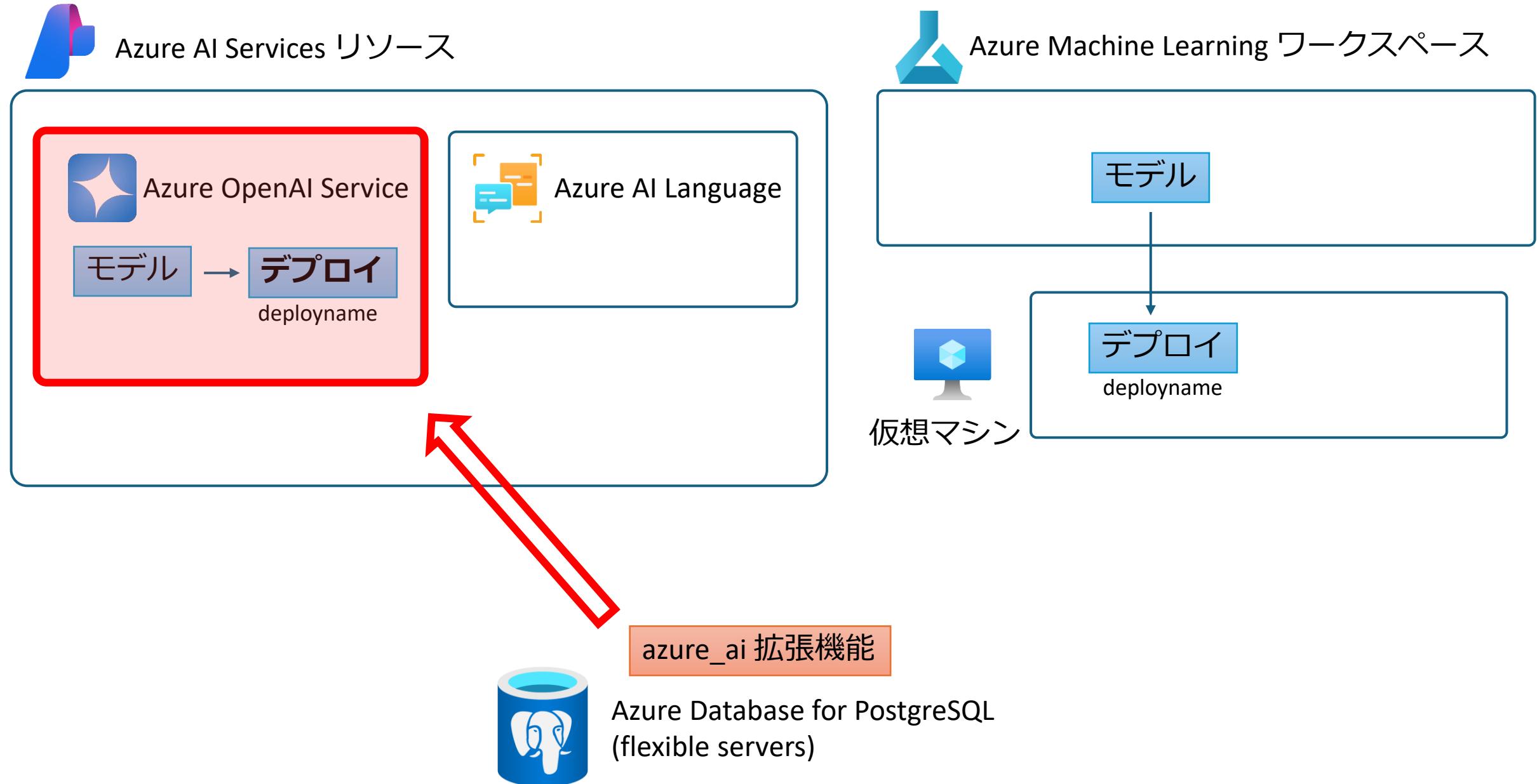
Azure Database for PostgreSQL でセマンティック検索を有効にする

1000 XP

24 分 • モジュール • 9 ユニット

このモジュールでは、セマンティック検索、ベクターの埋め込み、ベクターの類似性、および Azure Database for PostgreSQL - フレキシブル サーバーと PostgreSQL ベクターと Azure AI 拡張機能を使用して、セマンティック検索を使用して行を生成、格納、クエリします。

モジュール2では、 Azure OpenAI Serviceの利用について説明する。

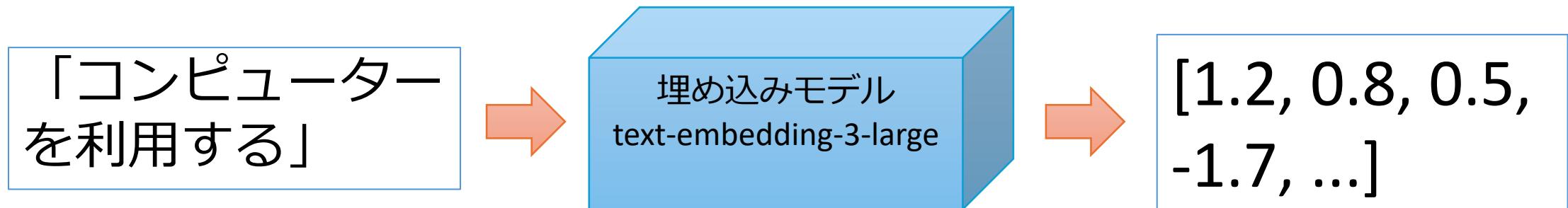


モジュール2 Azure Database for PostgreSQL でセマンティック検索を有効にする

- 「埋め込み」とは？
- 「埋め込み」と「ベクトル」の違い
- 「埋め込み」の計算
- 「埋め込み」の特徴
- コサイン類似度
- コサイン類似度の計算式と計算例
- コサイン類似度を計算する関数の例
- コサイン距離
- real[]型とvector型
- pgvector拡張機能
- ベクトル検索
- ベクトル検索の動作原理
- 最近傍探索アルゴリズム
- まとめ

埋め込み (embeddings) とは？

- ・「埋め込み」は、機械学習モデルと関連アルゴリズムで利用される特別な形式のデータ表現
- ・各「埋め込み」は浮動小数点数のベクトル（数値の配列）で表現される
 - ・ベクトルの例: [1.2, 0.8, 4.5, -1.7, ...]
- ・「text-embedding-3-large」などの「埋め込みモデル」を使用して、任意のテキストをベクトルに変換できる。



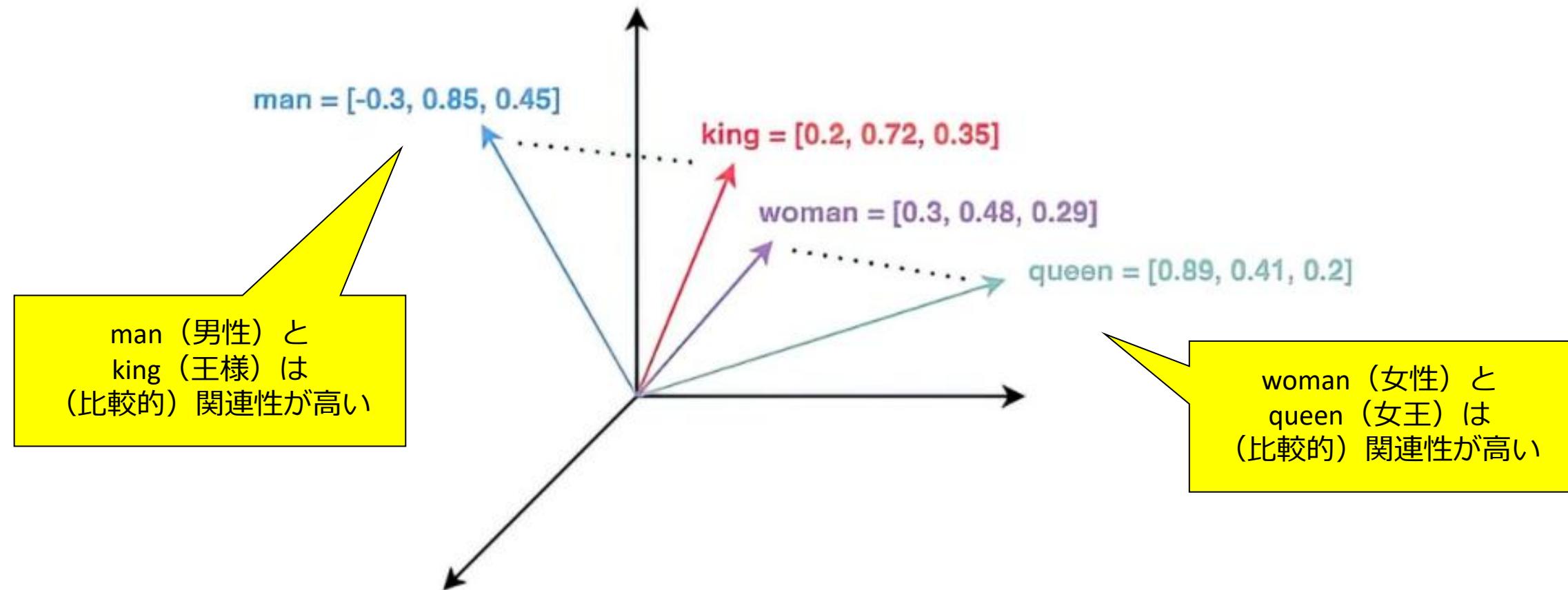
※実際には3072次元のベクトルとなる
(つまり、3072個の浮動小数点数となる)

「埋め込み」によって得られるベクトルのイメージ。

「man」「woman」「king」「queen」などの単語を3次元のベクトルとして表現した例。

関連性の高い単語はベクトルが似ている。

※実際の「埋め込み」は3072次元といった多次元データである



man、king、woman、queen という単語がマッピングされたベクトル空間。出典: [baeldung](#) ↗。

モジュール2 Azure Database for PostgreSQL でセマンティック検索を有効にする

- 「埋め込み」とは？
- 「埋め込み」と「ベクトル」の違い
- 「埋め込み」の計算
- 「埋め込み」の特徴
- コサイン類似度
- コサイン類似度の計算式と計算例
- コサイン類似度を計算する関数の例
- コサイン距離
- real[]型とvector型
- pgvector拡張機能
- ベクトル検索
- ベクトル検索の動作原理
- 最近傍探索アルゴリズム
- まとめ

(参考) 「埋め込み」と「ベクトル」の違い

- 埋め込み (embeddings)

- テキスト（等のデータ）の特徴を数値化したもの
- 似たようなテキスト（等のデータ）は同じような「埋め込み」に変換される
- OpenAIのtext-embedding-3-largeなどのモデルで、テキストの埋め込みを計算できる
- Azure AI VisionのAPIを使用して、画像の埋め込みを計算できる
- MetaのWav2Vecを使用して、音声の埋め込みを計算できる
- 埋め込みはベクトルとして表現・記録される**

- ベクトル、ベクター (vector)

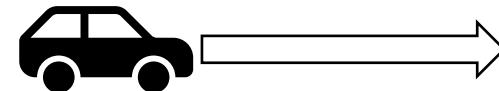
- 平面や空間での有向線分（長さと向きを持つ線分）
- コンピューターに記録された、数値の1次元配列
 - CPUのSIMD (Single Instruction, Multiple Data) 命令を使って高速に計算できる
 - データベースによってはベクトルを扱うためのVECTOR型などを持つ場合がある

[マルチモーダル埋め込みの概念 - Image Analysis 4.0 - Azure AI services | Microsoft Learn](#)

[Wav2vec](#)

(参考) 「埋め込み」と「ベクトル」の違い

- ・「埋め込み」は「ベクトル」として表現されるが、必ずしも「ベクトル」 = 「埋め込み」とは限らない。
- ・たとえばゲームの中であるキャラクターが動いているとき、その動きの向きや速度を表すのにも「ベクトル」が使用できる



- ・「ベクトル」はいろいろなもの（埋め込みやゲームキャラの動き）を表すのに使用できる
- ・ただし、AIや機械学習の文脈では、「ベクトル」は「埋め込み」によって得られたデータを表す言葉として使用されることが多い

モジュール2 Azure Database for PostgreSQL でセマンティック検索を有効にする

- 「埋め込み」とは？
- 「埋め込み」と「ベクトル」の違い
- 「埋め込み」の計算
- 「埋め込み」の特徴
- コサイン類似度
- コサイン類似度の計算式と計算例
- コサイン類似度を計算する関数の例
- コサイン距離
- real[]型とvector型
- pgvector拡張機能
- ベクトル検索
- ベクトル検索の動作原理
- 最近傍探索アルゴリズム
- まとめ

「埋め込み」の計算

- `azure_openai.create_embeddings` 関数を使用すると、テキストを埋め込み（ベクトル）に変換できる
- 戻り値の型は `real[]` 型である

`azure_openai.create_embeddings`関数で、テキストの埋め込み（ベクトル）を計算できる。

第1引数: 埋め込みを計算するためのモデルのデプロイ名。

第2引数: 埋め込みを計算する元のテキスト。

```
47    SELECT azure_openai.create_embeddings('text-embedding-3-large', 'こんにちは')
```

Data Output Messages Notifications



Showing rows: 1

	create_embeddings
	real[]
1	{0.0053383727,-0.0070507475,-0.018627813,0.018415974,0.017568612,0.033131804,-0.020845074,0.038328953,-0.00171

unnest関数を使用すると、結果 (real[]型) を表の形に展開できる。

```
45  SELECT unnest(azure_openai.create_embeddings('text-embedding-3-large', 'こんにちは'))
```

Data Output Messages Notifications

Showing rows: 1 to 1000

	unnest real	🔒
1	0.0053383727	
2	-0.0070507475	
3	-0.018627813	
4	0.018415974	
5	0.017568612	
6	0.033131804	
7	-0.020845074	
8	0.000000050	

Total rows: 3072 Query complete 00:00:01.667

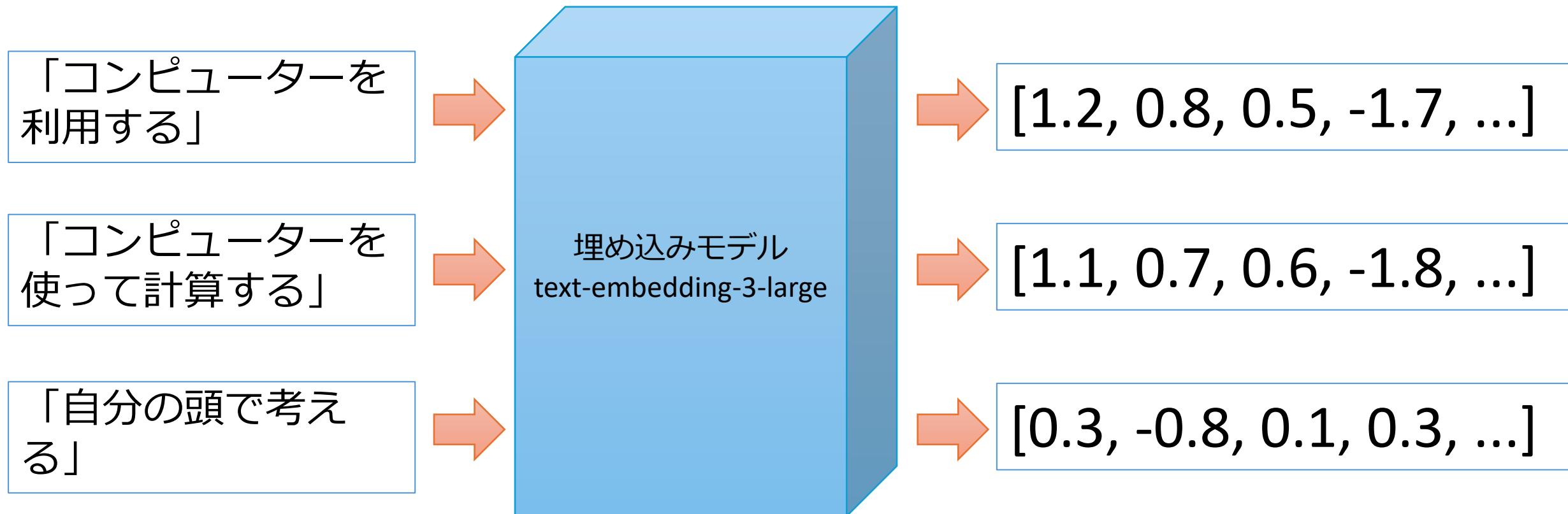
text-embedding-3-large モデル（のデプロイ）を使用した場合、デフォルトではテキストから3072次元の埋め込み（ベクトル）が得られる。

モジュール2 Azure Database for PostgreSQL でセマンティック検索を有効にする

- 「埋め込み」とは？
- 「埋め込み」と「ベクトル」の違い
- 「埋め込み」の計算
- 「埋め込み」の特徴
- コサイン類似度
- コサイン類似度の計算式と計算例
- コサイン類似度を計算する関数の例
- コサイン距離
- real[]型とvector型
- pgvector拡張機能
- ベクトル検索
- ベクトル検索の動作原理
- 最近傍探索アルゴリズム
- まとめ

「埋め込み」の特徴

- 意味が似ている2つの文章では、それらの埋め込み（ベクトル）も似たものとなる

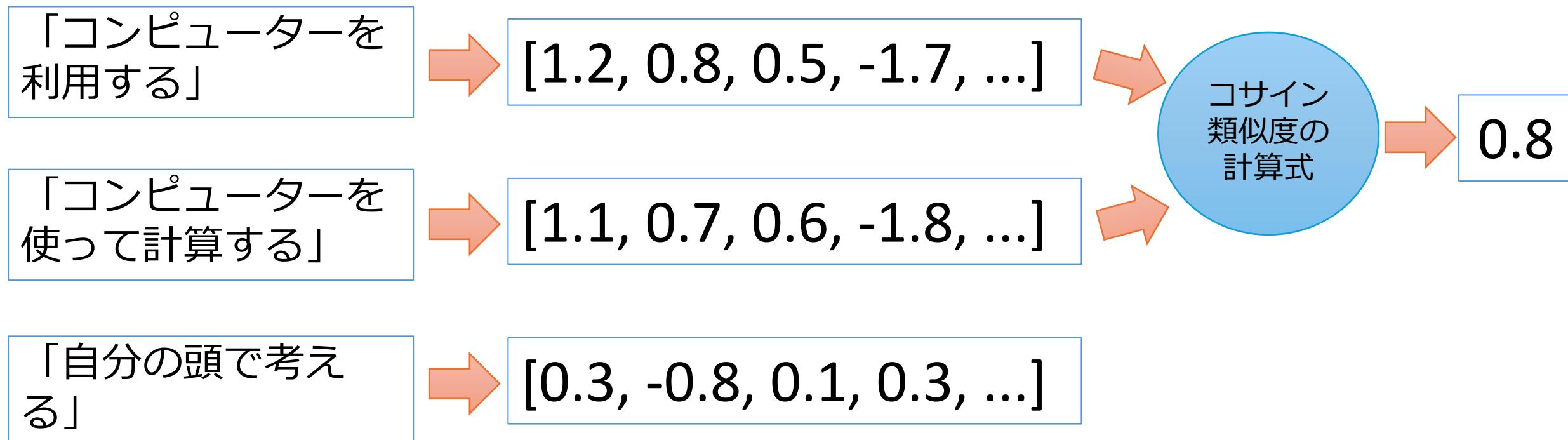


モジュール2 Azure Database for PostgreSQL でセマンティック検索を有効にする

- 「埋め込み」とは？
- 「埋め込み」と「ベクトル」の違い
- 「埋め込み」の計算
- 「埋め込み」の特徴
- コサイン類似度
- コサイン類似度の計算式と計算例
- コサイン類似度を計算する関数の例
- コサイン距離
- real[]型とvector型
- pgvector拡張機能
- ベクトル検索
- ベクトル検索の動作原理
- 最近傍探索アルゴリズム
- まとめ

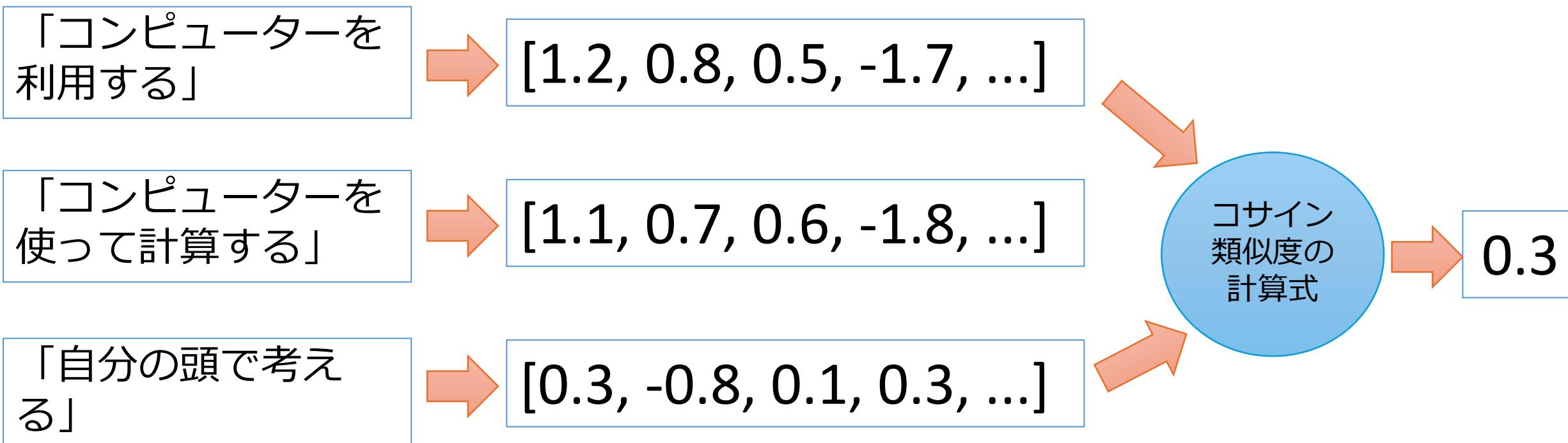
コサイン類似度

- ・2つのベクトルから「コサイン類似度」(-1~1) を計算することで、2つのデータがどの程度「似ているか」を1つの数で表現できる
- ・2つのベクトル(2つのテキスト)が「似ている」場合、「コサイン類似度」は1に近くなる



コサイン類似度

- ・2つのベクトルから「コサイン類似度」(-1~1) を計算することで、2つのデータがどの程度「似ているか」を1つの数で表現できる
- ・2つのベクトル(2つのテキスト)が「似ている」場合、「コサイン類似度」は1に近くなる



モジュール2 Azure Database for PostgreSQL でセマンティック検索を有効にする

- 「埋め込み」とは？
- 「埋め込み」と「ベクトル」の違い
- 「埋め込み」の計算
- 「埋め込み」の特徴
- コサイン類似度
- コサイン類似度の計算式と計算例
- コサイン類似度を計算する関数の例
- コサイン距離
- real[]型とvector型
- pgvector拡張機能
- ベクトル検索
- ベクトル検索の動作原理
- 最近傍探索アルゴリズム
- まとめ

コサイン類似度の計算式と計算例

① 内積 ($A \cdot B$) を求める

$$A \cdot B = (1 \times 4) + (2 \times 5) + (3 \times 6) = 4 + 10 + 18 = 32$$

$$\cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|}$$

② ノルム (長さ) を求める

$$\|A\| = \sqrt{1^2 + 2^2 + 3^2} = \sqrt{14}$$

$$\|B\| = \sqrt{4^2 + 5^2 + 6^2} = \sqrt{77}$$

ベクトル A = [1, 2, 3]

ベクトル B = [4, 5, 6]

③ 式に当てはめてコサイン類似度を計算

$$\cos(\theta) = \frac{32}{\sqrt{14} \cdot \sqrt{77}} = \frac{32}{\sqrt{1078}} \approx 0.9746$$

モジュール2 Azure Database for PostgreSQL でセマンティック検索を有効にする

- 「埋め込み」とは？
- 「埋め込み」と「ベクトル」の違い
- 「埋め込み」の計算
- 「埋め込み」の特徴
- コサイン類似度
- コサイン類似度の計算式と計算例
- コサイン類似度を計算する関数の例
- コサイン距離
- real[]型とvector型
- pgvector拡張機能
- ベクトル検索
- ベクトル検索の動作原理
- 最近傍探索アルゴリズム
- まとめ

コサイン類似度を計算する関数の例

```
CREATE OR REPLACE FUNCTION cosine_similarity(text1 TEXT, text2 TEXT)
RETURNS DOUBLE PRECISION AS $$

DECLARE
    vec1 VECTOR;
    vec2 VECTOR;

BEGIN
    vec1 := CAST(azure_openai.create_embeddings('text-embedding-3-large', text1) AS VECTOR);
    vec2 := CAST(azure_openai.create_embeddings('text-embedding-3-large', text2) AS VECTOR);
    RETURN 1 - (vec1 <=> vec2);
END;
$$ LANGUAGE plpgsql;
```

※実際には、テキストを比較するたびに埋め込みを計算するのは非効率であるため、あらかじめテキストを埋め込みに変換してテーブルの列に格納しておくことが多い。

cosine_similarity 関数を使用して2つのテキストを比較してみる

```
15   SELECT cosine_similarity('コンピューターを利用する', 'コンピューターを使って計算する')
```

```
16
```

Data Output Messages Notifications



Showir

	cosine_similarity	double precision
1	0.8070400838279789	

```
16   SELECT cosine_similarity('コンピューターを利用する', '自分の頭で考える')
```

```
17
```

Data Output Messages Notifications



	cosine_similarity	double precision
1	0.34848690936047944	

モジュール2 Azure Database for PostgreSQL でセマンティック検索を有効にする

- 「埋め込み」とは？
- 「埋め込み」と「ベクトル」の違い
- 「埋め込み」の計算
- 「埋め込み」の特徴
- コサイン類似度
- コサイン類似度の計算式と計算例
- コサイン類似度を計算する関数の例
- コサイン距離
- real[]型とvector型
- pgvector拡張機能
- ベクトル検索
- ベクトル検索の動作原理
- 最近傍探索アルゴリズム
- まとめ

(参考) コサイン距離

- 2つのベクトル `vec1` と `vec2` から「コサイン距離」を計算する
`<=>` 演算子と、 **cosine_distance** 関数がある。
 - `vec1 <=> vec2`
 - または **cosine_distance**(`vec1`, `vec2`)
- 1から「コサイン距離」を引くと「コサイン類似度」が得られる。
 - `1 - (vec1 <=> vec2)`
 - または `1 - cosine_distance(vec1, vec2)`
- なお「コサイン類似度」を計算する関数はPostgreSQL本体・
「`azure_ai`拡張機能」・「`pgvector`拡張機能」には定義されていないが、「コサイン距離」から簡単に計算できる。

モジュール2 Azure Database for PostgreSQL でセマンティック検索を有効にする

- 「埋め込み」とは？
 - 「埋め込み」と「ベクトル」の違い
 - 「埋め込み」の計算
 - 「埋め込み」の特徴
 - コサイン類似度
 - コサイン類似度の計算式と計算例
 - コサイン類似度を計算する関数の例
 - コサイン距離
- real[]型とvector型
 - pgvector拡張機能
 - ベクトル検索
 - ベクトル検索の動作原理
 - 最近傍探索アルゴリズム
 - まとめ

(参考) PostgreSQLの real[]型 と vector型

- azure_openai.create_embeddings関数は real[]型 の値を返す。
- 通常、ベクトルの演算（コサイン距離の計算）やベクトルのテーブル列への格納には real[] 型ではなく vector 型 が使用される。
- real[] 型 を vector 型 に変換するには CAST関数を使用する。

```
SELECT CAST(azure_openai.create_embeddings('text-embedding-3-large', 'こんにちは') AS vector)
```

モジュール2 Azure Database for PostgreSQL でセマンティック検索を有効にする

- 「埋め込み」とは？
- 「埋め込み」と「ベクトル」の違い
- 「埋め込み」の計算
- 「埋め込み」の特徴
- コサイン類似度
- コサイン類似度の計算式と計算例
- コサイン類似度を計算する関数の例
- コサイン距離
- real[]型とvector型
- pgvector拡張機能
- ベクトル検索
- ベクトル検索の動作原理
- 最近傍探索アルゴリズム
- まとめ

<=>演算子, cosine_distance関数, vector型などを使用する場合は「pgvector拡張機能」を有効にする。

(1)サーバーパラメーター「VECTOR」の有効化

The screenshot shows the 'pgsv1 | サーバー パラメーター' (Server Parameters) page in the Azure portal. The 'VECTOR' parameter is being configured under the 'azure.extensions' extension. The 'Value' dropdown contains 'vec' and 'VECTOR'. A red box highlights the 'VECTOR' option, which is checked. Red arrows point from the 'VECTOR' label in the text above to the 'VECTOR' entry in the dropdown and to the checked checkbox. The 'Save' button at the top right is also highlighted with a red box.

パラメータ名	値	パラメーターの型	説明
azure.extensions	2 項目…	Dynamic	Specific
vec			
<input checked="" type="checkbox"/> VECTOR			

(2)「pgvector」拡張機能のインストール

The screenshot shows the PostgreSQL Query Editor with the 'Messages' tab selected. A single query is listed: 'CREATE EXTENSION vector;'. Below the query, the message 'CREATE EXTENSION' is displayed, followed by 'Query returned successfully in 140 msec.'

※この拡張機能は一般的に「pgvector」と呼ばれることが多いが、拡張機能の名前は「vector」である。

モジュール2 Azure Database for PostgreSQL でセマンティック検索を有効にする

- 「埋め込み」とは？
- 「埋め込み」と「ベクトル」の違い
- 「埋め込み」の計算
- 「埋め込み」の特徴
- コサイン類似度
- コサイン類似度の計算式と計算例
- コサイン類似度を計算する関数の例
- コサイン距離
- real[]型とvector型
- pgvector拡張機能
- ベクトル検索
- ベクトル検索の動作原理
- 最近傍探索アルゴリズム
- まとめ

ベクトル検索（セマンティック検索）

- ・従来のデータ検索では、ユーザーが「検索キーワード」を指定し、そのキーワードを含む文書などを検索していた。
- ・検索キーワードを正しく入力しなければ、目的のテキストを見ることができなかった。
- ・埋め込み（ベクトル）とコサイン類似度を使用することで、ユーザーが指定したテキストに意味的に近い（関連性が高い）テキストを検索することが可能となった。
- ・これをベクトル検索（セマンティック検索）という。

モジュール2 Azure Database for PostgreSQL でセマンティック検索を有効にする

- 「埋め込み」とは？
- 「埋め込み」と「ベクトル」の違い
- 「埋め込み」の計算
- 「埋め込み」の特徴
- コサイン類似度
- コサイン類似度の計算式と計算例
- コサイン類似度を計算する関数の例
- コサイン距離
- real[]型とvector型
- pgvector拡張機能
- ベクトル検索
- ベクトル検索の動作原理
- 最近傍探索アルゴリズム
- まとめ

ベクトル検索の動作原理



出張で使用
できる予算
の上限は？

埋め込み
モデル

[1.2, 0.7, 0.6, -1.7, ...]

コサイン
類似度の
計算式

テキスト	埋め込み	コサイン類似度
今年の目標・・・	[0.5, -1.1, -0.3, 0.9, ...]	0.2
平素は格別のご高配を・・・	[-0.3, 0.4, 0.1, 0.5, ...]	
出張規程【2025】・・・	[1.5, 0.3, -1.1, 0.1, ...]	
会議の議事録・・・	[0.5, -1.1, -0.3, 0.9, ...]	
製品Xのマニュアル・・・	[0.2, 0.3, -1.2, -0.5, ...]	

ベクトル検索の動作原理



出張で使用
できる予算
の上限は？

埋め込み
モデル

[1.2, 0.7, 0.6, -1.7, ...]

コサイン
類似度の
計算式

テキスト	埋め込み	コサイン類似度
今年の目標・・・	[0.5, -1.1, -0.3, 0.9, ...]	0.2
平素は格別のご高配を・・・	[-0.3, 0.4, 0.1, 0.5, ...]	0.1
出張規程【2025】・・・	[1.5, 0.3, -1.1, 0.1, ...]	
会議の議事録・・・	[0.5, -1.1, -0.3, 0.9, ...]	
製品Xのマニュアル・・・	[0.2, 0.3, -1.2, -0.5, ...]	

ベクトル検索の動作原理



出張で使用
できる予算
の上限は？



[1.2, 0.7, 0.6, -1.7, ...]

コサイン
類似度の
計算式

テキスト	埋め込み	コサイン類似度
今年の目標・・・	[0.5, -1.1, -0.3, 0.9, ...]	0.2
平素は格別のご高配を・・・	[-0.3, 0.4, 0.1, 0.5, ...]	0.1
出張規程【2025】・・・	[1.5, 0.3, -1.1, 0.1, ...]	0.9
会議の議事録・・・	[0.5, -1.1, -0.3, 0.9, ...]	
製品Xのマニュアル・・・	[0.2, 0.3, -1.2, -0.5, ...]	

ベクトル検索の動作原理



出張で使用
できる予算
の上限は？



[1.2, 0.7, 0.6, -1.7, ...]

コサイン
類似度の
計算式

テキスト	埋め込み	コサイン類似度
今年の目標・・・	[0.5, -1.1, -0.3, 0.9, ...]	0.2
平素は格別のご高配を・・・	[-0.3, 0.4, 0.1, 0.5, ...]	0
出張規程【2025】・・・	[1.5, 0.3, -1.1, 0.1, ...]	0.9
会議の議事録・・・	[0.5, -1.1, -0.3, 0.9, ...]	0.3
製品Xのマニュアル・・・	[0.2, 0.3, -1.2, -0.5, ...]	

ベクトル検索の動作原理



出張で使用
できる予算
の上限は？



[1.2, 0.7, 0.6, -1.7, ...]

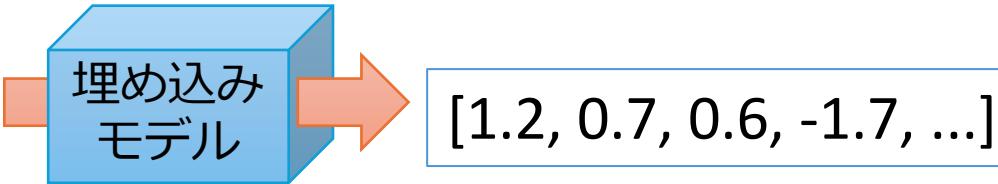
コサイン
類似度の
計算式

テキスト	埋め込み	コサイン類似度
今年の目標・・・	[0.5, -1.1, -0.3, 0.9, ...]	0.2
平素は格別のご高配を・・・	[-0.3, 0.4, 0.1, 0.5, ...]	0.1
出張規程【2025】・・・	[1.5, 0.3, -1.1, 0.1, ...]	0.9
会議の議事録・・・	[0.5, -1.1, -0.3, 0.9, ...]	0.3
製品Xのマニュアル・・・	[0.2, 0.3, -1.2, -0.5, ...]	0.2

ベクトル検索の動作原理



出張で使用
できる予算
の上限は？



テキスト	埋め込み	コサイン類似度
今年の目標・・・	[0.5, -1.1, -0.3, 0.9, ...]	0.2
平素は格別のご高配を・・・	[-0.3, 0.4, 0.1, 0.5, ...]	0.1
出張規程【2025】・・・	[1.5, 0.3, -1.1, 0.1, ...]	0.9
会議の議事録・・・	[0.5, -1.1, -0.3, 0.9, ...]	0.3
製品Xのマニュアル・・・	[0.2, 0.3, -1.2, -0.5, ...]	0.2

ベクトル検索の動作原理



出張で使用
できる予算
の上限は？



[1.2, 0.7, 0.6, -1.7, ...]

テキスト	埋め込み	コサイン類似度
今年の目標・・・	[0.5, -1.1, -0.3, 0.9, ...]	0.2
平素は格別のご高配を・・・	[-0.3, 0.4, 0.1, 0.5, ...]	0.1
出張規程【2025】・・・	[1.5, 0.3, -1.1, 0.1, ...]	0.9
会議の議事録・・・	[0.5, -1.1, -0.3, 0.9, ...]	0.3
製品Xのマニュアル・・・	[0.2, 0.3, -1.2, -0.5, ...]	0.2

モジュール2 Azure Database for PostgreSQL でセマンティック検索を有効にする

- 「埋め込み」とは？
- 「埋め込み」と「ベクトル」の違い
- 「埋め込み」の計算
- 「埋め込み」の特徴
- コサイン類似度
- コサイン類似度の計算式と計算例
- コサイン類似度を計算する関数の例
- コサイン距離
- real[]型とvector型
- pgvector拡張機能
- ベクトル検索
- ベクトル検索の動作原理
- 最近傍探索アルゴリズム
- まとめ

参考: ベクトルの最近傍探索アルゴリズム

- ・ 実際には、検索のたびに、全てのベクトルとの比較（類似度計算）を行うと、時間がかかりすぎる。
- ・ 実用的なシステム（ベクトルデータベース）では、指定したベクトルに近いベクトルを素早く探し出すために、より効率的なアルゴリズムとデータ構造（インデックス）が使用される
 - ・ ANN (Approximate Nearest Neighbor, 近似最近傍探索) : 近似的な方法でベクトルを高速に検索するアルゴリズム。数百万件といった大量のベクトルを高速に処理できる。
 - ・ HNSW (Hierarchical Navigable Small World, 階層的ナビゲーブルスモールワールド) : ANN検索の中でもよく使用されるアルゴリズム。階層的なグラフ構造を使って近いベクトルを効率的に探す。事前に階層インデックスを構築する。
- ・ Azure Database for PostgreSQLでもHNSWなどができる

興味がある方は
こちらをチェック

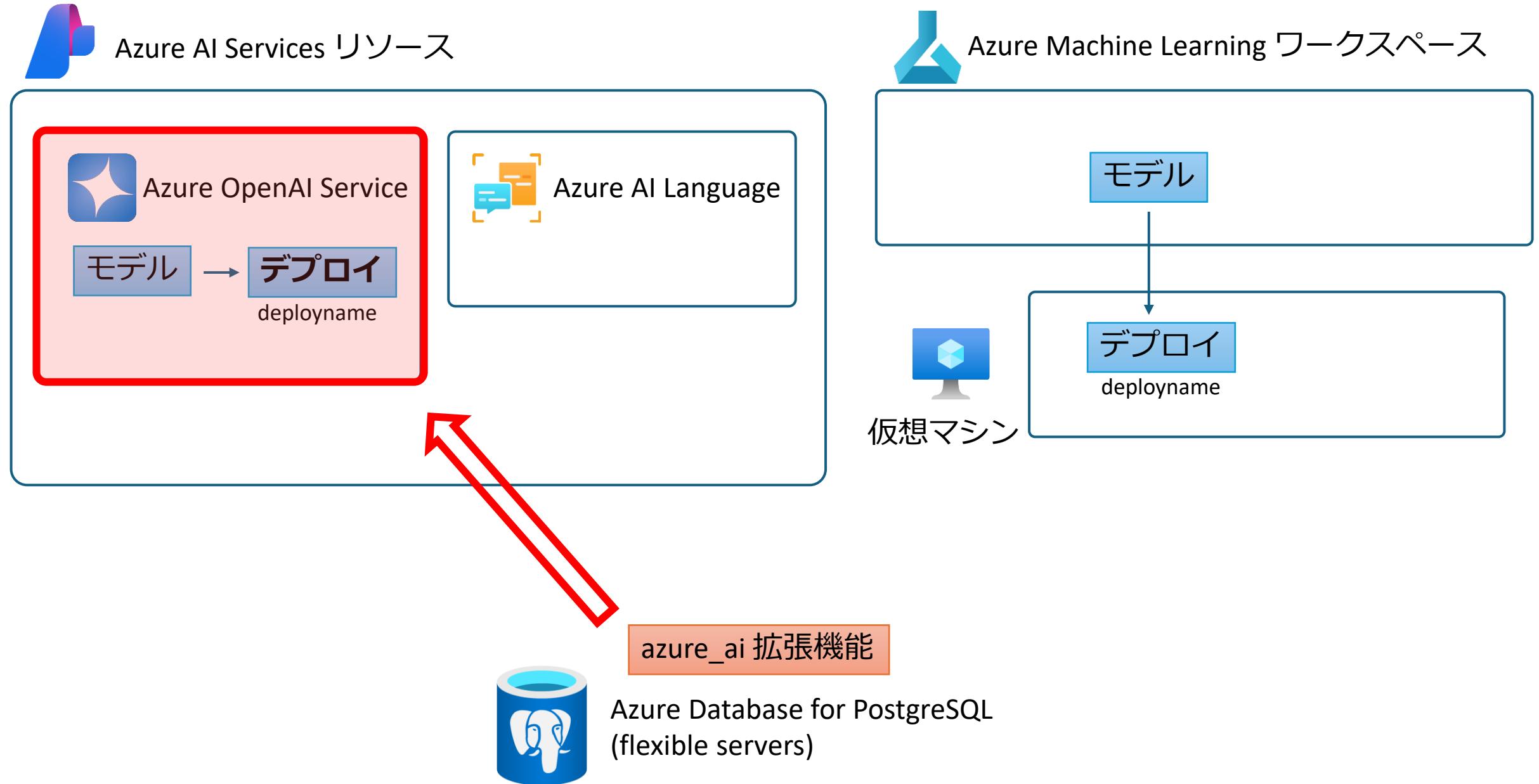
モジュール2 Azure Database for PostgreSQL でセマンティック検索を有効にする

- 「埋め込み」とは？
- 「埋め込み」と「ベクトル」の違い
- 「埋め込み」の計算
- 「埋め込み」の特徴
- コサイン類似度
- コサイン類似度の計算式と計算例
- コサイン類似度を計算する関数の例
- コサイン距離
- real[]型とvector型
- pgvector拡張機能
- ベクトル検索
- ベクトル検索の動作原理
- 最近傍探索アルゴリズム
- まとめ

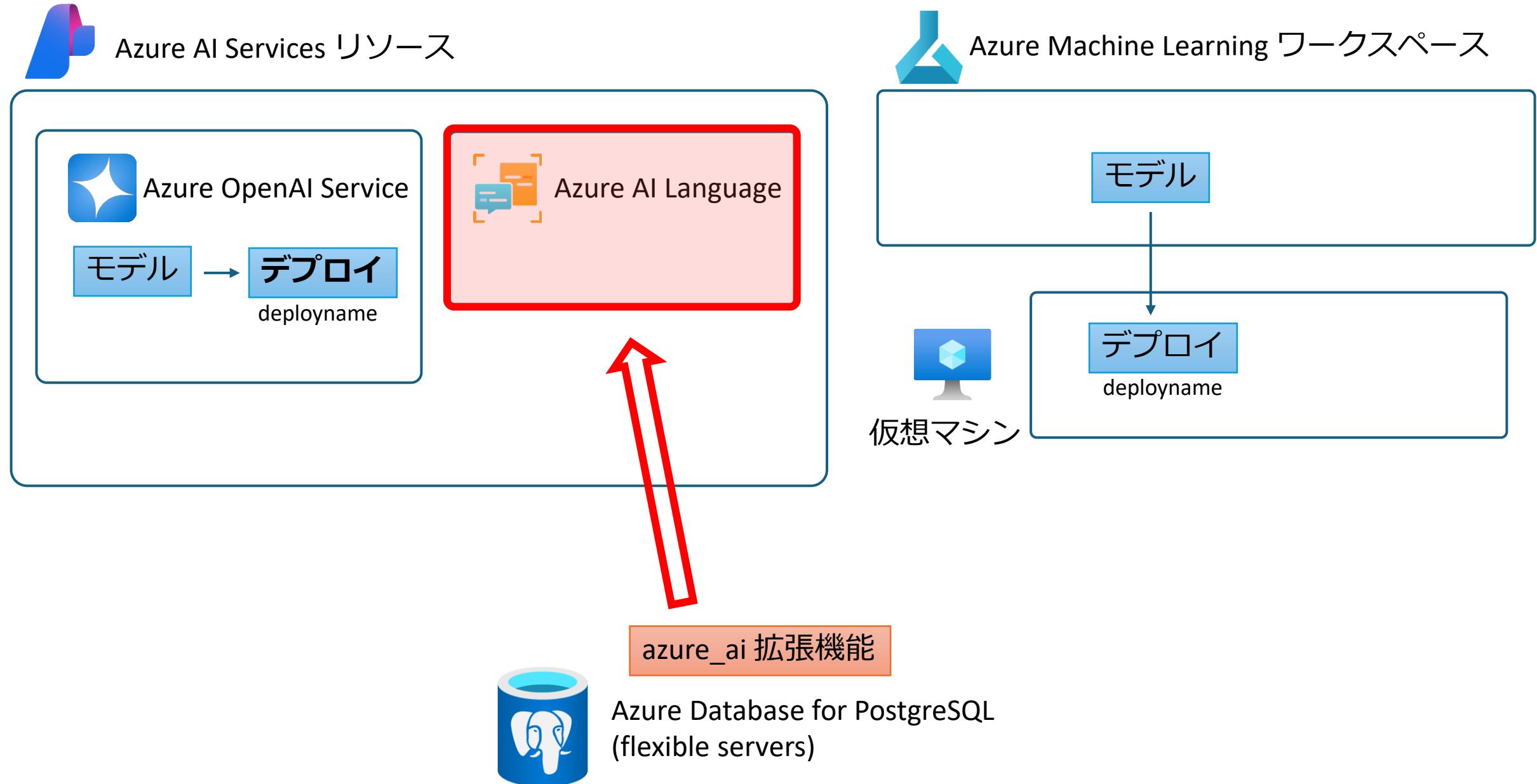
モジュール2のまとめ

埋め込みモデル	「text-embedding-3-large」などの「埋め込みモデル」を使用して、テキストなどのデータの「埋め込み」（ベクトル）を計算できる。
埋め込み (embeddings)	機械学習モデルと関連アルゴリズムで利用される浮動小数点数の配列（ベクトル）であり、元のデータの特徴を含む。
コサイン類似度	2つのデータの「埋め込み」（ベクトル）をそれぞれ計算し、そのコサイン類似度を計算することで、2つのデータの類似度（-1～1）を求めることができる。1に近いほど類似度が高い（意味的に近い）。
ベクトル検索 (セマンティック検索)	コサイン類似度などを使用して、類似度が高い（意味的に近い）データを検索する方法。
最近傍探索 アルゴリズム	ベクトルデータベースなどでは、HNSW (Hierarchical Navigable Small World)などの最近傍探索（近似最近傍探索）アルゴリズムとデータ構造を使用して、高速なベクトル検索を実現している。

モジュール2では、 Azure OpenAI Serviceの利用について説明した。



モジュール3～6では、 Azure AI Languageの利用について説明する。



モジュール 3

文章を要約する



Azure AI サービスと Azure Database for PostgreSQL を使用してデータを集計する

700 XP

21 分 • モジュール • 6 ユニット

このモジュールでは、Azure AI サービスと Azure Database for PostgreSQL - フレキシブル サーバー用の Azure AI 拡張機能を使用して、複雑なコンテンツを簡潔な要約に集約するための生成 AI 要約手法を有効にする方法について説明します。

モジュール3 Azure Database for PostgreSQL を使用して文章を要約する

- Azure AI Languageの「要約」とは
 - 「抽出要約」と「抽象要約」
 - 「抽出要約」の利用例
 - 「抽象要約」の利用例
-
- ※モジュール3～6は非常に短いため、連続で解説し、これらのモジュールのまとめはモジュール6の最後で行う

モジュール3 Azure Database for PostgreSQL を使用して文章を要約する

- Azure AI Languageの「要約」とは
 - 「抽出要約」と「抽象要約」
 - 「抽出要約」の利用例
 - 「抽象要約」の利用例

Azure AI Languageの「要約」とは

- ・入力された文から重要な情報を取り出してより短い文を作成
- ・2種類の方式「抽出要約」と「抽象要約」がある（後述）

モジュール3 Azure Database for PostgreSQL を使用して文章を要約する

- Azure AI Languageの「要約」とは
 - 「抽出要約」と「抽象要約」
 - 「抽出要約」の利用例
 - 「抽象要約」の利用例

「抽出要約」と「抽象要約」

方式	抽出要約 (Extractive Summarization)	抽象要約 (Abstractive Summarization)
概要	原文から重要な文をそのまま抽出	原文の内容をふまえて新しい文を生成
例	<p>「AIは情報処理を効率化します。これは多くの業界で活用されています。AIは多くの分野で効率化に役立っています」</p> <p style="text-align: center;">↓</p> <p>「AIは情報処理を効率化します」</p>	<p>「AIは情報処理を効率化します。これは多くの業界で活用されています。AIは多くの分野で効率化に役立っています」</p> <p style="text-align: center;">↓</p> <p>「AIは多くの業界で情報処理の効率化に役立っています」</p>
出力	原文の一部	原文に意味的に近い新しい文
正確さ	◎ (加工なし)	○ (新しい文章が生成されるため、不正確になる可能性がある)
情報量	△ (原文の一部のみ抽出)	◎ (元の文全体の情報が含まれる)

モジュール3 Azure Database for PostgreSQL を使用して文章を要約する

- Azure AI Languageの「要約」とは
- 「抽出要約」と「抽象要約」
- 「抽出要約」の利用例
- 「抽象要約」の利用例

抽出要約の利用例。

第1引数は処理対象のテキスト、第2引数は処理対象テキストの言語のコード、第3引数は出力する要約文章の数。

Query Query History

```
1 ▼ select azure_cognitive.summarize_extractive(  
2   'AIは情報処理を効率化します。これは多くの業界で活用されています。AIは多くの分野で効率化に役立っています', 'ja', 1)  
3  
4
```

Data Output Messages Notifications



Showing rows: 1 to 1



Page No: 1

of 1

	summarize_extractive azure_cognitive.sentence[]	🔒
1	{"(AIは情報処理を効率化します。,1)"}	

unnest関数を適用して、結果を表の形に変換すると、結果がわかりやすくなる

Query Query History

```
1 ✓ select * from unnest(azure_cognitive.summarize_extractive(  
2   'AIは情報処理を効率化します。これは多くの業界で活用されています。AIは多くの分野で効率化に役立っています。', 'ja', 1))  
3
```

Data Output Messages Notifications

Showing rows: 1 to 1 Page No: 1 of 1

	text	rank_score
1	AIは情報処理を効率化します。	1

モジュール3 Azure Database for PostgreSQL を使用して文章を要約する

- Azure AI Languageの「要約」とは
- 「抽出要約」と「抽象要約」
- 「抽出要約」の利用例
- 「抽象要約」の利用例

抽象要約の利用例。

Query History

```
1 select azure_cognitive.summarize_abstractive(  
2   'AIは情報処理を効率化します。これは多くの業界で活用されています。AIは多くの分野で効率化に役立っています', 'ja', 1)  
3  
4 |
```

Data Output Messages Notifications

Showing rows: 1 to 1 | Page No: 1 of 1

	summarize_abstractive	text[]
1	{AIは情報処理の効率化に役立ち、多くの業界と分野で活用されています。}	lock icon

モジュール 4



Azure Database for PostgreSQL を使用して感情分析とオピニオンマイニングを実行する

700 XP

15 分 • モジュール • 6 ユニット

このモジュールでは、Azure AI Services と Azure Database for PostgreSQL - フレキシブル サーバー用の Azure AI 拡張機能を使用して、感情分析とオピニオンマイニングでテキストデータから貴重な分析情報を抽出する方法について説明します。

モジュール4 Azure Database for PostgreSQL を使用して感情分析とオピニオンマイニ ングを実行する

- 「感情分析」 (sentiment analysis) とは
- 「azure_ai拡張機能」からの利用例
- (参考) オピニオンマイニング

「感情分析」 sentiment analysis とは？

- ・入力された文の感情を判定
- ・「ポジティブ」（肯定的）
- ・「ネガティブ」（否定的）
- ・「ニュートラル」（中立的）
- ・「ミックス」（肯定的な意見と否定的な意見が両方含まれている）

戻り値（感情分析の結果として得られるデータ）

- sentiment
 - 入力されたテキスト全体の感情を分類したもの
 - positive, negative, neutral, mixed のいずれか。
- positive_score
 - 0～1の実数（1に近いほど入力されたテキストは肯定的であることを表す）
- neutral_score
 - 0～1の実数（1に近いほど入力されたテキストは中立的であることを表す）
- negative_score
 - 0～1の実数（1に近いほど入力されたテキストは否定的であることを表す）

モジュール4 Azure Database for PostgreSQL を使用して感情分析とオピニオンマイニ ングを実行する

- ・「感情分析」（sentiment analysis）とは
- ・「azure_ai拡張機能」からの利用例
- ・（参考）オピニオンマイニング

結果が positive になる例。

```
48  SELECT * from azure_cognitive.analyze_sentiment(  
49    '山田先生の講義はわかりやすい', 'ja')  
50
```

Data Output Messages Notifications



SQL

	sentiment text	positive_score double precision	neutral_score double precision	negative_score double precision
1	positive	0.97	0.03	0

結果が negative になる例。

```
48  SELECT * from azure_cognitive.analyze_sentiment(  
49  '山田先生の講義はわかりにくい', 'ja')  
50
```

Data Output Messages Notifications



	sentiment text	positive_score double precision	neutral_score double precision	negative_score double precision
1	negative	0.14	0.02	0.85

結果が positive になる例。

```
48  SELECT * from azure_cognitive.analyze_sentiment(  
49  'スタッフは不親切だったが料理はすばらしかった', 'ja')  
50  |
```

Data Output Messages Notifications



	sentiment	positive_score	neutral_score	negative_score
	text	double precision	double precision	double precision
1	positive	0.99	0	0.01

結果が negative になる例。

```
48  SELECT * from azure_cognitive.analyze_sentiment(  
49  '料理はすばらしかったがスタッフは不親切だった', 'ja')  
50
```

Data Output Messages Notifications



	sentiment text	positive_score double precision	neutral_score double precision	negative_score double precision
1	negative	0.25	0	0.75

結果が mixed になる例。2つの文章があり、一つがPositive、もう一つがNegative。

```
48  SELECT * from azure_cognitive.analyze_sentiment(  
49  '料理はすばらしかった。スタッフは不親切だった。', 'ja')  
50
```

Data Output Messages Notifications



	sentiment text	positive_score double precision	neutral_score double precision	negative_score double precision
1	mixed	0.51	0.01	0.49

結果が neutral になる例。

```
48  SELECT * from azure_cognitive.analyze_sentiment(  
49    '吾輩は猫である。名前はまだない。', 'ja')  
50
```

Data Output Messages Notifications



SQL

	sentiment text	positive_score double precision	neutral_score double precision	negative_score double precision
1	neutral	0.03	0.91	0.06

モジュール4 Azure Database for PostgreSQL を使用して感情分析とオピニオンマイニ ングを実行する

- ・「感情分析」（sentiment analysis）とは
- ・「azure_ai拡張機能」からの利用例
- ・（参考）オピニオンマイニング

(参考) オピニオンマイニングとは

- ・「感情分析」よりも高度な分析機能。
- ・「料理はすばらしかったがスタッフは不親切だった」といった文章を複数の「主題」に分けて分析する。
- ・各「主題」ごとに「意見」と「感情」が得られる。
- ・"料理はすばらしかったが、スタッフは不親切だった"



主題 (Subject)	意見 (Opinion)	感情 (Sentiment)
料理	すばらしい	肯定的 (Positive)
スタッフ	不親切	否定的 (Negative)

※この機能は「Azure AI Language」には含まれているが、
「azure_ai 拡張機能」には対応する関数がないことに注意。
(つまり azure_ai 拡張機能からはオピニオンマイニングは利用できない)

モジュール 5



Azure AI Language と Azure Database for PostgreSQL を使用して分析情報を抽出する

800 XP

22 分 • モジュール • 7 ユニット

このモジュールでは、キー フレーズ抽出、エンティティ認識、個人を特定できる情報認識と編集、Azure Database for PostgreSQL - フレキシブル サーバーと Azure AI を使用してこれらのサービスにアクセスするなどの Azure AI Language ツールについて説明します。

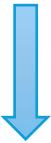
モジュール5 Azure AI Language と Azure Database for PostgreSQL を使用して分析情報を抽出する

- ・「キーフレーズ抽出」とは
- ・「azure_ai拡張機能」からの「キーフレーズ抽出」の利用例
- ・「エンティティ認識」とは
- ・「azure_ai拡張機能」からの「エンティティ認識」の利用例
- ・「エンティティリンク」とは
- ・「azure_ai拡張機能」からの「エンティティリンク」の利用例
- ・「個人を特定できる情報の検出」とは
- ・「azure_ai拡張機能」からの「個人を特定できる情報の検出」の利用例

「キーワード抽出」とは

- ・入力されたテキスト内の主要な概念を特定する。

"料理はすばらしく、スタッフはみな親切だった。"



"料理"



"スタッフ"

「キーワード抽出」の応用例

- ・メールが到着したら、メールの内容を分析し、「ラベル」を付与して、メールを分類する

モジュール5 Azure AI Language と Azure Database for PostgreSQL を使用して分析情報を抽出する

- ・「キーフレーズ抽出」とは
- ・「azure_ai拡張機能」からの「キーフレーズ抽出」の利用例
- ・「エンティティ認識」とは
- ・「azure_ai拡張機能」からの「エンティティ認識」の利用例
- ・「エンティティリンク」とは
- ・「azure_ai拡張機能」からの「エンティティリンク」の利用例
- ・「個人を特定できる情報の検出」とは
- ・「azure_ai拡張機能」からの「個人を特定できる情報の検出」の利用例

第1引数に処理対象の文章、第2引数に文章の言語のコードを指定する。

```
48  SELECT * from azure_cognitive.extract_key_phrases(  
49  '料理はすばらしく、スタッフはみな親切だった。', 'ja')  
50
```

Data Output Messages Notifications



extract_key_phrases



text[]

1

{料理,スタッフ}

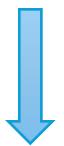
モジュール5 Azure AI Language と Azure Database for PostgreSQL を使用して分析情報を抽出する

- ・「キーフレーズ抽出」とは
- ・「azure_ai拡張機能」からの「キーフレーズ抽出」の利用例
- ・「エンティティ認識」とは
- ・「azure_ai拡張機能」からの「エンティティ認識」の利用例
- ・「エンティティリンク」とは
- ・「azure_ai拡張機能」からの「エンティティリンク」の利用例
- ・「個人を特定できる情報の検出」とは
- ・「azure_ai拡張機能」からの「個人を特定できる情報の検出」の利用例

「エンティティ認識」とは

- ・テキスト内の日時、場所などを特定。

"私は先週、北海道の登別温泉に旅行に行った。"



"先週" "北海道" "登別温泉" "旅行"
(日時) (場所) (場所) (イベント)

「エンティティ認識」とは

- ・会議の開催に関するメールが到着したら、メールの中に含まれている会議の日時を認識して、自動的にカレンダーにその会議の情報を登録する

モジュール5 Azure AI Language と Azure Database for PostgreSQL を使用して分析情報を抽出する

- ・「キーフレーズ抽出」とは
- ・「azure_ai拡張機能」からの「キーフレーズ抽出」の利用例
- ・「エンティティ認識」とは
- ・「azure_ai拡張機能」からの「エンティティ認識」の利用例
- ・「エンティティリンク」とは
- ・「azure_ai拡張機能」からの「エンティティリンク」の利用例
- ・「個人を特定できる情報の検出」とは
- ・「azure_ai拡張機能」からの「個人を特定できる情報の検出」の利用例

第1引数に処理対象の文章、第2引数に文章の言語のコードを指定する。

```
48  SELECT * from unnest(azure_cognitive.recognize_entities(  
49  '私は先週北海道の登別温泉に旅行に行った。', 'ja'))  
50
```

Data Output Messages Notifications

SQL

	text text	category text	subcategory text	confidence_score double precision
1	先週	DateTime	DateRange	0.84
2	北海道	Location	GPE	0.76
3	登別温泉	Location		0.98
4	旅行	Event		0.53

GPE = Geo Political Entity、地政学的エンティティ。地政学 = 地理と政治を合わせて考察する学問。

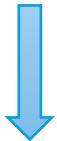
モジュール5 Azure AI Language と Azure Database for PostgreSQL を使用して分析情報を抽出する

- ・「キーフレーズ抽出」とは
- ・「azure_ai拡張機能」からの「キーフレーズ抽出」の利用例
- ・「エンティティ認識」とは
- ・「azure_ai拡張機能」からの「エンティティ認識」の利用例
- ・「エンティティリンク」**とは**
- ・「azure_ai拡張機能」からの「エンティティリンク」の利用例
- ・「個人を特定できる情報の検出」とは
- ・「azure_ai拡張機能」からの「個人を特定できる情報の検出」の利用例

「エンティティリンク」とは

- テキスト内のエンティティを認識してWikipediaページに関連付けする。

"私は先週、北海道の登別温泉に旅行に行った。"



"北海道" "登別温泉"



Wikipedia

[https://ja.wikipedia.org › wiki › 北海道](https://ja.wikipedia.org/wiki/北海道)

北海道

北海道（ほっかいどう）は、日本の北



Wikipedia

[https://ja.wikipedia.org › wiki › 登別温泉](https://ja.wikipedia.org/wiki/登別温泉) :

登別温泉

登別温泉（のぼりべつおんせん、英: Noboribetsu Onsen）

温泉の存在が知られており、明治時代に温泉宿が設けられ

モジュール5 Azure AI Language と Azure Database for PostgreSQL を使用して分析情報を抽出する

- ・「キーフレーズ抽出」とは
- ・「azure_ai拡張機能」からの「キーフレーズ抽出」の利用例
- ・「エンティティ認識」とは
- ・「azure_ai拡張機能」からの「エンティティ認識」の利用例
- ・「エンティティリンク」とは
- ・「azure_ai拡張機能」からの「エンティティリンク」の利用例
- ・「個人を特定できる情報の検出」とは
- ・「azure_ai拡張機能」からの「個人を特定できる情報の検出」の利用例

文章中に含まれる地名（「北海道」「登別」など）や
概念（「温泉」など）を認識し、Wikipediaへのリンクを付与する。
英語（en）とスペイン語（es）に対応。日本語（ja）は不可。

```
48  SELECT * from unnest(azure_cognitive.linked_entities(  
49    'I went on a trip to Noboribetsu Onsen in Hokkaido last week.', 'en'))  
50
```

Data Output Messages Notifications



Showing row

	name text	data_source_entity_id text	language text	data_source text	url text
1	Noboribetsu	Noboribetsu	en	Wikipedia	https://en.wikipedia.org/wiki/Noboribetsu
2	Onsen	Onsen	en	Wikipedia	https://en.wikipedia.org/wiki/Onsen
3	Hokkaido	Hokkaido	en	Wikipedia	https://en.wikipedia.org/wiki/Hokkaido

※ "Onsen" は "Sushi" や "Kawaii" と同様、英語として通用する

From Wikipedia, the free encyclopedia

In Japan, *onsen* (温泉) are hot springs and the bathing facilities and traditional inns around them. There are approximately 25,000 hot spring sources throughout Japan, and approximately 3,000 *onsen* establishments use naturally hot water from these geothermally heated springs.^[1]

Onsen may be either outdoor baths (露天風呂 or 野天風呂, *roten-buro* / *noten-buro*) or indoor baths (内湯, *uchiyu*). Traditionally, *onsen* were located outdoors, although many inns have now built indoor bathing facilities as well. Nowadays, as most households have their baths, the number of traditional public baths has decreased,^[2] but the number and popularity of hot spring resort towns (温泉街, *onsen-gai*) have increased since the end of Second World War.^[3] Baths may be either publicly run by a municipality or privately, often connecting to a lodging establishment such as a hotel, *ryokan*, or *minshuku*.

The presence of an *onsen* is often indicated on signs and maps by the symbol 湯, the kanji 湯 (*yu*, meaning "hot water"), or the simpler phonetic hiragana character ゆ (*yu*).



Outdoor *onsen* at Nakanoshima in Nachikatsuura, Wakayama



モジュール5 Azure AI Language と Azure Database for PostgreSQL を使用して分析情報を抽出する

- ・「キーフレーズ抽出」とは
- ・「azure_ai拡張機能」からの「キーフレーズ抽出」の利用例
- ・「エンティティ認識」とは
- ・「azure_ai拡張機能」からの「エンティティ認識」の利用例
- ・「エンティティリンク」とは
- ・「azure_ai拡張機能」からの「エンティティリンク」の利用例
- ・「個人を特定できる情報の検出」とは
- ・「azure_ai拡張機能」からの「個人を特定できる情報の検出」の利用例

「個人を特定できる情報の検出」とは

- ・テキストに含まれる電話番号やメールアドレスを特定
- ・特定した部分をマスク処理

"私のメールアドレスは yamada@contoso.com です"



"私のメールアドレスは ***** です"

モジュール5 Azure AI Language と Azure Database for PostgreSQL を使用して分析情報を抽出する

- ・「キーフレーズ抽出」とは
- ・「azure_ai拡張機能」からの「キーフレーズ抽出」の利用例
- ・「エンティティ認識」とは
- ・「azure_ai拡張機能」からの「エンティティ認識」の利用例
- ・「エンティティリンク」とは
- ・「azure_ai拡張機能」からの「エンティティリンク」の利用例
- ・「個人を特定できる情報の検出」とは
- ・「azure_ai拡張機能」からの「個人を特定できる情報の検出」の利用例

第1引数に処理対象の文章、第2引数に文章の言語のコードを指定する。

```
48  SELECT * from azure_cognitive.recognize_pi_entities(  
49  '私のメールアドレスは yamada@contoso.com です。', 'ja')  
50
```

Data Output Messages Notifications



	redacted_text text	entities azure_cognitive.entity[]
1	私のメールアドレスは ***** です。	{"(yamada@contoso.com,Email,\"\",0.8)"}

モジュール 6



Azure AI 翻訳と Azure Database for PostgreSQL を使用してテキストを翻訳する

700 XP

18 分 • モジュール • 6 ユニット

このモジュールでは、Azure AI 翻訳サービスを使用して Azure Database for PostgreSQL - フレキシブル サーバーでテキストデータを翻訳する手順について説明します。PostgreSQL にデータを読み込み、複数の言語に翻訳します。

モジュール6 Azure AI 翻訳と Azure Database for PostgreSQL を使用してテキストを翻訳する

- ・「Azure AI Language」の「翻訳」とは
- ・「azure_ai 拡張機能」からの利用例
- ・モジュール3～6のまとめ

「Azure AI Language」の「翻訳」とは？

- ・入力されたテキストを、指定した言語へと翻訳する
- ・入力テキストの言語は自動で判定される
 - ・（オプションで明示的に入力言語を指定することも可能）

"AIは人間の役に立ちます"



en (English)

"AI can help humans."

モジュール6 Azure AI 翻訳と Azure Database for PostgreSQL を使用してテキストを翻訳する

- ・「Azure AI Language」の「翻訳」とは
- ・「azure_ai 拡張機能」からの利用例
- ・モジュール3～6のまとめ

第1引数に翻訳対象の文章、第2引数に翻訳先の言語のコードを指定する

```
52     SELECT * from azure_cognitive.translate(  
53         'AIは人間の役に立ちます', 'en')  
54
```

Data Output Messages Notifications



SQL

translations

azure_cognitive.translation[]



detected_language

azure_cognitive.detected_language



1

{"(en,\"AI can help humans\"),"}

(ja,1)

第2引数に複数の翻訳先の言語のコードを指定することができる。
また第3引数に元の文章の言語のコードを指定することもできる（省略した場合は自動判定）

```
52  SELECT translations from azure_cognitive.translate(  
53  'AIは人間の役に立ちます', array['en', 'fr'], 'ja')  
54
```

Data Output Messages Notifications



translations

azure_cognitive.translation()



1 {"(en,\"AI can help humans\"),"(fr,\"L'IA peut aider les humains\")}

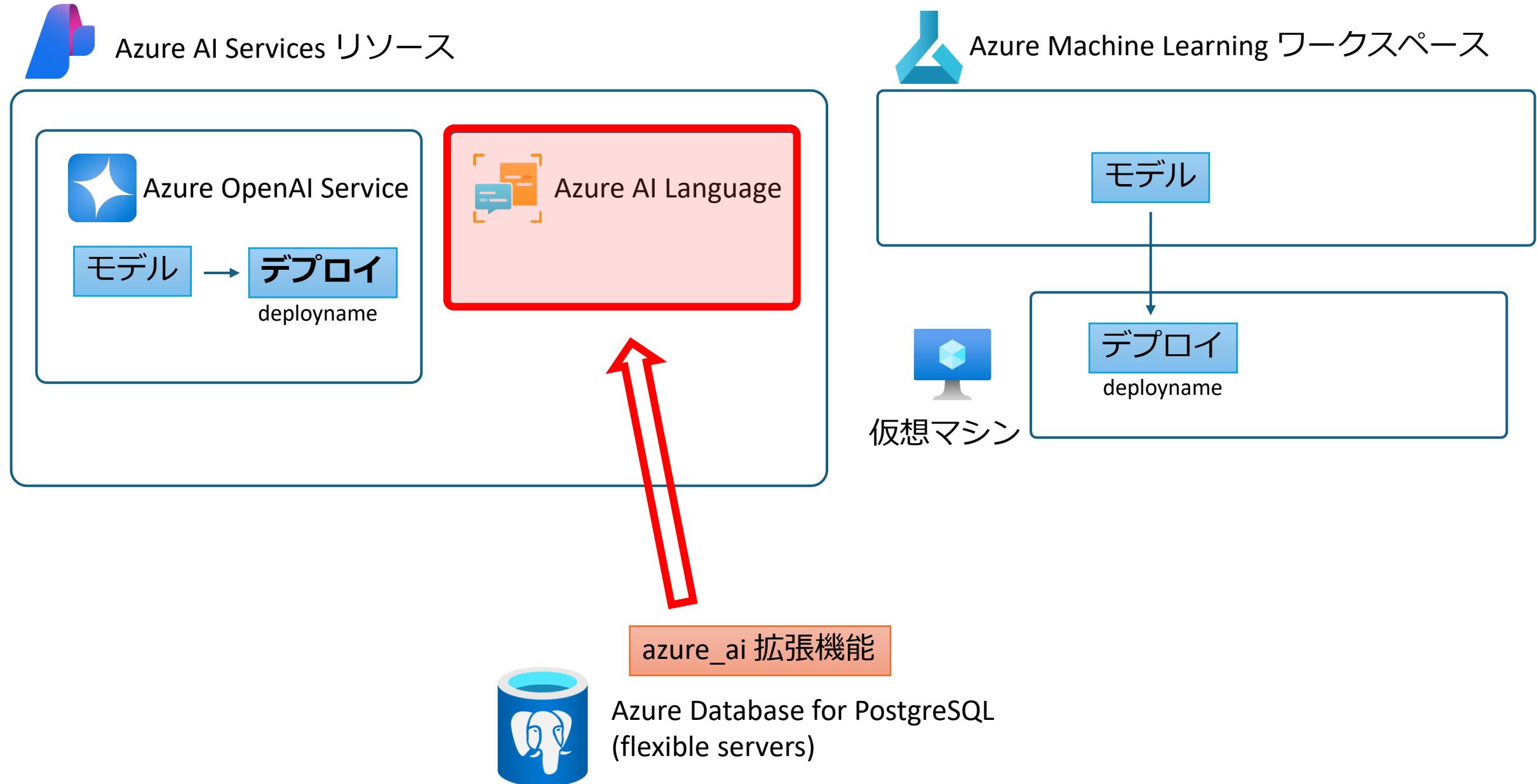
モジュール6 Azure AI 翻訳と Azure Database for PostgreSQL を使用してテキストを翻訳する

- ・「Azure AI Language」の「翻訳」とは
- ・「azure_ai 拡張機能」からの利用例
- ・モジュール3～6のまとめ

モジュール3~6まとめ

モジュール3	抽出要約（原文から重要な文をそのまま抽出）	azure_cognitive.summarize_extractive
	抽象要約（原文の内容をふまえて新しい文を生成）	azure_cognitive.summarize_abstractive
モジュール4	感情分析（テキストの感情をポジティブ・ネガティブ・ニュートラル・ミックスに判定）	azure_cognitive.analyze_sentiment
モジュール5	キーフレーズ抽出（テキスト内の主要な概念を特定）	azure_cognitive.extract_key_phrase
	エンティティ認識（テキスト内の日時、場所などを特定）	azure_cognitive.recognize_entities
	個人を特定できる情報の検出（テキストに含まれる電話番号やメールアドレスを特定）	azure_cognitive.recognize_pii_entities
	エンティティリンク（テキスト内のエンティティを認識してWikipediaページに関連付け）	azure_cognitive.linked_entities
モジュール6	翻訳（テキストを別の言語に翻訳）	azure_cognitive.translate

モジュール3～6では、 Azure AI Languageの利用について説明した。



モジュール7では、 Azure Machine Learningの利用について説明する。



Azure AI Services リソース



Azure OpenAI Service

モデル

→ デプロイ

deployname



Azure AI Language



Azure Machine Learning ワークスペース

モデル



仮想マシン

デプロイ

deployname



Azure Database for PostgreSQL
(flexible servers)



azure_ai 拡張機能

モジュール 7



Azure Database for PostgreSQL からの推論に Azure Machine Learning を使用する

700 XP

14 分 • モジュール • 6 ユニット

このモジュールでは、自動機械学習を使用した Azure Machine Learning モデルのトレーニングとデプロイについて説明します。次に、Azure Database for PostgreSQL - フレキシブル サーバーからモデルを直接呼び出します。

モジュール7 Azure Database for PostgreSQL からの推論に Azure Machine Learning を使 用する

- Azure Machine Learningとは？
- トレーニング済みの機械学習モデルの利用
- azure_ai 拡張機能からのAzure Machine Learningの利用例
- まとめ

Azure Machine Learningとは？

- ・ユーザー独自の「機械学習モデル」をトレーニング・運用するためのサービス。
 - ・今回の講義ではトレーニングの部分の解説は省略。
- ・利用例
 - ・回帰: 賃貸物件の住所や物件情報から、その物件の適切な家賃を算出する
 - ・分類: 健康診断データをもとに、病気かどうかを判定するモデルを開発し、患者の診察に役立てる
 - ・クラスタリング: 通販サイトの購買履歴に基づきユーザーをグループ化するモデルを開発し、グループごとにおすすめ商品を表示するのに役立てる

モジュール7 Azure Database for PostgreSQL からの推論に Azure Machine Learning を使 用する

- Azure Machine Learningとは？
- トレーニング済みの機械学習モデルの利用
- azure_ai 拡張機能からのAzure Machine Learningの利用例
- まとめ

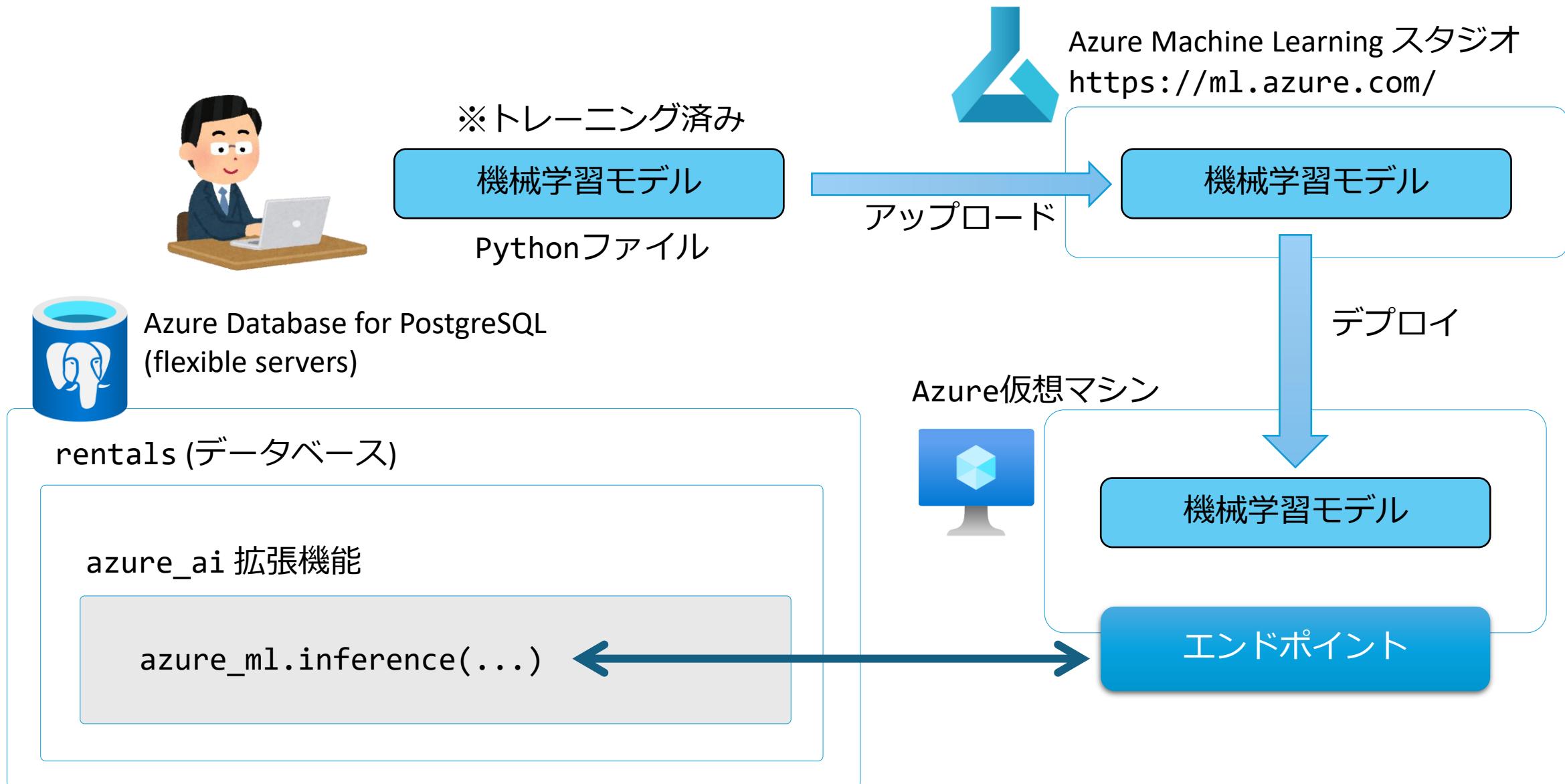
本講義では「賃貸物件の住所や物件情報から、その物件の適切な家賃を算出する」機械学習モデルを例として解説する。このモデルを使用すると、例えば、賃貸物件のオーナーが、物件の適切な家賃額を決定するのに役立つ。

トレーニング済みのモデルは以下から入手できる。

<https://github.com/MicrosoftLearning/mslearn-postgresql.ja-jp/blob/main/Allfiles/Labs/Shared/mlflow-model.zip>



トレーニング済みの機械学習モデルをアップロードして利用する方法



モジュール7 Azure Database for PostgreSQL からの推論に Azure Machine Learning を使 用する

- Azure Machine Learningとは？
- トレーニング済みの機械学習モデルの利用
- azure_ai 拡張機能からのAzure Machine Learningの利用例
- まとめ

azure_ai 拡張機能からの Azure Machine Learningの利用例

- 「Azure Machine Learningワークスペース」 リソースを作成
 - Azure Machine Learningスタジオにアクセス
 - 機械学習モデルをアップロード
 - 機械学習モデルをデプロイ
 - キーとエンドポイント入手
 - azure_ai 拡張機能にて、キーとエンドポイントを設定
 - azure_ml.inference 関数を呼び出す

☰ Microsoft Azure X

Azure サービス

+ リソースの作成

ストレージ アカウント

リソース

すべて サービス (99+) ▼ 増やす (5)

サービス もっと見る

- Azure Machine Learning
- Azure Cosmos DB
- Azure Database for MySQL
- Azure Cache for Redis

Marketplace もっと見る

- Azure Machine Learning
- Azure AI Foundry



Microsoft Azure

リソース、サービス、ドキュメントの検索 (G+/)

Copilot

ホーム >

Azure Machine Learning

Contoso (MngEnvMCAP959990.onmicrosoft.com)

+ Create

Recently deleted

ビューの管理

更新

CSV にエクスポート

クリエイティブ



New workspace

For ML projects and teams

この値と等しいすべて

+ フィルターの追加



New registry

For sharing ML assets across workspaces

グループ化なし

リストビ

右側 ↑↓

ソート グループ ↑↓

種類 ↑↓

場所 ↑↓

[ホーム > Azure Machine Learning >](#)

Azure Machine Learning

Machine Learning ワークスペースを作成する

[基本](#) [Inbound Access](#) [Outbound Access](#) [暗号化](#) [ID](#) [タグ](#) [確認および作成](#)

リソースの詳細

すべてのワークスペースは、課金が発生する Azure サブスクリプションに割り当てる必要があります。フォルダーなどのリソース グループを使用して、作成しようとしているワークスペースを含むリソースを整理および管理します。[Azure リソース グループの詳細情報](#)

サブスクリプション * [?](#) [×](#)リソース グループ * [?](#) [×](#)[新規作成](#)

ワークスペースの詳細

ストレージ接続、認証、コンテナーなどの基本的なワークスペース設定を構成します。[詳細情報](#)

名前 * [?](#) [✓](#)リージョン * [?](#) [×](#)[確認および作成](#)[< 前へ](#)[次へ: Inbound Access](#)

azure_ai 拡張機能からの Azure Machine Learningの利用例

- ・「Azure Machine Learningワークスペース」リソースを作成
- ・Azure Machine Learningスタジオにアクセス
- ・機械学習モデルをアップロード
- ・機械学習モデルをデプロイ
- ・キーとエンドポイント入手
- ・azure_ai 拡張機能にて、キーとエンドポイントを設定
- ・azure_ml.inference 関数を呼び出す

<https://ml.azure.com/>

Machine Learning Studio

mlwksp

+ 新規 表示のカスタマイズ

プロンプト フローを使用した生成型 AI





Multi-Round Q&A on Your Data

Create a chatbot that uses LLM and data from your own indexed files to ground multi-round question and answering capabilities in enterprise chat scenarios.

開始 クローン



Q&A on Your Data

Use LLM and data from your own indexed files to ground multi-round question and answering capabilities.

開始 クローン



Web Classification

Use LLM to classify URLs into multiple categories.

開始 クローン



Chat with

Create a chatbot to ground the...

開始

生成 AI モデル

すべて表示 < >

∞ Llama-4-Scout-17B-16E-Instruct ∞ Llama-4-Maverick-17B-128E-I... ∞ Llama-4-Scout-17B-16E ∞ Llama-

azure_ai 拡張機能からの Azure Machine Learningの利用例

- ・「Azure Machine Learningワークスペース」リソースを作成
- ・Azure Machine Learningスタジオにアクセス
- ・機械学習モデルをアップロード
- ・機械学習モデルをデプロイ
- ・キーとエンドポイント入手
- ・azure_ai 拡張機能にて、キーとエンドポイントを設定
- ・azure_ml.inference 関数を呼び出す

[← すべてのワークスペース](#)[ホーム](#)[モデル カタログ](#)**作成**[Notebooks](#)[自動 ML](#)[デザイナー](#)[プロンプト フロー](#)**アセット**[データ](#)[ジョブ \(Job\)](#)[コンポーネント](#)[パイプライン](#)[環境](#)[モデル](#)

Contoso > mlwksps > モデル

モデル一覧



- [ローカル ファイルから](#)
 - [ジョブ出力から](#)
 - [データストアから](#)
 - [ローカル ファイルから \(フレームワークに基づく\)](#)
- [+ 登録 ▾](#)

[Github リポジトリを探索する](#) [Azure Machine Learning チュートリアルを表示する](#)

バージョンとして作成

新しいバージョンであるモデルを検索するときに、モデルを登録することもできます。 [詳細情報](#)

← ローカルファイルからモデルを登録する ×



① モデルのアップロード

× モデルの設定

③ レビュー

モデルのアップロード

モデルを含むファイルまたはフォルダーをアップロードします。

モデルの種類 * ⓘ

MLflow

モデル フォルダーを選択する *

参照

5 個のファイルが選択されました。合計サイズ: 2.55 MB

ファイル名

ファイルサイズ

mlflow-model/conda.yaml

5.52 KB

mlflow-model/MLmodel

1.26 KB

mlflow-model/model.pkl

2.54 MB

mlflow-model/python_env.yaml

120 バイト

mlflow-model/requirements.txt

4.16 KB

戻る

次へ

キャンセル

← ローカルファイルからモデルを登録する ×



モデルのアップロード



モデルの設定



レビュー

モデルの設定

モデルの設定を構成します。

名前*

RentalListings



*

説明

バージョン ⓘ

このモデル名で次に使用可能なバージョン: 1

タグ

 : 追加

① タグ がありません

戻る

次へ

キャンセル

← ローカルファイルからモデルを登録する ×



モデルのアップロード



モデルの設定



③ レビュー

レビュー

選択内容をレビューまたは変更します。

モデルのアップロード



モデルの種類

MLflow

ファイル

5 個のファイルが選択されました。合計サイズ: 2.55 MB

mlflow-model/conda.yaml

mlflow-model/MLmodel

mlflow-model/model.pkl

mlflow-model/python_env.yaml

mlflow-model/requirements.txt

モデルの設定



名前

RentalListings

説明

--

バージョン

--

タグ

タグ がありません

戻る

登録

キャンセル

azure_ai 拡張機能からの Azure Machine Learningの利用例

- ・「Azure Machine Learningワークスペース」リソースを作成
- ・Azure Machine Learningスタジオにアクセス
- ・機械学習モデルをアップロード
- ・機械学習モデルをデプロイ
- ・キーとエンドポイント入手
- ・azure_ai 拡張機能にて、キーとエンドポイントを設定
- ・azure_ml.inference 関数を呼び出す

Contoso > mlwksp > モデル

< すべてのワークスペース

ホーム

モデルカタログ

作成

Notebooks

自動 ML

デザイナー

プロンプトフロー

アセット

データ

ジョブ (Job)

コンポーネント

パイプライン

環境

モデル

エンドポイント

管理

コンピューティング

モデル一覧

+ 登録

最新の情報に更新

削除

アーカイブ

...

...

最新バージョンのみを表示



検索

名前

☆

バージ...

種類



RentalListings

1

MLFLOW

リアルタイム エンドポイント
リアルタイム エンドポイント ウィザードを使用してモデルをデプロイする

バッチ エンドポイント
バッチ エンドポイント ウィザードを使用してモデルをデプロイする

Web サービス
Web サービスにデプロイする (フレームワークに基づくモデルのみ)

< 前へ

次へ >

25/ページ ▾

Contoso > mlwksp > モデル

✓ 成功: モデルは 'RentalListings:1' 正常に登録されました。

モデル一覧

+ 登録 ▾ 最新の情報に更新 削除

検索

1 RentalListings

仮想マシンの台数

仮想マシンのスペック

デプロイ名

RentalListings:1 のデプロイ

選択したモデルに対して、スコアリング スクリプトと環境が自動的に生成されます。
[詳細情報](#)

プロジェクト名
mlwksp

インスタンス数 *
1

仮想マシン *
Standard_DS2_v2 2 コア, 7 GB (RAM), 14 GB (ディスク), \$0.20/時間

エンドポイント

新規 既存

エンドポイント名 *
mlwksp-libnm

エンドポイント URL は、エンドポイントの作成後に生成されます。
[実行方法を調べる](https://mlwksp-libnm.japaneast.inference.ml.azure.com/score)

デプロイ名 *
rentallistings-1

推論データ収集 無効

パッケージ モデル 無効

デプロイ **キャンセル** **その他のオプション**

mlwksp-libnm

詳細 テスト 使用 監視 プレビュー ログ

+ デプロイの追加

⟳ 最新の情報に更新

✎ トライフィックを更新する

刪除

エンドポイント属性

サービス ID

mlwksp-libnm

説明

--

プロビジョニングの状態

成功

エラーの詳細

--

コンピューティングの種類

マネージド

作成者

System Administrator

作成日

Apr 6, 2025 6:46 PM

最終更新日時

展開の概要

ライブ トライフィックの割り当て

✓ rentallistings-1 (0%)

ミラー化されたトライフィックの割り当て

--

デプロイ rentallistings-1

名前

rentallistings-1

ライブ トライフィック

0%

スコアリングスクリプト

自動生成

プロビジョニングの状態

✓ 成功

azure_ai 拡張機能からの Azure Machine Learningの利用例

- ・「Azure Machine Learningワークスペース」リソースを作成
- ・Azure Machine Learningスタジオにアクセス
- ・機械学習モデルをアップロード
- ・機械学習モデルをデプロイ
- ・キーとエンドポイント入手
- ・azure_ai 拡張機能にて、キーとエンドポイントを設定
- ・azure_ml.inference 関数を呼び出す

すべてのワークスペース

ホーム

モデル カタログ

作成

Notebooks

自動 ML

デザイナー

プロンプト フロー

アセット

データ

Contoso > mlwksp > エンドポイント > mlwksp-libnm

mlwksp-libnm

詳細 テスト **使用** 閲覧 プレビュー ログ

基本消費量に関する情報

REST エンドポイント

`https://mlwksp-libnm.japaneast.inference.ml.azure.com/score`

認証

主キー

`.....`

再生成

セカンダリキー

`.....`

再生成

azure_ai 拡張機能からの Azure Machine Learningの利用例

- ・「Azure Machine Learningワークスペース」リソースを作成
- ・Azure Machine Learningスタジオにアクセス
- ・機械学習モデルをアップロード
- ・機械学習モデルをデプロイ
- ・キーとエンドポイント入手
- ・azure_ai 拡張機能にて、キーとエンドポイントを設定
- ・azure_ml.inference 関数を呼び出す

キーとエンドポイントを azure_ai 拡張機能にセットする。

```
SELECT azure_ai.set_setting('azure_ml.scoring_endpoint', 'https://i
```

```
SELECT azure_ai.set_setting('azure_ml.endpoint_key', '
```

azure_ai 拡張機能からの Azure Machine Learningの利用例

- ・「Azure Machine Learningワークスペース」リソースを作成
- ・Azure Machine Learningスタジオにアクセス
- ・機械学習モデルをアップロード
- ・機械学習モデルをデプロイ
- ・キーとエンドポイント入手
- ・azure_ai 拡張機能にて、キーとエンドポイントを設定
- ・`azure_ml.inference` 関数を呼び出す

azure_ml.inference関数の引数

```
azure_ml.inference(  
    'データ',  
    deployment_name=>'デプロイ名'  
)
```

予測を行うための元データ
(住所、物件条件など) を
JSON形式で指定

```
24 ▼ SELECT azure_ml.inference('{  
25   "input_data": {  
26     "columns": [  
27       "host_is_superhost",  
28       "host_has_profile_pic",  
29       "host_identity_verified",  
30       "neighbourhood_group_cleansed",  
31       "zipcode",  
32       "property_type",  
33       "room_type",  
34       "accommodates",  
35       "bathrooms",  
36       "bedrooms",  
37       "beds"  
38     ],  
39     "index": [0],  
40     "data": [[0, 0, 0, "Central Area", "98122", "House", "Entire home/apt", 4, 1.5, 3, 3]]  
41   }  
42 }', deployment_name=>'rentallistings-1')  
43
```

デプロイ名

予測を行うための元データ
(住所、物件条件など)

Data Output Messages Notifications



Showing rows: 1 to 1

	inference	jsonb
1	[249.29489644470468]	

モデルによって予測された値 (家賃)

```
24 ▼ SELECT azure_ml.inference('
25   "input_data": {
26     "columns": [
27       "host_is_superhost",
28       "host_has_profile_pic",
29       "host_identity_verified",
30       "neighbourhood_group_cleansed",
31       "zipcode",
32       "property_type",
33       "room_type",
34       "accommodates",
35       "bathrooms",
36       "bedrooms",
37       "beds"
38     ],
39     "index": [0, 1],
40     "data": [[0, 0, 0, "Central Area", "98122", "House", "Entire home/apt", 4, 1.5, 3, 3],
41               [1, 1, 1, "Queen Anne", "98109", "Apartment", "Private room", 2, 1, 1, 1]]
42   }
43 ', deployment_name=>'rentallistings-1')
```

Data Output

Messages Notifications



Showing rows: 1 to 1

	inference	jsonb
1	[249.294896444704]	[93.81349554262616]

複数のデータ
(住所、物件条件など) を
同時に投入して、
複数の予測値を得ることも可能。

モジュール7 Azure Database for PostgreSQL からの推論に Azure Machine Learning を使 用する

- Azure Machine Learningとは？
- トレーニング済みの機械学習モデルの利用
- Azure Machine Learningの基本的な利用の流れ
- azure_ai 拡張機能からのAzure Machine Learningの利用例
- まとめ

モジュール7のまとめ

- Azure Machine Learningは、機械学習モデルをトレーニング・運用するためのサービス。
 - モデルの例: 住所や物件情報から適正な家賃を予測するモデル
- モデルを仮想マシンにデプロイすることで、そのモデルを外部から利用できる。
- デプロイされたモデルにデータを送信すると、モデルによる処理結果（予測値など）が得られる（推論、inference）。
- azure_ai 拡張機能の `azure_ml.inference` を使用して、PostgreSQL からモデルを利用できる。

全体のまとめ (1/2)

モジュール 1	azure_ai 拡張機能の 概要	Azure Database for PostgreSQLでazure_ai 拡張機能を利用して、データベースから生成AI（埋め込み）・Azure AI Language（自然言語処理）・Azure Machine Learningなどの機能を利用できる。
モジュール 2	Azure OpenAI Serviceの 「埋め込み」の利用	埋め込みモデルを使用して、テキストのベクトルを計算できる。2つのテキストのベクトルのコサイン類似度を計算することで、意味的に近い文章を検索できる。
モジュール 3-6	Azure AI Language の利用 (自然言語処理)	<ul style="list-style-type: none">抽出要約（原文から重要な文をそのまま抽出）抽象要約（原文の内容を理解し、新しい文を生成）感情分析（入力されたテキストの感情をポジティブ・ネガティブ・ニュートラル・ミックスなどに判定）キーフレーズ抽出（テキスト内の主要な概念を特定）エンティティ認識（テキスト内の日時、場所などを特定）個人を特定できる情報の検出（テキストに含まれる電話番号やメールアドレスを特定）エンティティリンク（テキスト内のエンティティを認識して Wikipediaページに関連付け）翻訳（テキストを別の言語に翻訳）
モジュール 7	Azure Machine Learningの利用	Azure Machine Learningで、機械学習モデルを仮想マシンにデプロイし、データベースからのモデルの利用（推論）が可能。

全体のまとめ (2/2)

やりたいこと	使う機能
文章を短い文に要約したい！	抽出要約・抽象要約
文章を別の言語（日本語・英語など）に翻訳したい！	翻訳
文章から重要なキーワードを抜き出したい！	キーフレーズ抽出
文章の感情を判定したい！（アンケートから改善ポイントを抽出するなど）	感情分析
文章から「日時」や「場所」などを取り出したい！	エンティティ認識
文章から「地名」などの単語を取り出し、各単語の詳細を確認したい！	エンティティリンク
文章から電話番号・メールアドレスなどを検出・削除したい！	個人を特定できる情報の検出
入力された質問などに関連がある文章を検索したい！	ベクトル検索（埋め込みによるベクトル生成 + コサイン類似度による類似度判定）
機械学習モデルを呼び出してデータから予測値を計算したい！	Azure Machine Learningのモデルの呼び出し（推論）
上記以外のAI機能をPostgreSQLから呼び出したい！（GPTを使用して質問に対する回答を生成するなど）	関数を自作する