TextAI: From Curiosity to Prototype

A Learning Project with Practical Outcomes

Vision :

TextAI began as a personal curiosity project to explore:

- "Why read when I can listen?" — Adding **Text-to-Speech (TTS)** for faster content consumption
- "Most sites don't summarize well." — Implementing **AI-powered summarization**
- "What if I could process PDFs, URLs, images too?" — Supporting **multiple input formats**

Current Status: Functional prototype for personal use.

Tech Stack & Libraries

Component	Technology	Rationale
Frontend	Streamlit	Rapid UI prototyping in Python
Text Extraction	PyPDF2, BeautifulSoup, Tesseract	Extracts text from PDFs, web pages, and images
Summarization	Hugging Face (BART, DistilBART), GPT-3.5	Balance between open-source control and API convenience
Text-to-Speech	gTTS	Lightweight, no authentication required
Audio Sync	HTML/CSS with Timed Delays	Simulated highlighting to sync with audio

Hugging Face Models

Models Used

- facebook/bart-large-cnn: Main summarization model
    - Abstractive summarization
    - Chunked processing for memory efficiency (300-word segments)
- sshleifer/distilbart-cnn-12-6: Lightweight alternative
    - 40% smaller with minor tradeoffs in summary quality

Prototype Features

Core Functionality

- Adjustable-speed TTS (1x to 2x)
- Context-aware summarization (three-sentence summaries)
- Supports multiple input types: PDFs, URLs, screenshots
- Optional local-only processing with Hugging Face models

Technical Challenges and Solutions

Challenge	**Solution**
Audio-text synchronization	Simulated highlighting with timed delays
Model latency in local mode	DistilBART usage and GPU acceleration
OCR inaccuracies	Image preprocessing pipelines
Streamlit UI limitations	Custom HTML/CSS overlays

Performance Improvements

- **Speed**: Achieved 3x faster processing via chunk parallelization
- **Accuracy**: Combined Hugging Face and GPT-3.5 fallback for reliable summaries

- **Usability**: Added progress indicators and interactive user controls

Future Development

- **Browser Extension**: Summarize and read content directly from web pages
- **Voice Integration**: Add Whisper STT for voice command interaction
- **Domain Adaptation**: Fine-tuned models for specific fields like legal and medical documents

Key Insights

- **Technical**: Hugging Face models require significant optimization for responsiveness and scalability
- **Design**: Most development time was spent on handling edge cases—format inconsistencies, input errors, OCR issues
- **Next Steps**: Benchmarking performance on long-form documents (10,000+ words) and multilingual input support