

Dokumentenklassifikation: Text- vs. Layoutanalyse

Rupert Parzmair, Andreas Salminger, Clemens Zellinger
Hagenberg, 12.01.2026

Motivation / Problemstellung

Problem

- Riesige Mengen an unstrukturierten PDFs
- Manuelle Sortierung zu aufwendig und teuer

Ziel

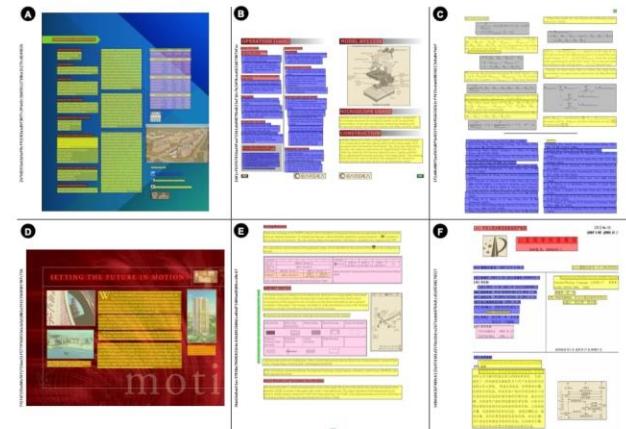
- Automatische Zuordnung in 6 Dokumentenklassen
- Steigerung der Genauigkeit und Zuverlässigkeit durch Kombination von Text- und Bilddaten

Kernfrage

- Reicht der Textinhalt oder brauchen wir zusätzlich das visuelle Layout?
- Vergleich: **Text** (TF-IDF) vs. **Layout** (YOLOv8) vs. **Hybrid**

Datensatz: DocLayNet

- IBM Research (2022)
- ~80.000 annotierte Seiten
- verschiedene Dokumentenlayouts
(Finanzberichte, wissenschaftl. Arbeiten, ...)



<https://dl.acm.org/cms/asset/8d780d5e-1dda-4c4f-aee4-9a32576d1753/3534678.3539043.key.jpg>

DocLayNet Core

- Dokumente als PNG-Bilder
- COCO-Format
- visuelle Struktur (Tabellen, Grafiken, ...)
- → YOLOv8 Objekterkennung

DocLayNet Extra

- Originale PDF-Dateien
- Textinhalte + Metadaten als JSON
- → TF-IDF Klassifikation

Methodik & Architektur

Split

- Split aus Datensatz übernommen
- Train/Val/Test: ca. 86/8/6

Pipeline A: textbasiert

- Aufbereitung JSON-Dateien von Doclaynet_extra in CSV
 - > CSV-Columns: page_id, original_filename, page_no, label, text
- Text in TF-IDF-Analyse (Term Frequency-Inverse Document Frequency)
- Modelle: LogisticRegression, LinearSVC, RF-Classifier

Methodik & Architektur

Pipeline B: layoutbasiert

- Aufbereitung COCO-Format für YOLO-Labels
- YOLO8n-Modell → PDF-LAYOUTS → Features → Modelle
 - > Features: count, area, mean_y, std_y, header/footer_ratio
 - > Modelle: LogisticRegression, LinearSVC, RF-Classifier

Pipeline C: hybrid

- Kombination der Features aus Pipeline A und B (Text + Layout)
- gleiche Modelle wie zuvor

Pipeline D: reine Bilderkennung

- 1 PNG pro Seite ohne Layoutinfos

Ergebnisse

