NLP Hackathon

Context-Aware Material Specification Extractor

Revolutionizing Engineering Document Analysis with AI

next slide →

Team Name: MlMonster

Shhaurya Jaiswal (Leader)
Devguru Tiwari
Rohan Choudhary
Shresth Shankhdhar

02

The Problem: The Engineer's Dilemma

- *Civil engineers and project managers spend countless hours manually sifting through hundreds of pages of dense technical documents (like CPWD specifications, IS codes, etc.).*
- *This process is slow, tedious, and highly prone to human error.*
- *Missing a single material specification can lead to compliance issues, budget overruns, and project delays.*

next slide →

03

thynk unlimited

Our Solution: An Intelligent Assistant

- *We have developed a web-based tool that automates the extraction of material specifications from any technical document (PDF or image).*
- *It uses a powerful combination of NLP and Generative AI to deliver accurate, structured, and ready-to-use data.*
- *Users can upload a document and instantly receive a detailed report, saving time and eliminating errors.*

next slide →

04

How It Works: The 4-Step Process

UPLOAD & PROCESS

The user uploads a document. Our system uses OCR (pytesseract) and PDF readers (pdfplumber) to digitize the content.

HYBRID EXTRACTION

We use a smart combination of keyword and semantic search to find all relevant information, even if it's phrased differently.

AI REFINEMENT

The extracted data is sent to a Generative AI (Gemini/Gemma) which acts as an expert engineer to clean, enrich, and standardize the information.

GENERATE REPORTS

The final, verified data is presented in a web table and is available for download as CSV or PDF reports.

next slide →

05

Core Innovation 1: The Hybrid Search Engine

- ***Keyword Search:*** *Fast and effective for finding exact matches (e.g., "Cement").*
- ***Semantic Search (Sentence-Transformers + FAISS)****: Our key innovation. It understands the meaning behind the words. It can find "rebar with high tensile strength" even if the keyword is "High strength deformed bars."*
- ***Result:*** *This hybrid approach ensures maximum accuracy and recall, capturing information that keyword-only systems would miss.*

next slide →

06

Core Innovation 2: AI-Powered Refinement

- *The extracted text is often messy and lacks context.*
- *Our ai_buddy.py module sends this data to a large language model (LLM).*
- *The AI is prompted to act as a domain expert, validating codes, providing clear definitions, and adding other relevant engineering insights.*
- *This step transforms basic text into actionable, high-quality information.*

next slide →

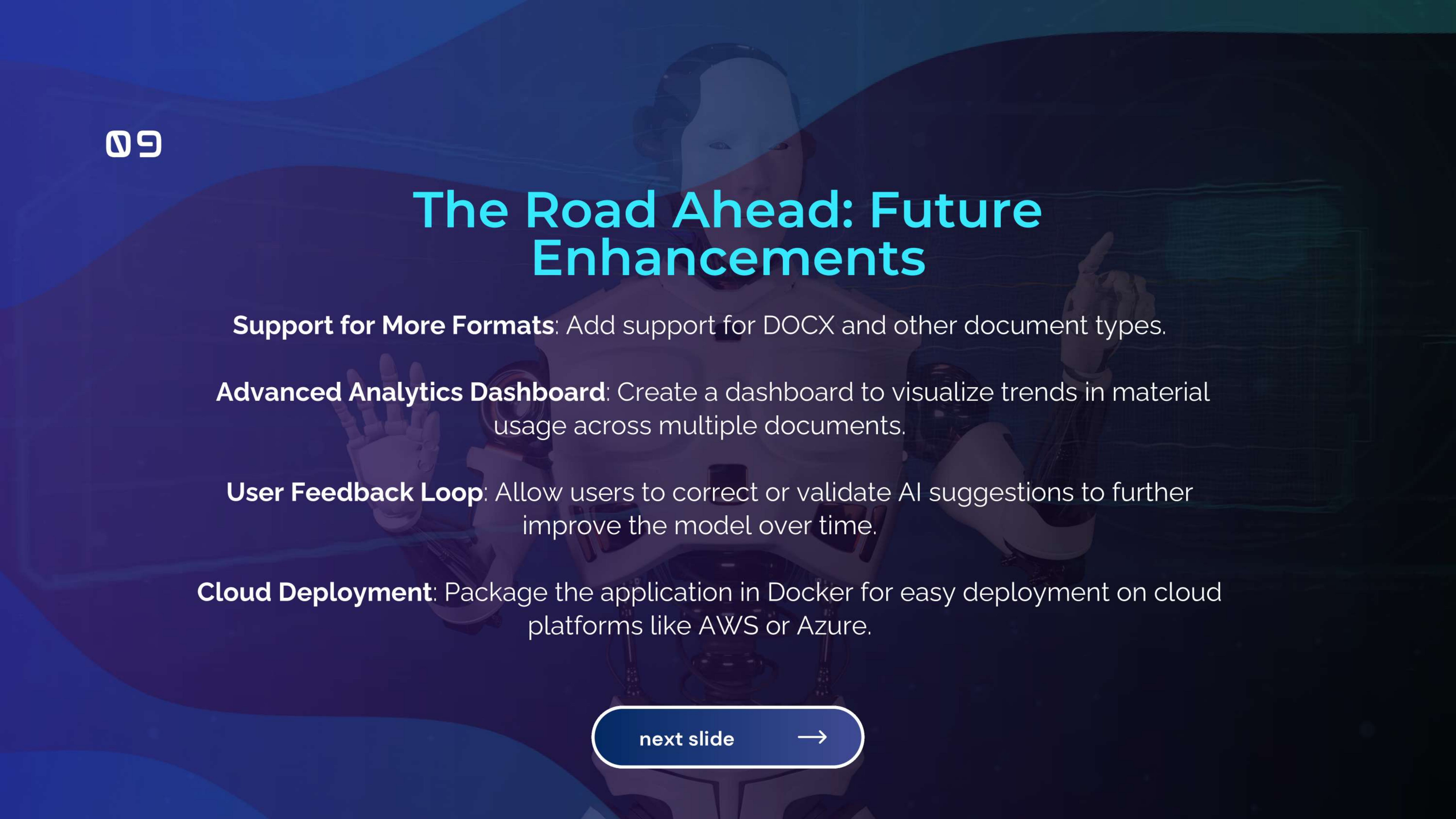
The Technology Behind the Magic

Backend: Python, Flask
NLP & Search: spaCy, Sentence-Transformers, Faiss
AI Integration: Google Generative AI (Gemini), OpenRouter (Gemma)
Document Processing: PyTesseract (OCR), PDFPlumber
Data Handling: Pandas, NumPy
Reporting: ReportLab

Key Achievements & Impact

- ***Drastic Time Reduction**: Reduces document analysis time from hours to seconds.*
- ***Enhanced Accuracy:** The hybrid search and AI refinement significantly reduce the risk of human error.*
- ***Structured Data Output**: Converts unstructured documents into ready-to-use CSV and PDF formats for easy integration into other workflows.*
- ***Scalable & Robust**: Built with industry-standard libraries and includes comprehensive error handling.*

next slide →



The Road Ahead: Future Enhancements

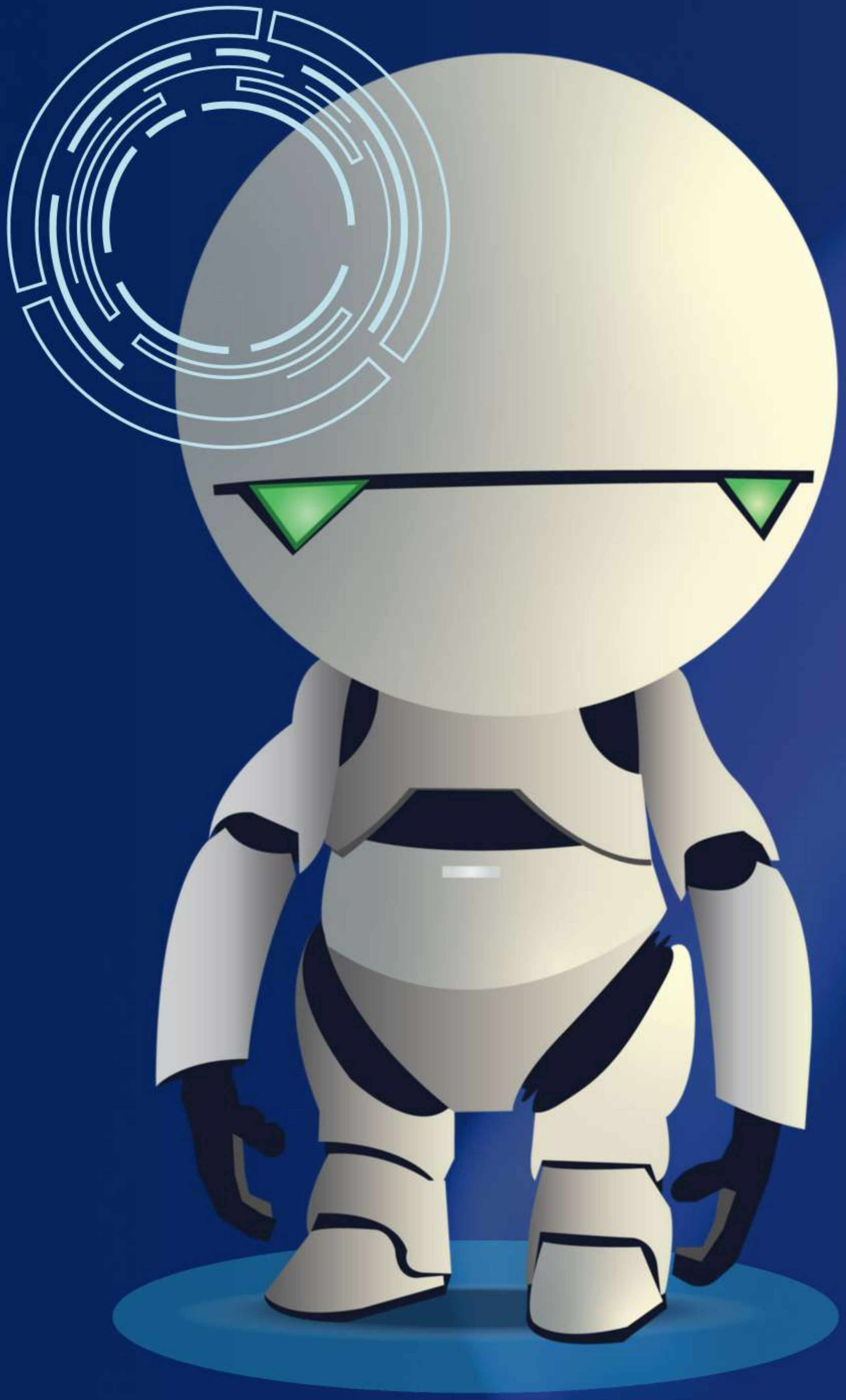
Support for More Formats: Add support for DOCX and other document types.

Advanced Analytics Dashboard: Create a dashboard to visualize trends in material usage across multiple documents.

User Feedback Loop: Allow users to correct or validate AI suggestions to further improve the model over time.

Cloud Deployment: Package the application in Docker for easy deployment on cloud platforms like AWS or Azure.

next slide →



Model Accuracy Assessment

***Precision** = TP / (TP + FP)*

- *Measures how much of the extracted data is actually correct.*
- *Ensures high trust and low noise in the output.*

***Recall** = TP / (TP + FN)*

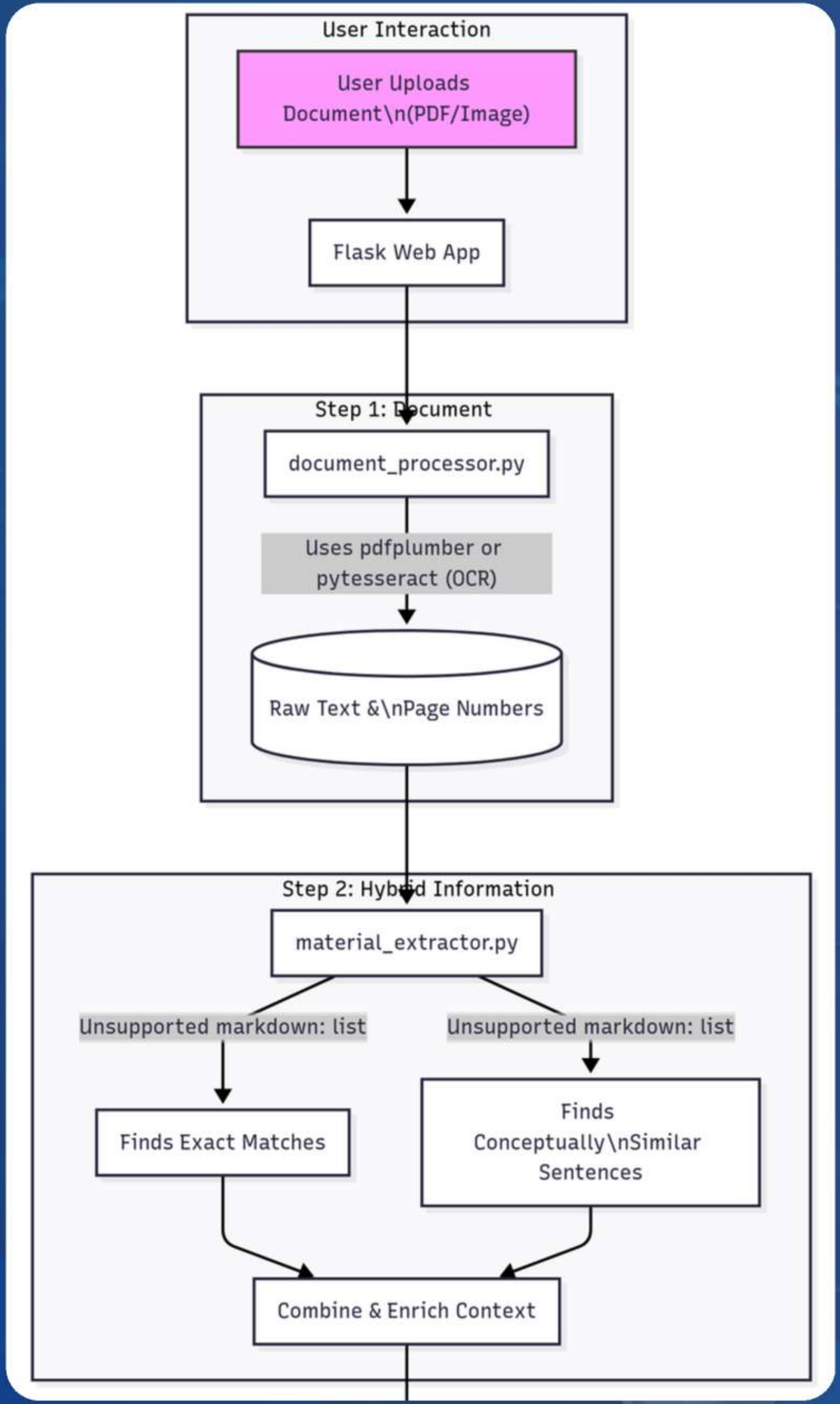
- *Measures how much of the relevant information was successfully extracted.*
- *Crucial for completeness and avoiding missed data.*

***F1-Score** = 2 × (P × R) / (P + R)*

- *Balances Precision and Recall for a single performance score.*
- *Best overall measure for real-world extraction quality.*

next slide →

Flowchart



User Interaction

User Uploads Document\n(PDF/Image)

Flask Web App

Step 1: Document

document_processor.py

Uses pdfplumber or pytesseract (OCR)

Raw Text &\nPage Numbers

Step 2: Hybrid Information

material_extractor.py

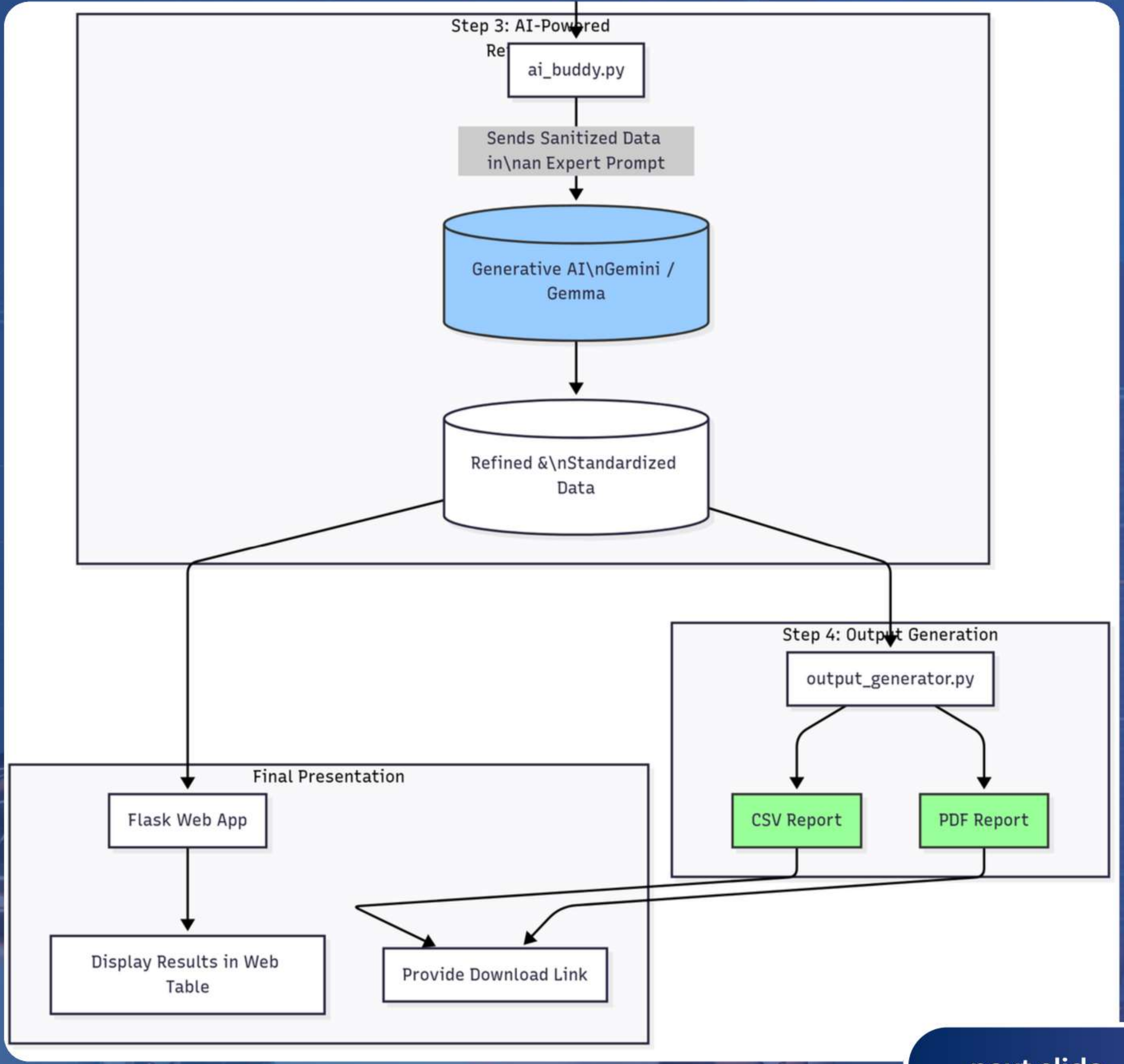
Unsupported markdown: list

Unsupported markdown: list

Finds Exact Matches

Finds Conceptually\nSimilar Sentences

Combine & Enrich Context



Step 3: AI-Powered

Re

ai_buddy.py

Sends Sanitized Data in\nan Expert Prompt

Generative AI\nGemini / Gemma

Refined &\nStandardized Data

Step 4: Output Generation

output_generator.py

CSV Report

PDF Report

Final Presentation

Flask Web App

Display Results in Web Table

Provide Download Link

next slide →

Thank You!