Structure-Preserving Extraction of Tabular Data from Engineering Manuals

Vinícius A. Aguiar[1,2] **, Gustavo L. B. Pereira** [1,2] **, Joao Pedro Gomes Fernandes**[1,2] **, Artur M. A. Novais**[1,2] **, Thiago Pedroso de Jesus** [1,2] **, Sávio S. T. de Oliveira**[1,2] **, Anderson S. Soares**[1,2]

[1]Institute of Informatics, Federal University of Goiás (UFG), Goiânia - Goiás - Brazil ,
[2]Center of Excellence in Artificial Intelligence (CEIA), Goiânia - Goiás - Brazil ,

{vinicius_aguiar2, gustavobueno, joaopedro234567}@discente.ufg.br

{artur.matos, thiagopedroso}@discente.ufg.br

savioteles@ufg.br, andersonsoares@ufg.br

Abstract. *Engineering manuals encode operational parameters in dense tables, scanned pages, hierarchical sections, and multi-page layouts. Conventional OCR and layout conversion pipelines often flatten these structures, weakening the association between parameters, values, and document context. This work proposes a structure-preserving extraction pipeline that uses multimodal page parsing, explicit semantic tags, deterministic validation, and row-level context injection to convert heterogeneous manuals into structured tabular records. On 60 pages from six protection-relay manuals, the pipeline produced usable output for allevaluated pages, identified 04 structured tables, assigned 197 captions, and reconstructed 26 multi-page tables. The evaluation focuses on extraction and structural preservation and does not assess end-to-end question answering or SQL execution.*

1. Introduction

Engineering workflows depend on technical manuals that encode protection parameters, configuration settings, and operational constraints. These manuals often circulate as PDF files with dense tables, hierarchical sections, scanned pages, and layouts that continue across page boundaries. Correct interpretation requires more than text recognition: the extraction process must preserve the association between each parameter, its value, and the section in which it appears. Losing this context may affect downstream interpretation in power system operation, where protection systems rely on many documented parameters [Horowitz et al. 2022, Terzija et al. 2011, Phadke and Thorp 2008].

Document understanding methods address parts of this problem, but engineering manuals combine several difficult cases in the same document. Layout-aware models incorporate spatial information, but typically depend on OCR input [Li et al. 2022, Huang et al. 2022, Hong et al. 2022]. OCR-free models such as Donut and Nougat process document images directly [Kim et al. 2022, Blecher et al. 2023], but generative decoding can produce inconsistent structures in irregular or multi-page layouts. DeepDeSRT

addresses table structure recognition [Schreiber et al. 2017], and TEDS measures structural similarity [Zhong et al. 2020]. Engineering use cases also require semantic grounding between rows, captions, pages, and section hierarchy.

This gap matters for RAG and structured-query applications. Typical pipelines segment documents into text chunks [Lewis et al. 2020]. When tables are flattened into text, chunking may separate values from their attributes or section context. Table-centric retrieval methods [Yu et al. 2025, Souza et al. 2024] and structured query generation approaches [Zhong et al. 2017] assume that tabular representations are available. In engineering manuals, this assumption often fails because the structured representation must first be extracted from heterogeneous PDFs.

This work proposes a pipeline that preserves structure while extracting tables from engineering manuals. The method uses multimodal page parsing, explicit semantic tags, deterministic validation, and row-level context injection to transform tables from unstructured documents into structured records. Retrieval is therefore treated as a downstream motivation, while the evaluation isolates extraction reliability and structural preservation.

This work makes four contributions. First, it defines a semantic tagging strategy that makes page boundaries, table blocks, captions, and multi-page continuations explicit during multimodal extraction. Second, it introduces a row-level context injection mechanism that preserves relationships among values, attributes, captions, pages, and document hierarchy. Third, it describes a structured storage representation for heterogeneous tables extracted from engineering manuals, supporting downstream querying without treating table rows as isolated text chunks. Fourth, it evaluates the pipeline on 60 pages from six protection-relay manuals. On the evaluated pages, the method produced usable output for all pages, identified 204 structured tables, assigned 197 captions, and reconstructed 26 multi-page tables.

2. Related Work

Document understanding models increasingly combine text, layout, and visual signals. MarkupLM models text and markup for visually rich documents [Li et al. 2022]. LayoutLMv3 combines text and image masking [Huang et al. 2022]. BROS focuses on text blocks and layout relations [Hong et al. 2022]. These models usually rely on OCR, which makes the pipeline sensitive to scan quality and layout degradation. OCR-free approaches such as Donut [Kim et al. 2022] and Nougat [Blecher et al. 2023] avoid an explicit OCR stage by generating structured output directly from document images, but their generative nature can introduce inconsistent markup when layouts are irregular.

Table extraction remains a distinct challenge because tabular meaning depends on row-column alignment, hierarchical headers, captions, and surrounding sections. DeepDeSRT [Schreiber et al. 2017] addresses table detection and structure recognition from document images. TEDS [Zhong et al. 2020] measures structural similarity between predicted and reference tables. PubTables-1M [Smock et al. 2022] provides a large annotated corpus for table extraction. FinTabNet [Zheng et al. 2021] offers another reference corpus for table recognition. These resources are useful for structural evaluation, but they do not cover all practical constraints of engineering manuals, especially heterogeneous manufacturer layouts, scanned pages, and table continuations across pages.

Document conversion systems combine OCR and layout reconstruction to create

machine-readable outputs. Docling [Livathinos et al. 2025] represents this class of practical parsers and provides a relevant baseline for conversion from PDF to structured text. However, such systems may flatten tables when layouts are dense or noisy, which can erase the association between parameters and values even when textual content is partially recovered.

RAG and table-centric retrieval motivate the need for structured extraction. Lewis et al. [Lewis et al. 2020] popularized retrieval-augmented generation over document collections, and TableRAG variants [Yu et al. 2025, Souza et al. 2024] explore retrieval over tabular inputs. Seq2SQL [Zhong et al. 2017] illustrates how language can be mapped to executable structured queries. These methods assume that reliable structured inputs exist. The present work addresses the upstream step: preserving table boundaries, captions, hierarchy, and row context while extracting tabular records from raw engineering manuals.

3. Methodology

Figure 1 summarizes the proposed pipeline. The method converts engineering manuals into structured tabular records through four stages: multimodal page extraction, semantic tagging, row-level context injection, and structured storage. The design follows two principles. First, table boundaries and document hierarchy should be made explicit during extraction rather than reconstructed only after parsing. Second, each row should carry enough context to remain interpretable outside the original PDF.

The extraction stage operates at the page level. Each page is treated as a visual input and processed by a vision-language model. The implementation evaluated in this paper uses Gemini 2.5 Flash for multimodal extraction. The model generates a Markdown representation under prompt constraints that request explicit structural markers. This choice makes the pipeline OCR-free at the extraction stage, while preserving compatibility with later deterministic parsing.

Generation uses semantic tags for structural elements. Each page receives a page marker, headings preserve document hierarchy up to a fixed depth, and detected tables receive explicit opening and closing tags. Captions are extracted when present or inferred from the closest hierarchical context. Tables that continue across pages are marked with continuation indicators, enabling the parser to reconnect segments as one logical table.

The extraction process also normalizes irregular tables commonly found in engineering manuals. Tables without clear column definitions are represented as Property-Value pairs, and nested headers are flattened through concatenation. This representation favors parseability while preserving the original hierarchical meaning in the column names.

After generation, a deterministic parser validates the intermediate representation. The parser checks whether tagged regions have valid boundaries, whether table rows can be parsed, and whether continuation markers can be associated with compatible table segments. Malformed structures are repaired only when a deterministic rule applies; otherwise, they are excluded from the validated structured table set. This prevents malformed generations from being inserted as trusted structured rows.

The validated Markdown representation is then parsed into tables. Instead of treating each table as an isolated grid, the pipeline augments each row with contextual

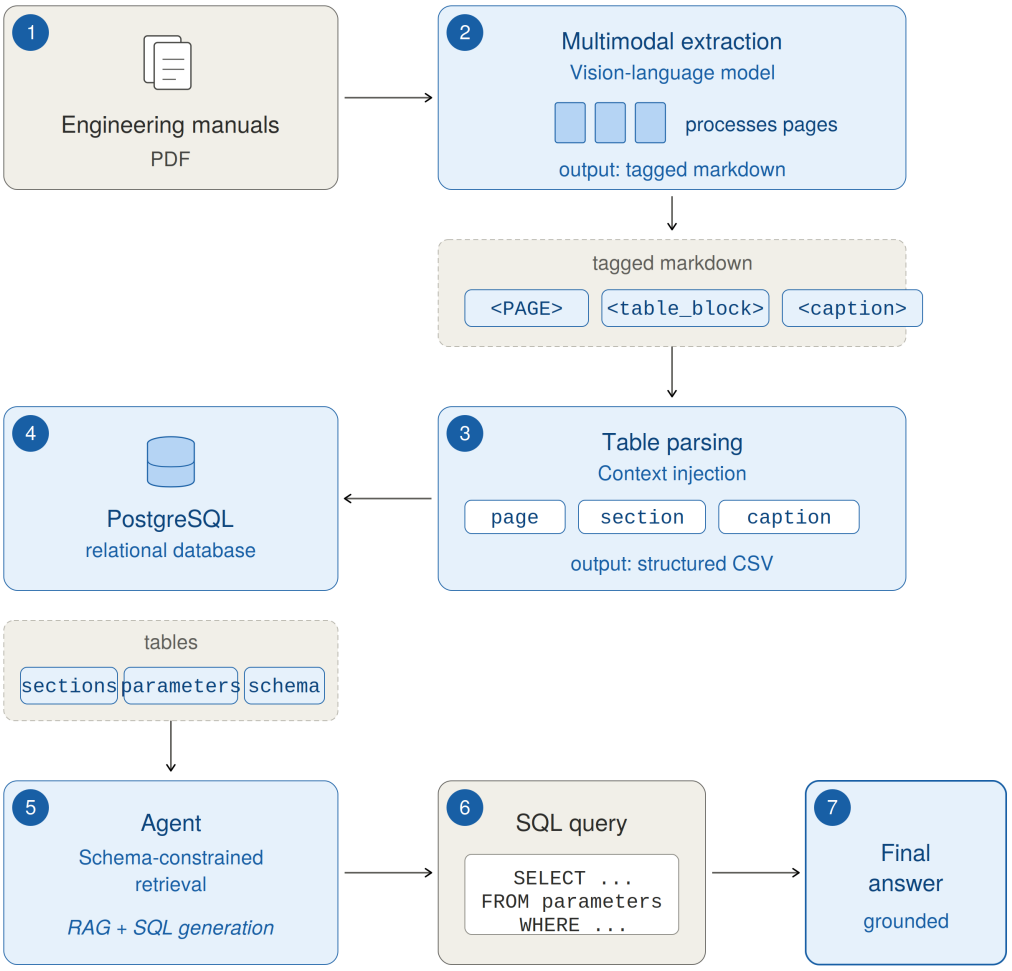


Figure 1. Overview of the proposed pipeline. Engineering manuals are processed through multimodal extraction with semantic tagging, validation, contextual structuring, and storage for downstream structured querying.

metadata: page number, section headings, table caption, and table index. This row-level grounding preserves relationships among values, attributes, and document context.

Each row is represented as:

$$r = (v_1, v_2, \ldots, v_n, c, p, h_1, h_2, \ldots, h_k, t), \tag{1}$$

where v_i denotes an extracted table value, c is the table caption, p is the page identifier, h_j represents hierarchical context, and t is the table index. This formulation allows an extracted row to be interpreted without recovering context from the original document.

The structured data is stored in a relational architecture designed for heterogeneous table schemas. A section-level table stores document metadata and semantic embeddings, a schema catalog stores table names, columns, and examples, and a sparse data table stores extracted rows with their contextual metadata. This design sacrifices strict schema uniformity to accommodate manuals from different manufacturers and table lay-

outs.

Structured querying motivates the representation but is not evaluated end to end in this paper. In a downstream workflow, a query system can inspect the stored schemas and retrieve rows under explicit table and section constraints. The evaluation below isolates the upstream question: whether the extraction pipeline preserves tabular structure and context well enough to populate such a structured representation.

3.1. Component Roles

The pipeline combines three components that address different failure modes in manual extraction. Semantic tags make page boundaries, tables, captions, and continuations explicit, giving the parser stable anchors in the generated Markdown. Without these tags, table content may still be present in the output, but downstream parsing must infer where a table starts, ends, or continues.

Row-level context injection addresses ambiguity after a table has been parsed. Engineering manuals often repeat parameter names across sections, pages, or device configurations. By attaching page, caption, section hierarchy, and table index to each row, the representation reduces the chance that a value is later interpreted without its original context.

The structured storage layer separates extraction from downstream querying. It does not bring retrieval accuracy into the measured scope, but it provides a consistent target for extracted rows and schemas. This separation keeps the evaluation focused on whether the document-processing stage can populate a structured representation from heterogeneous manuals.

4. Experimental Evaluation

The evaluation focuses on extraction reliability and structural preservation. It does not benchmark end-to-end question answering, SQL execution, or answer accuracy. This scope isolates the upstream step needed before structured querying: converting heterogeneous engineering manuals into parseable tabular records with preserved context.

The dataset consists of six protection-relay manuals from different manufacturers, including ABB, SEL, Alstom, and GE. From each document, ten pages were selected, resulting in 60 evaluated pages. The pages include natively digital PDFs and scanned documents with dense tables, hierarchical sections, and parameters distributed across page boundaries.

The baseline is Docling [Livathinos et al. 2025], a practical document conversion toolkit based on OCR and layout reconstruction. This comparison contrasts the proposed VLM-based structure-preserving pipeline with a traditional OCR/layout conversion pipeline. It should not be read as a comparison against another VLM baseline.

Table 1 reports coverage and structure-oriented indicators. Extraction Success Rate measures whether a page produced usable extracted content, defined as output with extractable text or tables and enough structure for downstream parsing. Pages dominated by empty content, image placeholders, or unparseable fragments are counted as failures. Scanned PDF Failures counts those cases among scanned inputs. Image Placeholders counts non-informative placeholder markers. Structured Tables counts table blocks with

explicit boundaries and parseable rows. Table Captions counts tables with an extracted or inferred caption. Cross-page Tables counts table reconstructions marked by continuation across pages. Output Size reports the generated output size, and Usable Lines counts non-empty extracted lines considered suitable for downstream structuring.

The outputs from both pipelines were inspected to identify extraction failures, loss of table boundaries, row-column degradation, and ambiguous parameter-value associations. Because the dataset is not a standardized benchmark and does not include cell-level ground truth, the results should be interpreted as evidence of structural preservation in the evaluated documents rather than as general benchmark performance.

5. Results and Discussion

Table 1. Comparison between Docling and the structure-preserving extraction pipeline.

Metric	Docling	Proposed
Extraction Success Rate	67%	100%
Scanned PDF Failures	33%	0%
Image Placeholders	38	0
Structured Tables	0	204
Table Captions	0	197
Cross-page Tables	0	26
Output Size (KB)	165.6	150.0
Usable Lines	3,367	4,220

Table 1 summarizes the comparison between Docling and the extraction pipeline on the evaluated pages. The main difference is structural: the method represents tables as explicit blocks with captions and continuation markers, while the baseline output often lacks table boundaries.

The extraction pipeline produced usable output for all evaluated pages. Docling failed on a subset of scanned inputs, with outputs dominated by non-informative image placeholders. This behavior is reflected by the 38 image placeholders in the baseline output and by the scanned PDF failure rate reported in Table 1.

The method identified 204 structured table blocks, assigned captions to 197 of them, and reconstructed 26 multi-page tables. These counts do not imply cell-level accuracy because the dataset does not include table ground truth. They show that the representation exposes structural units that can be parsed and stored, whereas the baseline does not provide explicit table blocks under the same criterion.

Output size and usable-line counts also show different extraction behavior. Docling produced a larger output, but part of this output consisted of repeated headers, formatting artifacts, and placeholder text. The extraction pipeline produced a smaller file while increasing the number of usable lines, suggesting a denser structured representation in the evaluated documents.

Parameter-value association The reported results should not be interpreted as a cell-level value fidelity benchmark, since the dataset does not include ground-truth annotations

for each table cell. The relevant evidence in Table 1 is structural: the representation keeps each extracted row tied to table, caption, page, and section metadata. This design reduces ambiguity in dense or continued tables, but exact value correctness remains outside the measured scope.

The results support the central design choice of preserving structure during extraction. In the evaluated manuals, explicit table boundaries, captions, and continuation markers made tabular content easier to parse and store than a flattened text representation. This distinction matters because engineering parameters are often meaningful only when their row, caption, page, and section context remain attached.

The comparison with Docling should be interpreted as a practical contrast with an OCR/layout conversion pipeline, not as a claim against all document understanding or VLM-based systems. Docling remains relevant because such tools are common first choices for PDF conversion. The observed failures show where a structure-preserving extraction strategy can be useful: scanned pages, dense tables, and layouts in which table continuation is part of the meaning.

The structured representation also supports downstream querying, but that stage remains outside the evaluation. The results instead indicate that structural preservation is an upstream requirement for query-oriented workflows, since errors introduced during extraction can propagate to later retrieval or query generation steps.

Several trade-offs remain. Prompt-constrained extraction depends on the behavior of the underlying VLM, and the current implementation was evaluated with Gemini 2.5 Flash. The pipeline is not tied to a specific output API, but changing the VLM would require revalidating prompt compliance and extraction quality. The sparse relational representation accommodates heterogeneous tables, but it does not enforce all column types or optimize queries as a manually designed schema would. These choices favor flexible ingestion of manuals from different manufacturers, while leaving broader benchmarking and query-time evaluation for future work.

5.1. Limitations

The evaluation has four main limitations. First, the dataset contains 60 pages from six manuals and does not cover the full variability of engineering documentation, including more degraded scans, diagrams, and complex merged cells. Since pages were selected rather than sampled from a standardized benchmark, selection bias is also possible. Second, the implementation uses Gemini 2.5 Flash, so performance may vary with other VLMs or under distribution shift. Third, the Markdown representation with custom semantic tags remains vulnerable to malformed generations, even though deterministic validation prevents invalid structures from entering the validated table set. Fourth, the evaluation does not include standardized cell-level metrics, TEDS, additional VLM baselines, or end-to-end retrieval metrics. The results should therefore be read as evidence of structural preservation in the evaluated manuals, not as a complete benchmark of document understanding or question answering.

6. Conclusion

This paper presented a structure-preserving extraction pipeline for engineering manuals. The approach combines Gemini 2.5 Flash-based page extraction, semantic tags, deter-

ministic validation, row-level context injection, and structured storage to preserve tabular relationships that are often lost in flattened PDF conversion.

On 60 evaluated pages from six protection-relay manuals, the pipeline produced usable output for all pages, identified 204 structured tables, assigned 197 captions, and reconstructed 26 multi-page tables. These results provide evidence of extraction reliability and structural preservation in the evaluated documents.

Future work should evaluate larger and more diverse manual collections, compare against additional VLM and table-aware baselines, add cell-level or TEDS-style ground truth where feasible, and quantify how extraction quality affects downstream structured querying.

Acknowledgments

This work has been funded by P&D CEMIG/ANEEL PD-04950- D0677/2023 supported by the Center of Excellence in Artificial Intelligence (CEIA), with financial resources from the MCTI grant number 057/2023, signed with EMBRAPII. This work was also supported by the National Institute of Science and Technology (INCT) in Responsible Artificial Intelligence for Computational Linguistics and Information Treatment and Dissemination (TILD-IAR) grant number 408490/2024-1.

Disclosure of Interests

The authors have no competing interests to declare that are relevant to the content of this article.

References

Blecher, L., Cucurull, G., Scialom, T., and Stojnic, R. (2023). Nougat: Neural optical understanding for academic documents. *arXiv preprint arXiv:2308.13418*.

Hong, T., Kim, D., Ji, M., Hwang, W., Nam, D., and Park, S. (2022). BROS: A pre-trained language model focusing on text and layout for better key information extraction from documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10767–10775.

Horowitz, S. H., Phadke, A. G., and Henville, C. F. (2022). *Power system relaying*. John Wiley & Sons, 5 edition.

Huang, Y., Lv, T., Cui, L., Lu, Y., and Wei, F. (2022). LayoutLMv3: Pre-training for document AI with unified text and image masking. In *Proceedings of the 30th ACM international conference on multimedia*, pages 4083–4091.

Kim, G., Hong, T., Yim, M., Nam, J., Park, J., Yim, J., Hwang, W., Yun, S., Han, D., and Park, S. (2022). OCR-free document understanding transformer. In *European Conference on Computer Vision*, pages 498–517. Springer.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., et al. (2020). Retrieval-augmented generation for

knowledge-intensive NLP tasks. *Advances in neural information processing systems*, 33:9459–9474.

Li, J., Xu, Y., Cui, L., and Wei, F. (2022). MarkupLM: Pre-training of text and markup language for visually rich document understanding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6078–6087.

Livathinos, N., Auer, C., Lysak, M., Nassar, A., Dolfi, M., Vagenas, P., Ramis, C. B., Omenetti, M., Dinkla, K., Kim, Y., et al. (2025). Docling: An efficient open-source toolkit for AI-driven document conversion. *arXiv preprint arXiv:2501.17887*.

Phadke, A. G. and Thorp, J. S. (2008). *Synchronized phasor measurements and their applications*, volume 1. Springer.

Schreiber, S., Agne, S., Wolf, I., Dengel, A., and Ahmed, S. (2017). DeepDeSRT: Deep learning for detection and structure recognition of tables in document images. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 01, pages 1162–1167.

Smock, B., Pesala, R., and Abraham, R. (2022). PubTables-1M: Towards comprehensive table extraction from unstructured documents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4624–4632.

Souza, E., da Silva, P. F., Gomes, D., Batista, V., Batista, E., and Pacheco, M. (2024). TableRAG: A novel approach for augmenting llms with information from retrieved tables. In *Proceedings of the 15th Brazilian Symposium in Information and Human Language Technology*, pages 182–191.

Terzija, V., Valverde, G., Cai, D., Regulski, P., Madani, V., Fitch, J., Skok, S., Begovic, M. M., and Phadke, A. (2011). Wide-area monitoring, protection, and control of future electric power networks. *Proceedings of the IEEE*, 99(1):80–93.

Yu, X., Jian, P., and Chen, C. (2025). TableRAG: A retrieval augmented generation framework for heterogeneous document reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 14074–14093.

Zheng, X., Burdick, D., Popa, L., Zhong, X., and Wang, N. X. R. (2021). Global table extractor (GTE): A framework for joint table identification and cell structure recognition using visual context. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 697–706.

Zhong, V., Xiong, C., and Socher, R. (2017). Seq2SQL: Generating structured queries from natural language using reinforcement learning. *CoRR*, abs/1709.00103.

Zhong, X., ShafieiBavani, E., and Jimeno-Yepes, A. (2020). Image-based table recognition: data, model, and evaluation. In *Computer Vision – ECCV 2020*, volume 12366 of *Lecture Notes in Computer Science*, pages 564–580. Springer.