**End-to-End Research Template**

[Grab your reader’s attention with a great quote from the document or use this space to emphasise a key point. To place this text box anywhere on the page, just drag it.]

*Extraction Evaluation → ML Evaluation → Publishable Paper*

*Use this as a single living document. Every section can be copied directly into a paper later.*

Student / Intern Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Supervisor(s): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Institution / Lab: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Start Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ End Date (planned): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Version: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# 1. Project Definition

Working Title (technical and precise):

Domain (e.g. scientific PDFs, reports, medical, energy, finance):

Core Problem (2 lines):

* Why unstructured PDFs are a bottleneck.
* Why this matters for downstream ML and decision making.

Success Criteria (what ‘done’ looks like):

* Extraction reaches target quality on held-out ground truth.
* ML shows stable performance with clear evaluation.
* Clear link is demonstrated between extraction quality and ML results.

# 2. Research Question and Hypothesis

Main Research Question:

How does the accuracy of structured information extraction from PDFs affect downstream machine-learning performance?

Hypothesis:

Improvements in extraction accuracy lead to measurable gains in ML model performance and stability.

# 3. Contributions (update continuously)

By the end, this should read like the contribution list in a paper.

* A validated pipeline for extracting structured variables from PDFs.
* A rigorous field-level evaluation framework for extraction quality.
* An empirical study linking extraction accuracy to downstream ML performance.

Your current contributions (add new bullets as you progress):

•

# 4. Data Description

## 4.1 PDF Dataset

* Number of PDFs:
* Source and domain:
* Native vs scanned breakdown:
* Languages (if relevant):
* Licensing and ethics notes:

## 4.2 Challenges Identified

* Layout variability (multi-column, headers, footers, mixed sections).
* OCR noise (if scanned).
* Tables, symbols, equations, units, and abbreviations.

# 5. Information Extraction Pipeline

## 5.1 Pipeline Overview

Insert pipeline diagram here (or paste an image screenshot of the pipeline).

Diagram location / link: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## 5.2 Extraction Method

* Tools used (PDF parser, OCR engine, LLM or rules, table extractor, etc.):
* OCR strategy (if any):
* Layout handling (reading order, columns, tables, figures):
* Entity and unit normalization approach:

## 5.3 Post-Processing

* Cleaning rules (whitespace, symbols, encoding).
* Unit normalization (e.g. mA to A, mg to g).
* Missing and ambiguous value handling (null rules, confidence thresholds).
* De-duplication and consistency checks.

# 6. Variable Definition (Critical Table)

This table can be used in the paper with minimal changes.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Variable | Description | Unit | Extraction Method | Notes (edge cases) |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Part I. Extraction Evaluation (this is the research)

## 7. Ground Truth Construction

* Number of manually annotated PDFs:
* Annotation guidelines (how to label each variable):
* Who annotated (single annotator or multiple):
* Consistency approach (double labeling, adjudication, agreement metric):
* Train / validation / test split at the PDF level (no leakage):

Ground truth storage format (CSV, JSON, database schema): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## 8. Extraction Evaluation Metrics

Evaluate per variable, not only globally.

* Precision, recall, F1 score (for categorical or entity extraction).
* Exact match or tolerance match (for numeric values, with defined tolerance).
* Coverage rate (percentage of PDFs where a field is extracted).
* Confidence calibration (optional if model outputs confidence).

Define tolerance rules for numeric fields (example: ±1 percent, or exact units match):

## 9. Extraction Results

### 9.1 Quantitative Results

Insert tables for: metric per variable, average performance, and variance across document types.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Variable | Precision | Recall | F1 | Coverage | Notes |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

### 9.2 Error Analysis

* Common failure modes (OCR, layout, table parsing, ambiguous wording, unit mismatch).
* Error taxonomy with counts and examples.
* Top 5 fixes and the expected impact.

Example errors (paste screenshots or snippets):

# Part II. Downstream Machine Learning

## 10. Feature Engineering

### 10.1 Raw Features

List the variables used directly as features:

•

### 10.2 Derived Features

Examples: ratios, aggregations, trends, interaction terms.

Derived feature list and justification:

•

## 11. ML Experimental Setup

* Task type (classification, regression, ranking, forecasting):
* Target variable and label definition:
* Models (baseline required, advanced optional):
* Training protocol (split strategy, cross-validation):
* Fixed random seeds and reproducibility settings:
* Hyperparameter strategy (grid, random, Bayesian):

## 12. ML Evaluation Metrics

Choose metrics that match your task.

* Classification: accuracy, F1, ROC-AUC, PR-AUC, confusion matrix.
* Regression: RMSE, MAE, R2, residual plots.
* Imbalance handling: macro F1, weighted F1, class-wise results.

## 13. ML Results

### 13.1 Baseline Results

Performance using extracted features.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model | Features | Metric 1 | Metric 2 | CV or Test | Notes |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

### 13.2 Feature Importance

* Permutation importance or SHAP (if appropriate).
* Top contributing features and interpretation.
* Stability of importance across folds.

# Part III. Extraction ↔ ML Link (the paper hook)

## 14. Impact of Extraction Quality on ML

This is the core scientific contribution. Design experiments that change extraction quality and measure ML impact.

* ML with raw extraction output.
* ML with cleaned or corrected extraction.
* ML with missing fields simulation (dropout by variable).
* ML with noise injection (numeric perturbation, entity swapping).

Key result statement template:

ML performance improves by \_\_\_\_ when extraction F1 improves from \_\_\_\_ to \_\_\_\_ (with confidence intervals).

## 15. Discussion

* When extraction quality matters most (which tasks, which variables).
* Which variables are robust vs fragile.
* Practical implications for real-world deployment pipelines.

## 16. Limitations

* Dataset size and representativeness.
* Domain specificity and generalization.
* OCR constraints and annotation bias.
* Assumptions made in evaluation.

How you will address limitations in future work:

•

## 17. Reproducibility Statement

* Code versioning (Git repository, tagged releases).
* Fixed seeds, environment capture (requirements, conda, Docker).
* Config files for runs and logging (experiment IDs).
* Clear rerun instructions (one command or notebook).

## 18. Conclusion and Future Work

Conclusion: summarize what was built, what was validated, and what was learned.

Future work: scaling dataset, improved extraction, extension to other domains.

# Publishability Checklist (how to know you are ready)

A project is typically publishable when most of the following are true:

1. Clear research question and a measurable hypothesis.
2. A dataset and ground truth that others can trust (documented labeling process).
3. Strong baselines and fair comparisons (not only one model).
4. Results are repeatable (same code, same seeds, same splits).
5. Extraction evaluation is field-level, not vague, and includes error analysis.
6. Downstream ML shows a meaningful, consistent gain, not a one-off spike.
7. You can explain why the gain happens (linking extraction errors to ML failure).
8. Limitations are stated honestly and do not break the main claim.
9. The work offers a reusable artifact: pipeline, evaluation protocol, or dataset.

What would make this work more publishable in your case:

•

# Weekly Research Log (internal, but powerful)

Fill this every week. It makes your work easy to defend and easy to write up.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Week | What I built | What I evaluated | What improved | What failed | Next step |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

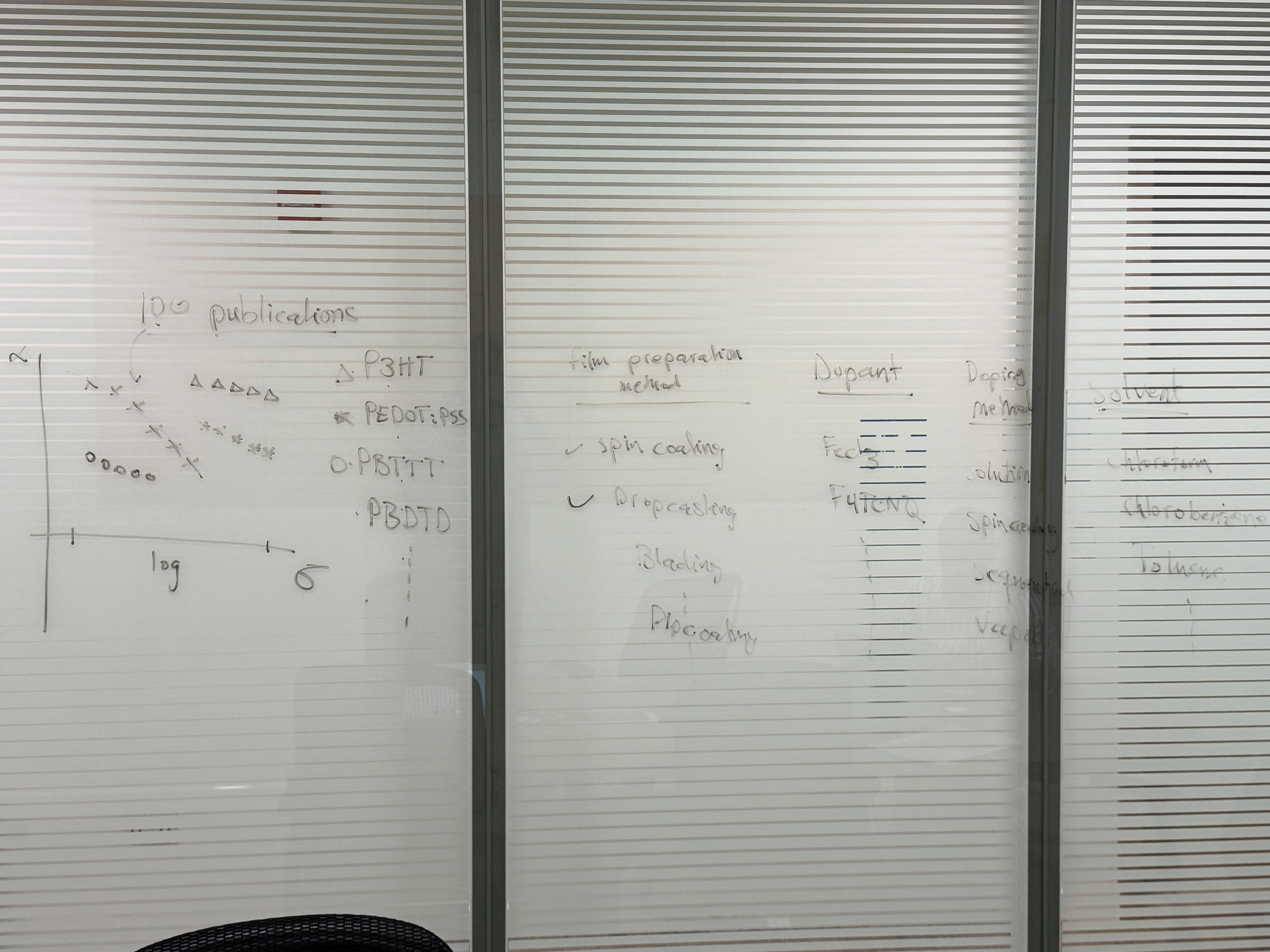
Notes and meeting action items:

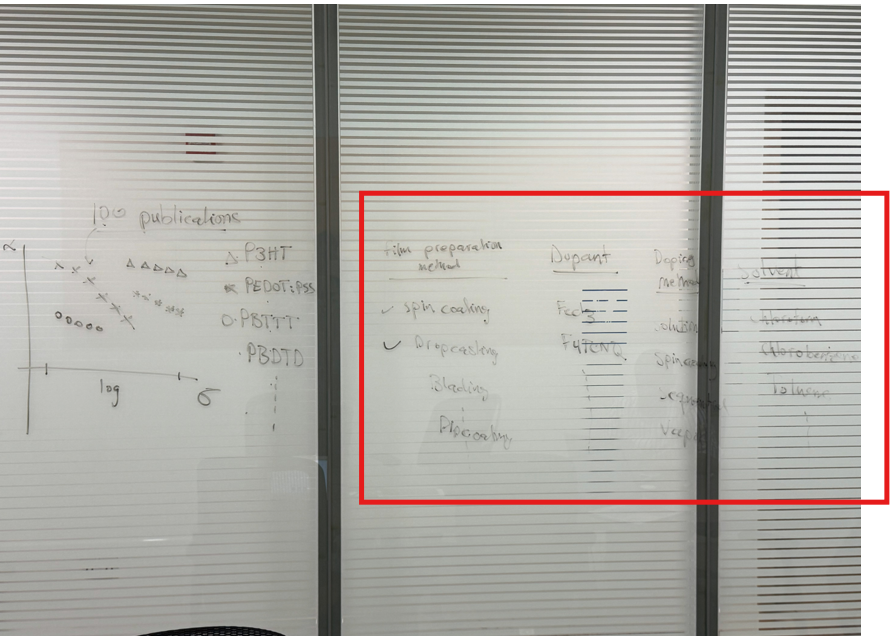
•

A graph of different types of data

AI-generated content may be incorrect.

<https://onlinelibrary.wiley.com/doi/pdf/10.1002/marc.201700727>





Add new column called reference and add check the other review papers

A white board with writing on it

AI-generated content may be incorrect.

A white board with lines and words on it

AI-generated content may be incorrect.

***Using structure to analyze trends***