Amlgo Labs Task 2

Task: Web application that can translate PDF files between Hindi and English.

Approach:

- **Programming Language**: Python with version 3.10
- **UI or Frontend**: Streamlit (for quick deployment + simple UI)
- **PDF Handling**: PyMuPDF, Fitz, pdfplumber, or pdfminer.six for text extraction
- **PDF generation**: reportlab or PyMuPDF for PDF generation
- **Translation:** Google Translate API (free tier) or Hugging Face Transformers and we have Deep translator Python Library
- **Translation Model English to Hindi**: Helsinki-NLP opus-mt-en-hi
- **Translation Model Hindi to English**: Helsinki-NLP/opus-mt-hi-en
- **Deployment**: Streamlit Cloud (free and fast)
- Architecture: Extract text to pdf – Translation – Reconstruct PDF

Additionally, I can do translation pdf without open source (Hugging face) or APIs using Custom Translator Transformer that I do from scratch.

I used Deep translator that give me free functionality to use unlimited translator services (Libra, Google, Microsoft) without APIs.

Folder Structure:

How to Run:

conda create --name pdftrans python=3.10

conda activate pdftrans

Pip install -r requirements.txt

Streamlit run app.py

Pipeline

📁Src/config.py

- Contain Supported languages, Translation directions, File upload settings, Text processing settings, PDF generation settings, Error messages, Success messages

📁Src/pdf_reader.py

- Check validation
- Extract blocks
- Extract Text
- Extract Pdf Information for showing on UI
- Check Extractable type

Improvement:

- ❖ Extract text statistics for UI like total characters, total words, avg. words per page, estimated language, colors, size
- ❖ Pre-processing

📁Src/postprocessing.py

- Skip words to translate like abbreviations, numbers, symbols, punctuation, File name, Version
- Return Dictionary that generate TRUE and FALSE with word that will be translate or not.
- Remove important spaces at the beginning or end and introduce extra spaces accidentally.

Improvement:

- ❖ Add values in abbreviations Dictionary

📁Src/translator.py

- Get object with select language
- Text Translate with preprocessing
- Check Language

📁Src/pdfwriter.py

- It takes an existing pdf with text region and writes the translated text there, then returns the new PDF as bytes.
- Maintain Font style, colour and layout
- Maintain paragraphs or bullet points

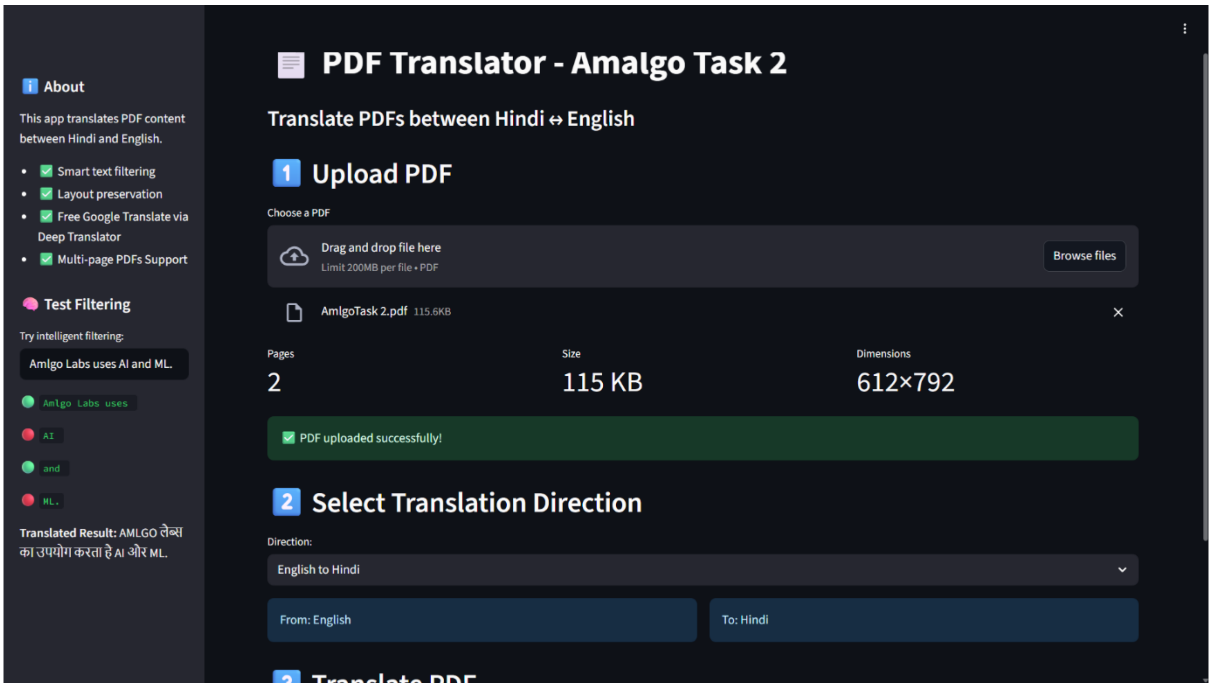
- Maintain Margin and Width of text
- Combine all of these and return new pdf

App.py (Streamlit)

- Sidebar contain text about project
- Filtering testing
- Header
- File Upload
- Translation
- Download

Improvement:

- ❖ Improve Filtering testing
- ❖ Add more data about project in sidebar
- ❖ Add more things about data



ℹ️ About

This app translates PDF content between Hindi and English.

- ✅ Smart text filtering
- ✅ Layout preservation
- ✅ Free Google Translate via Deep Translator
- ✅ Multi-page PDFs Support

🧠 Test Filtering

Try intelligent filtering:

Amlgo Labs uses AI and ML.

🟢 `Amlgo Labs uses`
🔴 `AI`
🟢 `and`
🔴 `ML.`

Translated Result: AMLGO लेब्स का उपयोग करता है AI और ML.

📄 PDF Translator - Amalgo Task 2

Translate PDFs between Hindi ↔ English

1️⃣ Upload PDF

Choose a PDF

Drag and drop file here
Limit 200MB per file • PDF

AmlgoTask 2.pdf 115.6KB

Pages	Size	Dimensions
2	115 KB	612×792

✅ PDF uploaded successfully!

2️⃣ Select Translation Direction

Direction: English to Hindi

From: English

To: Hindi

Testing with PDF contain

- Highlight text
- Images
- Colors text
- Links
- Skip translation like AI, NASA, ML, etc.
- Multi page

Problems and Improvement

- Multiple PDF once at a time
    - Batch Translations
- Convert Code into Classes and Object
- Add logging functionally and Exception handling
- Add DevOps Concept
- Add Database for uploaded PDF
- Showcase more info about data statistics in UI
- Add abbreviations Dictionary
- Can make a simple Version
- Without any library or API
- Bold text will increase size
- Multiple users trying
    - Using Session and timeout
- Pdf contain Hindi + English language
    - Fix by Detect language
- Can define Parameter like default size, default margin, line height, etc.
- Enhance formatting, Page structure, Tables
    - layout recognition engines: UniLM DiT
    - OCR engines: PaddleOCR
- Slow Speed or takes time
    - Batch Translations
    - Use Multithreading and parallelize translations
    - Avoid Re-rendering Unchanged Elements
    - Try different Open source and services and find best one