More Power, More Accuracy? An Energy-Efficiency Analysis of Intelligent Document Processing

Adam Grossenbacher
adam.grossenb@hes-so.ch
iCoSys, HEIA-FR, HES-SO University of Applied Sciences and Arts Western Switzerland
Fribourg, Switzerland

Loïc Guibert
iCoSys, HEIA-FR, HES-SO University of Applied Sciences and Arts Western Switzerland
Fribourg, Switzerland
loic.guibert@hes-so.ch

Beat Wolf
iCoSys, HEIA-FR, HES-SO University of Applied Sciences and Arts Western Switzerland
Fribourg, Switzerland
beat.wolf@hes-so.ch

Jean Hennebert
iCoSys, HEIA-FR, HES-SO University of Applied Sciences and Arts Western Switzerland
Fribourg, Switzerland
jean.henneber@hes-so.ch

Abstract

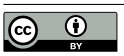
Document AI (or Intelligent Document Processing) is a rapidly evolving field widely used to parse complex documents, yet its computational and environmental costs remain largely unquantified. To address this gap, this paper empirically evaluates the trade-offs between accuracy and energy consumption across the three most representative categories of state-of-the-art open-source Document AI systems: pipeline tools, general-purpose Vision-Language Models (VLMs), and specialized VLMs. Using the reference benchmark OmniDocBench v1.5 and a controlled measurement protocol validated by physical wattmeters, we assess parsing performance alongside key sustainability metrics, including throughput, energy usage, and CO2eq emissions. Our findings reveal no strong correlation between parsing accuracy and energy consumption. Notably, specialized VLMs emerge as the most compelling category, with GLM-OCR achieving the best balance between accuracy and energy usage. In contrast, larger general-purpose models consume significantly more energy without necessarily delivering superior results. Moreover, pipeline tools demonstrate lower accuracy despite matching the energy footprint of specialized models. Finally, our extensive profiling shows that efficiency is heavily influenced by the hardware platform (CPU vs. GPU), the inference backend, and document characteristics, specifically length and complexity.

CCS Concepts

• **Applied computing** → **Document analysis**; • **Hardware** → **Power and energy**; • **General and reference** → **Evaluation**; • **Computing methodologies** → *Information extraction*; *Computer vision.*

Keywords

Intelligent Document Processing, Document AI, Vision-Language Models, Sustainable AI, Benchmarking



This work is licensed under a Creative Commons Attribution 4.0 International License.
DocEng '26, Fribourg, Switzerland
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2786-3/2026/08
https://doi.org/10.1145/3820755.3821481

ACM Reference Format:
Adam Grossenbacher, Loïc Guibert, Beat Wolf, and Jean Hennebert. 2026. More Power, More Accuracy? An Energy-Efficiency Analysis of Intelligent Document Processing. In *Proceedings of the 2026 ACM Symposium on Document Engineering (DocEng '26), August 25–28, 2026, Fribourg, Switzerland.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3820755.3821481

1 Introduction

The digital transformation is generating ever-increasing volumes of data [23, 53], with a significant proportion consisting of complex, unstructured documentary data [18]. The automatic processing, interpretation, and utilization of this information have become essential for organizations seeking to enhance operational efficiency and improve decision reliability [8, 18, 26]. Consequently, Document AI [28], also known as Intelligent Document Processing (IDP) [4], has emerged as a rapidly evolving field focused on enabling machines to parse diverse documents.

At the core of Document AI is document parsing, a crucial step designed to reliably extract structural and semantic information. This process aims to resolve persistent issues such as layout variability (e.g., blocks, columns, reading order), varying scan quality, and the accurate recognition of complex elements (e.g., text, tables, formulas, or graphs) [39, 65, 69]. By successfully converting complex documents into machine-readable formats, document parsing supports information retrieval systems and enhances the capabilities of Large Language Models (LLMs) [1, 67], particularly when integrated with Retrieval-Augmented Generation (RAG) pipelines [36].

Recent document parsing research predominantly follows three representative technological categories. The first approach relies on pipeline tools composed of specialized expert models [17, 19, 47, 61]. Although these methods offer competitive performance, they are constrained by integration complexity, cumulative error propagation, and difficulties in handling highly complex layouts. The second category uses end-to-end specialized Vision-Language Models (VLMs) [15, 20, 41, 43, 50, 51, 63]. While aiming to simplify the processing chain, these models often struggle to correctly restore the reading order and may generate hallucinations when faced with long or complex document structures [15, 40]. The third category uses general-purpose VLMs, designed for versatility across a wide

range of tasks [1, 13, 67]. While sharing several limitations with specialized VLMs, they tend to consume significantly more energy due to their general-purpose nature.

Despite the exponential growth of the Document AI market, a rapidly evolving field projected to exceed $9 billion by 2029 [62], its expansion is increasingly constrained by the computational demands of modern systems [54, 55]. The systematic adoption of large-scale deep learning models to achieve state-of-the-art accuracy raises significant sustainability concerns, as high computational requirements translate into increased energy consumption, financial costs, and environmental impact [55]. It is therefore imperative to evaluate the trade-offs between parsing accuracy and energy consumption, alongside economic and ecological sustainability [7]. Consequently, mitigating the energy footprint of Document AI systems is crucial not only for reducing environmental impact, but also for controlling escalating operational costs and addressing the constraints of battery-powered devices such as mobile and embedded systems [52]. While several influential studies propose methodologies to assess the environmental cost of machine learning models, primarily during inference [3, 33, 35, 57], their computational and environmental costs remain largely unquantified in this context. Indeed, there remains a notable gap: no empirical evidence currently exists specific to Document AI that quantifies the trade-offs between parsing accuracy and energy consumption.

To address this gap, this paper empirically evaluates the trade-offs between accuracy and energy consumption across pipeline tools, specialized VLMs, and general-purpose VLMs. Using a controlled measurement protocol validated by physical wattmeters, we assess parsing performance alongside key sustainability metrics, including throughput, energy usage, and Carbon Dioxide equivalent (CO2eq) emissions. Our findings reveal the most compelling Document AI solutions and provide a data-driven basis for sustainable deployment at scale.

2 Methodology

To quantify the accuracy–energy trade-off and key sustainability indicators in Document AI systems, our experimental framework evaluates both parsing performance and energy consumption through a multi-faceted approach that combines rigorous energy measurement methods, dedicated datasets, and a controlled inference protocol.

2.1 Energy Estimation

The assessment of energy consumption and CO2eq emissions, which are closely correlated [34], relies on three main strategies. The first strategy, direct measurement via wattmeter [22], is used in this study as the ground-truth reference. The second, analytical estimation [33, 35], relies on static models and offers limited accuracy when accounting for real-world power fluctuations. The third strategy, on-chip monitoring [9, 14], uses interfaces such as NVML or RAPL to provide real-time power estimates, though it often fails to capture full system consumption. Within this on-chip monitoring paradigm, we adopt CodeCarbon [14], a software-based tool selected for its maturity and widespread adoption within the community (1.8k stars on GitHub). In our setup, CodeCarbon is configured to estimate the power usage of the entire machine rather than isolated processes, ensuring comparability with physical wattmeter measurements. The carbon intensity factor used in this study corresponds to the global average electricity mix (445 gCO2/kWh) [29], notwithstanding significant regional variations (e.g., 35 gCO2/kWh in Switzerland [12]).

2.2 Benchmarks and Datasets

A wide range of benchmarks exist to evaluate performance across Document AI tasks [18, 30, 37, 42, 48, 58, 66, 68], including OCR, layout analysis, table detection and reconstruction, form and formula understanding, key information extraction, and document visual question answering. Among them, OmniDocBench [46] has become one of the most representative benchmarks in recent publications introducing state-of-the-art Document AI tools [15, 20, 43, 63], often complemented by Ocean-OCR [10] and olmOCR-bench [49].

Developed by Shanghai AI Lab, OmniDocBench comprises 1,651 document images (JPG/PNG) and is designed as a comprehensive benchmark for evaluating document parsing in real-world conditions. The dataset covers a wide variety of document types, layout structures, and language settings, including academic papers, financial reports, newspapers, textbooks, exam papers, magazines, handwritten notes, and historical documents. It provides rich annotations at multiple levels, including block- and span-level categories, recognition targets for text, formulas, and tables, as well as reading-order information and fine-grained page and block attributes. In addition to its diversity, OmniDocBench is supported by an evaluation suite for both end-to-end and module-level assessment, making it an ideal candidate for analyzing the performance of complete parsing systems as well as individual components such as layout analysis, OCR, formula recognition, and table parsing. The benchmark also places particular emphasis on annotation quality through a multi-stage process combining automatic tools, manual refinement, expert inspection, and large-model-assisted verification.

While OmniDocBench was originally designed to benchmark parsing precision, this study leverages its end-to-end evaluation protocol to assess accuracy concurrently with energy consumption, carbon emissions, and system performance. We chose this benchmark because, unlike many single-task alternatives, it reflects the full breadth of Document AI tasks, covers diverse and challenging real-world documents, and offers a comprehensive evaluation framework with well-defined modern metrics, including CDM [60] for formula evaluation.

2.3 Hardware

Experiments were conducted on a workstation equipped with an NVIDIA RTX 6000 Blackwell Max-Q GPU with 96 GB of memory capped at 300 W, an Intel Core i7-9700K processor, and 32 GB of RAM, running Ubuntu 24.04.3 LTS (Noble Numbat).

2.4 Protocol

Our measurement protocol is designed to isolate energy consumption strictly during the inference phase, excluding initialization costs, such as model weight loading into VRAM, and auxiliary processing. This exclusion ensures a fair comparison across heterogeneous architectures, since large general-purpose VLMs may incur substantial one-time initialization costs that would otherwise skew

the results relative to lightweight pipeline tools. It also better reflects industrial deployments, where systems are typically evaluated under steady-state workloads and fixed initialization overheads are amortized over large volumes of documents.

To validate the robustness of our empirical evaluation, each tool performed the benchmark three separate times at different intervals, rather than consecutively. This precaution was taken to mitigate the potential impact of unforeseen background system processes, thereby improving the overall consistency and reliability of the measurements. The results presented in this paper represent the average of these three independent runs. Energy consumption is monitored with CodeCarbon [14] and validated using an external physical wattmeter placed between the computer and the electrical outlet to measure total system power draw, both sampled every five seconds. Moreover, to enhance the granularity of our profiling, CPU, GPU, and memory usage are simultaneously tracked using the Python libraries psutil, resource, and pynvml.

Documents are submitted sequentially to isolate inference costs, while internal parallelization strategies for multi-page inputs or page elements (e.g., blocks, regions) are governed by the native implementations of each tool. To ensure measurement fidelity, no competing processes external to the workflow were active beyond standard system services. Model configurations follow standard inference protocols, with general-purpose VLMs (e.g., Qwen) specifically adhering to the reference implementations provided by OmniDocBench [46]. Finally, our profiling captures over thirty indicators per measurement, assessing parsing performance alongside key sustainability metrics, including inference time (s), throughput (pages/s), CO2eq emissions (g/page), energy consumption (J or Wh), energy efficiency (pages/J), and CPU/GPU/memory utilization (% and MiB).

3 Experiments

Our experiments empirically evaluate accuracy, energy consumption, and CO2eq emissions using the controlled methodology detailed in Section 2. We consider a broad range of state-of-the-art open-source Document AI systems, grouped into the three most representative categories. First, for pipeline tools, we assess Docling (via RapidOCR) [19], Marker [47], MinerU2-pipeline [61], and PP-StructureV3 (via PP-OCRv5_mobile) [17]. These represent the most recent and widely adopted systems in this category as of late 2025 and early 2026. Second, for general-purpose Vision-Language Models (VLMs), we evaluate Qwen2.5-VL [5] across several quantized model sizes (AWQ, 4-bit): 3B, 7B, 32B, and 72B. Released in early 2025, Qwen2.5-VL is recognized as one of the best-performing open-source VLMs [5]. Third, for specialized VLMs, we assess MinerU2.5 [43], MinuerU2.5-Pro [59], PaddleOCR-VL [15], PaddleOCR-VL-1.5 [16], DeepSeek-OCR [63], DeepSeek-OCR 2 [64] and GLM-OCR [20]. Released in late 2025 and early 2026, these models are among the most accurate available and currently occupy the top ranks on the reference benchmark OmniDocBench v1.5 [46]. Notable alternatives excluded from our evaluation include proprietary models like GPT-5 [45] and Gemini-3 [24], and open-source models like HunyuanOCR [56], Nanonets-OCR-s [41], olmOCR2 [50], and Qwen-3-VL [67].

Finally, we rely on the full OmniDocBench benchmark to evaluate parsing performance (using reported values or computing missing scores) alongside key sustainability metrics, including throughput, energy usage, and CO2eq emissions. Following the methodology established by OmniDocBench, the Parsing Quality (PQ) metric, referred to as "Overall" within the benchmark, serves as the primary measure of document parsing accuracy. This metric provides a global evaluation of the model's ability to extract the full page content into a comprehensive Markdown format. As defined in Equation 1, it is calculated as the arithmetic mean of three dedicated sub-metrics: Text Edit Distance (where a lower value indicates fewer text extraction errors), Table TEDS (measuring table structural reconstruction quality), and Formula CDM (evaluating mathematical formula recognition). Consequently, a poor performance in any individual component directly impacts the global PQ score.

$$\mathrm{PQ} = \frac{(1 - \mathrm{Text}^{\mathrm{EditDistance}}) \cdot 100 + \mathrm{Table}^{\mathrm{TEDS}} + \mathrm{Formula}^{\mathrm{CDM}}}{3} \tag{1}$$

Preprocessing Constraints for Docling: While all systems were intended to be evaluated on the complete OmniDocBench dataset (1,651 images), technical constraints within Docling-2.58.0 required an adjustment. Memory management issues during the decompression of high-resolution images caused Out-Of-Memory (OOM) crashes for the largest documents. As the tool currently lacks native configuration parameters or workarounds to mitigate these memory spikes, Docling was evaluated on a subset of 1,600 images, excluding the 51 most computationally demanding files. All other tools were evaluated on the full dataset to preserve benchmark integrity, providing Docling's metrics as an indicative baseline. Consequently, its low energy footprint and high throughput must be interpreted with caution.

4 Results

The results presented in Table 1 and Figure 1 evaluate the trade-offs between accuracy and energy consumption across the three categories of Document AI systems under study. Our main findings derived from this evaluation are summarized below. Notably, our profiling reflects steady-state performance under a continuous full workload, excluding idle or intermittent power consumption. Moreover, the energy per page values correspond to the estimations generated by CodeCarbon rather than the measurements validated by physical wattmeters. However, as analyzed in Section 5.5, despite a systematic underestimation by CodeCarbon, both methods produce identical trends and system rankings.

4.1 Accuracy vs. Energy Trade-off

Our empirical evaluation reveals critical insights into the trade-offs between parsing performance and sustainability metrics in Document AI, leading to four key takeaways.

Notably, Specialized VLMs Emerge as the Most Compelling Category: Specialized models emerge as the optimal paradigm, successfully decoupling high parsing accuracy from high energy usage. Systems such as MinuerU2.5-Pro and GLM-OCR achieve state-of-the-art parsing performance while maintaining a minimal energy footprint. Specifically, GLM-OCR achieves the best balance

Table 1: Evaluation of pipeline tools, general VLMs, and specialized VLMs on OmniDocBench v1.5. Parsing quality scores are sourced from [16] for pipelines, and the best of [16, 59] for general VLMs, except for Docling and Qwen2.5-VL-XXB-AWQ (our evaluation). We compute all remaining metrics. All tools are evaluated on a GPU, with VLMs running on vLLM. Bold and underlined values denote the best and second-best results; missing values indicate unmeasured metrics.

Type	Method	Params	Parsing Quality ↑ (overall)	Throughput ↑ (pages/s)	Energy ↓ (kJ/page)	CO2eq ↓ (g/page)
Pipeline	PP-StructureV3 [17]	–	86.73	0.546	<u>0.262</u>	<u>0.032</u>
	MinerU2-pipeline [61]	–	75.51	0.325	0.369	0.046
	Marker-1.8.2 [47]	–	71.30	0.128	1.716	0.212
	Docling-2.58.0 [19]	–	49.23	**0.581**	**0.101**	**0.012**
General VLMs	Gemini-3 Pro [25]	–	92.85	–	–	–
	Qwen3-VL-235B-Instruct [67]	235B	89.78	–	–	–
	Qwen2.5-VL-72B-AWQ [5]	72B	88.37	0.025	13.024	1.610
	GPT-5.2 [44]	–	86.52	–	–	–
	Qwen2.5-VL-32B-AWQ [5]	32B	82.72	0.050	6.300	0.779
	Qwen2.5-VL-7B-AWQ [5]	7B	80.37	0.179	1.655	0.205
	Qwen2.5-VL-3B-AWQ [5]	3B	71.35	0.225	1.124	0.139
Specialized VLMs	MinuerU2.5-Pro [59]	1.2B	**95.69**	0.362	0.611	0.076
	GLM-OCR [20]	0.9B	<u>95.15</u>	<u>0.562</u>	0.317	0.039
	PaddleOCR-VL-1.5 [16]	0.9B	94.87	0.468	0.347	0.043
	PaddleOCR-VL [15]	0.9B	94.11	0.475	0.339	0.042
	MinerU2.5 [43]	1.2B	92.98	0.352	0.617	0.076
	DeepSeek-OCR 2 [64]	3B	90.17	0.393	0.659	0.081
	Deepseek-OCR [63]	3B	87.01	0.455	0.577	0.071

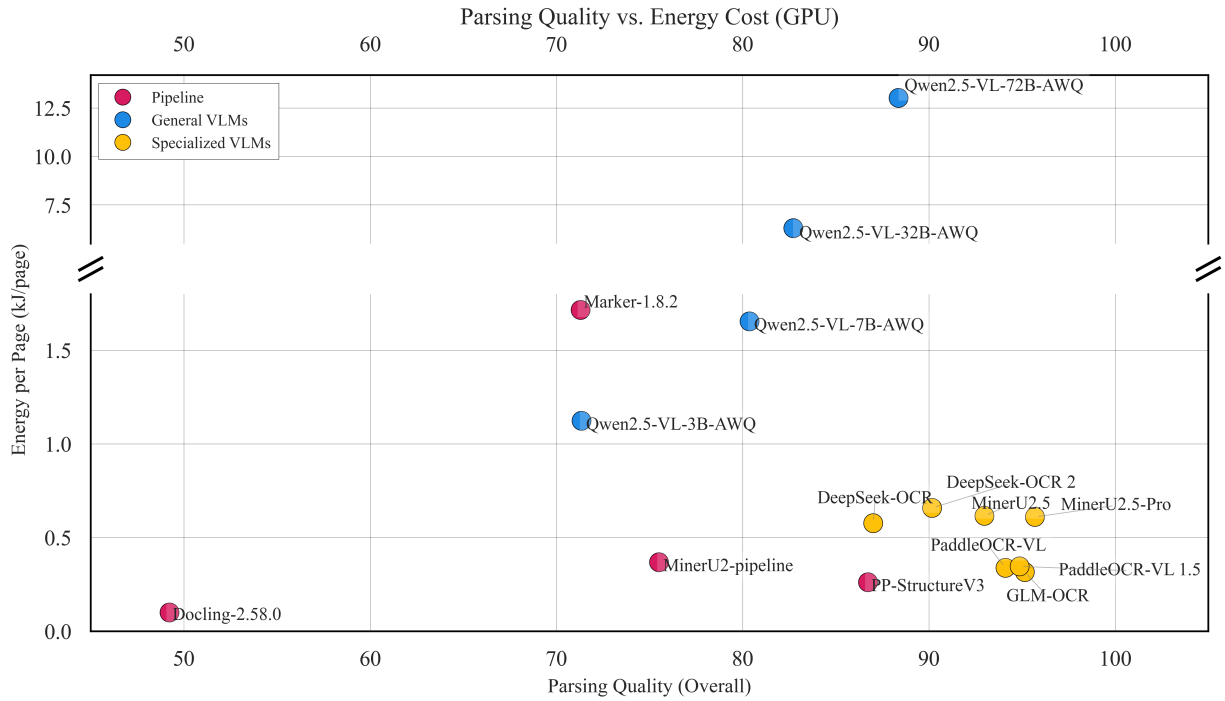


Figure 1: Accuracy–energy trade-offs across pipeline tools, general VLMs, and specialized VLMs.

between accuracy and energy usage, reaching a top-tier accuracy score of 95.15 while requiring 0.317 kJ/page.

In Contrast, Larger General-Purpose VLMs Consume Significantly More Energy: Massive general-purpose models incur prohibitive computational and environmental costs. Relying on sheer parameter scale consumes orders of magnitude more energy per page without necessarily delivering superior results. For instance, Qwen2.5-VL-72B-AWQ consumes over 40x more energy than GLM-OCR while exhibiting lower accuracy, despite having 80x more parameters. This observation aligns with established scaling trends [11, 27, 31], but underscores their inefficiency for this specific task.

Pipeline Tools Demonstrate Lower Accuracy: Traditional pipeline tools exhibit unmatched throughput and minimal CO2eq emissions, making them highly attractive for sustainability. Docling, for example, achieves the lowest overall energy consumption (0.101 kJ/page) and carbon footprint, yet delivers a significantly lower parsing quality score of 49.23. Consequently, while these systems are highly efficient, their limited accuracy restricts their viability to resource-constrained applications where precision is secondary.

No Strong Correlation Between Parsing Accuracy and Energy Consumption: Ultimately, our findings corroborate that, within our results, there is no strong correlation between raw parameter count and task-specific performance. Sustainable and highly accurate document parsing is driven by targeted architectural specialization rather than brute-force scaling. Specialized VLMs reach equal or greater accuracy than large general-purpose models while being significantly less energy-intensive.

4.2 Orders of Magnitude & Limitations

To provide concrete reference points, a dishwasher cycle consumes approximately 1.16 kWh (≈ 4176 kJ) [2]; with this energy, GLM-OCR could process approximately 13,000 pages. Similarly, regarding CO2eq emissions, an average European passenger car emits about 130 gCO2/km [21]; driving a single kilometer therefore generates emissions equivalent to processing approximately 3,300 pages with the same model.

Moreover, our extensive profiling shows that the hardware platform (CPU vs. GPU) critically impacts performance (see Section 5.1), and that the choice of inference backend (e.g., vLLM, Transformers) plays a major role, potentially increasing throughput and reducing energy consumption (see Section 5.2). Furthermore, document characteristics, specifically the complexity and length of the processed document, have a direct impact on throughput and energy efficiency (see Sections 5.3 and 5.4). Finally, CodeCarbon exhibits certain limitations, notably a tendency to underestimate actual energy usage (see Section 5.5).

5 Discussion

While the previous section established the primary accuracy-energy trade-offs, practical efficiency in Document AI is heavily influenced by several granular technical variables. Drawing on our empirical profiling, this section details the key factors influencing throughput and energy usage. We first examine the impact of the hardware platform (CPU vs. GPU) and the inference backend. We then evaluate how document characteristics, specifically length and complexity, affect parsing performance. Finally, we discuss the methodological limitations of our study, notably the constraints of software-based energy estimation via CodeCarbon.

5.1 Impact of Hardware: CPU vs. GPU

An evaluation was conducted to assess the influence of hardware architecture on parsing performance and sustainability metrics, using the same measurement protocol and dataset as our primary benchmarks. As shown in Table 2, the GPU results align with the values previously reported in Table 1, providing a consistent baseline for comparison.

Table 2: Performance comparison between GPU and CPU for pipeline tools.

Method	Hardware	Throughput (pages/s)	Energy (kJ/page)
Docling	GPU	0.581	0.101
	CPU	0.025	1.418
PP-StructureV3	GPU	0.546	0.262
	CPU	0.018	5.180
MinerU2-pipeline	GPU	0.325	0.369
	CPU	0.041	2.802

Due to the prohibitive latency associated with CPU-based inference, this comparison was limited to the three most efficient pipeline tools. Notably, the observed disparity reveals a clear trend likely applicable to VLMs: CPU architectures are inherently inefficient for Document AI tasks. Moreover, across all tested methods, CPU execution results in a drastic efficiency collapse, rendering it suboptimal for large-scale document processing.

5.2 Influence of Inference Backends

An evaluation was conducted to assess the impact of different inference backends (vLLM, Transformers/PyTorch, and native implementations) on the efficiency of two specialized VLMs (MinerU2.5 and PaddleOCR-VL) and one general-purpose VLM (Qwen2.5-VL-32B-AWQ). This model selection reflects the chronology of our experiments, which were conducted before the release of newer iterations and architectures, such as MinerU2.5-Pro, GLM-OCR, and PaddleOCR-VL-1.5, thereby accounting for the discrepancy with the systems evaluated in our primary results. While preserving the controlled measurement protocol used in the primary benchmarks, this evaluation relies on a smaller 4-page in-house dataset, comprising complex documents with images, tables, text, and formulas, illustrated in Figure S1 of the supplementary material, rather than the full OmniDocBench, since processing the entire benchmark was unnecessary for this analysis given its prohibitive time and energy footprint. Accordingly, as this constitutes a secondary profiling rather than the central study, these findings should be interpreted with caution. Table 3 summarizes the obtained results.

Our findings reveal that for PaddleOCR-VL, the vLLM GPU backend increases throughput by 17.4× and reduces energy usage per page by 12.2× compared to the native PaddlePaddle GPU setup. Similarly, for MinerU2.5, vLLM GPU delivers a 15.5× higher throughput

Table 3: Performance comparison across multiple VLM inference backends. †Refers to the Qwen2.5-VL-32B-AWQ model.

Method	Platform	Throughput (pages/s)	Energy (kJ/page)
MinerU2.5	Transformers CPU	0.004	22.25
	Transformers GPU	0.047	2.41
	vLLM GPU	0.730	0.23
PaddleOCR-VL	PaddlePaddle CPU	0.003	27.19
	PaddlePaddle GPU	0.069	1.71
	vLLM GPU	1.202	0.14
Qwen2.5-VL†	Transformers GPU	0.011	23.30
	vLLM GPU	0.057	5.23

and reduces energy usage per page by 10.5× relative to the Transformers GPU setup. In contrast, for the larger general-purpose Qwen2.5-VL-32B, gains are lower, with throughput improved by 5.18× and energy footprint per page reduced by 4.46×.

This acceleration is primarily attributed to vLLM's PagedAttention mechanism [32], which eliminates wasteful key-value cache memory allocation and reduces fragmentation by using small, dynamically allocated blocks. This structure allows for sharing blocks between sequences and avoids recalculating identical tokens. Finally, our extensive profiling shows that through this memory optimization and effective batch size increase, vLLM significantly enhances GPU utilization, yielding substantial throughput gains (typically 2× to 4×, but notably higher in our empirical results) over standard implementations.

5.3 Impact of Document Complexity

An evaluation was conducted to assess the impact of document complexity on throughput and energy consumption. Consistent with the rationale outlined in Section 5.2, the model selection and the adoption of an in-house dataset reflect the experimental chronology and the necessity of minimizing the computational footprint of this secondary profiling. While maintaining the controlled measurement protocol of our primary benchmarks, this analysis utilizes two representative PDFs from an in-house dataset (illustrated in Figure S2 of the supplementary material) to facilitate a direct comparison between simple and complex document structures. The simple document consists of a single page comprising only one-column text (4,939 characters, 735 words), whereas the complex document integrates text, a table, a formula, a figure, and two columns (4,298 characters, 800 words). Table 4 summarizes the resulting metrics.

Our findings reveal that document complexity significantly affects throughput and energy usage. Pipeline tools demonstrate a slowdown ranging from 2.6× to 5.9× on complex documents. Notably, within this limited evaluation setting, Marker stands out as an exception, achieving an approximate 1.35× speedup on complex documents. In contrast, general-purpose VLMs show a more moderate degradation, with slowdown factors between 1.6× and 2.1×. Specialized VLMs emerge as the most robust category, exhibiting only a limited performance drop of approximately 1.4×. Finally, we emphasize that throughput must be weighed against parsing accuracy, as speed alone does not determine the optimal tool.

Table 4: Impact of document complexity on the efficiency of pipeline tools, general VLMs, and specialized VLMs. Evaluated on GPU (VLMs via vLLM). Bold and underlined denote best and second-best results. † Suffix -AWQ is omitted for brevity.

Type	Method	Throughput (pages/s)		Energy (kJ/page)	
		Simple	Complex	Simple	Complex
Pipeline	Docling-2.58.0	**2.089**	0.357	**0.05**	<u>0.31</u>
	Marker-1.8.2	0.052	0.070	5.04	2.80
	MinerU2-pipeline	1.184	<u>0.389</u>	<u>0.09</u>	**0.29**
	PP-StructureV3	<u>1.292</u>	**0.491**	<u>0.09</u>	**0.29**
General VLMs	Qwen2.5-VL-3B†	0.057	0.034	2.41	3.94
	Qwen2.5-VL-7B†	0.066	0.032	2.62	4.99
	Qwen2.5-VL-32B†	0.032	0.020	7.09	11.82
	Qwen2.5-VL-72B†	0.025	0.014	12.85	21.88
Specialized VLMs	MinerU2.5	0.373	0.270	0.40	0.55
	PaddleOCR-VL	0.441	0.309	0.34	0.52

5.4 Impact of Document Length

An evaluation was conducted to assess the impact of document length on throughput and energy consumption. Consistent with the rationale outlined in Section 5.2, the model selection and the adoption of an in-house dataset reflect the experimental chronology and the necessity of minimizing the computational footprint of this secondary profiling. While maintaining the controlled measurement protocol of our primary benchmarks, this analysis utilizes two representative PDFs from an in-house dataset (illustrated in Figure S2 of the supplementary material) to facilitate a direct comparison between short and long document structures. The short document corresponds to the same simple document used in Section 5.3 (a single page of simple text), whereas the long document is obtained by replicating this content ten times, resulting in a ten-page PDF comprising identical repeated content. We further verified that no caching mechanism in the assessed tools could bias the measured throughput or energy consumption. Table 5 summarizes the resulting metrics.

Our findings reveal that document length heavily influences throughput and energy usage. Pipeline tools demonstrate speedups ranging from 1.2× to 4.5× when processing longer documents. In contrast, general-purpose VLMs benefit significantly more from increased document length, achieving throughput gains of approximately 9× to 10×. Specialized VLMs also show substantial improvements, with speedups exceeding 6×.

Moreover, this change is due, among other factors, to fixed processing costs that persist even when process initialization and loading are excluded from the measurements. These largely content-independent costs are amortized across multiple pages but become dominant for single-page documents, mechanically reducing throughput. It is also possible that certain GPU optimizations, such

Table 5: Impact of document length on the efficiency of pipeline tools, general VLMs, and specialized VLMs. Evaluated on GPU (VLMs via vLLM). Bold and underlined denote best and second-best results. † Suffix -AWQ is omitted for brevity.

Type	Method	Throughput (pages/s)		Energy (kJ/page)	
		Short	Long	Short	Long
Pipeline	Docling-2.58.0	**2.089**	**4.149**	**0.05**	**0.03**
	Marker-1.8.2	0.052	0.236	5.04	1.20
	MinerU2-pipeline	1.184	2.149	<u>0.09</u>	<u>0.07</u>
	PP-StructureV3	<u>1.292</u>	1.601	<u>0.09</u>	0.10
General VLMs	Qwen2.5-VL-3B†	0.057	0.511	2.41	0.36
	Qwen2.5-VL-7B†	0.066	0.620	2.62	0.37
	Qwen2.5-VL-32B†	0.032	0.305	7.09	1.04
	Qwen2.5-VL-72B†	0.025	0.240	12.85	1.49
Specialized VLMs	MinerU2.5	0.373	2.288	0.40	0.08
	PaddleOCR-VL	0.441	<u>2.745</u>	0.34	0.09

as parallelization strategies, only become effective when processing multi-page documents. Finally, these discrepancies warrant further investigation but remain beyond the scope of this study.

5.5 Limitations

While this study empirically evaluates the trade-offs between accuracy and energy consumption in Document AI, it is subject to certain scope and methodological limitations. First, our energy measurements rely on a single hardware configuration; although validated by physical wattmeters, absolute values may vary across different infrastructures. Second, while we assessed a representative sample of state-of-the-art open-source Document AI systems covering the main technological categories, this selection is not exhaustive. Notably, it excludes the energy profiling of proprietary APIs, such as GPT or Gemini, due to the opacity of their server-side consumption. Third, several secondary profiling experiments rely on smaller in-house datasets rather than the full OmniDocBench; these analyses were designed to isolate specific factors and reduce the computational footprint, but their results should be interpreted as complementary rather than fully generalizable.

Finally, regarding energy profiling, software-based on-chip monitoring tools present inherent measurement boundaries. To quantify this, an evaluation was conducted to measure the error between the power consumption estimated by CodeCarbon and the physical measurements obtained using a wattmeter. This analysis utilizes the measurements from Table 1 and Table 2, supplemented by additional data gathered under the same controlled measurement protocol and dataset as our primary benchmarks. Table 6 summarizes the obtained results.

Our findings reveal that CodeCarbon underestimates energy consumption by an average of 22.98% on GPU and 29.63% on CPU. These findings are consistent with the results reported by Fischer [22], who identify a typical underestimation of approximately 20–30% on average, with a more pronounced bias on CPUs.

Table 6: CodeCarbon underestimation vs. Wattmeter.

Hardware	Min (%)	Max (%)	Mean (%)
GPU	19.11	30.92	22.98
CPU	27.40	31.86	29.63

The overall measurement bias mainly arises from the omission or approximation of non-profiled components [22]. CodeCarbon estimates CPU, GPU, and RAM energy consumption using software telemetry or heuristics: GPU power is tracked through NVML, CPU energy is read from RAPL when available, and RAM power is approximated from the memory configuration. It therefore excludes or only indirectly captures other system-level contributions, including PSU losses, storage, motherboard components, and cooling. Since the wattmeter captures the total outlet consumption, these omissions and approximations mechanically lead to underestimation. Moreover, telemetry sampling frequency and smoothing may miss transient power spikes. Cooling, which can represent more than 10% of the total energy footprint of an AI workload [6], is not accounted for by CodeCarbon.

The underestimation is more pronounced on CPUs compared to GPUs, which can be attributed to the limitations of CPU power estimation boundaries and the disproportionate impact of unmeasured system components [22]. While the unmeasured system baseline remains roughly constant, CPU inference consumes significantly less power than GPU inference; consequently, this baseline represents a much larger relative share of the total energy usage. Furthermore, architectural differences in power reporting exacerbate this gap. GPU telemetry captures the entire graphics board, including VRAM and cooling fans. In contrast, CPU telemetry isolates the processor socket, completely excluding system RAM and the CPU cooler. This discrepancy inherently widens the gap with physical wattmeter measurements for CPUs.

Consequently, while CodeCarbon provides an accessible software solution for evaluating energy consumption and comparative benchmarking, our reported sustainability metrics should be interpreted as robust approximations rather than absolute ground truths. Our findings reveal that both measurement methods produce identical trends and system rankings; therefore, this study reports CodeCarbon values, although the corresponding physical wattmeter measurements would have led to the same comparative conclusions. Moreover, this choice reflects the greater practical scalability of software-based profiling, which can be deployed across larger experimental campaigns more readily than physical wattmeter measurements.

6 Conclusion

This empirical evaluation of state-of-the-art open-source Document AI systems reveals no strong correlation between parsing accuracy and energy consumption. Our findings demonstrate that pipeline tools and specialized VLMs offer superior efficiency compared to larger general-purpose models. Notably, specialized VLMs emerge as the most compelling category, with GLM-OCR achieving the

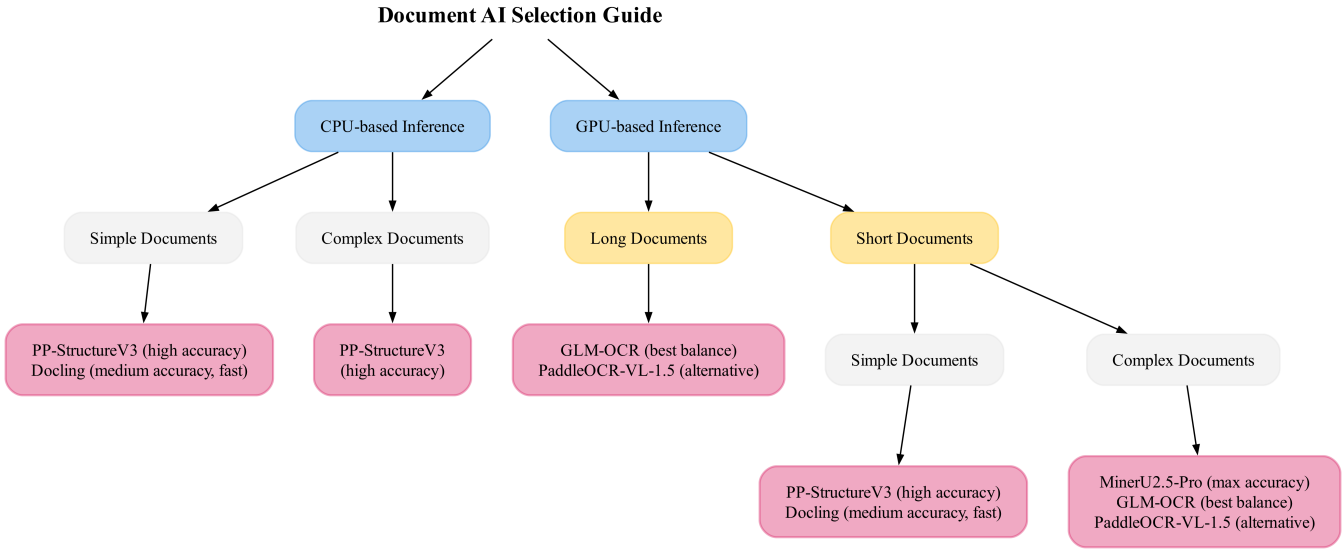


Figure 2: Decision guide for selecting open-source Document AI systems based on hardware (CPU/GPU) and document characteristics (length and complexity). The "best balance" indicator highlights the optimal accuracy-energy trade-off.

best balance between accuracy and energy usage. Beyond these primary trade-offs, our extensive profiling underscores the limitations of software-based energy estimation, and shows that efficiency is heavily influenced by the hardware platform (CPU vs. GPU), the inference backend, and document characteristics, specifically length and complexity. Consequently, these findings confirm that a paradigm shift toward targeted architectural specialization, rather than brute-force scaling, is essential for cost-effective and sustainable Document AI, although we caution that significant parsing performance disparities persist among these specialized systems. To assist users, we propose a practical decision guide (Figure 2) for selecting the optimal system based on specific hardware and document constraints.

7 Ethical Statement

The primary motivation of this work is to promote sustainable and cost-effective AI by quantifying the hidden computational and environmental costs of Document AI. By demonstrating that specialized VLMs can outperform larger general-purpose models, we aim to encourage the deployment of systems that minimize both energy footprints and operational expenses. Regarding data privacy, our empirical evaluation relies on a single public benchmark, the reference benchmark OmniDocBench v1.5, alongside three in-house datasets derived from documents extracted from DocBank [38], which is also a public dataset. This ensures that no personally identifiable information was involved. Moreover, our extensive profiling highlights that most state-of-the-art open-source Document AI systems evaluated in this study originate from a single geographic region, specifically China. While this concentration does not imply inherent bias, it underscores a lack of global diversity that could influence training data focus, language adaptation, or digital sovereignty. Practitioners should thus remain attentive to these contextual factors when deploying these models across different linguistic or critical applications.

Acknowledgments

This work was supported by the University of Applied Sciences and Arts (HES-SO) and conducted at the Institute of Artificial Intelligence and Complex Systems (iCoSys). We also extend our gratitude to the open-source community, the creators of OmniDocBench, and the CodeCarbon team for providing the systems, reference benchmarks, and energy monitoring tools essential to this empirical evaluation.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* (2023). doi:10.48550/arXiv.2303.08774

[2] Agence de la transition écologique (ADEME). 2025. *Électricité : quels appareils consomment le plus dans votre maison.* https://agirpourlatransition.ademe.fr/particuliers/maison/economies-denergie-deau/electricite-appareils-consomment-plus-maison

[3] Mauricio Fadel Argerich and Marta Patiño-Martínez. 2024. Measuring and improving the energy efficiency of large language models inference. *IEEE Access* 12 (2024), 80194–80207.

[4] AWS. 2025. What is Intelligent Document Processing (IDP)? https://aws.amazon.com/what-is/intelligent-document-processing/

[5] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. 2025. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923* (2025). doi:10.48550/arXiv.2502.13923

[6] Luiz André Barroso, Urs Hölzle, Parthasarathy Ranganathan, et al. 2013. The datacenter as a computer: Designing warehouse-scale machines. *Margen & Claypool* (2013).

[7] Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big?. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*. 610–623.

[8] Aline FS Borges, Fernando JB Laurindo, Mauro M Spínola, Rodrigo F Gonçalves, and Claudia A Mattos. 2021. The strategic use of artificial intelligence in the digital era: Systematic literature review and future research directions. *International journal of information management* 57 (2021), 102225.

[9] Semen Andreevich Budennyy, Vladimir Dmitrievich Lazarev, Nikita Nikolaevich Zakharenko, Aleksei N Korovin, OA Plosskaya, Denis Valer'evich Dimitrov, VS Akhripkin, IV Pavlov, Ivan Valer'evich Oseledets, Ivan Segundovich Barsola, et al. 2022. Eco2ai: carbon emissions tracking of machine learning models as the first step towards sustainable ai. In *Doklady mathematics*, Vol. 106. Springer, S118–S128. doi:10.1134/S1064562422060230

[10] Song Chen, Xinyu Guo, Yadong Li, Tao Zhang, Mingan Lin, Dongdong Kuang, Youwei Zhang, Lingfeng Ming, Fengyu Zhang, Yuran Wang, et al. 2025. Ocean-ocr: Towards general ocr application via a vision-language model. *arXiv preprint arXiv:2501.15558* (2025). doi:10.48550/arXiv.2501.15558

[11] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. 2023. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2818–2829.

[12] Climatiq. 2021. Electricity supplied from grid – residual mix. Climatiq Emission Factor Database. https://www.climatiq.io/data/emission-factor/ed6ef6e0-c51f-4237-9639-bce6a31a2e80

[13] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261* (2025). doi:10.48550/arXiv.2507.0626

[14] Benoit Courty, Victor Schmidt, Sasha Luccioni, Goyal-Kamal, MarionCoutarel, Boris Feld, Jérémy Lecourt, LiamConnell, Amine Saboni, Inimaz, supatomic, Mathilde Léval, Luis Blanche, Alexis Cruveiller, ouminasara, Franklin Zhao, Aditya Joshi, Alexis Bogroff, Hugues de Lavoreille, Niko Laskaris, Edoardo Abati, Douglas Blank, Ziyao Wang, Armin Catovic, Marc Alencon, Michał Stęchły, Christian Bauer, Lucas Otávio N. de Araújo, JPW, and MinervaBooks. 2024. *mlco2/codecarbon: v2.4.1*. doi:10.5281/zenodo.11171501

[15] Cheng Cui, Ting Sun, Suyin Liang, Tingquan Gao, Zelun Zhang, Jiaxuan Liu, Xueqing Wang, Changda Zhou, Hongen Liu, Manhui Lin, et al. 2025. PaddleOCR-VL: Boosting Multilingual Document Parsing via a 0.9 B Ultra-Compact Vision-Language Model. *arXiv preprint arXiv:2510.14528* (2025).

[16] Cheng Cui, Ting Sun, Suyin Liang, Tingquan Gao, Zelun Zhang, Jiaxuan Liu, Xueqing Wang, Changda Zhou, Hongen Liu, Manhui Lin, et al. 2026. PaddleOCR-VL-1.5: Towards a Multi-Task 0.9 B VLM for Robust In-the-Wild Document Parsing. *arXiv preprint arXiv:2601.21957* (2026).

[17] Cheng Cui, Ting Sun, Manhui Lin, Tingquan Gao, Yubo Zhang, Jiaxuan Liu, Xueqing Wang, Zelun Zhang, Changda Zhou, Hongen Liu, Yue Zhang, Wenyu Lv, Kui Huang, Yichao Zhang, Jing Zhang, Jun Zhang, Yi Liu, Dianhai Yu, and Yanjun Ma. 2025. PaddleOCR 3.0 Technical Report. arXiv:2507.05595 [cs.CV] doi:10.48550/arXiv.2507.05595

[18] Lei Cui, Yiheng Xu, Tengchao Lv, and Furu Wei. 2021. Document AI: Benchmarks, Models and Applications. arXiv:2111.08609 [cs.CL] doi:10.48550/arXiv.2111.08609

[19] Deep Search Team. 2024. *Docling Technical Report*. Technical Report. arXiv:2408.09869 doi:10.48550/arXiv.2408.09869

[20] Shuaiqi Duan, Yadong Xue, Weihan Wang, Zhe Su, Huan Liu, Sheng Yang, Guobing Gan, Guo Wang, Zihan Wang, Shengdong Yan, Dexin Jin, Yuxuan Zhang, Guohong Wen, Yanfeng Wang, Yutao Zhang, Xiaohan Zhang, Wenyi Hong, Yukuo Cen, Da Yin, Bin Chen, Wenmeng Yu, Xiaotao Gu, and Jie Tang. 2026. GLM-OCR Technical Report. arXiv:2603.10910 [cs.CL] https://arxiv.org/abs/2603.10910

[21] European Environment Agency. 2023. *CO2 emissions from new passenger cars in Europe — Data Explorer*. European Environment Agency (EEA). https://co2cars.apps.eea.europa.eu/

[22] Raphael Fischer. 2025. Ground-Truthing AI Energy Consumption: Validating CodeCarbon Against External Measurements. arXiv:2509.22092 [cs.AI] doi:10.48550/arXiv.2509.22092

[23] John Gantz and David Reinsel. 2012. The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east. *IDC iView: IDC Analyze the future* 2007, 2012 (2012), 1–16.

[24] Google. 2025. A new era of intelligence with Gemini 3. Google Blog. https://blog.google/products/gemini/gemini-3/

[25] Google DeepMind. 2025. Gemini 3.0. https://blog.google/products-and-platforms/products/gemini/gemini-3-collection/

[26] Varun Grover, Roger HL Chiang, Ting-Peng Liang, and Dongsong Zhang. 2018. Creating strategic business value from big data analytics: A research framework. *Journal of management information systems* 35, 2 (2018), 388–423.

[27] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556* (2022).

[28] IBM. 2025. What is Document AI? https://www.ibm.com/think/topics/document-ai

[29] International Energy Agency (IEA). 2025. Electricity 2025 — Emissions. https://www.iea.org/reports/electricity-2025/emissions

[30] Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. 2019. Funsd: A dataset for form understanding in noisy scanned documents. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, Vol. 2. IEEE, 1–6. doi:10.1109/ICDARW.2019.10029

[31] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361* (2020).

[32] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles* (Koblenz, Germany) *(SOSP '23)*. Association for Computing Machinery, New York, NY, USA, 611–626. doi:10.1145/3600006.3613165

[33] Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. 2019. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700* (2019). doi:10.48550/arXiv.1910.09700

[34] Loïc Lannelongue, Jason Grealey, and Michael Inouye. 2021. Green algorithms: quantifying the carbon footprint of computation. *Advanced science* 8, 12 (2021), 2100707.

[35] Loïc Lannelongue, Michael Inouye, Jason Grealey, Even Matencio, and ongcp97. 2025. Green Algorithms: Towards Environmentally Sustainable Computational Science. https://www.green-algorithms.org/

[36] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 9459–9474. https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf

[37] Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, Ming Zhou, and Zhoujun Li. 2020. Tablebank: Table benchmark for image-based table detection and recognition. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*. 1918–1925. doi:10.48550/arXiv.1903.01949

[38] Minghao Li, Yiheng Xu, Lei Cui, Shaohan Huang, Furu Wei, Zhoujun Li, and Ming Zhou. 2020. Docbank: A benchmark dataset for document layout analysis. *arXiv preprint arXiv:2006.01038* (2020). doi:10.48550/arXiv.2006.01038

[39] Jian Liang, David Doermann, and Huiping Li. 2005. Camera-based analysis of text and documents: a survey. *International Journal of Document Analysis and Recognition (IJDAR)* 7, 2 (2005), 84–104.

[40] Hanchao Liu, Wenyuan Xue, Yifei Chen, Dapeng Chen, Xiutian Zhao, Ke Wang, Liping Hou, Rongjun Li, and Wei Peng. 2024. A survey on hallucination in large vision-language models. *arXiv preprint arXiv:2402.00253* (2024).

[41] Souvik Mandal. 2025. Nanonets OCR Small. https://nanonets.com/research/nanonets-ocr-s/

[42] Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. 2021. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*. 2200–2209. doi:10.48550/arXiv.2007.00398

[43] Junbo Niu, Zheng Liu, Zhuangcheng Gu, Bin Wang, Linke Ouyang, Zhiyuan Zhao, Tao Chu, Tianyao He, Fan Wu, Qintong Zhang, et al. 2025. MinerU2. 5: A Decoupled Vision-Language Model for Efficient High-Resolution Document Parsing. *arXiv preprint arXiv:2509.22186* (2025). doi:10.48550/arXiv.2509.22186

[44] OpenAI. 2025. GPT-5.2 System Card. https://cdn.openai.com/pdf/3a4153c8-c748-4b71-8e31-aecbde944f8d/oai_5_2_system-card.pdf

[45] OpenAI. 2025. Introducing GPT-5. OpenAI Blog. https://openai.com/index/introducing-gpt-5/

[46] Linke Ouyang, Yuan Qu, Hongbin Zhou, Jiawei Zhu, Rui Zhang, Qunshu Lin, Bin Wang, Zhiyuan Zhao, Man Jiang, Xiaomeng Zhao, et al. 2025. Omnidocbench: Benchmarking diverse pdf document parsing with comprehensive annotations. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 24838–24848. doi:10.48550/arXiv.2412.07626

[47] Vik Paruchuri. 2025. marker: Convert PDF to Markdown + JSON quickly with high accuracy. https://github.com/datalab-to/marker

[48] Birgit Pfitzmann, Christoph Auer, Michele Dolfi, Ahmed S Nassar, and Peter Staar. 2022. Doclaynet: A large human-annotated dataset for document-layout segmentation. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*. 3743–3751. doi:10.1145/3534678.3539043

[49] Jake Poznanski, Aman Rangapur, Jon Borchardt, Jason Dunkelberger, Regan Huff, Daniel Lin, Christopher Wilhelm, Kyle Lo, and Luca Soldaini. 2025. olmocr: Unlocking trillions of tokens in pdfs with vision language models. *arXiv preprint arXiv:2502.18443* (2025). doi:10.48550/arXiv.2502.18443

[50] Jake Poznanski, Luca Soldaini, and Kyle Lo. 2025. olmOCR 2: Unit Test Rewards for Document OCR. arXiv:2510.19817 [cs.CV] doi:10.48550/arXiv.2510.19817

[51] rednote hilab. 2025. *dots.ocr: Multilingual Document Layout Parsing in a Single Vision-Language Model*. https://github.com/rednote-hilab/dots.ocr

[52] Charlotte Rodriguez, Laura Degioanni, Laetitia Kameni, Richard Vidal, and Giovanni Neglia. 2024. Evaluating the energy consumption of machine learning: Systematic literature review and experiments. *arXiv preprint arXiv:2408.15128* (2024). doi:10.48550/arXiv.2408.15128

[53] David Reinsel-John Gantz-John Rydning, John Reinsel, and John Gantz. 2018. The digitization of the world from edge to core. *Framingham: International Data*

Corporation 16 (2018), 1–28.

[54] Roy Schwartz, Jesse Dodge, Noah A Smith, and Oren Etzioni. 2020. Green ai. *Commun. ACM* 63, 12 (2020), 54–63.

[55] Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th annual meeting of the association for computational linguistics*. 3645–3650.

[56] Hunyuan Vision Team, Pengyuan Lyu, Xingyu Wan, Gengluo Li, Shangpin Peng, Weinong Wang, Liang Wu, Huawen Shen, Yu Zhou, Canhui Tang, et al. 2025. Hunyuanocr technical report. *arXiv preprint arXiv:2511.19575* (2025).

[57] Arya Tschand, Arun Tejusve Raghunath Rajan, Sachin Idgunji, Anirban Ghosh, Jeremy Holleman, Csaba Kiraly, Pawan Ambalkar, Ritika Borkar, Ramesh Chukka, Trevor Cockrell, et al. 2025. MLPerf Power: Benchmarking the Energy Efficiency of Machine Learning Systems from μWatts to MWatts for Sustainable AI. In *2025 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. IEEE, 1201–1216. doi:10.48550/arXiv.2410.12032

[58] Jordy Van Landeghem, Rubèn Tito, Łukasz Borchmann, Michał Pietruszka, Pawel Joziak, Rafal Powalski, Dawid Jurkiewicz, Mickaël Coustaty, Bertrand Anckaert, Ernest Valveny, et al. 2023. Document understanding dataset and evaluation (dude). In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 19528–19540. doi:10.48550/arXiv.2305.08455

[59] Bin Wang, Tianyao He, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Tao Chu, Yuan Qu, Zhenjiang Jin, Weijun Zeng, Ziyang Miao, et al. 2026. MinerU2.5-Pro: Pushing the Limits of Data-Centric Document Parsing at Scale. *arXiv preprint arXiv:2604.04771* (2026).

[60] Bin Wang, Fan Wu, Linke Ouyang, Zhuangcheng Gu, Rui Zhang, Renqiu Xia, Botian Shi, Bo Zhang, and Conghui He. 2025. Image Over Text: Transforming Formula Recognition Evaluation with Character Detection Matching. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 19681–19690.

[61] Bin Wang, Chao Xu, Xiaomeng Zhao, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Rui Xu, Kaiwen Liu, Yuan Qu, Fukai Shang, Bo Zhang, Liqun Wei, Zhihao Sui, Wei Li, Botian Shi, Yu Qiao, Dahua Lin, and Conghui He. 2024. MinerU: An Open-Source Solution for Precise Document Content Extraction. arXiv:2409.18839 [cs.CV] doi:10.48550/arXiv.2409.18839

[62] Abdul Wasay. 2025. *Intelligent Document Processing Global Market Report 2025*. Technical Report. The Business Research Company. https://www.thebusinessresearchcompany.com/report/intelligent-document-processing-global-market-report

[63] Haoran Wei, Yaofeng Sun, and Yukun Li. 2025. DeepSeek-OCR: Contexts Optical Compression. arXiv:2510.18234 [cs.CV] doi:10.48550/arXiv.2510.18234

[64] Haoran Wei, Yaofeng Sun, and Yukun Li. 2026. DeepSeek-OCR 2: Visual Causal Flow. arXiv:2601.20552 [cs.CV] https://arxiv.org/abs/2601.20552

[65] Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2020. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*. 1192–1200. doi:10.1145/3394486.3403172

[66] Yiheng Xu, Tengchao Lv, Lei Cui, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, and Furu Wei. 2022. XFUND: A benchmark dataset for multilingual visually rich form understanding. In *Findings of the association for computational linguistics: ACL 2022*. 3214–3224. doi:10.18653/v1/2022.findings-acl.253

[67] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388* (2025).

[68] Xu Zhong, Elaheh ShafieiBavani, and Antonio Jimeno Yepes. 2020. Image-based table recognition: data, model, and evaluation. In *European conference on computer vision*. Springer, 564–580. doi:10.1007/978-3-030-58589-1_34

[69] Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. 2019. Publaynet: largest dataset ever for document layout analysis. In *2019 International conference on document analysis and recognition (ICDAR)*. IEEE, 1015–1022. doi:10.1109/ICDAR.2019.00166