Data Science Practicum Proposal: Document Intelligence System

Hana Mengistu
Data Science Program
Regis University, Denver, CO, USA
hmengistu@regis.edu

I. Abstract:

This practicum proposal focuses on the design and development of a document intelligence system capable of automatically extracting both structured and unstructured information from a wide variety of documents, including scanned PDFs and images. Many modern organizations continue to rely on manual or semi-manual processes to convert physical documents and digital PDFs into structured formats such as spreadsheets or databases in order to manage records, invoices, forms, and reports. These processes are time-consuming, error-prone, and difficult to scale. While commercial document understanding platforms exist, they are often limited in flexibility, constrained to specific document types, or prohibitively expensive for small and medium-sized organizations.

The proposed project aims to build an end-to-end document processing pipeline that ingests heterogeneous documents and produces structured outputs such as raw text, detected tables, and form-like key–value pairs. The system will integrate computer vision techniques, optical character recognition (OCR), and machine learning–based layout analysis to identify and extract meaningful document components. Emphasis will be placed on modular system design, interpretability, and the use of open-source tools to replicate the core functionality of large-scale document AI platforms while remaining accessible and extensible.

The expected outcome of this project is a working prototype capable of processing at least two distinct document types, demonstrating effective document understanding across varying layouts and formats. The results will illustrate how modern data science and machine learning techniques can be applied to automate document analysis tasks. This project addresses a real-world problem with direct relevance to healthcare, finance, and government domains, where efficient and accurate document processing is critical.

Index Terms

Document Intelligence, OCR, Computer Vision, Layout Analysis, Automated Data Extraction.

II. Introduction/Background:

Organizations across industries continue to depend on documents as a primary medium for information exchange. When document workflows become complex, data scientists are often called in to translate unstructured or semi-structured document data into digital formats. Current AI systems, such as ChatGPT and other document-processing tools, are capable of reading and interpreting documents from PDFs or images. However, these systems are typically designed for general purposes and are prone to errors when translating complex documents.

Building a system specifically focused on document intelligence, trained to handle a variety of document types consistently, has a higher probability of success. Extracting meaningful information from documents remains a long-standing challenge due to variations in layout, quality, language, and structure. While commercial platforms such as Amazon Textract, Google Document AI, and Microsoft Azure Document Intelligence have made significant progress, each has its own strengths and weaknesses. For example, in manual labeling tasks, the Azure approach tends to be faster and more accurate compared to Google, but if the system detects data incorrectly, there is often no easy way to provide corrections [1].

This demonstrates that although powerful systems exist from major tech companies, there is still significant room for growth and improvement in document intelligence solutions. A dedicated, focused system that integrates OCR, computer vision, and machine learning techniques can address these limitations and improve the accuracy and reliability of automated document processing.

III. Problem Statement:

The central challenge of this practicum is automating the extraction of meaningful information from diverse, unstructured documents such as PDFs, images, invoices, forms, reports, and tables.

This project will build an end-to-end document intelligence system capable of accurately extracting text, tables, and key–value pairs from heterogeneous documents. The stages of development include:

- **Data Collection:** acquiring diverse document types
- **Preparation:** preprocessing PDFs and images
- **Exploration:** analyzing layouts and content
- **Modeling:** applying OCR, computer vision, and machine learning
- **Reporting:** outputting structured, actionable information

A dedicated, modular system focused on document understanding can reduce manual labor, improve accuracy, and accelerate workflows. For example, in the legal domain, law firms process massive quantities of paperwork, including contracts, briefs, and personal letters. Traditionally, lawyers or paralegals had to review every document manually, a time-consuming and error-prone process [2]. Automating this workflow can save hundreds of hours per month and significantly reduce human errors.

IV. Related Work:

Significant work in document AI includes the development of Intelligent Document Processing (IDP) systems. Research indicates that while OCR is a mature technology, the spatial understanding of key-value pairs remains an active area of improvement [3].

V. Methodology/Approach

The methodology for this project focused on developing a model capable of interpreting table structures from PDF inputs, alongside handling standard text-based PDFs.

Data Acquisition and Preprocessing

The dataset consists of diverse document types, including government records, bankruptcy filings, handbooks, and scanned memos. To ensure data quality, I performed the following preprocessing steps:

- **Format Conversion:** Converted multi-page PDFs into individual `.jpg` files for per-page processing.
- **Noise Reduction:** Used `cv2` thresholding to remove artifacts and visual noise.

- **Classification:** Categorized the data into three groups based on visual complexity:
  - *OCR-Ready:* Pure text-based documents.
  - *Medium:* Documents containing minimal imagery.
  - *Noisy:* Documents containing significant image content.

Annotation and Model Training

I utilized Roboflow to manually annotate the OCR-ready dataset. The labels included: `header`, `paragraph`, `table`, `table_info`, `table_row_header`, and `table_column_header`.

The pipeline integrates three primary models:

1) **Tesseract:** For initial text recognition.
2) **Table Transformer:** To identify cell boundaries and understand table structure.
3) **TrOCR:** For final text transcription.

Training Specifications

Due to the manual nature of the labeling, the training set consisted of 109 documents, split into training (77), validation (20), and testing (12) sets. To improve model performance, I utilized a "crop-based" training strategy, generating 10,000 crops for training and over 3,000 for validation. These crops focus on individual table cells to enhance the model's structural identification capabilities.

VI. Data Analysis

Training Progress and Convergence

The model's training performance was monitored over 3,000 steps to evaluate the reduction in Character Error Rate (CER). As illustrated in the performance trend, the initial CER of 51% at step 500 was progressively reduced through iterative training.

By the conclusion of the 5th epoch, the model achieved a final CER of 40.8%. This steady downward trajectory indicates effective model convergence, though it highlights the ongoing need for optimization in high-precision scenarios.

Data Analysis Future Optimization Path

To build upon these results, the next phase of development will focus on aggressive error reduction and domain expansion:

1) **Extended Training:** Implementing 10-20 additional epochs with a target CER of less than 20%.
2) **Diversification:** Expanding the training set to include diverse document domains such as legal contracts, invoices, and medical records.
3) **Architecture Refinement:** Aiming for a sub-10% CER on core content types through more robust data augmentation and hyperparameter tuning.

VII. Expected Outcomes:

This project will produce a working prototype capable of extracting text, understanding table structures, and identifying key–value pairs from diverse documents. The system will use OCR for text extraction, computer vision to detect tables and document structure, machine learning to identify key–value pairs, and open-source tools such as OpenCV and Python ML libraries.

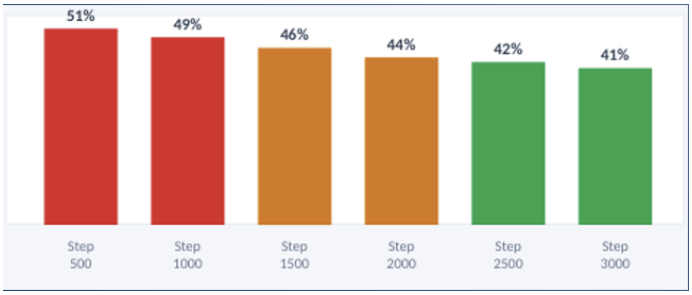

Fig. 1. Enter Caption

The expected impact includes reducing manual labor, improving extraction accuracy, and accelerating workflows in industries such as healthcare, finance, legal services, and more. Academically, this project demonstrates the application of multiple MSDS program skills, including data collection, cleaning, analysis, visualization, and reporting.

Limitations include potential errors on poorly scanned or highly complex documents. Future work may expand the system to support additional languages, document types, or real-time processing, further increasing its applicability and robustness.

VIII. Timeline:

With the help of AI I have created a Timeline to better help the project stay on track.

Week	Plan	Milestones
Week 1	Define project scope, write proposal, identify objectives, select tools	Submit Week 1 Project Proposal
Week 2	Collect sample PDFs, images, and forms (3 days); explore document types and layouts	Preliminary dataset ready; first progress update
Week 3	Clean and preprocess documents; convert scanned PDFs to images; normalize formats	Data preprocessing completed; pipeline tested on small dataset
Weeks 4–5	Implement OCR pipeline (Tesseract or similar); extract raw text; debugging and code refinement	OCR prototype working on 1–2 document types; progress updates
Week 6	Apply computer vision for table detection; ML models for key–value pairs	Prototype can extract tables/key-values from 2 types
Week 7	Combine OCR, table detection, and key–value extraction into a unified system	Integrated end-to-end prototype; preliminary results ready
Week 8	Evaluate accuracy, fix errors, optimize models, finalize report and presentation	Submit final report, Regis Portfolio showcase (Wednesday)

IX. Conclusion

This project successfully created a OCR pipeline capable of integrating `Tesseract` for initial text identification, the `Table Transformer` for structural table analysis, and `TrOCR` for character transcription. Through a 3,000-step training , the model achieved a final Character Error Rate (CER) of 40.8%, which demonstrates the ability to understand table structure and non table structure.

While the current performance provides a functional foundation, the model's still has some limitations as well as imporvments that can be taken into account:

Current Limitations

- **Input Constraints:** The model is currently limited to printed text and does not support handwriting.
- **Dependency on External Label:** Reliance on pre-labeled YOLO bounding boxes increases the complexity of the pipeline.
- **Domain Specificity:** Current performance is optimized for the bankruptcy dataset; retraining is required for generalization to other document types.

Moving forward, the primary objective is to scale the model's domain applicability by expanding the training dataset to include diverse document while targeting a CER of below 10%.

REFERENCES

[1] F. Wu, "Comparison of ai ocr tools," https://persumi.com/u/fredwu/tech/e/blog/p/comparison-of-ai-ocr-tools-microsoft-azure-ai-document-intelligence-google-cloud-document-ai-aws-textract-and-others, 2024.

[2] P.-U. Ricoh, "Ai in document management," https://www.pfu-us.ricoh.com/blog/ai-in-document-management, 2024.

[3] Tech.us, "How to improve document processing accuracy using document ai," https://tech.us/blog/how-to-improve-document-processing-accuracy-using-document-ai, 2024.

[4] Anthropic, "Claude 3.5 sonnet," 2026, aI language model used as a coding and technical assistant.