IndiaAI Intelligent Document Processing (IDP) Challenge – Stage 1

1. Overview of the Architecture

The proposed solution implements a modular, end-to-end Intelligent Document Processing (IDP) pipeline that ingests, validates, extracts, structures, analyzes, translates, and summarizes information from heterogeneous, multilingual, and low-quality government documents.

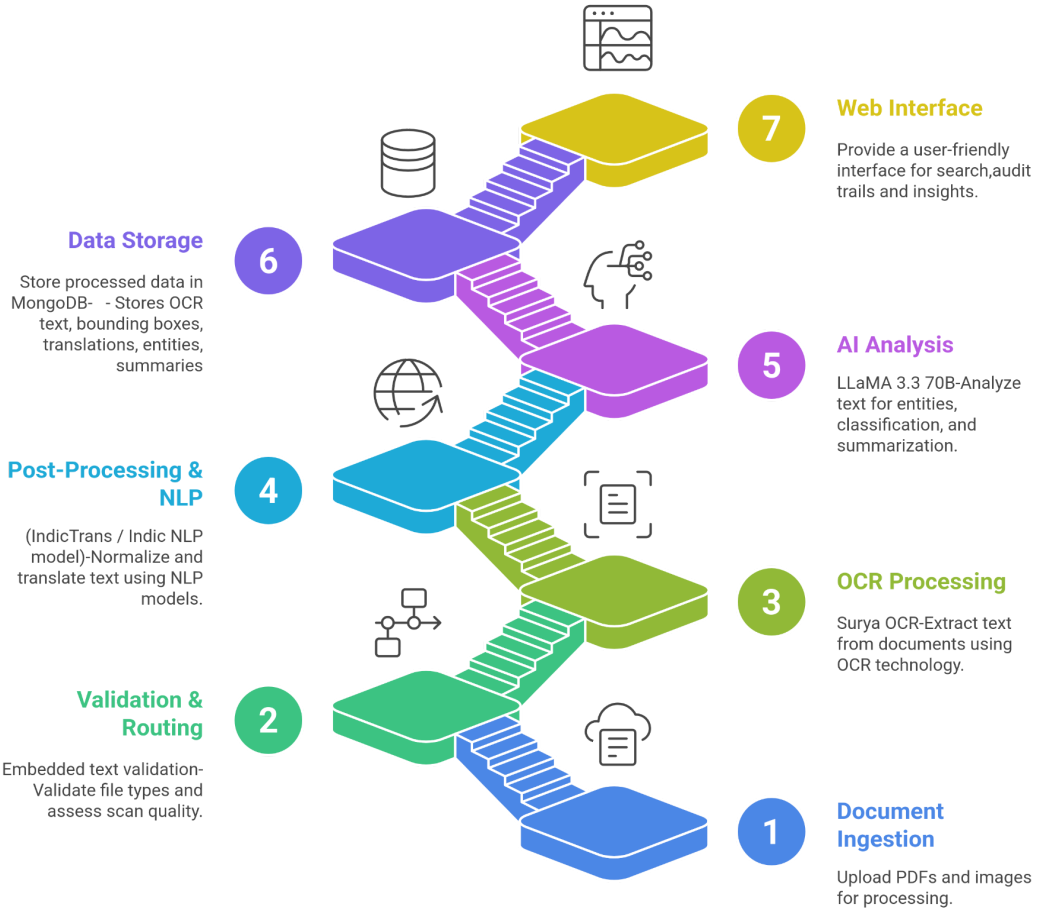


Figure 1: End-to-End OCR-based Intelligent Document Processing Architecture

Figure 1 presents the OCR-centric IDP architecture, in which documents move through clearly defined processing stages: ingestion, validation and routing, OCR, post-processing and NLP, AI analysis, data storage, and a web interface for search and audit trails. Each stage is encapsulated as an independent service, allowing technologies

such as Surya OCR, Indic NLP/translation models, and LLM-based analysis to be integrated or upgraded without disrupting the overall system.

The pipeline-based design ensures scalability and operational efficiency by enabling horizontal scaling of compute-intensive components (OCR and AI workers) based on workload, while lighter services such as routing and storage remain stable. This modularity increases explainability—each transformation step is observable and logged—and makes the solution ready for secure on-premise or cloud deployment in public-sector environments where robustness, cost control, and auditability are mandatory

2. High-Level Architectural Flow

2.1. User / Admin

Entry point of the system where authorised users or administrators upload documents for processing. Users may submit PDFs or scanned images from diverse government workflows, without any assumption about document quality, structure, or language.

2.2. Document Ingestion

Validates incoming files before processing. Performs file-type checks, size limits, corruption detection, virus scanning (if available), and metadata capture such as source system, timestamp, and declared document type. Validated documents are securely routed to the backend so that only compliant inputs enter the pipeline.

2.3. File validation and routing

Validated documents are securely routed to the backend, ensuring only compliant inputs enter the pipeline.

2.4. OCR Processing

Converts documents into machine-readable text using dynamically selected OCR engines based on format and quality. Digitally generated PDFs, scanned pages, and low-quality images are routed to the most suitable OCR pipeline to maximise recognition accuracy while preserving layout information and bounding boxes.

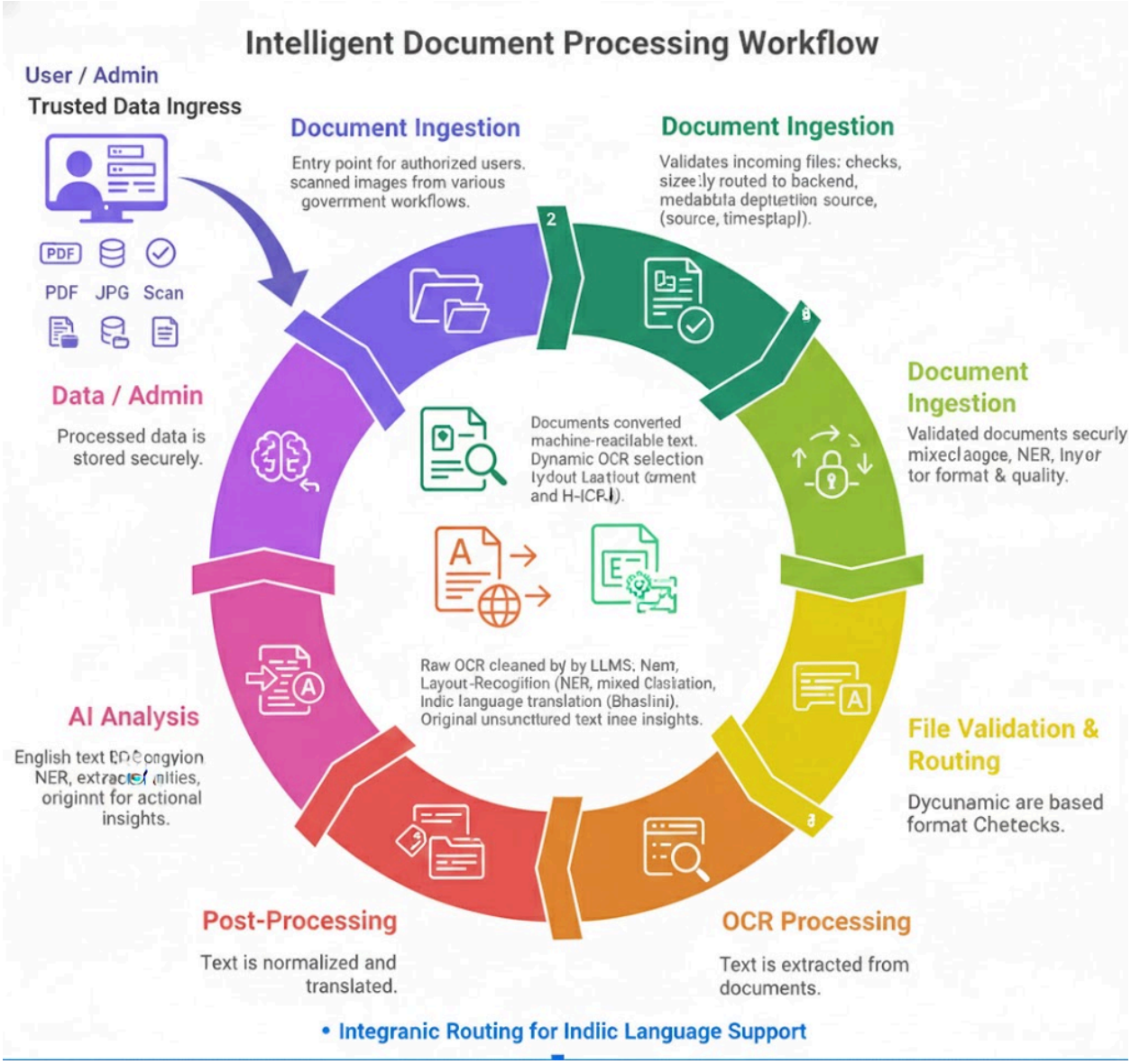


Figure 2: Document Processing Workflow from Ingestion to Output

2.5. Post-Processing

Cleans and normalises raw OCR output. The system detects language, handles mixed-language content, and translates Indic languages into English using Indic translation models. Original text is always retained alongside translated text to support traceability, human verification, and audit requirements.

2.6. AI Analysis

Analyses standardised English text with large language models to extract structured information. Performs named entity recognition, document classification, and concise, section-aware summarisation so that unstructured documents become actionable, decision-ready insights for officers.

2.7. Data Storage

Stores all processing artefacts—OCR text, translations, extracted entities, summaries, layout metadata, and logs—in a structured database optimised for fast retrieval and full-text search. The storage layer maintains an immutable audit trail that links every AI output back to its source content.

2.8. Web Interface

Provides users and administrators with search, review, and inspection capabilities over processed documents. Presents extracted entities, summaries, and original content in an accessible view, enabling efficient verification, collaboration, and downstream decision-making.

Throughout the pipeline, the system dynamically routes documents based on format and quality, ensuring optimal OCR selection, efficient downstream processing, and consistent handling of multilingual, heterogeneous government documents.

3. Document Ingestion Layer

Purpose
Securely accept and validate documents originating from multiple government systems and user channels before they enter the processing pipeline.

Supported Inputs

- PDF documents
- Image files (JPG, JPEG, TIFF, BMP)
- Local uploads and cloud-based or system-to-system uploads

Key Functions

- File type and size validation
- Corruption and basic integrity checks
- Metadata capture (document type, source, timestamp, uploader)
- Secure routing of validated documents to the processing backend

This layer ensures robustness and compatibility with real-world government document submission patterns, including heterogeneous formats and sources.

4. Processing Backend – File Validation & Routing Logic

Purpose

- To intelligently route documents to the appropriate OCR pipeline.

Routing Decisions

- PDF vs image-based document
- Presence of embedded text
- Scan quality and text density

Benefits

- Avoids unnecessary OCR computation
- Reduces processing cost
- Improves overall system latency

5. OCR Processing Layer

5.1 PDF Document Processing

For digitally generated or text-heavy PDFs:

- Marker OCR is used
- Preserves layout structure
- Extracts text along with bounding box coordinates

5.2 Image & Scanned Document Processing

For scanned images or image-based PDFs:

- PDFs are converted into images
- Surya OCR is applied for:
    - Low-resolution scans
    - Skewed or noisy images
    - Multilingual content

Output of OCR Layer

- Extracted text
- Bounding box coordinates
- Page-level layout information

This structured OCR output enables accurate downstream processing.

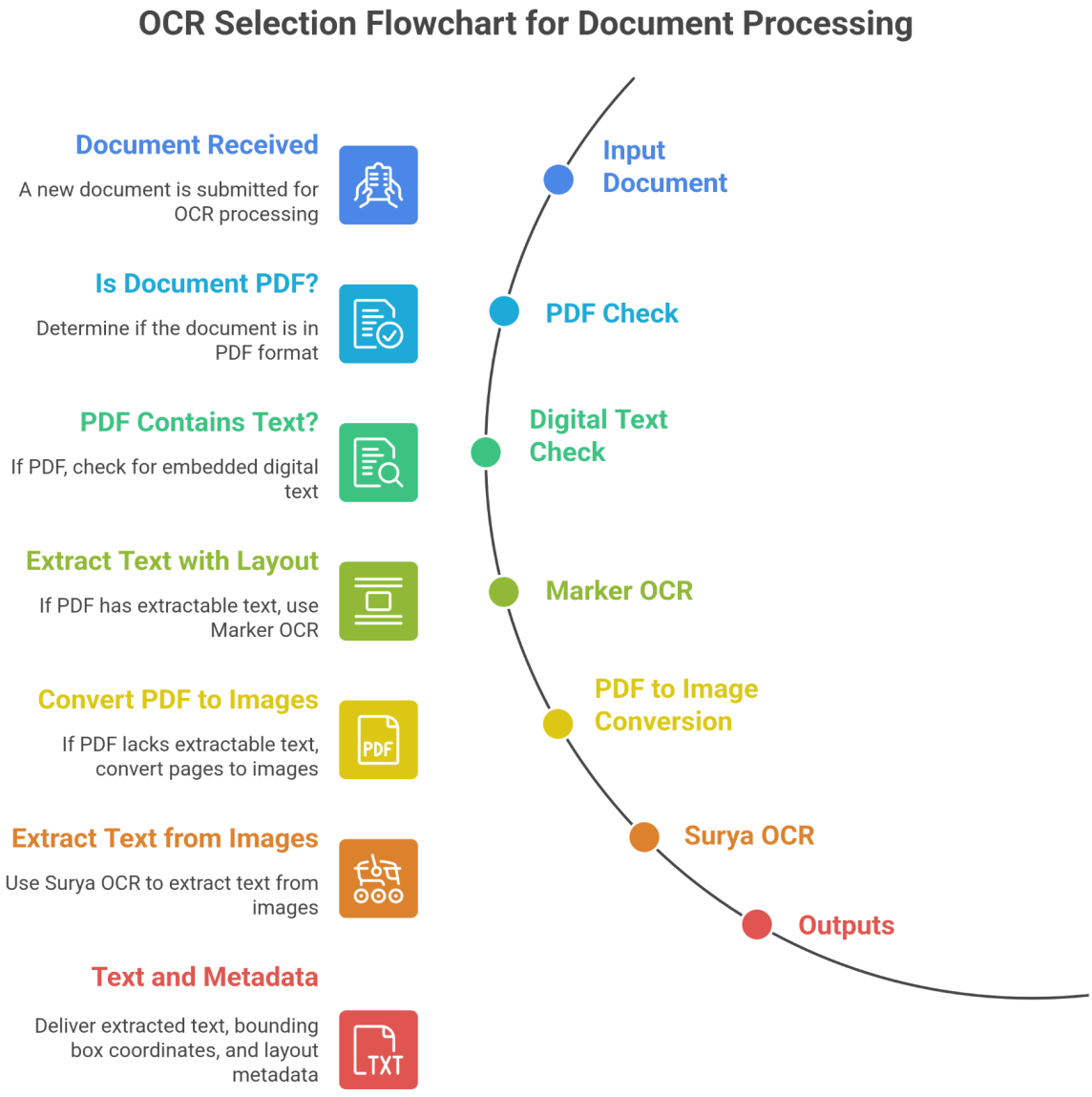


Figure 3: OCR Selection Logic Based on Document Type and Quality

6. Post-Processing Layer

The post-processing layer converts raw OCR output into structured and usable information.

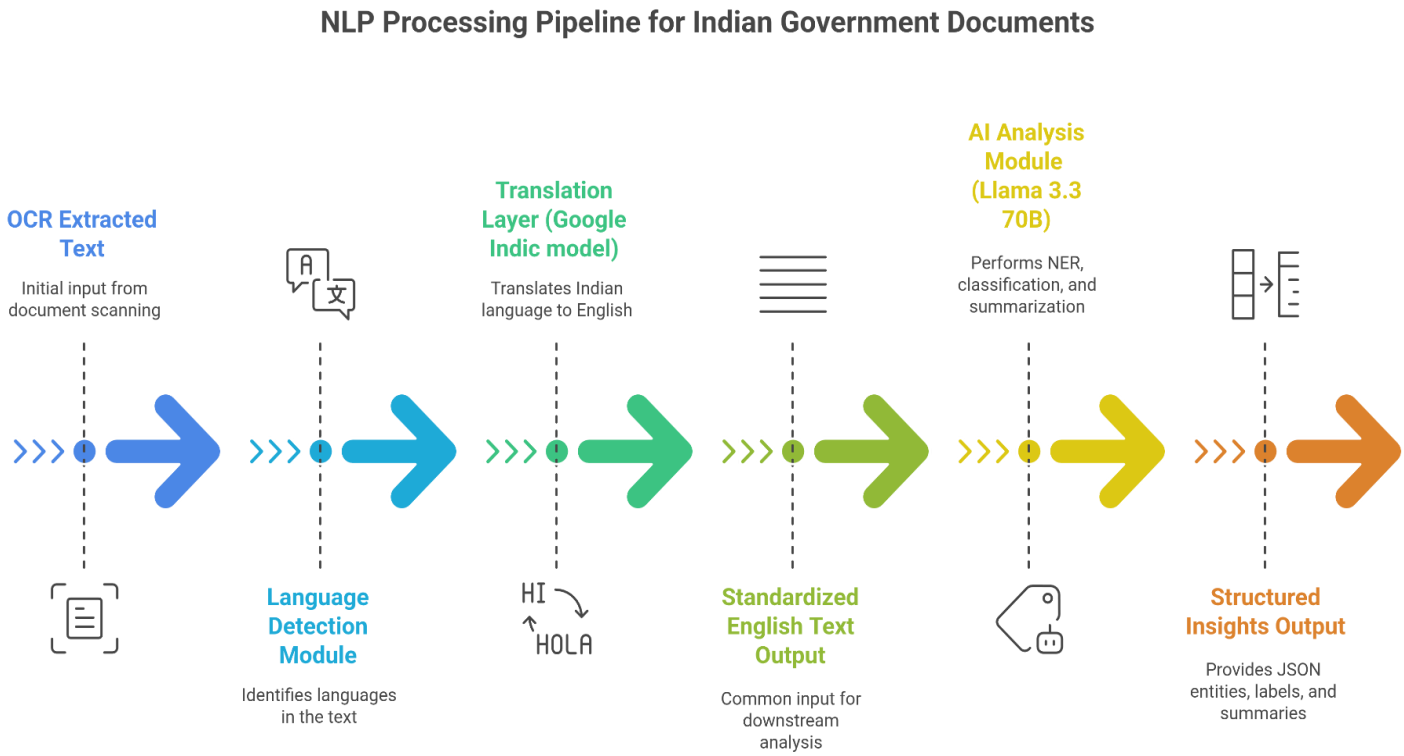


Figure 4: Language Detection, Translation, and AI Analysis Workflow

6.1 Language Detection

Purpose

To automatically identify the language(s) present in the document.

Capabilities

- Detects Indian languages and English
- Supports mixed-language documents
- Enables correct translation and NLP processing

6.2 Translation (Indian languages → English)

Purpose

To standardize document content for analysis and verification.

Approach

- Indian language translation models
- Original text retained for auditability
- Translated text used for downstream NLP tasks

This ensures consistency across multilingual documents.

6.3 AI Analysis & Information Extraction

Tasks Performed

- Named Entity Recognition (NER)
- Document classification
- Context-aware summarization

Outputs

- Key entities (names, dates, IDs, authorities)
- Document type labels
- Concise, section-aware summaries

This layer transforms unstructured text into **decision-ready insights**.

7. Data Storage & Management Layer

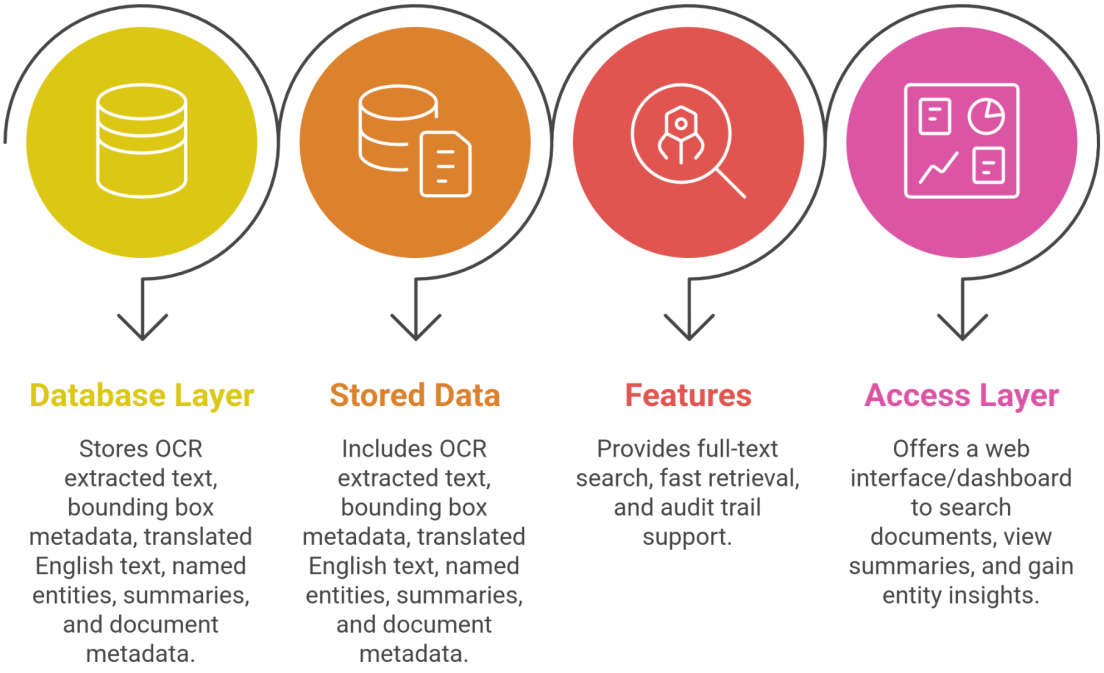


Figure 5: Document Metadata, Text, and Insight Storage Architecture

Database Used: MongoDB

Stored Artifacts

- OCR text and layout metadata
- Extracted entities
- Summaries
- Document-level metadata

Features

- Full-text search
- Fast retrieval
- Audit trail preservation

This design supports transparency and explainability.

8. Web Interface Layer

Purpose

To provide a user-friendly interface for administrators and officers.

Features

- Interactive dashboard
- Search and query functionality
- Document-wise insights and summaries

The interface enables efficient document review and decision-making.

9. Architectural Design Principles

The solution is designed with the following principles:

- **Modularity** – Each component can evolve independently
- **Scalability** – Supports high-volume document processing
- **Cost Efficiency** – CPU-friendly, no heavy training pipelines
- **Explainability** – Retains original text and traceable outputs
- **Security Ready** – Can integrate encryption and access control

10. Model Usage & Training Strategy

At Stage 1, the solution operates as an **inference-driven pipeline**.

Key Points

- No task-specific supervised training is performed
- Pre-trained OCR and NLP models are used
- No labeled dataset is created by the team

Implication

A traditional train–test split is **not applicable** at this stage.

11. Evaluation Approach

Evaluation Datasets

- FUNSD: 50 noisy scanned forms (entity extraction benchmark)
- SynthLayout-20: 20 synthetic documents (layout segmentation)
- CNN/DailyMail: 20 news articles (summarization benchmark)

Key Insights & Roadmap

Strengths:

- Excellent summarization (BERTScore 86.0%)
- Strong character accuracy (77.7%)
- Perfect text-line positioning (±5px Y-accuracy)

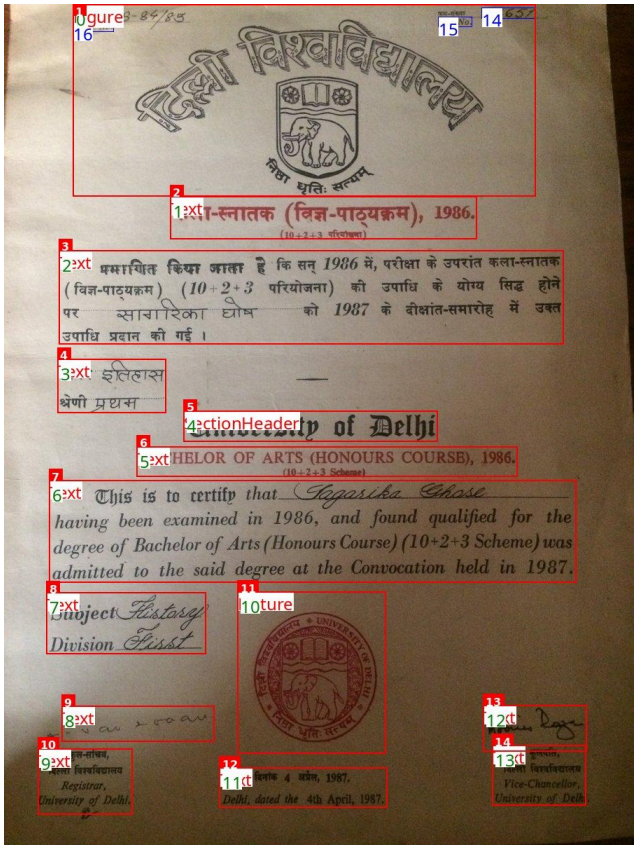
Challenges:

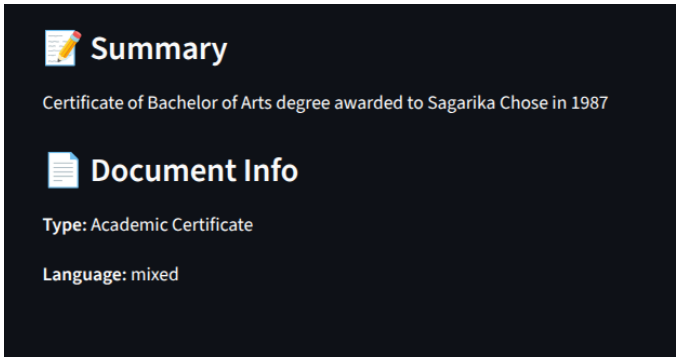
- Layout mIoU reflects text-line (28px) → layout-block (370px) conversion
- Entity F1 limited by strict bbox matching

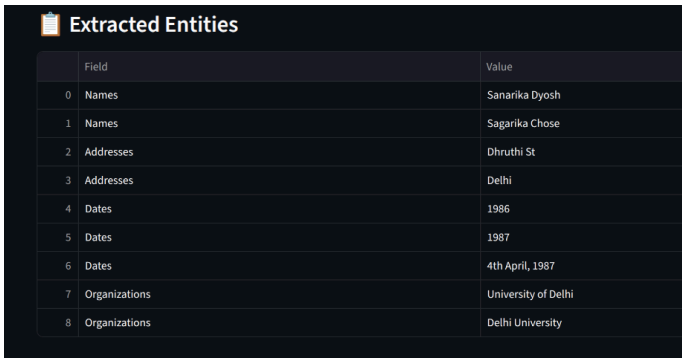
Stage 2 Roadmap:

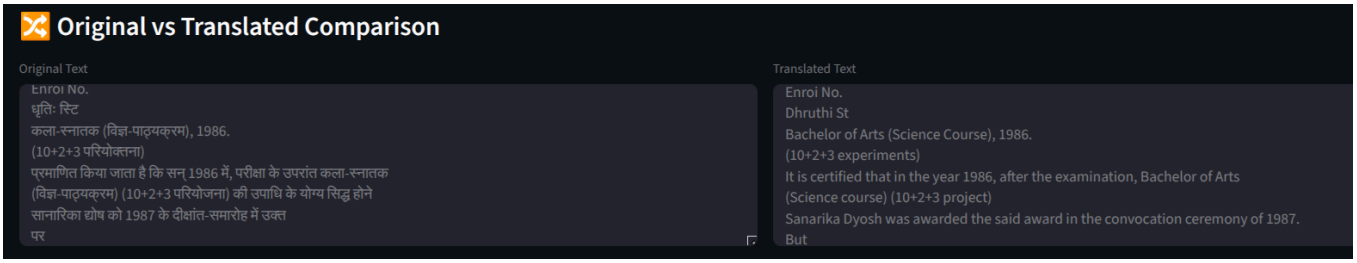
- DBSCAN clustering → 70%+ mIoU
- LayoutLMv3 fine-tuning → 65%+ F1
- Cross-document validation

Faded mixed languages low lighting condition example









Original_text_extracted from document in json:

": "## Page 1\n52651\nTHE STATE S-84\nRoll No.\nEnroi No.\nधृतिः स्टि\nकला-स्नातक (विज्ञ-पाठ्यक्रम), 1986.\n(10+2+3 परियोक्तना)\nप्रमाणित किया जाता है कि सन् 1986 में, परीक्षा के उपरांत कला-स्नातक\n(विज्ञ-पाठ्यक्रम) (10+2+3 परियोजना) की उपाधि के योग्य सिद्ध होने\nसानारिका द्योष को 1987 के दीक्षांत-समारोह में उक्त\nपर\nउपाधि प्रदान की गई।\nविषय इतिहास\nश्रेणी प्रथम\nUniversity of Delhi\nBACHELOR OF ARTS (HONOURS COURSE), 1986.\n(10+2+3 Scheme)\nThis is to certify that Sagarika Chose\nhaving been examined in 1986, and found qualified for the\ndegree of Bachelor of Arts (Honours Course) (10+2+3 Scheme) was\nadmitted to the said degree at the Convocation held in 1987.\nSubject Flistory\nDivision First\n· van + saav\nकुल-सचिव,\nदिल्ली विश्वविद्यालय\nबिल्ली

विश्वविद्यालय\nदिल्ली, दिनांक 4 अप्रेल, 1987.\nVice-Chancellor,\nRegistrar,\nUniversity of Delhi.\nDelhi, dated the 4th April, 1987.\nUniversity of Delhi.\n\n"

12. Alignment with IndiaAI IDP Requirements

Requirement	Architectural Support
Data Extraction	OCR with text and bounding boxes
Document Segmentation	Layout-aware processing
Structured Output	JSON-based entity extraction
Summarization	NLP-driven contextual summaries
Multilingual Support	Language detection and translation
Scalability	Modular pipeline
Responsible AI	Transparency and auditability

13. Future Enhancement Roadmap

Stage 2

Dataset-specific fine-tuning

Pre-trained OCR and NLP models will be fine-tuned on curated, domain-specific government document datasets (e.g., land records, court orders, licenses). This improves accuracy for entity extraction, classification, and summarization by adapting models to document layouts, terminology, and linguistic patterns specific to each department.

Formal train–test split

A labeled dataset will be created and divided into training, validation, and test sets following standard ML practices. This enables objective performance measurement, reproducibility, and comparison across model versions, supporting evidence-based model improvements and procurement evaluations.

Cross-document verification

Information extracted from multiple related documents (e.g., applicant forms, supporting certificates, and historical records) will be cross-validated to detect inconsistencies, missing fields, or conflicts. This adds reliability checks without replacing human decision-making.

Advanced compliance validation

Rule-based and AI-assisted validation layers will be introduced to automatically check extracted data against regulatory, policy, or procedural requirements (e.g., mandatory fields, date validity, authority signatures), supporting compliance reviews and reducing manual verification effort.

This roadmap ensures future readiness while maintaining current feasibility.

14. Conclusion

The proposed architecture provides a robust, scalable Intelligent Document Processing system tailored for Indian public-sector requirements. By combining OCR, multilingual NLP, structured extraction, and explainable summarization, the solution strongly aligns with IndiaAI's vision. Cost Leadership: Unlike cloud OCR services charging ₹1+ per page, our open-source solution requires one-time setup with fixed operational cost(₹0.09/page electricity). Cost stays constant regardless of volume - ideal for high-throughput government workflows processing lakhs of pages daily.