🧠Building a Research Assistant with AI: From Tool to Platform

🚩The Challenge

I set out to solve a simple but common problem:

"How can I make research faster, smarter, and less painful when working with long academic or technical documents?"

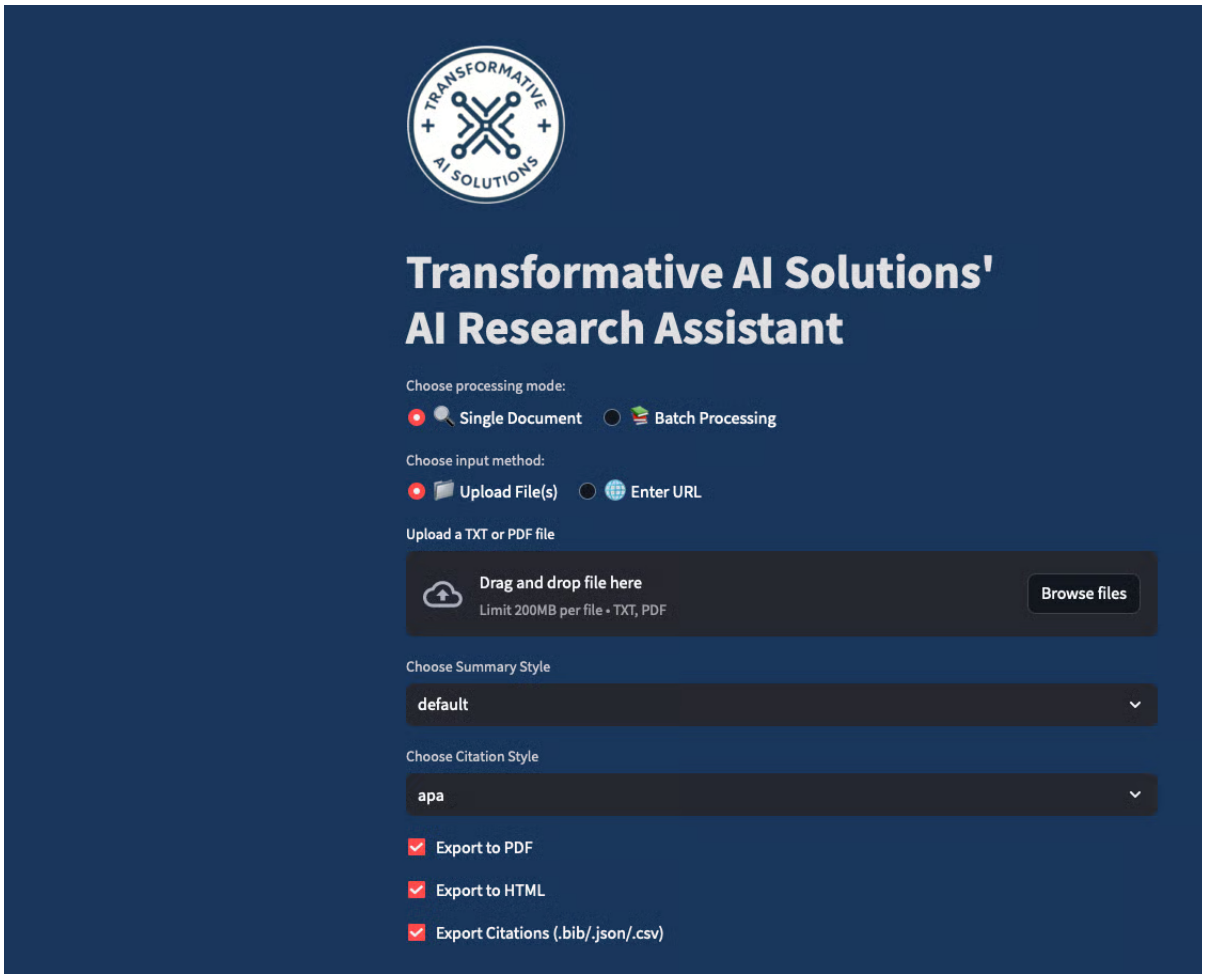
The original idea was to build a lightweight tool for summarizing PDFs using GPT. But the project quickly evolved into something more—a **modular, extensible AI-powered research assistant** that supports multiple formats, automates citation handling, and exports professional summaries with just a few clicks.

🔧The Solution

I built a full-featured **AI-powered web app using Streamlit** that integrates:

- 🧾 PDF and TXT file uploads

- 🌐 URL processing for online articles or PDFs
- 🤖 GPT-4 Turbo summarization (multiple styles)
- 📚 Citation detection and formatting (APA, MLA)
- 🧠 Topic detection
- 📤 Export to TXT, PDF, Markdown, HTML, BibTeX, CSV, and JSON
- 🔐 Local login system (with future encrypted database support)
- 🧪 Batch file processing
- 🗂️ Timestamped outputs and in-app previews



Research Assistant User Interface

🔍Feature Highlights

✨ Smart Summaries

Users can choose summary styles, reprocess specific sections, or ask the assistant to "Explain this further."

📄 Citation Wizard

The assistant detects in-text citations, fetches metadata via DOI/URL, and reformats them into APA or MLA automatically.

📂 Batch Processing

Upload multiple files at once (coming soon with progress bar support). Each document is summarized and cited separately.

📤 Flexible Exporting

Users can export results to multiple formats with a single click. File names are automatically timestamped for easy organization.

🧰 The Stack

Core Tools

- Streamlit for the interface
- OpenAI GPT-4 Turbo for summarization and explanations
- pdfplumber, PyPDF2, and BeautifulSoup4 for parsing
- fpdf for PDF export
- YAML + streamlit-authenticator for login and user config (Temporary, Soon to be replaced with SQLite)
- Custom Python utilities for citation parsing, formatting, and file management

```
ser", "content": prompt}]

].message.content

e="default"):
ot(text, style)
)


n more detail:\n\n{text}"
)

:
t summary or context of this source based on the title alone:\n\n{title}"
)


rompt(text)
)

):
rom a web page URL"""


zilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome,


et(url, headers=headers, timeout=10)
atus()

response.content, 'html.parser')

tyle elements
script", "style", "nav", "header", "footer", "aside"]):
)

)


 for line in text.splitlines())
() for line in lines for phrase in line.split("  "))
 for chunk in chunks if chunk)



s.RequestException as e:
or fetching URL: {str(e)}")

or parsing content: {str(e)}")
```

Sample Python Code

💡Technical Decisions That Paid Off

- **Modularity**: I kept the codebase clean by separating citation logic, export utilities, and AI interaction modules.
- **Scalability-first**: Even before SaaS considerations, I built with growth in mind—batch processing, export

options, and file management all scale well.
- **User-first UX**: I prioritized ease of use: checkbox controls, file previews, and download buttons make this accessible to non-technical users.

📈Impact & Use Cases

- While it started as a personal project, the app has real potential for:

    - **Students** writing annotated bibliographies
    - **Researchers** conducting literature reviews
    - **Consultants** extracting insights from dense technical or industry reports
    - **Writers and journalists** summarizing long-form research

  “Finally, a research tool that doesn’t just summarize, but also tells me where the information came from.” — hypothetical future fan

🚀What’s Next?

- I'm preparing the app for a potential SaaS offering. Future additions include:

    - ✅ **Encrypted database for user sessions & projects**
    - ✅ **Project save/load features**
    - 🌍 **Multilingual summarization**
    - 📊 **Visual dashboards and infographics**
    - 🧩 **Prompt customization**
    - 📋 **End-to-end research project workflow support**

💼Work With Me

- This project is just one example of how AI tools—when applied thoughtfully—can solve real-world workflow problems

problems.

If you're a:

- Research-heavy organization
- Educational platform
- Knowledge-based consultant or analyst
- Startup building custom AI workflows
- Experienced Python Coder

👋 I'd love to work with you.

💬 Let’s Talk About AI Solutions
🗂️ Explore More Projects

Insights

Explore AI consulting resources and product information.

Contact:
admin@transformativeaisolutions.com

Connect

Your Email Address

Enter your email her

Submit Your Inquir