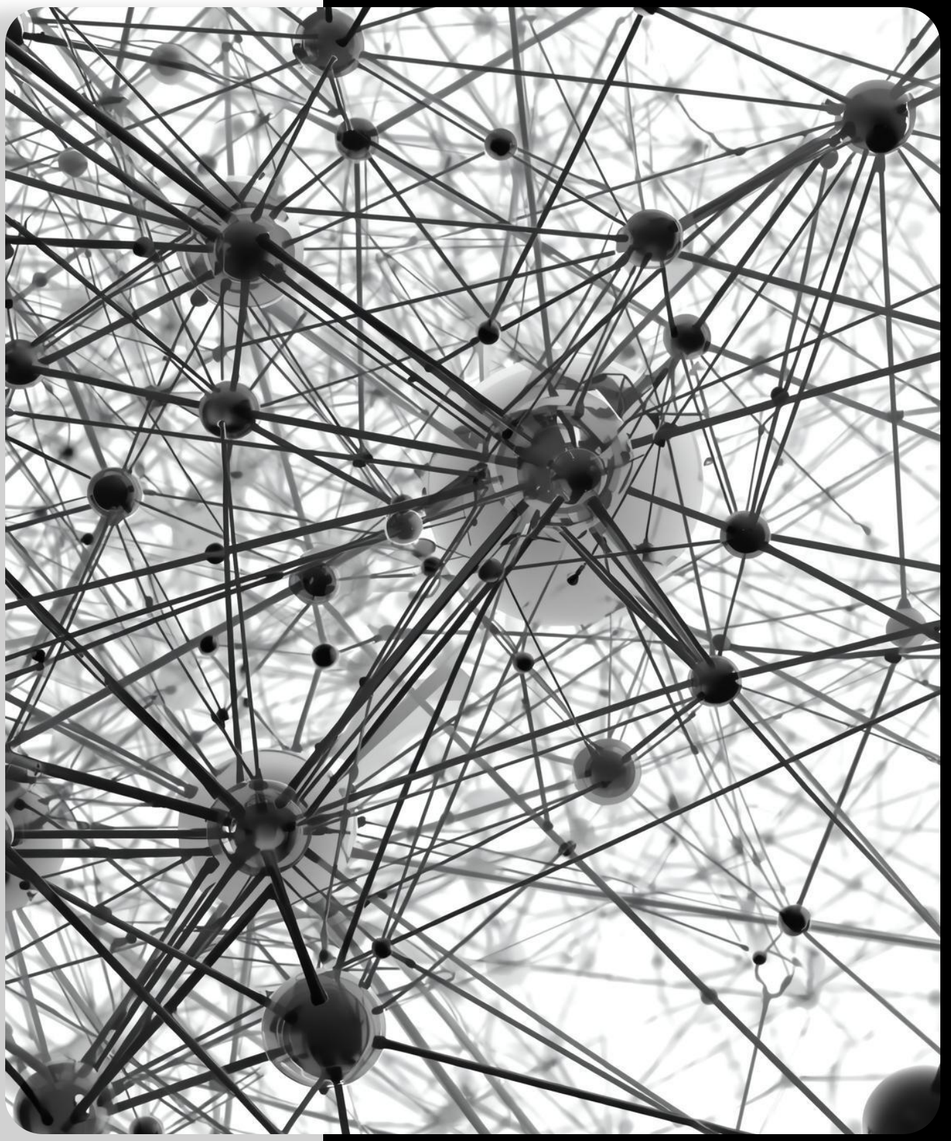


NLP-Driven Workflow Automation for Inspection Readiness

Document Control Intelligence

Juri Moon | juri.moon.03@gmail.com

Problem Overview

FDA Document Request for Pre Approval Inspection

My Contribution:

- Created a **master summary table** consolidating all **deviations** in response to the **FDA Formatting**

64494

```
/Project_64493/
├ Deviations/
|   ├ DEV-2022-224.pdf
|   ├ DEV-2022-226.pdf
|   ├ DEV-2023-101.pdf
|   ├ DEV-2023-108.pdf
|   ├ DEV-2023-112.pdf
|   ├ DEV-2023-118.pdf
|   ├ DEV-2023-125.pdf
|   ├ DEV-2023-129.pdf
|   ├ DEV-2023-135.pdf
|   └ DEV-2023-142.pdf
```

1. 50+ Deviations
2. Short Deadline
3. Legacy Project (8 yrs)

Current Need

*Automated system to summarize and classify deviation— enabling rapid, traceable access that meets **FDA Pre-Approval Inspection (PAI)** expectations and reduces manual review time.*

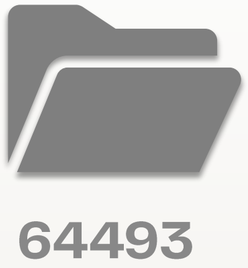
Objective Overview

Develop an NLP-based system that can

1. **Read Scanned PDFs**
2. **Auto-classify by type (Equipment, Facility, Materials)**
3. **Export structured summaries into a clear database in compliance to FDA**

System Architecture

INPUT LAYER




```
/Project_64493/
├ Deviations/
|   ├ DEV-2022-224.pdf
|   ├ DEV-2022-226.pdf
|   ├ DEV-2023-101.pdf
|   ├ DEV-2023-108.pdf
|   ├ DEV-2023-112.pdf
|   ├ DEV-2023-118.pdf
|   ├ DEV-2023-125.pdf
|   ├ DEV-2023-129.pdf
|   ├ DEV-2023-135.pdf
|   └ DEV-2023-142.pdf
```

PROCESSING LAYER

Extractor
- Extract
- Cleans (¬Æ, â€™)
- Organize

Summarization
- BART-Large-CNN

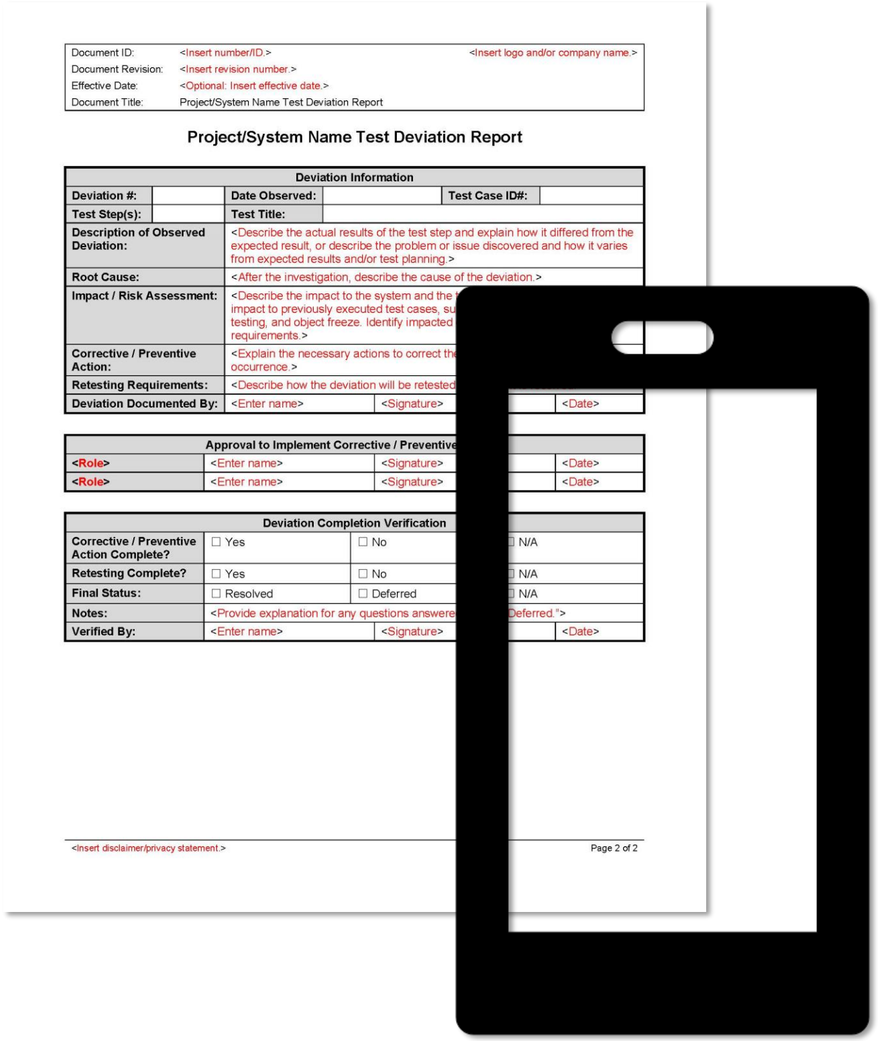
Classification
- BART-Large-MNLI
- zero-shot classification

OUTPUT LAYER

Summary Database
(FDA Format)

Taxonomy Dashboard
(per classification)

DEMO



Document ID:	<Insert number/ID.>	<Insert logo and/or company name.>
Document Revision:	<Insert revision number.>	
Effective Date:	<Optional: Insert effective date.>	
Document Title:	Project/System Name Test Deviation Report	

Project/System Name Test Deviation Report

Deviation Information					
Deviation #:		**Date Observed:**		**Test Case ID#:**	
Test Step(s):		**Test Title:**			
Description of Observed Deviation:	<Describe the actual results of the test step and explain how it differed from the expected result, or describe the problem or issue discovered and how it varies from expected results and/or test planning.>				
Root Cause:	<After the investigation, describe the cause of the deviation.>				
Impact / Risk Assessment:	<Describe the impact to the system and the i impact to previously executed test cases, su testing, and object freeze. Identify impacted requirements.>				
Corrective / Preventive Action:	<Explain the necessary actions to correct th occurrence.>				
Retesting Requirements:	<Describe how the deviation will be retested				
Deviation Documented By:	<Enter name>	<Signature>			<Date>

Approval to Implement Corrective / Preventive				
<Role>	<Enter name>	<Signature>		<Date>
<Role>	<Enter name>	<Signature>		<Date>

Deviation Completion Verification				
Corrective / Preventive Action Complete?	☐ Yes	☐ No	☐ N/A	
Retesting Complete?	☐ Yes	☐ No	☐ N/A	
Final Status:	☐ Resolved	☐ Deferred	☐ N/A	
Notes:	<Provide explanation for any questions answere		Deferred.*>	
Verified By:	<Enter name>	<Signature>		<Date>

<Insert disclaimer/privacy statement.>

Page 2 of 2

Dummy Example: Power Outage

DEV-2022-109

Event Overview

- Date/Time: 5/21/22, 03:39 – 08:05
- Site-wide power outage; Eurotherm system lost temperature/vacuum data
- No data recorded for ~4.5 hours

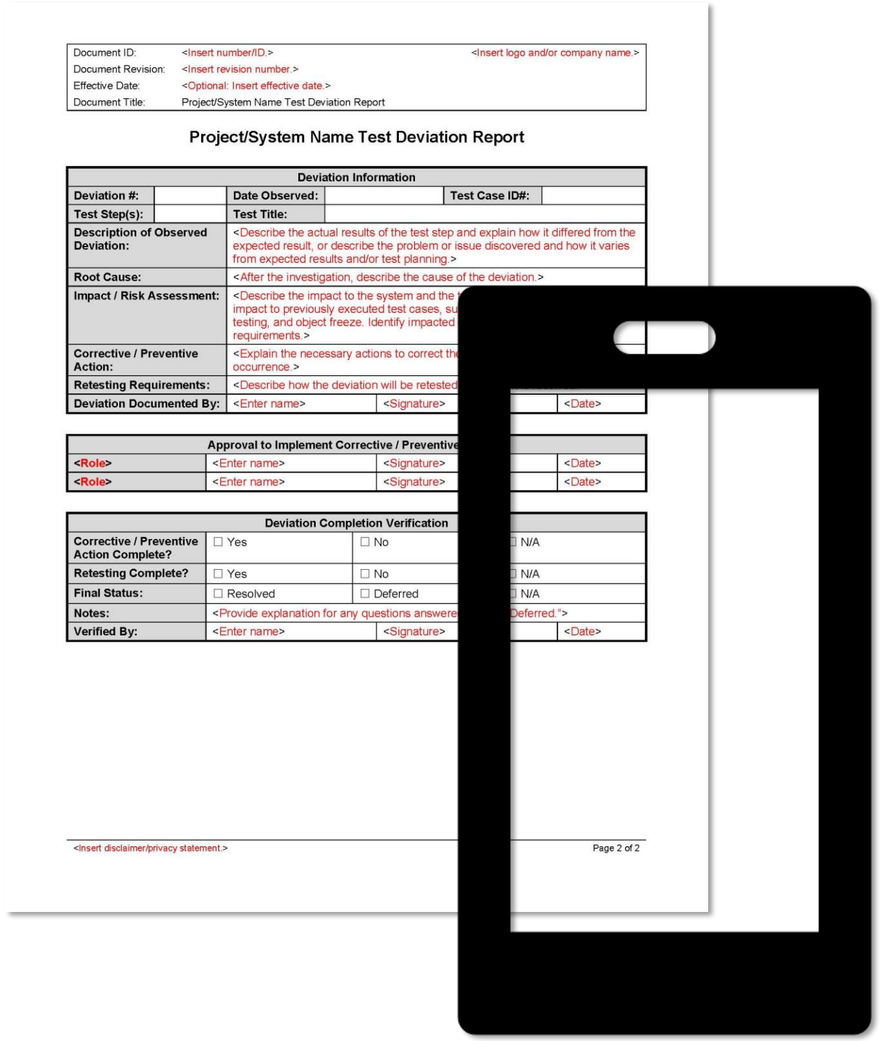
Impact Assessment

- Pre-failure temp: 36.9 °C → Post-failure temp: 32.0 °C
- No temperature specification in batch record; drying under vacuum is the only requirement
- No impact expected on process or product quality

Outcome

- Batch met all release criteria:
  - Yield 85%
  - Purity 99.82%
  - Assay 97 wt%

DEMO



Document ID:	<Insert number/ID.>	<Insert logo and/or company name.>
Document Revision:	<Insert revision number.>	
Effective Date:	<Optional: Insert effective date.>	
Document Title:	Project/System Name Test Deviation Report	

Project/System Name Test Deviation Report

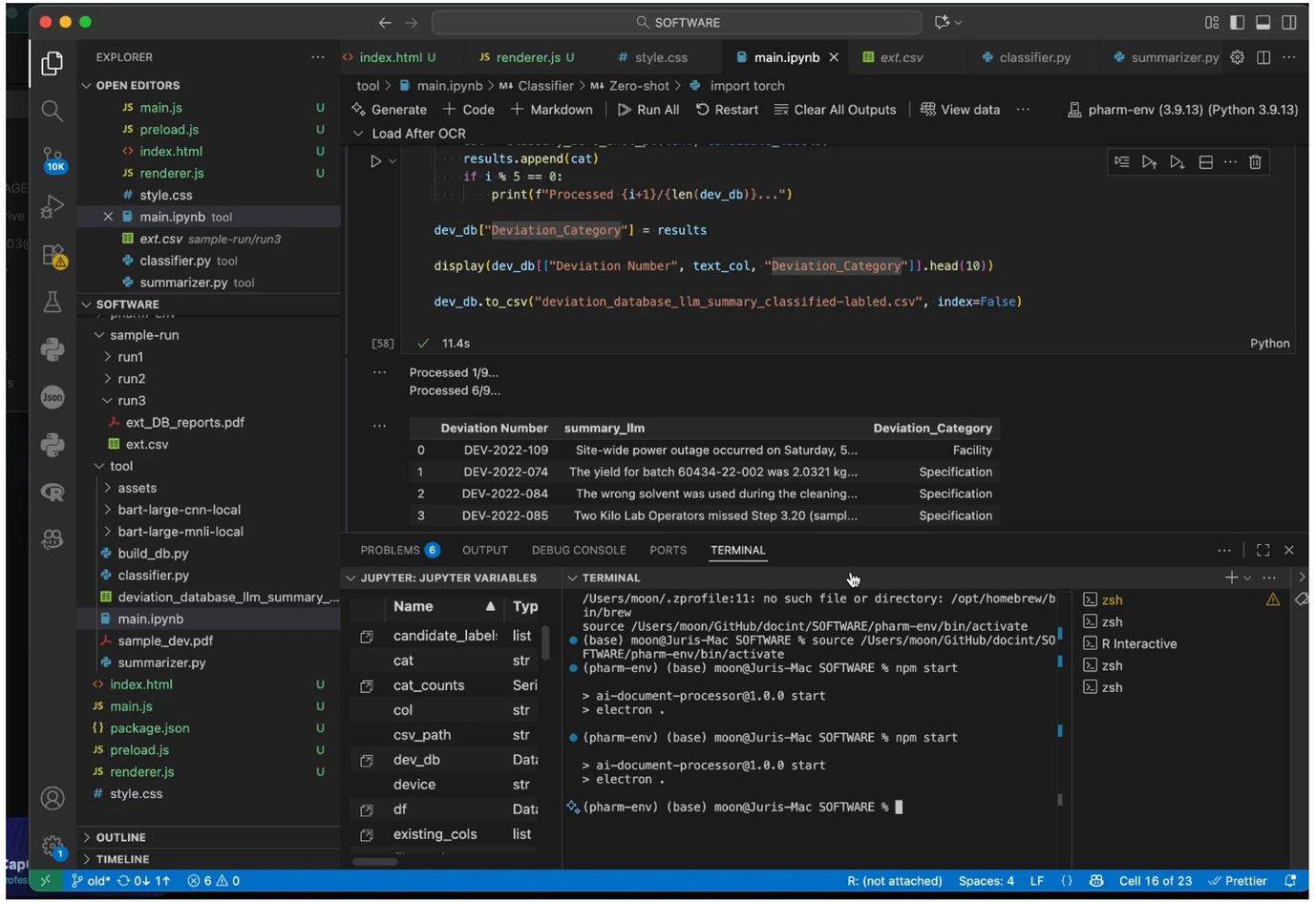
Deviation Information					
Deviation #:		**Date Observed:**		**Test Case ID#:**	
Test Step(s):		**Test Title:**			
Description of Observed Deviation:	<Describe the actual results of the test step and explain how it differed from the expected result, or describe the problem or issue discovered and how it varies from expected results and/or test planning.>				
Root Cause:	<After the investigation, describe the cause of the deviation.>				
Impact / Risk Assessment:	<Describe the impact to the system and the f… impact to previously executed test cases, su… testing, and object freeze. Identify impacted … requirements.>				
Corrective / Preventive Action:	<Explain the necessary actions to correct th… occurrence.>				
Retesting Requirements:	<Describe how the deviation will be retested				
Deviation Documented By:	<Enter name>	<Signature>		<Date>	

Approval to Implement Corrective / Preventive				
<Role>	<Enter name>	<Signature>		<Date>
<Role>	<Enter name>	<Signature>		<Date>

Deviation Completion Verification				
Corrective / Preventive Action Complete?	☐ Yes	☐ No	☐ N/A	
Retesting Complete?	☐ Yes	☐ No	☐ N/A	
Final Status:	☐ Resolved	☐ Deferred	☐ N/A	
Notes:	<Provide explanation for any questions answere…		Deferred.”>	
Verified By:	<Enter name>	<Signature>		<Date>

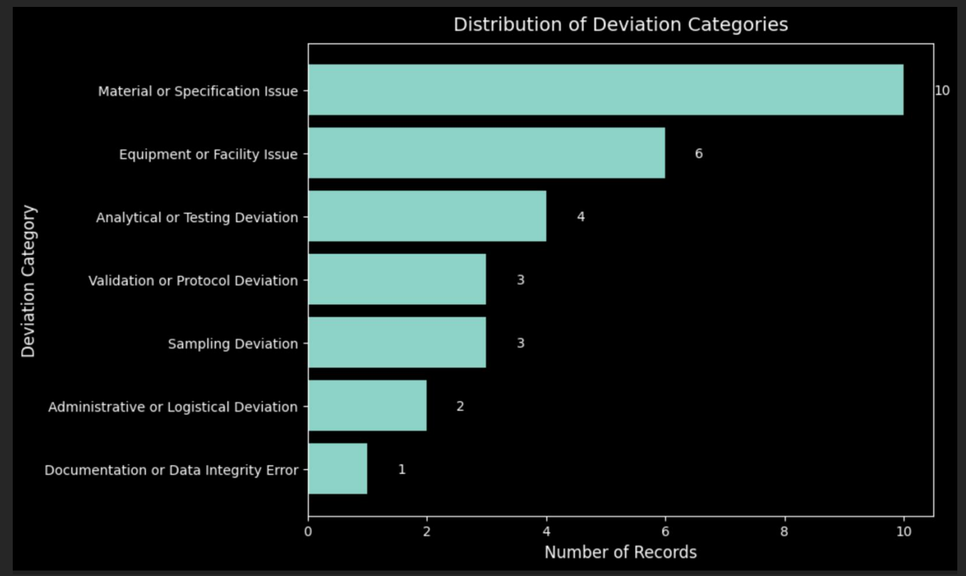
<Insert disclaimer/privacy statement.>

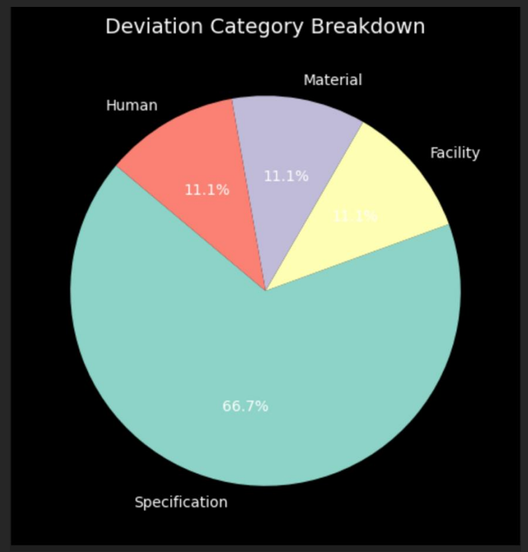
Page 2 of 2

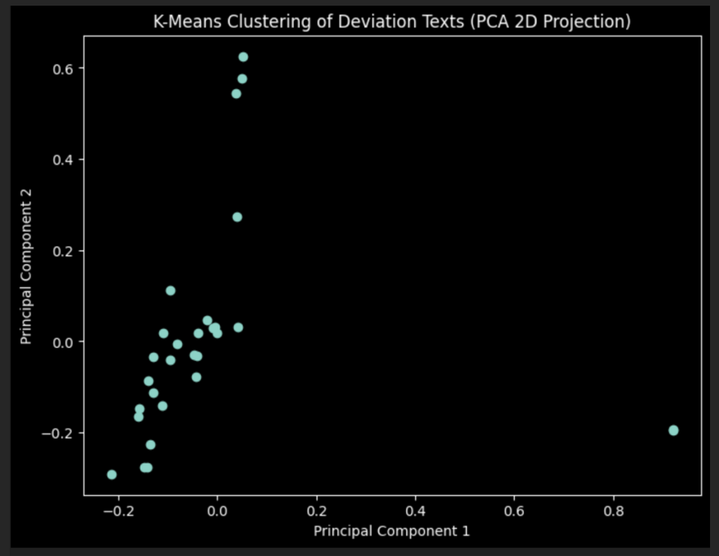


Classifier

```python
# --- Define your candidate labels ---
candidate_labels = [
    "Analytical or Testing Deviation",
    "Sampling Deviation",
    "Documentation or Data Integrity Error",
    "Equipment or Facility Issue",
    "Manufacturing Process Deviation",
    "Material or Specification Issue",
    "Validation or Protocol Deviation",
    "Safety or Environmental Event",
    "Training or Human Error",
    "Administrative or Logistical Deviation"
]
```







Results & Impact

1. Now, We Are Able to Answer Questions Like:

Deviation Insights

- What is the high-level view on **current deviation?**
- What are the **most frequent deviation categories**

  (equipment, process, documentation)?

2. Quantifying “Convenience”

Metric	Manual	Automated
Summary Creation (FDA Requirement)	2 days - 3 days	20min + 1 day (review)
Compliance traceability	None	Full
Cognitive Clicks	~5-10	Evaluation In-process
New Member Training Burden	High	Evaluation In-process

Technical Approach

Summarizer Engine

- **Uses BART-Large-CNN transformer** developed by Meta / Facebook AI Research
- Runs fully offline
- Trained on 300,000+ human-written documents
- Designed to read long, technical text and generate concise summaries

Classifier

- Uses **BART-Large-MNLI for zero-shot** text classification
- Learns relationships between sentences to understand meaning, not just keywords

Challenges & Solutions

1. Technical

Challenges	Solution	Status
• Inconsistent document formats	• Regex normalization	• Completed
• OCR noise from scanned forms	• Bilateral filtering	• Completed
• Validation Plan	• Domain fine-tuning using deviation dataset	• In process

2. Cross Functional Adoption

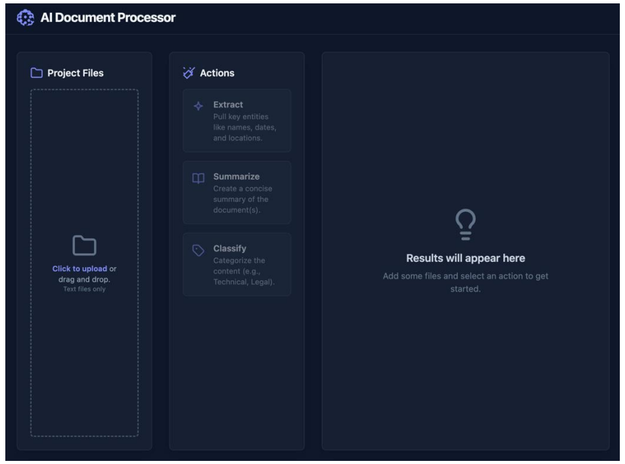
Challenges	Solution	Status
• Single-user workflow; tool not yet accessible to QA/RA teams	• Develop user-friendly GUI for non-technical access	• In process

Next: Validation Plan

- **ROUGE-1 / ROUGE-L:** Measures textual overlap with human-written reference summaries.
- **Acceptance:** ≥ 0.35 (industry benchmark for technical summarization).
- **BERTScore:** Semantic similarity to human reference (context-level meaning).
- **Acceptance**: ≥ 0.85 (used in biomedical summarization validation

Lewis et al., *“BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension”*, ACL 2020.

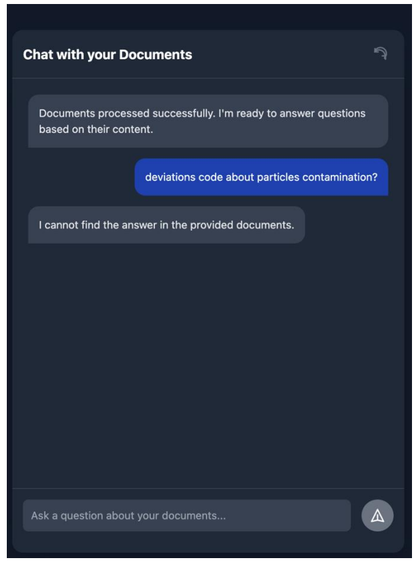
Next & Implementation



1. User Software (In process)

Develop a full software:

- Expand users (QA, RA, Engineers)



2. Search Assistant (Chatbot)

- Support natural-language search
- Query within database - What happened to Batch X?

3. Cross-Functional Feeback

- Validate automated summaries
- Collect user feedback to refine accuracy & usability

Lessons

1. Technical

- Begin with a working pipeline, focus on functionality before perfection.
- Always start from the big picture, then narrow down to accuracy and optimization.

2. Cross Functional Collaboration

- Design with empathy:

“How can this make my manager’s or teammate’s work easier?”

- Listen and observe openly to get user’s needs/insights/feedback

3. Personal Growth

- Be Proactive:

Don’t hesitate — just start

- When Stuck, break it down to step by step