

アーキテクチャ図

[ユーザー]

|



[Streamlit UI (TransEmbPic)]

|

| (1) 画像アップロード (PNG/JPG)



[Langchain (統括システム: OCR エージェント、翻訳エージェント、埋込・保存エージェントのロジックを内包)]

| |

| | (2) 画像データを OCR 処理指示

| | \_\_\_\_\_> [Azure Computer Vision (OCR API)]

| | \_\_\_\_\_|

| | (3) 抽出された英語テキスト

| | \_\_\_\_\_|

| | \_\_\_\_\_|

| | (4) 英語テキストを翻訳指示

| | \_\_\_\_\_> [Azure Translator (Text API)]

| | \_\_\_\_\_|

| | (5) 翻訳された日本語テキスト

| | \_\_\_\_\_|

| | \_\_\_\_\_|

| | (6) 日本語テキストをベクトル化指示

| | \_\_\_\_\_> [Azure OpenAI (Embedding API)]

| | \_\_\_\_\_|

| | (7) 生成されたベクトルデータ

| | \_\_\_\_\_|

| | \_\_\_\_\_|

| | (8) 元画像、翻訳テキスト、ベクトルデータ、メタ情報を保存指示

| | \_\_\_\_\_> [Azure Cosmos DB (JSON ドキュメント + ベクトルインデックス)]

| | \_\_\_\_\_| (翻訳文、メタ情報、ベクトル保存)

| | \_\_\_\_\_|

| | \_\_\_\_\_|

| | (9) 元画像と翻訳テキストで画像合成(Pillow)し、保存指示

| | \_\_\_\_\_> [Azure Blob Storage (加工済み画像保存)]

| | \_\_\_\_\_|

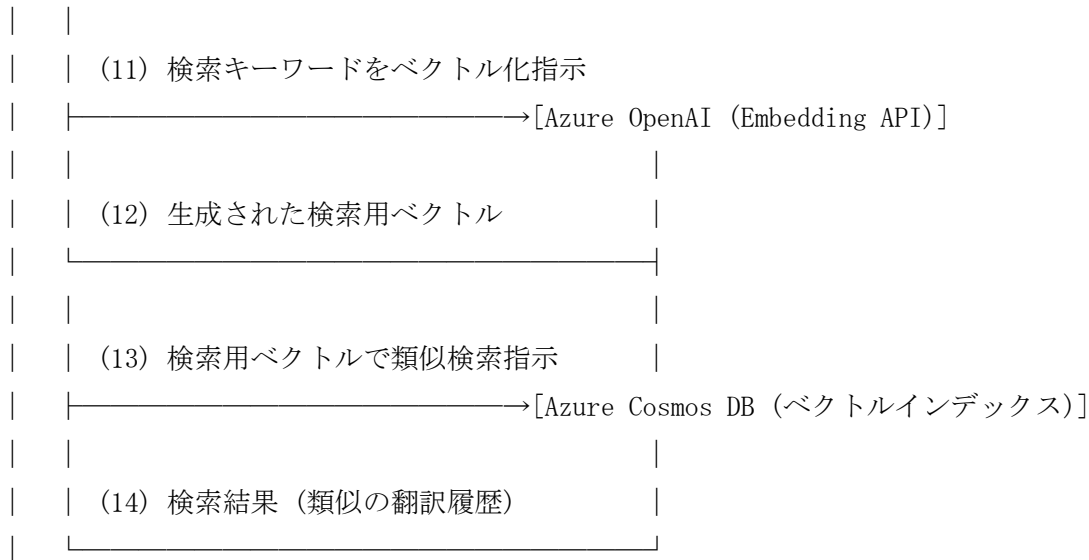
| (10) 翻訳履歴の検索キーワード入力 (UI 経由)

・タイトル: TranslateEmbAgent (翻訳埋込エージェント)

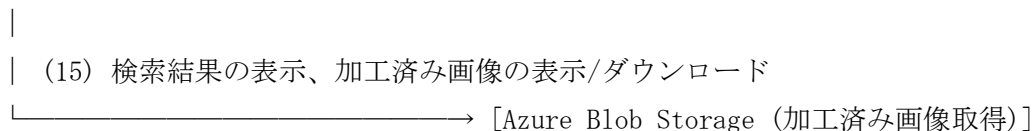
・プロダクト名: TransEmbPic (トランスエンブピック)



[Langchain (統括システム)]



[Streamlit UI (TransEmbPic)]



#### フロー説明:

1. **画像のアップロード:** ユーザーは **Streamlit UI (TransEmbPic)** を通じて、翻訳したい外国語（英語）が含まれる PNG/JPG 画像をアップロードします。
2. **OCR 処理指示:** **Langchain**（統括システム）はアップロードされた画像を受け取り、内包する OCR エージェントのロジックに基づき、**Azure Computer Vision (OCR API)** に画像内のテキスト抽出を指示します。
3. **英語テキスト取得:** **Azure Computer Vision** から抽出された英語テキストが **Langchain** に返されます。
4. **翻訳指示:** **Langchain** は、翻訳エージェントのロジックに基づき、抽出された英語テキストを **Azure Translator (Text API)** に渡し、日本語への翻訳を指示します。
5. **日本語テキスト取得:** **Azure Translator** から翻訳された日本語テキストが **Langchain** に返されます。
6. **ベクトル化指示:** **Langchain** は、埋込・保存エージェントのロジックに基づき、翻訳された日本語テキストを **Azure OpenAI (Embedding API)** に渡し、テキストのベクトル表現 (Embedding) の生成を指示します。
7. **ベクトルデータ取得:** **Azure OpenAI** から生成されたベクトルデータが **Langchain** に返されます。
8. **データ保存 (Cosmos DB):** **Langchain** は、元画像のメタ情報、翻訳された日本語テキスト、そして生成されたベクトルデータを **Azure Cosmos DB** に保存します。Cosmos DB は JSON ドキュメン

トとして情報を格納し、ベクトルデータは検索用のインデックスとしても利用されます（CRUD 操作の Create）。

9. **画像合成と保存 (Pillow & Blob Storage):** Langchain は、埋込・保存エージェントのロジックと Pillow ライブラリを使用して、元の画像に翻訳された日本語テキストを埋め込みます。完成した加工済み画像は **Azure Blob Storage** に保存されます。
10. **翻訳履歴検索:** ユーザーは **Streamlit UI** を通じて、過去の翻訳履歴を検索するためのキーワードを入力します。
11. **検索キーワードのベクトル化:** Langchain は入力された検索キーワードを受け取り、**Azure OpenAI (Embedding API)** を使ってキーワードをベクトル化します。
12. **検索用ベクトル取得:** **Azure OpenAI** から検索キーワードのベクトルデータが **Langchain** に返されます。
13. **類似検索指示 (Cosmos DB):** Langchain は、生成された検索用ベクトルを使って **Azure Cosmos DB** のベクトルインデックスに対し、類似する翻訳履歴の検索を指示します（CRUD 操作の Read）。
14. **検索結果取得:** **Azure Cosmos DB** から類似度の高い翻訳履歴データ（過去の翻訳文、メタ情報、関連する加工済み画像の参照など）が **Langchain** に返されます。
15. **結果表示と画像アクセス:** Langchain は検索結果を **Streamlit UI** に渡し、ユーザーに表示します。ユーザーは結果に基づいて、**Azure Blob Storage** に保存されている関連の加工済み画像を表示したりダウンロードしたりできます。