Benchmarking PDF Extraction Tools: A Comprehensive Study

Date: Dec 19, 2024 Version: 1.0

Abstract

This study evaluates multiple PDF extraction tools to identify the most suitable solution for document processing needs. By comparing tools across diverse PDF types and performance metrics, we provide a detailed analysis of their capabilities, strengths, and limitations. The benchmarking process highlights key metrics such as text, table, and image extraction accuracy, as well as OCR performance, markdown conversion, and logical reading order preservation.

1. Introduction

PDF extraction tools are pivotal in automating document processing workflows, forming the backbone of modern AI applications such as Retrieval-Augmented Generation (RAG), Generative AI, and intelligent agents. These systems thrive on the accurate extraction of structured and unstructured content, enabling efficient information processing and generation. For instance, RAG systems depend on clean, well-structured content to retrieve and augment responses, while intelligent agents in automation workflows require reliable data extraction to execute logical operations effectively.

Given the diverse use cases for PDF extraction—spanning academic research, financial reporting, and beyond—it is essential to evaluate these tools across varying content types and real-world scenarios. This study benchmarks seven leading PDF extraction tools, systematically analyzing their performance in content extraction and deployment. By highlighting strengths and weaknesses, this research provides actionable insights for selecting the right tool for specific applications, advancing both AI-driven workflows and automation capabilities.

Disclaimer
The views and feedback shared in this article are based on internal testing and evaluations conducted by Actualize's engineering team. This study does not intend to criticize, guarantee ownership, or take any responsibility for the performance or effectiveness of the tools discussed. Our aim is to transparently share the findings from our testing process without bias, providing insights for informational purposes only.

2. Objectives

The primary objective of this study is to evaluate and compare PDF extraction tools to determine their suitability for specific use cases. The key evaluation criteria include:

- Text extraction accuracy
- Table extraction accuracy
- Image extraction clarity and positioning
- Markdown conversion fidelity
- OCR capabilities for scanned documents
- Logical reading order preservation
- Resource utilization across CPU, MPS, and GPU platforms

3. Evaluation Framework

3.1 Content Diversity

The benchmarking process covers various document types:

- Academic papers
- Financial reports
- FAQs
- Catalogs
- Medical records
- Regional language documents
- Scanned documents

Each document type poses unique challenges, such as handling complex table structures, extracting code snippets and equations, or performing accurate OCR on non-English text.

3.2 Benchmarking Basis

Academic Paper Benchmarking

The paper used for benchmarking is an academic paper focused on retrieval-augmented generation (RAG). It comprises diverse elements such as tables, images, mathematical equations, code snippets, and formatted text. Accurate extraction of these elements is essential for downstream AI applications like document understanding and automated knowledge augmentation.

Performance for the academic paper was analyzed using two evaluation approaches:

- **LLM as Judge**: Scoring was standardized with evaluations performed using GPT-4o and Claude 3.5 Sonnet.
- **Human Evaluation**: Independent human evaluators cross-validated the results for accuracy and quality.

Other PDF Types Benchmarking

A variety of PDF types were evaluated, including catalogs, FAQs, financial reports, regional language documents, medical records, and scanned documents. The benchmarking for these document types focused on specific content needs, such as structured text, tables, images, OCR performance, and Markdown formatting. To ensure an objective evaluation, the benchmarking process relied on GPT-4o (a large language model) as the primary judge, providing standardized scoring across metrics like text extraction accuracy, table extraction, reading order preservation, and Markdown conversion.

3.3 Tool Selection

The evaluated tools include:

- **MinerU**
- **Xerox**
- **Docling**
- **Llama Parse**
- **Marker**
- **Unstructured**
- **Markitdown**

Tools were selected based on their deployment modes (local vs. API-based), feature sets, and reported performance in prior studies.

Deployment Modes of PDF Extraction Tools

Locally Run Tools

- **Docling, MinerU, Marker, and Markitdown**: These tools operate entirely on local systems, requiring no API connectivity. They are ideal for environments where offline processing is a priority or internet access is limited.
- **Unstructured**: This tool has a flexible deployment mode and can operate both locally and as an API. In this evaluation, Unstructured was tested in local mode.

API-Based Tools

- **Xerox**: Requires an API key for operation and is fully dependent on API calls, making internet connectivity a critical requirement.

- **Llama Parser**: Similar to Xerox, it requires an API key and relies on external API servers for processing PDFs.

3.4 Performance Metrics

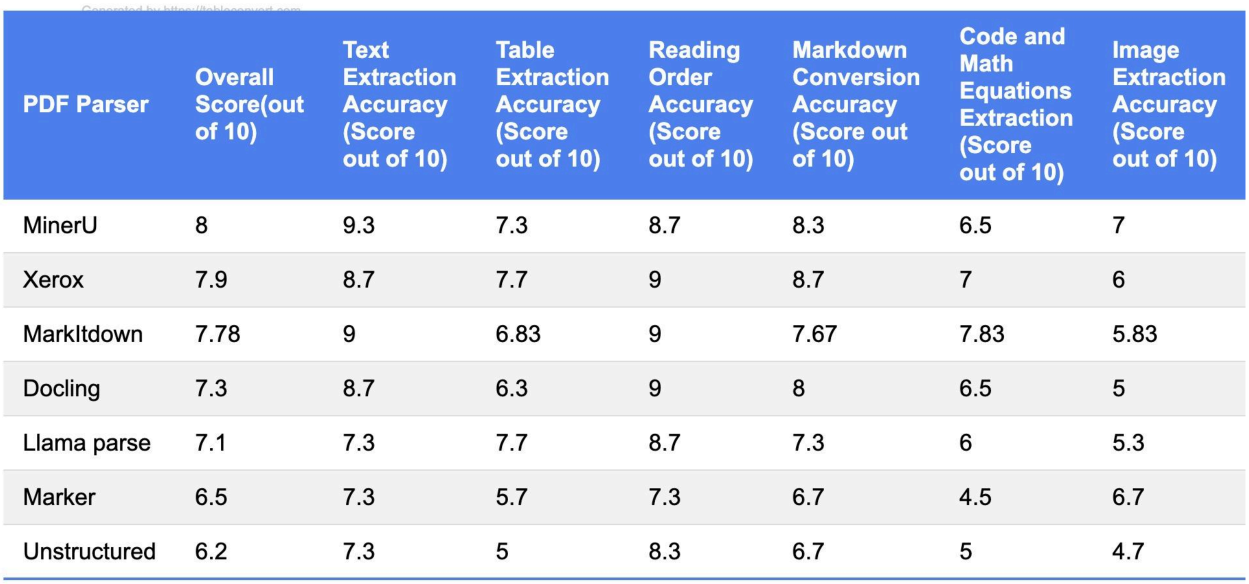
Metrics were standardized to ensure consistency:

- **Text Extraction Accuracy**: Evaluates how accurately the library extracts plain text from PDFs, preserving the original content without errors or distortions.
- **Performance**: Measures the speed of the extraction process and the tool's resource utilization on both CPU and MPS systems. Includes considerations such as processing time, system usage, and scalability.
- **Table Extraction Accuracy**: Assesses the ability to correctly extract tables, including their structure, alignment, and formatting. Accuracy in handling complex and nested tables is also evaluated.
- **Reading Order Accuracy**: Checks whether the tool preserves the logical sequence of content, ensuring the extracted text maintains the flow of the original PDF.
- **Markdown Conversion Accuracy**: Tests the tool's capability to convert PDF content into Markdown format. Includes evaluation of formatting elements such as titles, bullet points, lists, and other layout features.
- **Code and Math Equations Extraction**: Determines the effectiveness of the tool in extracting code snippets and mathematical equations from PDFs. Focuses on maintaining syntax, structure, and readability.
- **Image Extraction**: Evaluates the ability to extract images with clarity and accuracy in positioning within the document structure.
- **OCR Capabilities and Accuracy**: For tools with OCR functionality, assesses their ability to extract text from scanned PDFs. Includes metrics for OCR accuracy and performance in image-to-text conversion.
- **Resource Utilization**: Measured only for tools run locally (Docling, MinerU, Marker, and Unstructured in local mode) on CPU and MPS systems. GPU resource utilization tests (using A40 GPUs) were performed on Docling, MinerU, and Marker. Unstructured does not support GPU acceleration.

4. Results and Analysis

4.1 Overall Tool Performance

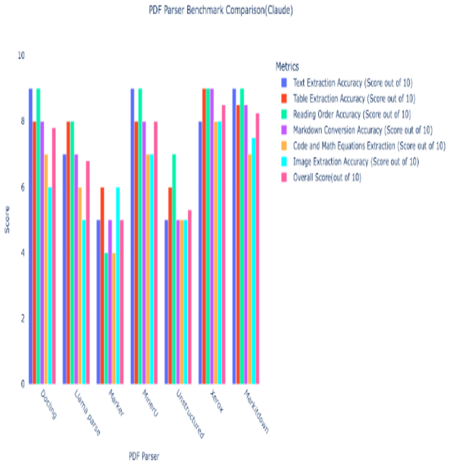
Overall tool performance was assessed by taking every performance metric aspect as a whole, including resource utilization. Each tool's performance across text extraction, table extraction, reading order accuracy, markdown conversion, code and math equations extraction, image extraction, and OCR capabilities contributed to its final score.

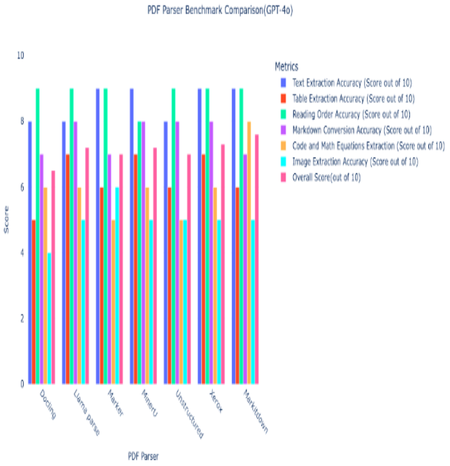


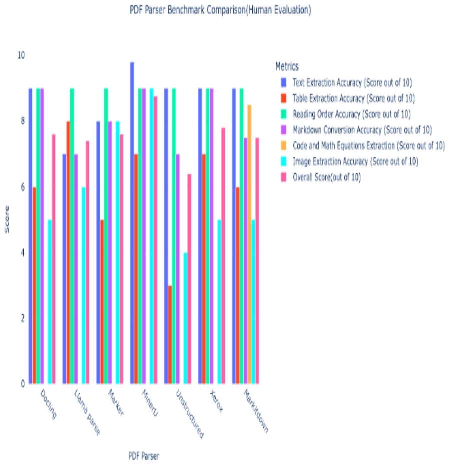
PDF Parser	Overall Score(out of 10)	Text Extraction Accuracy (Score out of 10)	Table Extraction Accuracy (Score out of 10)	Reading Order Accuracy (Score out of 10)	Markdown Conversion Accuracy (Score out of 10)	Code and Math Equations Extraction (Score out of 10)	Image Extraction Accuracy (Score out of 10)
MinerU	8	9.3	7.3	8.7	8.3	6.5	7
Xerox	7.9	8.7	7.7	9	8.7	7	6
MarkItdown	7.78	9	6.83	9	7.67	7.83	5.83
Docling	7.3	8.7	6.3	9	8	6.5	5
Llama parse	7.1	7.3	7.7	8.7	7.3	6	5.3
Marker	6.5	7.3	5.7	7.3	6.7	4.5	6.7
Unstructured	6.2	7.3	5	8.3	6.7	5	4.7

4.2 Academic Paper Benchmark Results

Results for academic paper benchmarking across GPT-4o, Claude 3.5 Sonnet, and human evaluation are presented below:

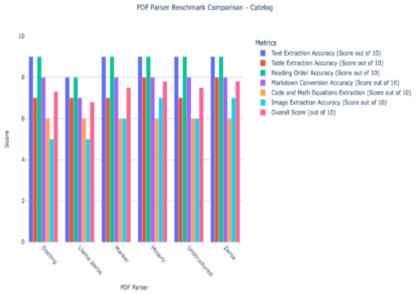


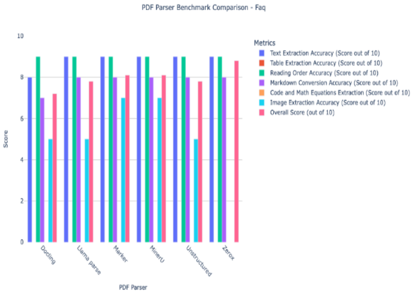


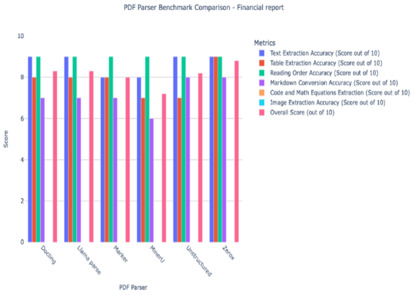


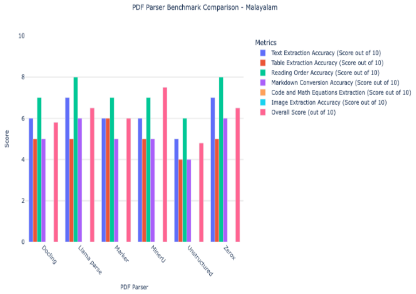
4.3 Performance on Other Document Types

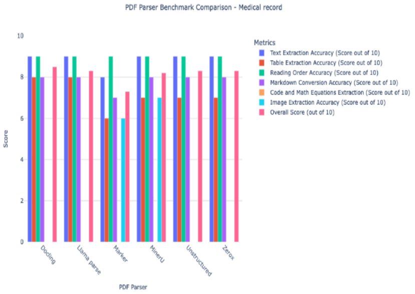
The tools were evaluated on catalogs, FAQs, financial reports, regional language documents, medical records, and scanned documents.

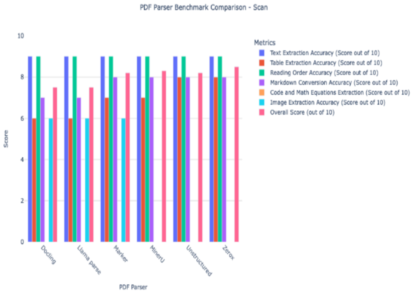






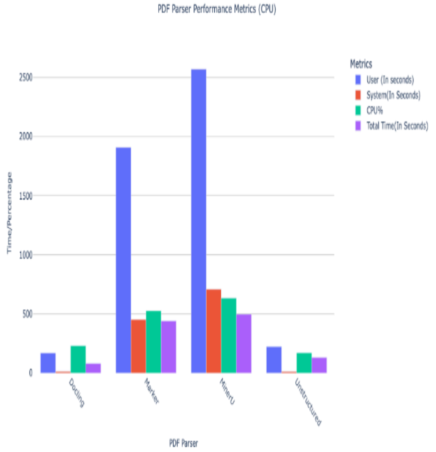


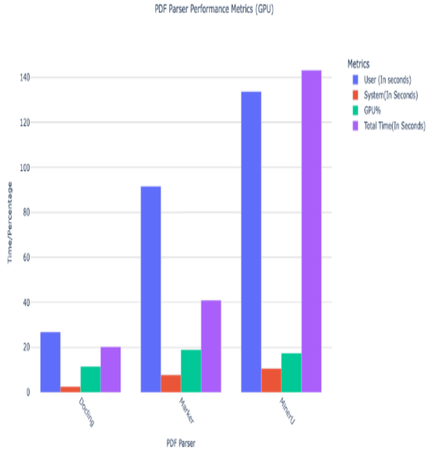


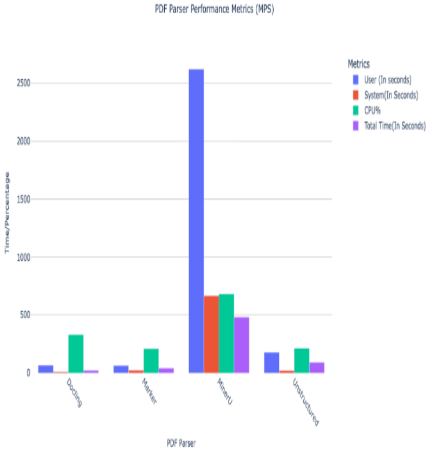


4.4 Resource Utilization Analysis

The tools' resource utilization was benchmarked on CPU, MPS, and GPU systems. The evaluation measured processing speed, memory consumption, and scalability for local tools (Docling, MinerU, Marker, and Unstructured). GPU tests were performed on MinerU, Docling, and Marker using A40 GPUs, while Unstructured does not currently support GPU acceleration.







5. Discussion

The evaluation demonstrates that accurate content extraction is essential for AI products, particularly for RAG systems, generative models, and intelligent agents. Tools like MinerU and Xerox excel in text and OCR extraction, ensuring high-quality input data for AI models. For table-intensive workflows, Llama Parse’s performance in extracting structured data is particularly noteworthy. While Markitdown demonstrated exceptional speed and markdown conversion accuracy, its table and image extraction results require further improvement. Future advancements in GPU support and resource optimization will further enhance these tools' deployment efficiency in AI workflows.

6. Conclusion and Future Work

MinerU emerged as the best all-rounder for content extraction, particularly excelling in text and Markdown conversion, which is critical for AI products like RAG and generative models. Markitdown exhibited the fastest PDF conversion performance but needs refinement in table and image extraction capabilities. Xerox's OCR performance makes it ideal for scanned documents and knowledge retrieval workflows. Moving forward, integration with advanced AI pipelines, resource optimization on GPUs, and enhancing table recognition capabilities will ensure PDF extraction tools align more effectively with AI-driven applications.

7. Read More

1. GitHub
2. Google Collab
3. Docling
4. Llama Parse
5. Marker
6. MinerU
7. Unstructured
8. Zerox
9. Markitdown