

Streamlit×Gemini API 箇条書き→レポート 自動変換アプリ - プレゼン原稿

発表時間: 約4分

作成者: Manus AI

スライド1: タイトルスライド (30秒)

皆さん、こんにちは。本日は「Streamlit×Gemini API 箇条書き→レポート自動変換アプリ」について発表させていただきます。

このアプリケーションは、メモや箇条書きの内容を入力するだけで、AIが自動的に読みやすいレポート形式の文章に変換してくれるWebアプリケーションです。

使用技術は、Streamlit、Gemini API、Python、そして自然言語処理技術を組み合わせており、シンプルなWebアプリで文章作成を効率化することを目的としています。

スライド2: アプリケーション概要 (40秒)

まず、このアプリケーションが解決する課題と提供する価値について説明します。

多くの方が経験されていると思いますが、箇条書きから文章を作るのに時間がかかったり、文章構成を考えるのが難しかったり、自然な日本語表現に悩むことがあります。また、レポート作成の効率化が求められる場面も多いでしょう。

このアプリケーションは、これらの課題に対して具体的な解決策を提供します。ワンクリックで自動変換が可能で、AIが最適な文章構成を提案し、自然で読みやすい日本語に変換することで、作業時間を大幅に短縮できます。

特に学生、研究者、ビジネスパーソンの文章作成を効率化することを目指しており、日常的な文書作成業務の生産性向上に貢献します。

スライド3: 技術スタック（30秒）

次に、使用している技術スタックについて説明します。

まず、Streamlitです。これはWebアプリケーションフレームワークで、簡単なUI作成、リアルタイム更新、そしてPythonのみで開発できるという特徴があります。

次に、Gemini APIです。これはGoogleの生成AI APIで、高性能な言語モデル、日本語対応、そして優れた文章生成・変換能力を持っています。

そして、Pythonプログラミング言語です。シンプルな構文、豊富なライブラリ、そしてAI開発に最適な環境を提供します。

これらの技術を選択した理由は、開発効率の観点からStreamlitにより最小限のコードでWebアプリを構築でき、AI性能の観点からGemini APIの高品質な日本語文章生成能力を活用できるからです。

スライド4: コード構造と処理フロー（40秒）

コードの構造と処理フローについて詳しく見てみましょう。

完全なソースコードは非常にシンプルで、わずか15行程度で実装されています。まず、必要なライブラリをインポートし、Gemini APIキーを設定します。次に、Streamlitでタイトル、テキストエリア、ボタンを表示し、ユーザーの入力を受け取ります。

処理フローは4つのステップで構成されています。

第一に、初期設定として、Gemini APIキーの設定とライブラリのインポートを行います。

第二に、UI表示として、タイトル、テキストエリア、ボタンの表示を行います。

第三に、AI処理として、プロンプト作成とGemini APIへのリクエストを実行します。

第四に、結果表示として、生成されたレポートをWebページに表示します。

このシンプルな構造により、保守性が高く、理解しやすいコードになっています。

スライド5: UIコンポーネント（30秒）

Streamlitで使用しているUIコンポーネントについて説明します。

まず、タイトル表示です。st.titleを使用してアプリケーションのメインタイトルを表示します。

次に、テキストエリアです。st.text_areaを使用して、ユーザーが箇条書きを入力するための大きなテキストボックスを提供します。高さは200ピクセルに設定しています。

そして、実行ボタンです。st.buttonを使用して、レポート生成処理を開始するトリガーとなるボタンを配置します。

最後に、結果表示です。st.markdownとst.writeを使用して、生成されたレポートをMarkdown形式で表示します。

Streamlitの特徴として、わずか数行のコードで直感的なWebUIを作成でき、リアルタイムでの状態管理と自動更新により、ユーザーフレンドリーなインターフェースを簡単に実現できます。

スライド6: Gemini API統合（30秒）

Gemini APIの統合とプロンプトエンジニアリングについて説明します。

API設定では、google.generativeaiライブラリをインポートし、StreamlitのsecretsでAPIキーを安全に管理します。そして、gemini-1.5-flashモデルを初期化し、generate_contentメソッドでコンテンツを生成します。

プロンプトエンジニアリングでは、効果的なプロンプト設計が重要です。明確な変換指示、レポート形式の指定、日本語での出力指定、そしてユーザーの箇条書きを適切に挿入することで、高品質な変換結果を得ることができます。

具体的なプロンプトは、「以下の箇条書きを、まとめたレポート形式の日本語文章に変換してください」という指示に、ユーザーの入力を組み合わせたシンプルな構造になっています。

Gemini 1.5 Flashは、高速で高品質な日本語文章生成を実現し、自然で読みやすい文章を短時間で生成できます。

スライド7: 実行結果デモ（30秒）

実際のアプリケーション画面をご覧ください。

このデモでは、太宰治の『斜陽』に関する箇条書きを入力例として使用しています。ユーザーが箇条書きを入力し、「レポート生成」ボタンをクリックすると、AIが自動で文章構成を整理し、読みやすいレポート形式で出力されます。

変換の特徴として、箇条書きから自然な文章への変換、論理的な構成での段落分け、読みやすい日本語表現、そして文学作品の内容を適切に要約する能力が確認できます。

処理時間は数秒でレポート生成が完了し、ユーザーは待ち時間なく結果を確認できます。この高速性により、実用的なツールとして日常的に使用することが可能です。

スライド8: まとめと今後の展開（30秒）

最後に、このアプリケーションの利点と今後の改善・拡張案についてまとめます。

アプリケーションの利点として、まず開発効率があります。わずか15行のコードで実用的なWebアプリケーションを構築できました。次に使いやすさです。直感的なUIで誰でも簡単に文章変換が可能です。そして高品質です。Gemini APIによる自然で読みやすい文章生成を実現しています。

今後の改善・拡張案として、文章スタイル選択機能、ファイル入出力機能、多言語対応、テンプレート機能、履歴管理機能などが考えられます。これらの機能により、より幅広いユースケースに対応できるツールに発展させることができます。

StreamlitとAIを組み合わせることで、文章作成を革新し、より効率的で創造的な作業環境を提供できることを実証できました。

ご清聴ありがとうございました。

時間配分詳細

- スライド1（タイトル）：30秒
- スライド2（アプリ概要）：40秒
- スライド3（技術スタック）：30秒
- スライド4（コード構造）：40秒
- スライド5（UIコンポーネント）：30秒
- スライド6（Gemini API統合）：30秒
- スライド7（実行結果デモ）：30秒
- スライド8（まとめ）：30秒

合計: 4分20秒

発表のポイント

- 技術的な詳細と実用性のバランスを保つ
- 実際のデモ画面を活用して具体性を持たせる
- シンプルなコードで高機能を実現していることを強調
- 今後の拡張可能性を示すことで発展性をアピール

- ・ 聴衆が実際に使ってみたくになるような魅力を伝える