Benchmarking Table Extraction: Multimodal LLMs vs Traditional OCR

[1,2]**Guilherme G.M. Nunes**, [1]**Vitor Rolla**, [1]**Duarte Pereira**, [1]**Vasco Alves**,
[1]**André Carreiro**, [2]**Márcia Lourenço Baptista**
[1]Fraunhofer AICOS, Portugal
[2]Information Management School (IMS)
Universidade Nova de Lisboa, Portugal
{guilherme.nunes, vitor.rolla}@fraunhofer.pt

Abstract

This paper compares two approaches for table extraction from images: deep learning computer vision and Multimodal Large Language Models (MLLMs). Computer vision models for table extraction, such as the Table Transformer model (TATR), have enhanced the extraction of complex table structural layouts by leveraging deep learning for precise structural recognition combined with traditional Optical Character Recognition (OCR). Conversely, MLLMs, which process both text and image inputs, present a novel approach by potentially bypassing the limitations of TATR plus OCR methods altogether. Models such as GPT-4o, Phi-3 Vision, and Granite Vision 3.2 demonstrate the potential of MLLMs to analyze and interpret table images directly, offering enhanced accuracy and robust extraction capabilities. A state-of-the-art metric like Grid Table Similarity (GriTS) evaluated these methodologies, providing nuanced insights into structural and text content effectiveness. Utilizing the PubTables-1M dataset, a comprehensive and widely used benchmark in the field, this study highlights the strengths and limitations of each approach, setting the stage for future innovations in table extraction technologies. Results show that deep learning computer vision techniques still have a slight edge when extracting table structural layout, but in terms of text cell content, MLLMs are far better.

1 Introduction

With the increasing volume of digital documents, such as records, manuals, and scientific papers, processing and transforming them into representations that allow proper extraction of information has become highly challenging (Staar et al., 2018). Many of these documents contain tables, as they help represent data in an organized, readable, and straightforward manner. However, automatically identifying and extracting structural layout and content information becomes more complex, which can be crucial in scientific and business applications (Chen et al., 2023; Burdick et al., 2020).

This work explores and compares two strategies for extracting tables contained in images in a structured manner: (a) a deep learning computer vision model, Table Transformer (TATR), combined with Optical Character Recognition (OCR) and (b) the novel Multimodal Large Language Models (MLLMs). These approaches were evaluated using metrics that capture how well they extract the tables' structural and text content.

The remainder of this paper is structured as follows. Section 2 provides an overview of existing and related work. Section 3 outlines the followed methodology and experiment details and Section 4 discusses the obtained results. Finally, conclusions and limitations are drawn in Sections 5 and 6.

2 Related Work

This section reviews key literature on Optical Character Recognition (OCR) and table extraction, and LLMs.

2.1 Table Extraction & OCR

OCR is fundamental in extracting text from tables within images (Li et al., 2024). Traditional OCR methods, including Tesseract (Smith, 2007) and Paddle-OCR (Du et al., 2020), follow a two-step process of text detection and recognition but often struggle with extracting complex table structural layouts due to diverse fonts and layouts (Ranjan et al., 2021; Zhong et al., 2020).

Recent developments in OCR technology have introduced bounding box detection, significantly improving word localization and integration with table structure recognition (Smock et al., 2023). Models such as TableNet (Paliwal et al., 2019), which utilize features for segmenting table regions, and Microsoft's TATR Transformer-based models (Smock et al., 2021), which perform end-to-end table detection and structural layouts recognition,

Proceedings of the 1st Joint Workshop on Large Language Models and Structure Modeling (XLLM 2025), pages 8–15
August 1, 2025

have shown promising results. Challenges like OCR errors, computational costs, and handling intricate structures like merged cells remain despite advancements.

2.2 Multimodal LLMs for Table Extraction

Multimodal LLMs can accomplish a wide range of tabular tasks (Zheng et al., 2024). These models can bypass OCR for table extraction, providing more efficient and accurate table extraction (Sui et al., 2024). Models such as LLaVA (Liu et al., 2023) and GPT-4o (Yenduri et al., 2023) can incorporate image and text processing, leveraging their capabilities for improved table recognition. Current research investigates representations and prompting strategies like chain-of-thought to evaluate the table's structural understanding capabilities of LLMs (Deng et al., 2024; Sui et al., 2024).

GPT-4 Omni (GPT-4o) (Yenduri et al., 2023) was launched in May of 2024 by OpenAI. It introduced several significant innovations as a foundation model, dwarfing the other models. It has a massive number of parameters — estimated to be well over 1 trillion - compared to GPT-3, at 175 billion parameters, and GPT-1, at an estimated 117 million parameters (Shahriar et al., 2024). It can process text, audio, and images at considerable speeds, which grants it remarkable multimodal capabilities. It was pre-trained using data up to October 2023, including data from public datasets and private partnerships.

Table LLaVA (Zheng et al., 2024; Liu et al., 2023) is a LLaVA model fine-tuned on the MMTab (Zheng et al., 2024) dataset. This enables it to do table-based question answering and data interpretation tasks. Regarding its limitations, Table LLaVA focuses mainly on single tables in English, and the resolution of input images is relatively low. MiniCPM-V (Yao et al., 2024) has strong image capabilities, supporting up to 1.8M pixels (high-resolution image perception) and robust OCR. It has multilingual support, covering over 30 languages. Phi-3-Vision (Microsoft, 2024) was trained on a diverse multimodal instruction tuning dataset encompassing 500 billion tokens. The Phi was trained primarily on English text. Languages other than English will experience worse performance. The resolution of input images is relatively low, similar to Table LLaVa. In multiple vision-language benchmarks, it surpasses previous models. In most benchmarks, Granite Vision 3.2 (GraniteVision, 2025) outperforms Phi-3-Vision. This model was trained on a curated dataset comprising approximately 13 million images and 80 million instructions from public and synthetic datasets. Granite Vision 3.2 is a streamlined and effective vision-language model tailored for comprehending visual documents. It facilitates the automated extraction of information from tables, charts, infographics, plots, and diagrams. The resolution of input images is medium, greater than Table LLaVa and Phi-3Vision.

Challenges persist, including accurately interpreting visual data, understanding complex table formats, and designing practical input and prompting strategies (Sui et al., 2024). Models must efficiently handle table serialization and adapt to various representation formats, ensuring accurate extraction and reasoning.

2.3 Datasets

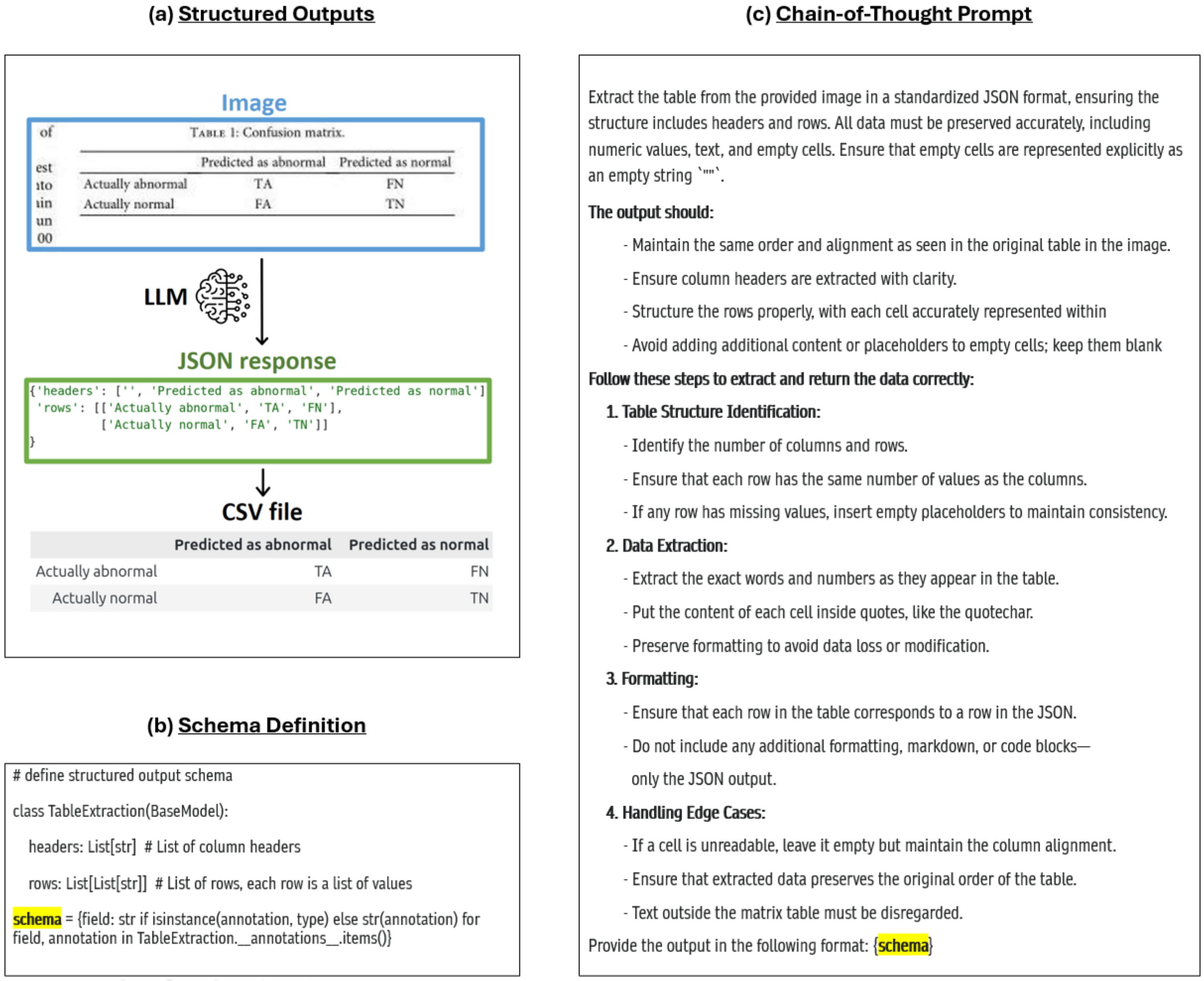
Several datasets with images of tables exist, including SciTSR (Chi et al., 2019), TableBank (Li et al., 2019), and PubTabNet (Zhong et al., 2020). With nearly one million tables, PubTables-1M (Smock et al., 2021) stands out due to its extensive scale and detailed annotations, making it the most recent and complete dataset. PubTables-1M's rich annotations, including spatial coordinates and OCR ground truth, enable models to learn how to recognize tables' structural and textual aspects. In the present study, the data used in the experiments is a subset of the PubTables-1M dataset (Smock et al., 2021).

3 Methodology

This section first explains the methodology used to retrieve table outputs from large language models. Then, in the second subsection, the evaluation metrics used are briefly detailed. Finally, in the third subsection, the specific details of the experiment's execution are provided.

3.1 Prompting LLMs for Tables

Evaluating the TATR model on the PubTables-1M dataset is straightforward, as the ground truth and the model's output prediction are essentially in the same format. In contrast, submitting tables to a large language model expecting structured output introduces additional challenges, such as ensuring proper formatting and dealing with potential response inconsistencies due to LLMs' generative nature.



(a) Structured Outputs

Table 1: Confusion matrix.

	Predicted as abnormal	Predicted as normal
Actually abnormal	TA	FN
Actually normal	FA	TN

```
{'headers': ['', 'Predicted as abnormal', 'Predicted as normal']
 'rows': [['Actually abnormal', 'TA', 'FN'],
          ['Actually normal', 'FA', 'TN']]
}
```

	Predicted as abnormal	Predicted as normal
Actually abnormal	TA	FN
Actually normal	FA	TN

(b) Schema Definition

```
# define structured output schema
class TableExtraction(BaseModel):
    headers: List[str]  # List of column headers
    rows: List[List[str]]  # List of rows, each row is a list of values
schema = {field: str if isinstance(annotation, type) else str(annotation) for
field, annotation in TableExtraction.__annotations__.items()}
```

(c) Chain-of-Thought Prompt

Extract the table from the provided image in a standardized JSON format, ensuring the structure includes headers and rows. All data must be preserved accurately, including numeric values, text, and empty cells. Ensure that empty cells are represented explicitly as an empty string `""`.

The output should:

- Maintain the same order and alignment as seen in the original table in the image.
- Ensure column headers are extracted with clarity.
- Structure the rows properly, with each cell accurately represented within
- Avoid adding additional content or placeholders to empty cells; keep them blank

Follow these steps to extract and return the data correctly:

1. Table Structure Identification:

- Identify the number of columns and rows.
- Ensure that each row has the same number of values as the columns.
- If any row has missing values, insert empty placeholders to maintain consistency.

2. Data Extraction:

- Extract the exact words and numbers as they appear in the table.
- Put the content of each cell inside quotes, like the quotechar.
- Preserve formatting to avoid data loss or modification.

3. Formatting:

- Ensure that each row in the table corresponds to a row in the JSON.
- Do not include any additional formatting, markdown, or code blocks—only the JSON output.

4. Handling Edge Cases:

- If a cell is unreadable, leave it empty but maintain the column alignment.
- Ensure that extracted data preserves the original order of the table.
- Text outside the matrix table must be disregarded.

Provide the output in the following format: {schema}

Figure 1: (a) Example of table extraction with LLM structured output. The LLM converts the input image into a structured JSON response, extracting the table attributes - headers and rows - whilst ignoring content outside the table. This JSON is then converted into a comma-separated values (CSV) file for evaluation. (b) Structured output schema definition. (c) The chain-of-thought prompt is used with the structured output technique.

An initial approach to extracting table information involved prompting the models to produce an output in a comma-separated values (CSV) format. This method was primarily effective for GPT models, with performance varying based on the prompts used; incorporating chain-of-thought instructions generally enhanced the outcomes. Alternatively, the Markdown format was tested for table extraction. However, the absence of a standardized Markdown structure across different models made it challenging to evaluate and compare outputs consistently.

Ultimately, OpenAI's Structured Output functionality was implemented alongside the chain-of-thought instructions prompt (see Figure 1(c)), ensuring compliance with a predefined JSON schema (Figure 1(b). This approach established a standardized format across all model outputs, facilitating a more straightforward structural layout and cell content evaluation. Figure 1(a) illustrates the transformations applied to the tabular data and the chain-of-thought prompt.

3.2 Evaluation

Several evaluation metrics were implemented to evaluate the detection of the table's structural layouts and the content present in its cells. The structural layout metrics aim to classify the model's ability to detect and preserve the table's organization, including its headers, rows, and columns. The content metrics are based on string similarity, which evaluates the accuracy and relevance of the extracted information within the table cells.

Table shape accuracy evaluates how closely predicted table dimensions align with the actual ones, calculated as the harmonic mean of row and column accuracy (Eq. 1). Significant inaccuracies in

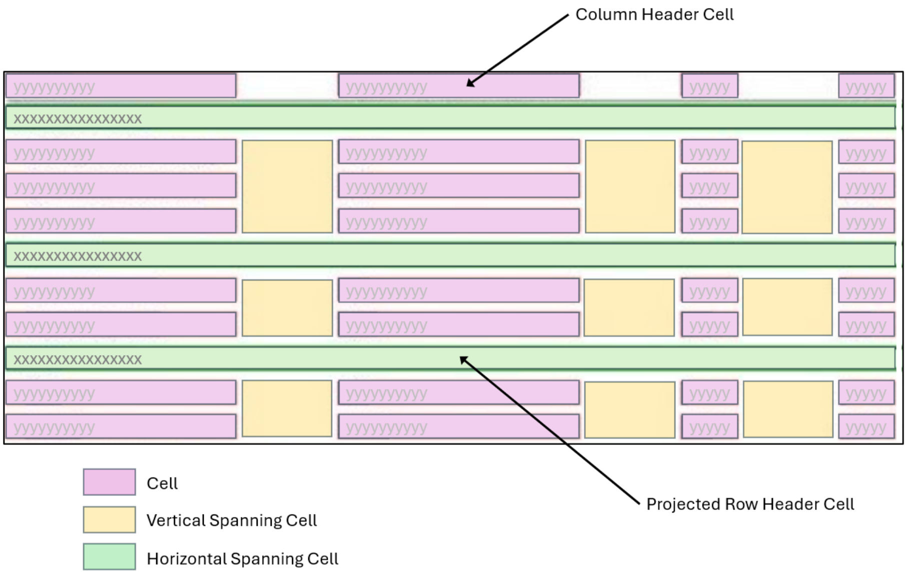


Figure 2: Example of projected row header cell and spanning cell, inspired by (Smock et al., 2021)

either dimension can significantly impact overall accuracy.

$$\text{Shape Acc} = \frac{2}{\frac{1}{\text{Row Acc}} + \frac{1}{\text{Column Acc}}} \quad (1)$$

A vital metric utilized in this evaluation is the F1 Score, which balances Precision and Recall to gauge the accuracy of the extracted tables. The F1 Score represents a harmonic mean of Precision and Recall, ensuring that a model's effectiveness in extracting table data considers both correctness and completeness.

Precision measures how many of the extracted cells are accurate, in comparison to the total number of cells that were extracted.

Recall assesses how many ground truth cells were accurately matched, guaranteeing a high retrieval rate.

The F1 Score (Eq. 2) integrates both metrics to provide a comprehensive evaluation of the model's capability to correctly extract tables:

$$F_1 = \frac{2 \cdot P \cdot R}{P + R} \quad (2)$$

,where P is precision and R is recall.

Various thresholds were utilized to determine the similarity between predicted and ground truth cell content to evaluate text-based table extraction precisely.

- Threshold = 0 → Assesses structural layout accuracy only, disregarding text content(Structural Layout F1 Score).
- Thresholds = 75, 85, 95, 100 → Evaluate content similarity by using fuzzy matching on the strings (text), requiring progressively higher levels of textual accuracy (Cell content F1 Score).

For example, a threshold of 75 permits minor text variations (e.g., typing errors), while a threshold of 100 demands an exact correspondence between the predicted and the ground truth content.

On the other hand, Grid Table Similarity (GriTS) (Smock et al., 2022) evaluates tables in their matrix form, accommodating topology, content, and positioning within a unified framework. GriTS operates by first computing the longest common subsequence (LCS) between the ground truth and predicted sequences. This step identifies which items are missing in the truth sequence and which are extra in the prediction, allowing for calculating precision, recall, and, ultimately, the F1 score based on these discrepancies.

All metrics offer valuable insights, with GriTS potentially providing more comprehensive results due to its holistic assessment capabilities.

3.3 Experiment Details

PubTables-1M (Smock et al., 2021) has 94,000 samples of table images, of which 44,000 are classified as non-complex - they do not present spanning cells or projected rows. Spanning cells are merged cells that span horizontally or vertically from multiple cells. At the same time, projected rows usually subdivide tables encompassing situations where a specific row is not aligned with the other rows in the table, acting as a subtitle (Smock et al., 2021; Xiao et al., 2025) as shown in Figure 2.

Non-complex tables were selected because they can effectively be represented as CSV files for structural layout comparison. A final selection of 1,000 samples ensured statistical compatibility with the partial non-complex dataset. This subset size was chosen considering the execution time and costs required for processing images with local and cloud-hosted LLMs.

Five different models were used in the experiments: Granite Vision 3.2 (GraniteVision, 2025), Phi-3-Vision (Microsoft, 2024), GPT-4o (OpenAI, 2024), GPT-4o-mini (OpenAI, 2024), and TATR-OCR (Smock et al., 2023; Du et al., 2020). Granite & other (Microsoft, 2024) was executed on a system equipped with an NVIDIA V100 GPU with 32 GB of memory. Table 1 presents the models' parameters and availability.

Model	Parameters	Availability
Granite Vision 3.2	2.8 Billion	Free (Open Source)
Phi-3-Vision	3.8 Billion (approx.)	Free (Open Source)
GPT-4o	1.8 Trillion (estimated)	Paid
GPT-4o-mini	8 Billion (approx.)	Paid
TATR	28 Million (approx.)	Free (Open Source)

Table 1: Models parameters and availability.

As shown in Figure 1, an advanced prompting technique was implemented to ensure structured and interpretable outputs from LLMs. This approach guided the models to generate structured responses, facilitating consistent evaluation across different architectures (LLMs vs. TATR-OCR). The performance of the models was measured using well-defined evaluation metrics, ensuring an objective comparison.

4 Results

The results of the evaluation comparison, depicted in Figure 3, provide a comprehensive overview of the performance of the five models. These models are TATR-OCR, Granite, Phi-3-Vision, GPT-4o-mini, and the standout performer GPT-4o. The analysis begins in Figure 3(a) with the GriTS F1 Score, where GPT-4o stands out with an impressive 89.6%, closely followed by TATR-OCR at 87.8%. GPT-4o-mini and Granite yield more moderate scores at 74.6% and 76.3%, respectively, while Phi-3-Vision records a relatively lower score of 65.2%.

The Structural Layout F1 Score in Figure 3(a) further differentiates the models, with TATR-OCR achieving a remarkable 98.2% and GPT-4o also performing strongly at 94.9%. Granite and GPT-4o-mini demonstrate similar mid-tier performance levels at 81.5% and 81.9%, respectively, and Phi-3-Vision lay behind at 70.8%. In terms of Table Shape Accuracy, the pattern is similar: GPT-4o and TATR-OCR excel with accuracies of 95.2% and 98.4%, respectively, while Granite and GPT-4o-mini maintain comparable scores of 82.3% and 82.6%. Phi-3-Vision again underperforms with only 71.3% accuracy.

Beyond these overall metrics, the analysis delves into structural layout errors in Figure 3(b). These errors refer to discrepancies in the arrangement of elements within the document, examining both missing and extra rows and columns. GPT-4o performs the best, with only 0.3% missing rows. TATR-OCR, Phi-3-Vision, and GPT-4o-mini show moderate performance, missing around 2.7-3.2% of rows. Granite is the least reliable, missing 7.9% of table rows, which could lead to major data loss. For the extra rows, TATR-OCR and Granite are tied as best performance with a percentage of 0.7%. Followed by Phi-3-Vision with almost 2%, the worst performers are GPT-4o with 5.2% and GPT-4o-mini with 8.2%.

Looking at the missing columns GPT-4o and TATR-OCR both have a 0.3%. Phi-3-Vision and GPT-4o-mini are in mid-performance 1-1.5%, respectively. The worst is Granite with 4.2% missing columns. Regarding the extra columns percentage Granite is again the worst with almost 15%, Phi-3-Vision and GPT-4o-mini are in mid-performance 4.7 - 5.4%. TATR-OCR is close to GPT-4o with 0%.

Overall, the analysis of structural layout errors highlights significant differences in performance among the models. GPT-4o consistently demonstrates the best overall accuracy, only worse by a high percentage of extra rows. TATR-OCR also performs well, particularly in handling columns. Phi-3-Vision and GPT-4o-mini exhibit moderate performance, showing errors in missing and extra elements. However, Granite proves to be the least reliable, with the highest percentage of missing rows and columns and a substantial number of extra columns. These discrepancies in structural layout accuracy can significantly impact data integrity, reinforcing the importance of selecting the most precise model for document processing tasks.

The analysis of string matching thresholds in Figure 3(c) reveals distinct performance variations

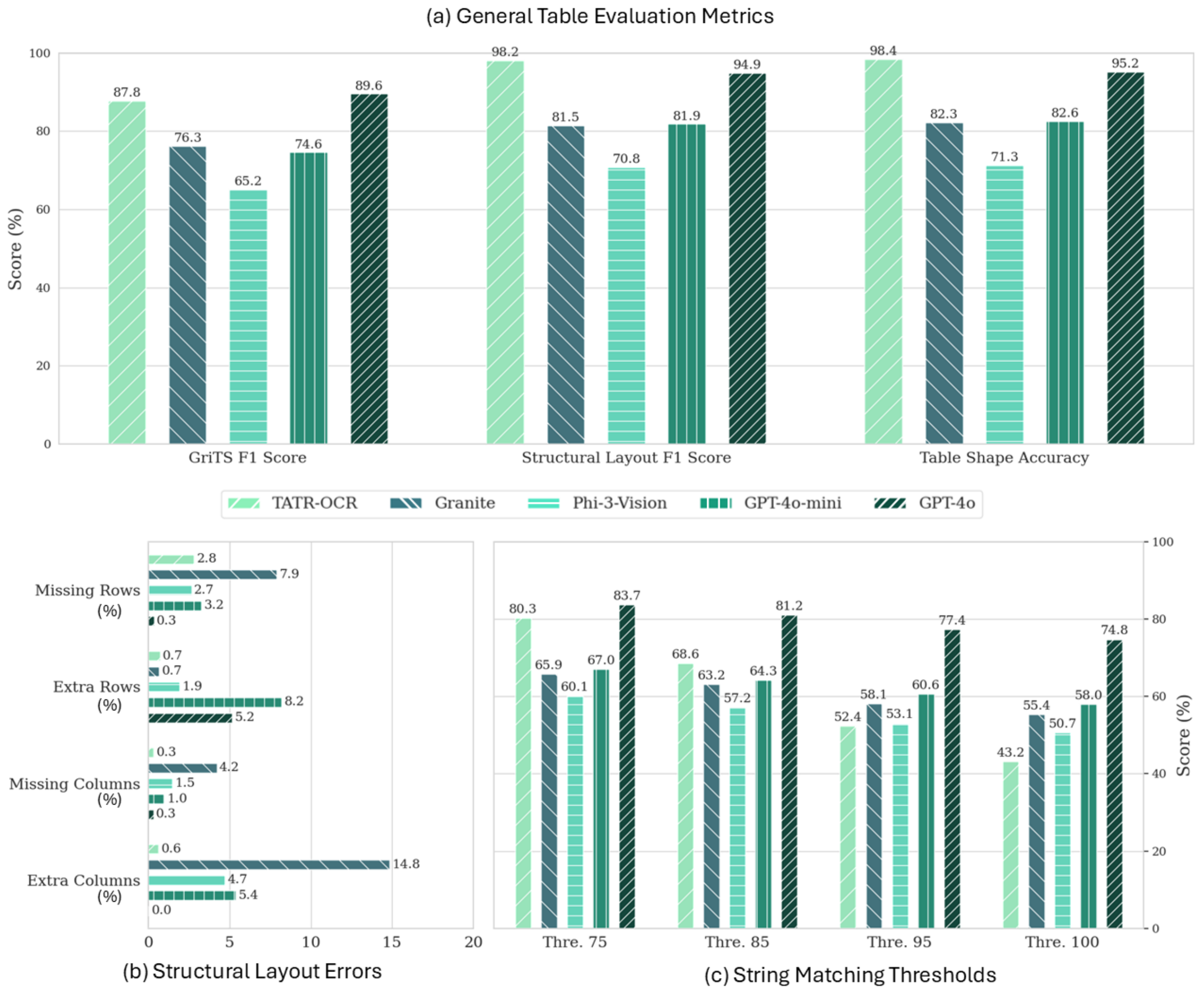


Figure 3: (a) General Table Evaluation Metrics: GriTS F1 Score - table similarity based on structural layout and cell text content, Structural Layout F1 Score - evaluate structural cell positions, and Table Shape Accuracy - evaluate the harmonic mean of the table's rows and columns accuracy. (b) Structural Layout Errors: percentage of missing/extra rows and columns. (c) String Matching Thresholds: Cell content F1 Score of the cell position and content with different thresholds for cell string match.

among the five models. These thresholds are crucial as they determine the level of similarity required for a match, with a higher threshold indicating a stricter matching condition. GPT-4o consistently achieves the highest scores across all thresholds, demonstrating superior accuracy in string matching. TATR-OCR follows closely behind, maintaining a competitive performance. Phi-3-Vision and GPT-4o-mini exhibit moderate results, while Granite consistently underperforms, scoring the lowest in most cases. As the matching threshold increases from 75 to 100, all models experience a decline in performance, indicating that stricter criteria lead to greater difficulty in identifying matches. Despite this trend, GPT-4o remains the most reliable, maintaining high accuracy even at the strictest threshold. These findings highlight the varying robustness of different models and emphasize the importance of selecting the most suitable one based on the required level of precision.

In conclusion, the analysis underscores the importance of selecting the most suitable model for the task at hand. GPT-4o consistently outperforms the other models across all key metrics, exhibiting superior structural accuracy and text recognition capabilities. GPT-4o-mini, on the other hand, emerges as a strong alternative, showcasing its potential with slightly lower performance. TATR-OCR and Phi-3-Vision deliver similar mid-range results, while Granite shows the least favorable performance with significant structural errors and lower text accuracy. These findings suggest that GPT-4o and GPT-4o-mini are the most reliable choices for table extraction tasks, whereas the other models may benefit from additional post-processing steps to improve their accuracy.

5 Conclusions

TATR outperforms the LLMs when only the table structural layout is considered based on the metric results. If the exact content of the cell is not a top priority, using TATR with OCR is a viable choice, keeping in mind that the OCR may not correctly identify all the cells. However, when the text content of the cells is considered, the LLMs perform better than TATR with OCR. GPT-4o is by far the best among the LLMs tested, but is also the largest and most expensive model. Smaller models can be a good option depending on the specific use case and the volume of data to be processed.

6 Limitations

The primary limitation of this study is that it does not use complex tables from the original dataset in the experiments, particularly those with spanning cells and projected rows, because of the difficulty in representing them as a matrix. This constraint affects the generalization of the findings to more intricate table layouts. Additionally, inconsistencies were observed in the responses generated by MLLMs, despite employing structured output formats. This impacted metrics negatively, as conversion to JSON/CSV was not achievable in such cases.

To address these limitations, future work could explore advanced prompting techniques to mitigate inconsistencies in LLM outputs. One-shot and few-shot learning, fine-tuning, and reflection-based approaches (e.g., Haystack framework (Pietsch et al., 2019)) can improve output consistency and reliability. Also, alternative structured output (to include more complex table structures, especially with spanning cells) and prompting strategies (e.g., a two-step process with a preliminary table generation followed by parsing) could be investigated, and performing a sensitivity analysis of the model's decoding parameters should be considered. Finally, examining whether improvements in prompting (like chain-of-thought or step-by-step reasoning) might further enhance structured outputs could also provide meaningful insights.

While TATR was chosen as the OCR option for this study, exploring additional traditional methods or hybrid systems (combining OCR with LLM strategies) could yield a more comprehensive comparison. A discussion on computational cost versus performance can also be raised, especially noting that while models like GPT-4o have trillions of parameters, simpler models like TATR operate at a much lower price.

Furthermore, future work must consider complex tables, but also incorporate more diverse "other" tabular data sources, such as SciTSR (Chi et al., 2019), TableBank (Li et al., 2019), and PubTabNet (Zhong et al., 2020), which would provide a more comprehensive evaluation of table extraction performance across different domains and table complexities.

Finally, experimenting with new LLMs specifically fine-tuned for table extraction - such as Gemini 2.5 pro (Gemini-Team, 2025), Table LLaVA (Zheng et al., 2024), Mistral OCR (et. al., 2023), and MiniCPM-V (Yao et al., 2024) - could also improve table structural layout recognition and content extraction accuracy. A more detailed qualitative analysis of errors could pinpoint where each model fails, thereby offering insights for further improvements. These approaches would contribute to a more robust and adaptable table extraction framework.

References

Douglas Burdick, Marina Danilevsky, Alexandre V Evfimievski, Yannis Katsis, and Nancy Wang. 2020. Table extraction and understanding for scientific and enterprise applications. *Proc. VLDB Endow.*, 13(12):3433–3436.

Leiyuan Chen, Chengsong Huang, Xiaoqing Zheng, Jinshu Lin, and Xuanjing Huang. 2023. TableVLM: Multi-modal pre-training for table structure recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2437–2449, Toronto, Canada. Association for Computational Linguistics.

Zewen Chi, Heyan Huang, Heng-Da Xu, Houjin Yu, Wanxuan Yin, and Xian-Ling Mao. 2019. Complicated table structure recognition.

Naihao Deng, Zhenjie Sun, Ruiqi He, Aman Sikka, Yulong Chen, Lin Ma, Yue Zhang, and Rada Mihalcea. 2024. Tables as texts or images: Evaluating the table reasoning ability of LLMs and MLLMs. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 407–426, Bangkok, Thailand. Association for Computational Linguistics.

Yuning Du, Chenxia Li, Ruoyu Guo, Xiaoting Yin, Weiwei Liu, Jun Zhou, Yifan Bai, Zilin Yu, Yehua Yang, Qingqing Dang, and Haoshuang Wang. 2020. Pp-ocr: A practical ultra lightweight ocr system.

Albert Q. Jiang et. al. 2023. Mistral 7b.

Gemini-Team. 2025. Gemini: A family of highly capable multimodal models.

Team GraniteVision. 2025. Granite vision: a lightweight, open-source multimodal model for enterprise intelligence.

Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, Ming Zhou, and Zhoujun Li. 2019. Tablebank: A benchmark dataset for table detection and recognition.

Yiming Li, Qiang Wei, Xinghan Chen, Jianfu Li, Cui Tao, and Hua Xu. 2024. Improving tabular data extraction in scanned laboratory reports using deep learning models. *Journal of Biomedical Informatics*, 159.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning.

Microsoft. 2024. Phi-3 technical report: A highly capable language model locally on your phone.

OpenAI. 2024. Gpt-4o system card. `https://arxiv.org/abs/2410.21276`. Accessed: 2025-03-30.

Shubham Singh Paliwal, D. Vishwanath, Rohit Rahul, Monika Sharma, and Lovekesh Vig. 2019. Tablenet: Deep learning model for end-to-end table detection and tabular data extraction from scanned document images. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, pages 128–133.

Malte Pietsch, Timo Möller, Bogdan Kostic, Julian Risch, Massimiliano Pippi, Mayank Jobanputra, Sara Zanzottera, Silvano Cerza, Vladimir Blagojevic, Thomas Stadelmann, Tanay Soni, and Sebastian Lee. 2019. Haystack: the end-to-end nlp framework for pragmatic builders.

Ashish Ranjan, Varun Nagesh Jolly Behera, and Motahar Reza. 2021. *OCR Using Computer Vision and Machine Learning*, pages 83–105. Springer International Publishing, Cham.

Sakib Shahriar, Brady D. Lund, Nishith Reddy Mannuru, Muhammad Arbab Arshad, Kadhim Hayawi, Ravi Varma Kumar Bevara, Aashrith Mannuru, and Laiba Batool. 2024. Putting gpt-4o to the sword: A comprehensive evaluation of language, vision, speech, and multimodal proficiency. *Applied Sciences*, 14(17).

Ray Smith. 2007. An overview of the tesseract ocr engine. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2:629–633.

Brandon Smock, Rohith Pesala, and Robin Abraham. 2021. Pubtables-1m: Towards comprehensive table extraction from unstructured documents. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2022-June:4624–4632.

Brandon Smock, Rohith Pesala, and Robin Abraham. 2022. Grits: Grid table similarity metric for table structure recognition. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 14191 LNCS:535–549.

Brandon Smock, Rohith Pesala, and Robin Abraham. 2023. Aligning benchmark datasets for table structure recognition. In *Document Analysis and Recognition - ICDAR 2023*, pages 371–386, Cham. Springer Nature Switzerland.

Peter W J Staar, Michele Dolfi, Christoph Auer, and Costas Bekas. 2018. Corpus conversion service: A machine learning platform to ingest documents at scale. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '18. ACM.

Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. 2024. Table meets llm: Can large language models understand structured table data? a benchmark and empirical study. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, WSDM '24, page 645–654, New York, NY, USA. Association for Computing Machinery.

Bin Xiao, Murat Simsek, Burak Kantarci, and Ala Abu Alkheir. 2025. Rethinking detection based table structure recognition for visually rich document images. *Expert Systems with Applications*, 269:126461.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, Qianyu Chen, Huarong Zhou, Zhensheng Zou, Haoye Zhang, Shengding Hu, Zhi Zheng, Jie Zhou, Jie Cai, Xu Han, Guoyang Zeng, Dahai Li, Zhiyuan Liu, and Maosong Sun. 2024. Minicpm-v: A gpt-4v level mllm on your phone.

Gokul Yenduri, Ramalingam M, Chemmalar Selvi G, Supriya Y, Gautam Srivastava, Praveen Kumar Reddy Maddikunta, Deepti Raj G, Rutvij H Jhaveri, Prabadevi B, Weizheng Wang, Athanasios V. Vasilakos, and Thippa Reddy Gadekallu. 2023. Generative pre-trained transformer: A comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions.

Mingyu Zheng, Xinwei Feng, Qingyi Si, Qiaoqiao She, Zheng Lin, Wenbin Jiang, and Weiping Wang. 2024. Multimodal table understanding.

Xu Zhong, Elaheh ShafieiBavani, and Antonio Jimeno Yepes. 2020. Image-based table recognition: Data, model, and evaluation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12366 LNCS:564–580.