**BookVision Project: Full Expansion Report**

**Objective:**  
To expand BookVision into a comprehensive document analysis platform by enhancing its OCR capabilities, semantic understanding, data handling, visualization, and deployment strategy.

**1. Project Overview**

BookVision is a lightweight, intelligent document analyzer that allows users to upload documents in various formats (PDF, DOCX, Images), extract their content using OCR, and analyze the sentiment, themes, and structure of the content. The expanded version aims to support richer data formats, smarter processing, and visual insights.

**2. Key Functional Modules**

**2.1 File Input**

* **Support:** PDF, DOCX, JPG/PNG
* **Libs:** pdfplumber, python-docx, Pillow, pdf2image

**2.2 OCR Engine**

* **Primary OCR:** EasyOCR
* **Backup:** pytesseract
* **PDF Layering:** OCRmyPDF
* **Optional (Paid):** Google Cloud Vision OCR

**2.3 Semantic Analysis**

* **Base Model:** sentence-transformers
* **Tasks:** Text embedding, keyword extraction, similarity comparison
* **Additional:** nltk, spacy, keybert

**2.4 Sentiment Analysis**

* **Libs:** transformers (BERT-based), textblob, vaderSentiment

**2.5 Data Storage**

* **Storage Type:** TinyDB (NoSQL, JSON-based)
* **Use:** Save document metadata, OCR output, analysis results

**2.6 Visualization and Dashboards**

* **Libs:** matplotlib, seaborn, plotly, wordcloud
* **Visuals:** Pie charts, sentiment timelines, word clouds

**2.7 Backend Framework**

* **Base Framework:** Flask (Micro web backend)
* **Routes:** Upload, Process, Results, Admin Dashboard

**2.8 Frontend (Optional UI Upgrades)**

* **Template Engine:** Jinja2 (HTML + Bootstrap)
* **Optional:** Move to Streamlit for faster UI dev

**2.9 Deployment**

* **Free Hosting:** Render or Streamlit Cloud
* **Alternative:** Railway.app, Heroku
* **Paid Hosting (Optional):** AWS Lightsail, Google Cloud Run

**3. Library Compatibility Matrix**

All below libraries have been verified as compatible with Python 3.9–3.13:

* Document Handling: python-docx, pdfplumber, pdf2image, Pillow
* OCR: pytesseract, easyocr, ocrmypdf, google-cloud-vision
* NLP: nltk, spacy, sentence-transformers, textblob, vaderSentiment, keybert
* Database: tinydb
* Visualization: matplotlib, seaborn, plotly, wordcloud
* Web: flask, flask-cors, jinja2
* Utils: tqdm, numpy, pandas, scikit-learn, requests

**4. Roadmap & Milestones**

**Phase 1: Core Setup**

* Port codebase to VS Code
* Install all required libraries
* Set up Flask routing

**Phase 2: OCR Layer**

* Integrate EasyOCR and Tesseract with fallback
* Add OCRmyPDF for scanned PDFs
* Optional: Integrate Google Cloud Vision OCR

**Phase 3: Text Analysis**

* Sentiment analysis with VADER/TextBlob
* Keyword extraction with KeyBERT
* Improve semantic similarity with Sentence-BERT

**Phase 4: Data Handling**

* Store results in TinyDB
* Add search/filter by document metadata

**Phase 5: Visualization**

* Create pie charts, bar graphs
* Add word clouds
* Sentiment timeline (matplotlib/plotly)

**Phase 6: Deployment**

* Deploy to Render or Streamlit Cloud
* Enable user upload logging + feedback

**5. Advanced Features (Future Expansion)**

* **User Authentication:** Basic login system with Flask-Login
* **Document Classification:** Train model to classify document types
* **Named Entity Recognition:** Use spacy for people/org detection
* **Summarization:** Integrate HuggingFace transformers for abstractive summary
* **Audio Conversion:** Convert text to speech with gTTS or pyttsx3

**6. Recommendations for Expansion**

**6.1 Libraries to Consider**

* docx2txt, langdetect, pyglet (audio), dash (web UI), streamlit-chat

**6.2 Optional Cloud Services**

* **Storage:** Firebase / Supabase for document backup
* **OCR API:** Google Vision for enterprise accuracy
* **Model Hosting:** HuggingFace Spaces for custom models

**7. Summary**

This expansion will transform BookVision from a simple document reader into a powerful, intelligent analysis suite with multi-format input, smart OCR, and analytics-ready output. The current roadmap provides a balance between open-source tools and optional premium integrations for future scalability.

**Prepared By:** ChatGPT - Project Expansion Advisor  
**Date:** August 7, 2025