Report: Approaches for Automatic Document Classification and Splitting from PDFs

# Problem Statement

We have PDF files containing multiple unknown documents, where:  
- The number of documents is unknown.  
- The number of pages per document can vary.  
- The document types (e.g., Aadhaar, PAN, Payslip, Bank Passbook) are unknown.  
- Goal: Automatically split PDFs into individual documents and optionally classify them into categories.

# Approach 1: Supervised Classification

Description:

Requires labeled training data: PDFs or pages labeled with document types.  
Steps:  
1. Split PDF into pages.  
2. Extract text/images from pages.  
3. Train a classifier (Logistic Regression, Random Forest, XGBoost, or deep learning) to predict document type.  
4. Group consecutive pages with the same predicted label into documents.

Advantages:

- High accuracy if enough labeled data is available.  
- Can predict document types directly.  
- Efficient for production pipelines with known document types.

Limitations:

- Requires labeled dataset.  
- Cannot detect new/unknown document types.  
- Not flexible for variable-length documents unless preprocessing is robust.

Use Cases:

- Large enterprise document management systems with predefined document types.  
- HR portals, finance departments, or government portals where document types are fixed.

# Approach 2: Rule-Based / Heuristic Methods

Description:

Uses manual rules based on:  
- Page patterns (titles, headers, footers, page numbers).  
- Word count, line count, or specific keywords.  
- Signature, date, or other markers.  
Detects document boundaries using heuristics and predefined patterns.

Advantages:

- Does not require labeled data.  
- Simple and fast to implement.  
- Works well for structured documents with consistent formatting.

Limitations:

- Poor generalization: fails for unseen layouts or scanned documents.  
- Heuristics may need frequent tuning.  
- Cannot handle semantic similarity between pages.

Use Cases:

- Scenarios with standardized forms (e.g., payslips, invoices, contracts).  
- Organizations with fixed templates for each document type.

# Approach 3: Unsupervised Semantic Embedding + Clustering

Description:

Uses text embeddings (Sentence-BERT, OpenAI embeddings) to represent each page.  
Steps:  
1. Split PDF into pages.  
2. Convert page text into embeddings.  
3. Measure similarity between consecutive pages.  
4. Detect boundaries using similarity drop or adaptive threshold.  
5. Optionally, apply clustering (HDBSCAN/KMeans) to discover document types.  
6. Group consecutive pages with same cluster → form documents.

Advantages:

- Fully unsupervised: no prior knowledge of document types required.  
- Handles variable-length documents.  
- Can discover new document types automatically.  
- Supports scanned documents with OCR integration.

Limitations:

- Requires computational resources for embeddings.  
- Threshold tuning may be needed for very noisy PDFs.  
- Clustering quality depends on semantic similarity of pages.

Use Cases:

- PDFs with unknown or dynamic document types.  
- Document ingestion pipelines in banks, finance, or government where incoming PDFs vary.  
- Any system requiring unsupervised splitting and grouping of multi-document PDFs.

# Enhanced Techniques

1. Enhanced Boundary Detection:  
- Combines semantic similarity + content pattern features (titles, headers, signatures, word count, dates).  
- Uses adaptive thresholds and weighted scoring.  
- Reduces false positives and improves splitting accuracy.  
  
2. Clustering + Sequence Smoothing:  
- Cluster all pages using HDBSCAN.  
- Merge consecutive pages of the same cluster → documents.  
- Supports repeated document types and variable page lengths.  
  
3. Visualization:  
- UMAP to reduce embeddings to 2D → visualize clusters.  
- Helps inspect document type distribution.  
  
4. OCR Integration:  
- Use Tesseract for scanned/image PDFs.  
- Ensures text extraction for embeddings and boundary detection.

# Conclusion

- If you know document types and have labeled data → Use Supervised Classification.  
- If you have standardized templates → Use Rule-Based / Heuristic methods.  
- If you don’t know document types, page counts, or layout → Use Unsupervised Embedding + Clustering, optionally with enhanced boundary detection and OCR.  
  
This approach is most flexible and production-ready for real-world scenarios with unknown, mixed, or variable-length documents.