

PAPER2WEB: LET’S MAKE YOUR PAPER ALIVE!

Yuhang Chen^{1*}, Tianpeng Lv^{1*}, Siyi Zhang¹, Yixiang Yin¹, Yao Wan^{1†},
Philip S. Yu², Dongping Chen^{3‡}

¹ONE Lab, Huazhong University of Science and Technology,

²University of Illinois Chicago, ³University of Maryland

{u202315752, wanyao}@hust.edu.cn, dongping@umd.edu

* Equal Contribution. † Corresponding author. ‡ Project Lead.

🌐 Project Website: https://francischen3.github.io/P2W_Website

🔗 Code Repository: <https://github.com/YuhangChen1/Paper2All>

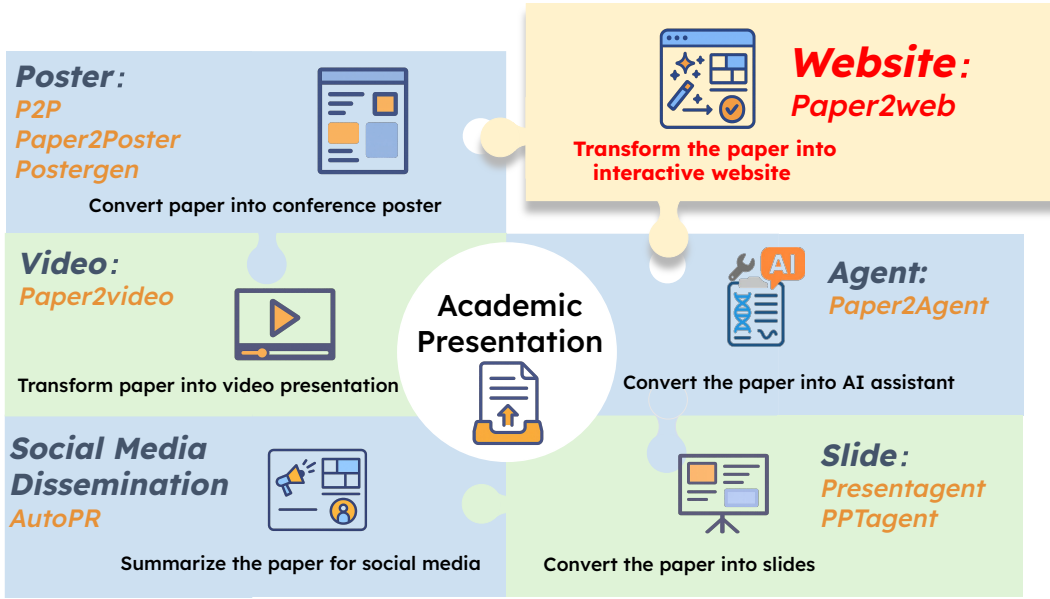


Figure 1: Our work, PAPER2WEB, constitutes an important piece of the puzzle for the presentation and dissemination of academic papers. We build a unified platform to streamline all academic presentation at Paper2All.

ABSTRACT

Academic project websites can more effectively disseminate research when they clearly present core content and enable intuitive navigation and interaction. However, current approaches such as direct *Large Language Model* (LLM) generation, templates, or direct HTML conversion struggle to produce layout-aware, interactive sites, and a comprehensive evaluation suite for this task has been lacking. In this paper, we introduce PAPER2WEB, a benchmark dataset and multi-dimensional evaluation framework for assessing academic webpage generation. It incorporates rule-based metrics like *Connectivity*, *Completeness* and human-verified LLM-as-a-Judge (covering interactivity, aesthetics, and informativeness), and PaperQuiz, which measures paper-level knowledge retention. We further present PWAGENT, an autonomous pipeline that converts scientific papers into interactive and multimedia-rich academic homepages. The agent iteratively refines both content and layout through MCP tools that enhance emphasis, balance, and presentation quality. Our experiments show that PWAGENT consistently outperforms end-to-end baselines like template-based webpages and arXiv/alphaXiv versions by a large margin while maintaining low cost, achieving the Pareto-front in academic webpage generation.

1 INTRODUCTION

Research papers are predominantly distributed in PDF format, conveying information solely through static text and images (Tkaczyk et al., 2015; Li et al., 2020; Clark & Divvala, 2016; Lo et al., 2020). However, PDFs offer limited support for interactivity and multimedia content (W3C Web Accessibility Initiative, 2018; Government Digital Service and Central Digital and Data Office, 2024; NHS Digital, 2025; Kumar & Wang, 2024), resulting in substantial information loss during dissemination (Tkaczyk et al., 2015; Li et al., 2020). As a result, transforming academic papers into more visual and accessible formats has emerged as a promising direction for enhancing scholarly communication and accelerating knowledge dissemination (Fischhoff, 2013; Thorlacius, 2007).

Recently, growing efforts have sought richer and more efficient ways to transform scholarly articles—such as converting papers into concise posters with Paper2Poster (Pang et al., 2025), presentation slides with PresentAgent (Shi et al., 2025), videos with Paper2Video (Zhu et al., 2025), public-facing content with AutoPR (Chen et al., 2025a). However, these approaches either discard the fine-grained details present in the original text or retain only the main ideas while overlooking the communicative advantages of multimedia content such as videos and animated graphics. This creates a gap for formats that preserve core textual knowledge while seamlessly integrating multimedia to enhance scientific communication across diverse communities.

Compared with the above methods, an online web page can integrate textual content with multimedia and present in a coordinated and navigable manner. As illustrated in Figure 6, well-designed webpages can bridge the gap between scholarly content and interactive digital presentation, thereby enabling broader and more effective dissemination of research outcomes. However, this poses challenges in requiring deliberate spatial organization to accommodate rich media and interactive components. Recent efforts have explored converting full academic papers into web pages to broaden accessibility and dissemination. The arXiv HTML initiative (Frankston et al., 2024) is one representative example, yet such approaches often produce disordered layouts and redundant text, reducing readability, precision, and cross-device accessibility. As illustrated in Figure 2, common failure modes include rigid figure grids with inconsistent scaling, detached captions, missing responsiveness, and limited interactivity. AlphaXiv leverages LLMs for content condensation and layout optimization, yet it still limits author control over multimedia placement and visual design, resulting in largely static presentations that fail to fully exploit interactive capabilities. As noted by prior work (Frankston et al., 2024), these issues stem from TeX–HTML pipelines that emulate LaTeX behavior without executing a full TeX engine, leading to missing structures and visual inconsistencies. On the other hand, directly LLM-driven webpage generation also struggles to process long contexts (Liu et al., 2024b; Hsieh et al., 2024) and to effectively integrate multimedia content while maintaining robust interactivity (Xiao et al., 2024a).

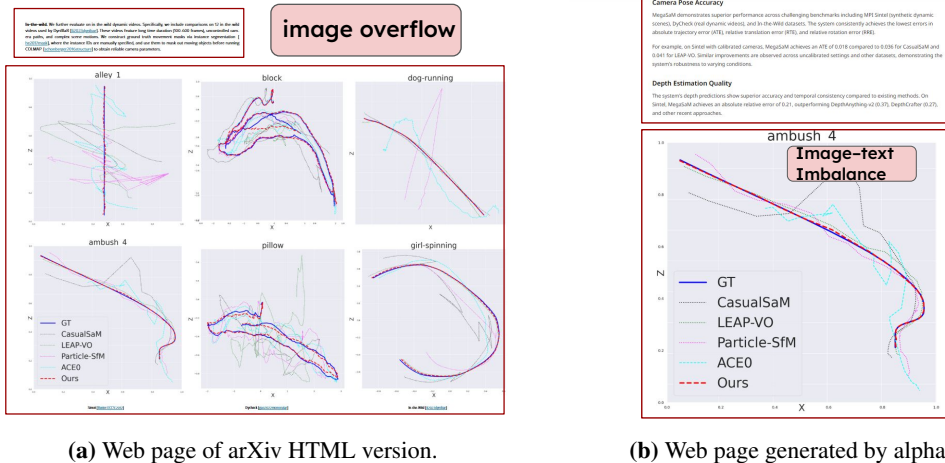


Figure 2: Problems in current scholar web page generation, including distorted layout and limited interactivity.

Our Work: PAPER2WEB. In this paper, we introduce PAPER2WEB, a new task that aims to transform full academic papers into interactive websites that preserve core content while integrating multimedia and improving usability. We begin by constructing a dataset of paired academic papers and their corresponding webpages. Specifically, we crawl accepted papers from selected conferences, parse their texts to extract reliable metadata such as authorship, and augment each record with citation counts from Semantic Scholar. We then perform multi-stage filtering: using cues in the paper body and related code repositories to locate candidate project homepages, employing an LLM to assess their relevance, and relying on human annotators to resolve ambiguous cases. This pipeline yields 10,700 papers with verified homepages, forming the basis for our analysis of effective research websites.

To address these limitations, we propose PWAGENT, a multi-agent framework for transforming scholarly documents into structured and interactive web content. PWAGENT first decompose a paper into structured assets and organize links and executable artifacts under a unified schema. It then performs *Model Context Protocol* (MCP) ingestion to construct a semantically aligned resource repository enriched with relational metadata and exposed through standardized tools for downstream use. A content-aware allocation heuristic estimates each asset’s spatial footprint and assigns provisional layout budgets to guide rendering and navigation. Finally, agent-driven iterative refinement drafts an initial website, inspects rendered views, and issues targeted edits via tool calls to correct visual imbalance, enhance information hierarchy, and appropriately anchor multimedia elements. This loop alternates between global assessment and localized adjustment, linking segmented screenshots to corresponding HTML fragments for precise editing.

Using the PAPER2WEB dataset, we also construct a benchmark for PAPER2WEB. We introduce the first metric to measure the interactivity and dynamic elements of the webpage, as well as *Connectivity* and *Completeness*, human-assisted *MLLM-as-a-Judge* for comprehensive assessments. Furthermore, we propose PaperQuiz to evaluate knowledge transfer from webpage screenshots through both verbatim and interpretive questions, incorporating a verbosity penalty to discourage overly text-heavy pages. On this benchmark, PWAGENT improves connectivity and completeness by roughly 12% on average across methods, achieving a 28% gain over the arXiv HTML baseline. It also yields an 18% average improvement via MLLM-as-a-Judge and **triples** the average score of the strongest end-to-end baseline and remains competitive with template-assisted variants.

Contributions. The key contributions of this paper are as follows:

- **A New Task, Dataset and Evaluation Suite.** We build the PAPER2WEB dataset, a large-scale corpus that links scientific papers to their corresponding project homepages, enabling quantitative analysis of web-based academic dissemination.
- **Comprehensive Benchmark.** We establish a benchmark with autonomous metrics aligned well with human preference to comprehensively assess the quality of web page generation, reveal problems within current automatic webpage generation methods.
- **A State-of-the-Art Automatic Approach.** We propose PWAGENT, a MCP-based agent for the end-to-end transformation of academic papers into structured, interactive pages.

2 PAPER2WEB: A NEW TASK AND DATASET

Since no dataset exists for analyzing academic website content and layout, we collect data from recent AI papers. We harvest project links from papers and code repositories, then crawl the corresponding webpage. Finally, we collect a comprehensive dataset covering multiple conferences and

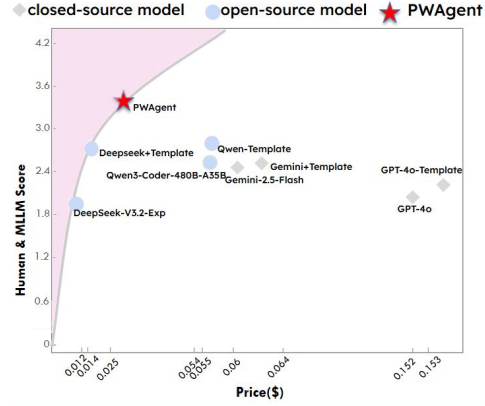


Figure 3: Pareto-front comparison of each website generation methods. Our PWAGENT achieve the highest quality with moderate and affordable cost.

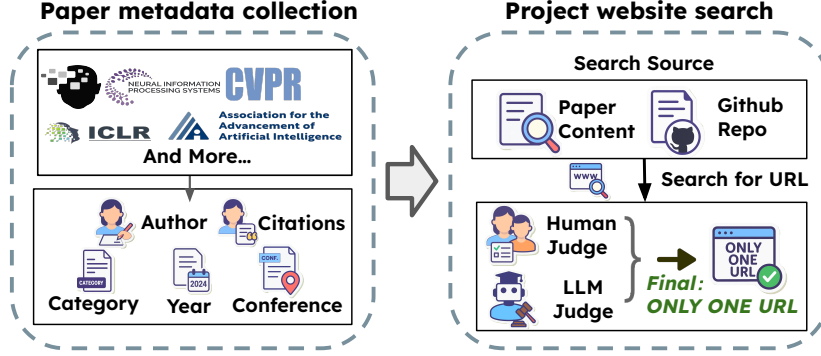


Figure 4: To transform static papers into exploratory web pages, we collect the first paper-webpage dataset by crawling across multiple top-tier conferences and filtering by online search and human annotators.

categories with 10,716 papers and their human-created project homepages. Figure 4 presents our data collection pipeline.

2.1 DATA COLLECTION

Paper Metadata Collection. We focus on AI papers as they are recent, peer-reviewed, cover diverse subfields with varied modalities, and attract attention that motivates high-quality dissemination. Using automated tools, we collect papers from major AI conferences (ICML, NeurIPS, WWW, ICLR, etc., 2020-2025). We extract source links, parse full texts for metadata (title, authors, venue, year), and retrieve citation counts from Semantic Scholar. Each paper’s introduction is submitted to an LLM that assigns one of thirteen topical categories (Figure 6, right panel), enabling standardized cross-paper analysis.

Project Website Search. Our pipeline retrieves external links from each paper and its code repository, scanning the paper body and README files. We parse local context around each link, crawl the target HTML, and use an LLM to analyze the content. Human reviewers resolve ambiguous cases to ensure each paper maps to at most one canonical project website. Papers lacking relevant links in either source are defined as having no project homepage.

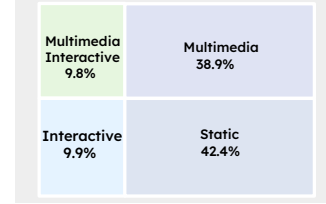


Figure 5: PAPER2WEB dataset statistics.

2.2 DATA CHARACTERISTICS

Finally, we curate a comprehensive dataset comprising 10,716 papers with human-created project homepages and 85,843 without. We group papers into 13 categories following ICML/NeurIPS/ICLR conference taxonomies. The right panel of Figure 6 shows computer vision has the strongest demand for project websites, with homepage adoption rising steadily in recent years. To characterize webpage features, we manually audited 2,000 samples. We define interactive sites as pages with dynamic behaviors and explorable components responding to user intent; multimedia pages as those embedding rich media like videos; and static sites as pages delivering primarily text and still images in linear presentation. Figure 5 shows the distribution by feature set. While many pages remain static, multimedia dissemination through embedded videos and animations is notable. Interactive capabilities that enhance user experience remain comparatively rare and unevenly implemented, representing the first systematic characterization of interactive behavior and multimedia orchestration in academic webpages.

3 EVALUATION METRICS

To systematically assess the quality of generated academic web pages, we introduce the PAPER2WEB benchmark. This comprehensive metric suite centers on the dual principles of information efficiency and a balanced text–visual composition. The framework evaluates web pages across three key dimensions: (1) Connectivity & Completeness, (2) Holistic Evaluation via

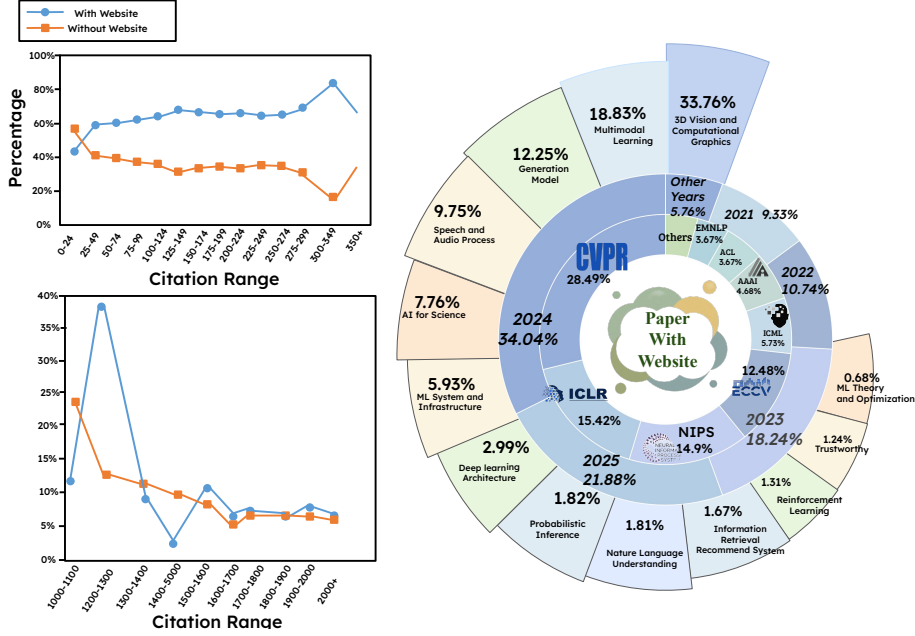


Figure 6: The right panel shows the categorization of our data. We divided the dataset into 13 categories and counted items in each. In addition, we show distributions by conference and by year. The top-left panel presents, for each category, the relative proportions of papers without and with a website among papers with low citation counts. The bottom-left panel depicts the distribution of papers without and with a website restricted to highly cited papers (those with over 1,000 citations).

Human/MLLM-as-a-Judge, and (3) PaperQuiz, which measures how effectively the website transfers knowledge.

3.1 CONNECTIVITY & COMPLETENESS

This metric jointly evaluates the hyperlink quality and structural fidelity of generated web pages. Both indicators are assessed through LLM analysis of the HTML source code, supplemented by human evaluation for reliability. For connectivity, we examine how effectively the webpage links internal and external resources to support coherent navigation and information access. To reduce evaluation bias, a dedicated URL parser is employed to count and verify valid hyperlinks, ensuring objective measurement of link quality. For completeness, we measure how well the generated webpage reproduces the core sections of the source paper. To enhance consistency, two quantitative priors, image–text balance and information efficiency, are applied to further evaluate structural integrity and content compactness.

Image–Text Balance Prior. Let D denote the weighted deviation between the observed image–text ratio and the ideal 1:1 balance, and let $\gamma > 0$ be a scaling factor (Pang et al., 2025). We define the penalty term and score as:

$$\zeta = \frac{5}{1 + \gamma \cdot D}, \quad S_{\text{img-txt}} = 5 - \zeta. \quad (1)$$

Information Efficiency Prior. To encourage concise, information-dense presentation, let $r = L/W$ denote the ratio between the generated text length L and the median human-designed length W , with $\beta > 0$ a scaling factor (e.g., $\beta=0.6$) (Tufte & Graves-Morris, 1983). We define the efficiency as:

$$p(r) = \frac{5}{1 + \beta \cdot \max(0, r - 1)}. \quad (2)$$

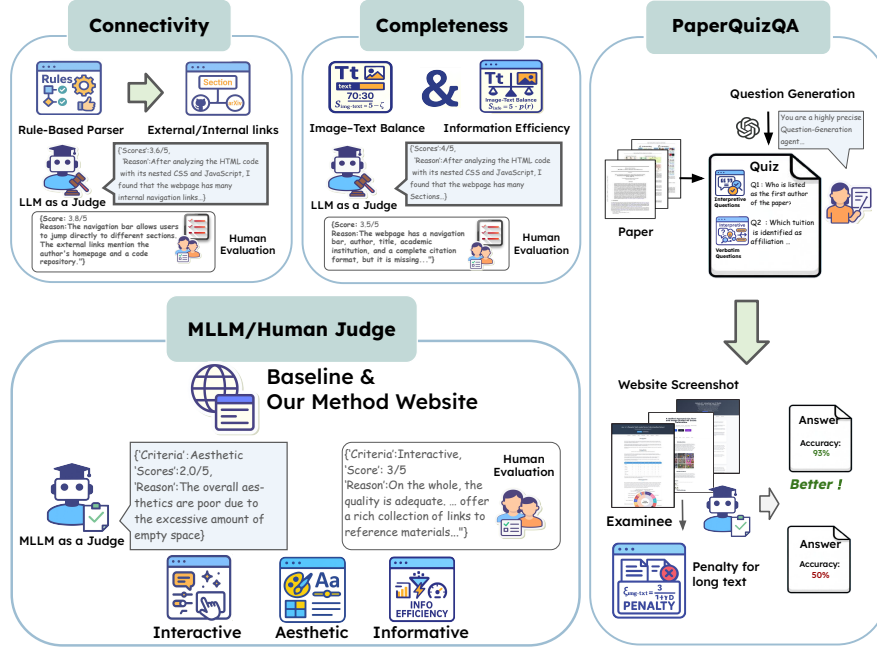


Figure 7: Our evaluation metrics include multiple modules: (1) *Connectivity* and *Completeness* by parsing HTML links and structure with image-text balance and information-efficiency priors, (2) an MLLM/Human Judge to rate interactivity, aesthetics, and informativeness in a holistic manner, and (3) a QA PaperQuiz on webpage screenshots with a verbosity penalty.

3.2 HOLISTIC EVALUATION WITH HUMAN-VERIFIED MLLM-AS-A-JUDGE

To evaluate the overall effectiveness of web pages at a holistic level, we employ a MLLM as an automated judge, combined with human verification to mitigate bias. The model outputs a quantitative score ranging from 1 to 5 for each webpage. Specifically, it evaluates three key dimensions: *Interactive*, which measures element responsiveness, saliency emphasis, and overall usability; *Aesthetic*, which assesses element quality, layout balance, and visual appeal; and *Informative*, which evaluates the clarity and logical coherence of webpage content. See Appendix B for scoring guidelines.

3.3 PAPERQUIZ

Inspired by Paper2Poster (Pang et al., 2025), we focus on the academic web page and acknowledge its central role in communicating research as a dynamic bridge between authors and a broader audience. Therefore, we design an evaluation protocol that simulates this knowledge-transfer scenario. We first employ an LLM as an examiner to generate a comprehensive set of 50 questions from the source paper. These questions are divided into two types: 25 Verbatim questions, which are directly answerable from specific text, figures, or tables on the webpage, and 25 Interpretive questions, which require a higher-level comprehension of the paper’s core contributions, methodology, and results. In the second stage, we present a screenshot of the rendered webpage to a diverse panel of MLLMs (including both open and closed source models). These models are tasked with answering the quiz based solely on the provided webpage content. By comparing the quiz scores across different generated web pages, we can quantitatively assess which one most effectively conveys the original paper’s essential information. To prevent high scores resulting from excessive text transfer, we introduce a penalty term ζ , defined in Eq. 1, to discount for verbosity.

4 PWAGENT: A STRONG BASELINE

To address the core challenges of the PAPER2WEB, we introduce PWAGENT, an automated pipeline for converting scientific papers into project homepages. The core of our approach involves parsing the paper’s content into a structured format managed by an MCP server (Hou et al., 2025; Ehtesham et al., 2025; Krishnan, 2025). This server encapsulates key paper assets, along with predefined

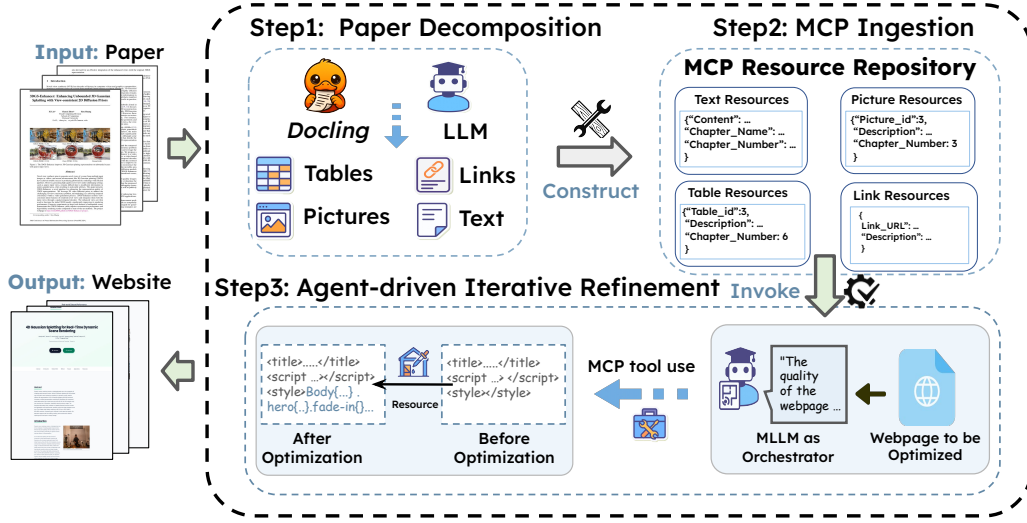


Figure 8: PWAGENT turns papers into interactive and multimedia-rich project homepages. Papers are deconstructed via Docling/Marker + LLM into multiple assets and stored in an MCP repository. An agent drafts a page, then iteratively optimizes until layout and UX are solid.

prompts for webpage generation and stylistic refinement, organizing them into a centralized resource repository. During this process, the agent leverages the tool-use capabilities of the MCP to access the resource repository, enabling a continuous optimization loop. The overall process includes the following key stages: (1) *Paper Decomposition*, which isolates key contributions from the paper. (2) *MCP Ingestion*, which encapsulates these contributions as a resource repository managed by the MCP server. (3) *Agent-driven Iterative Refinement*, which connects the MCP server to LLM-based agents that autonomously perform content matching and optimization through tool calls.

4.1 PAPER DECOMPOSITION

We first deconstruct an unstructured scientific paper into structured intellectual assets that populate the MCP Resource Repository. Starting from the source PDF, the document is converted to Markdown using tools such as MARKER¹ or DOCLING². An LLM then performs semantic decomposition that extracts metadata, reconstruct tables, and model detailed page layout and reading order, yielding a machine-readable representation like JSON or Markdown that captures the paper’s key contributions.

Instead of summarizing, the LLM analyzes the Markdown text against a predefined schema to identify, isolate, and organize the paper’s key assets. These assets fall into three categories: (1) *Textual Assets*: each logical section is represented as a distinct resource object containing its title, LLM-generated synopsis, full text, and metadata; (2) *Visual Assets*: figures and tables are extracted as images and linked to their original captions, labels, and textual references to preserve context; and (3) *Link Assets*: external URLs and internal citations are systematically captured and categorized to provide structured access to supplementary materials and related work.

4.2 MCP INGESTION

Here we apply the MCP to the task of transforming scholarly papers into structured, queryable resources. We first instantiate a fully instrumented MCP server, which converts static assets into queryable resources with stable IDs and standardized tool access points. The server is responsible for resource construction, materializing assets with relational metadata and provisional layout budgets, and for tool registration, exposing a minimal, consistent API for downstream retrieval, composition, and editing.

¹<https://github.com/datalab-to/marker>

²<https://github.com/docling-project/docling>

We enrich the parsed outputs with cross-modal semantics: **(1)** An LLM is used to align each visual element with its most relevant textual description and adds back-references to the citing paragraphs. **(2)** Link assets are typed by function to support structured cross-references. To achieve a coherent visual presentation, a content-aware spatial allocation heuristic estimates each asset’s footprint and assigns a proportional layout budget to balance visual density across the page.

These enriched records are then committed to MCP server as MCP Resource Repository, where each resource is stored with a unique rid and fields for grounding and navigation. Concretely, the text resource stores the full paragraph and an LLM-generated synopsis; a Visual resource stores the image and its caption; and a Link resource stores the URL, its semantic role, and a short descriptor. Together, these resources form a structured, cross-referenced repository that serves as the foundation for webpage synthesis. Finally, the server registers a compact tool suite that provides enumeration of resource IDs, access to grounded content and metadata for rendering, typed references for connectivity-aware placement, and initial layout allocation. This lightweight yet expressive interface is sufficient to synthesize a well-grounded HTML first draft for subsequent refinement by the multi-agent workflow.

4.3 AGENT-DRIVEN ITERATIVE REFINEMENT

Finally, we propose an agent-driven iterative refinement mechanism to progressively enhance the layout, visual coherence, and semantic alignment of generated webpages. The process begins with initial page generation, where the agent retrieves essential metadata and relevant assets from the resource repository using MCP tools. Based on this information, it rapidly constructs a foundational webpage that serves as the baseline for subsequent refinement.

Following initialization, the system enters an iterative refinement loop that continues until no further corrective actions are needed or a predefined iteration limit is reached. At its core is an MLLM acting as the *Orchestrator Agent*, which conducts holistic visual assessments of the rendered webpage and invokes MCP tools to fix detected flaws. To address complex layout and visual consistency issues, the Orchestrator performs joint global–local reasoning and coordinates targeted optimizations through tool calls. To reduce hallucinations during long-range reasoning, the agent segments the rendered page into independent visual tiles linked to their corresponding HTML fragments, sequentially analyzing each to detect imbalances and misalignments and propose precise edits. After each round of local refinement, adjacent tiles are merged, borrowing the spirit of merge sort. Therefore, neighboring regions can be jointly optimized by integrating their HTML and imagery. This aggregation allows the MLLM to capture inter-section dependencies and prevent visual artifacts such as overflow, occlusion, or cross-section drift. Finally, the Orchestrator performs a global pass to assess overall content completeness and visual harmony, realizing a part-to-whole optimization path that further mitigates hallucinations. The process terminates once optimization is complete or the maximum refinement cycles are reached.

5 HOW PWAGENT MAKE PAPER ALIVE?

5.1 EXPERIMENT SETUPS

We evaluate four distinct baseline methodologies to rigorously assess the performance of our proposed approach. These serve as crucial benchmarks for gauging information dissemination efficacy and human-centered friendliness. **(1) Oracle Method**, original websites created by authors. They serve as the gold standard for optimal presentation and content delivery; **(2) End-to-End Generation**, where GPT-4o, *Gemini-2.5-Flash* (Gemini), *DeepSeek-V3.2-Exp* (DeepSeek) and *Qwen3-Coder-480B-A35B* (Qwen) generate websites either through text-based rendering from scratch or by adapting the widely adopted Nerfies academic website template (Park et al., 2021) (The above models combined with a template will be referred to respectively as GPT-4o-Template, Gemini-Template, DeepSeek-Template, and Qwen-Template); **(3) Existing HTML Versions**, where research papers from arXiv and alphaXiv provide public HTML versions, we scrape their screenshots and source code, noting some lack official web formats; **(4) PWAGENT (Our)**, where Qwen3-30B-A3B is responsible for paper deconstruction and MCP ingestion, while the Orchestrator Agent is powered by the Qwen2.5-VL-32B model.

Table 1: Detailed comparison between PAPER2WEB and other baselines across *Completeness*, *Connectivity* and holistic MLLM evaluation.

Methods	Connectiveness				Completeness				Holistic Evaluation								
									Interactive			Aesthetic			Informative		
	Rule ↑	LLM ↑	Human ↑	Avg. ↑	Rule ↑	LLM ↑	Human ↑	Avg. ↑	MLLM ↑	Human ↑	Avg. ↑	MLLM ↑	Human ↑	Avg. ↑	MLLM ↑	Human ↑	Avg. ↑
Original Website	3.20	3.47	2.99	3.22	3.17	3.93	4.00	3.70	1.70	3.37	2.54	3.14	3.63	3.39	4.49	3.86	4.18
<i>Model end-to-end methods</i>																	
GPT-4o	1.81	2.07	2.05	1.98	2.11	3.15	3.43	2.90	0.53	1.85	1.19	2.61	2.13	2.37	2.01	2.56	2.29
Gemini-2.5-flash	2.26	2.11	2.16	2.18	2.72	3.56	3.43	3.24	1.30	2.15	1.73	2.80	2.41	2.61	3.63	2.68	3.16
DeepSeek-V3.2-Exp	1.83	2.09	2.16	2.03	2.09	3.21	3.51	2.94	0.54	2.01	1.28	2.63	2.20	2.42	2.21	2.61	2.41
Qwen3-Coder-480B-A35B	2.52	3.05	2.82	2.80	2.79	3.58	3.62	3.33	1.44	2.43	1.94	2.74	2.49	2.62	3.92	2.81	3.37
<i>Model end-to-end methods + Template</i>																	
GPT-4o-Template	1.83	2.26	2.77	2.29	2.25	3.37	3.54	3.05	0.56	1.47	1.02	2.63	2.35	2.49	3.87	2.58	3.23
Gemini-Template	2.47	2.87	2.78	2.71	2.73	3.72	3.78	3.41	1.47	1.58	1.53	2.75	2.46	2.61	4.28	2.67	3.48
DeepSeek-Template	2.38	2.91	2.80	2.70	2.75	3.68	3.84	3.42	1.45	1.60	1.53	2.74	2.46	2.60	4.26	2.67	3.47
Qwen-Template	3.01	3.21	2.87	3.03	2.88	3.90	3.80	3.53	1.47	1.58	1.53	2.77	2.93	2.85	4.31	3.22	3.77
<i>Automated generation methods</i>																	
arXiv (HTML)	3.70	2.23	1.34	2.42	2.49	3.81	3.75	3.35	1.05	1.51	1.28	2.72	2.65	2.69	4.01	3.06	3.54
alphaXiv	3.43	3.01	2.91	3.12	2.88	3.95	3.85	3.56	1.25	1.61	1.43	2.73	2.80	2.77	4.20	3.46	3.83
PWAGENT (Our)	3.06	3.30	2.94	3.10	2.91	4.02	3.86	3.56	1.39	3.16	2.28	2.82	3.35	3.09	4.31	3.56	3.93

Table 2: PaperQuiz evaluation on the PAPER2WEB, based on open and closed-source MLLMs. The evaluation metrics include Raw Score and Score with Penalty under two settings: “*Verbatim*” and “*Interpretive*”.

Methods	Verbatim			Interpretive			Avg	Score with Penalty			
	open-source ↑	closed-source ↑	V-Avg ↑	open-source ↑	closed-source ↑	I-Avg ↑		Penalty ↓	V_avg ↓	I_avg ↓	Avg ↑
Original Website	2.94	2.14	2.54	3.81	3.09	3.45	3.00	1.43	1.11	2.02	1.57
<i>Model end-to-end methods</i>											
GPT-4o	2.53	1.46	1.99	3.38	2.32	2.85	2.42	3.03	-0.93	-0.18	-0.56
Gemini-2.5-flash	2.60	1.59	2.10	3.14	2.72	2.93	2.52	2.18	-0.19	0.71	0.24
DeepSeek-V3.2-Exp	2.55	1.54	2.00	3.21	2.55	2.88	2.44	2.26	-0.26	0.62	0.18
Qwen3-Coder-480B-A35B	2.65	1.64	2.15	3.22	3.02	3.12	2.64	2.12	0.03	1.00	0.52
<i>Model end-to-end methods + Template</i>											
GPT-4o-Template	2.58	1.42	2.00	3.48	2.25	2.87	2.43	2.50	-0.50	0.37	-0.07
Gemini-Template	3.62	3.36	3.49	4.40	4.45	4.42	3.96	2.01	1.48	2.41	1.95
DeepSeek-Template	3.55	3.19	3.37	4.11	4.25	4.18	3.78	1.96	1.41	2.22	1.82
Qwen-Template	3.70	3.44	3.57	4.52	4.41	4.47	4.02	2.00	1.57	2.47	2.02
<i>Automated generation methods</i>											
arXiv (HTML)	3.62	3.42	3.52	4.52	4.43	4.47	4.00	2.87	0.65	1.60	1.13
alphaXiv	3.57	3.60	3.58	4.58	4.54	4.56	4.07	1.97	1.61	2.59	2.10
PWAGENT (Our)	3.76	3.42	3.59	4.56	4.40	4.48	4.04	2.00	1.59	2.48	2.03

5.2 MAIN RESULTS

Completeness & Connectivity. As shown in the left half of Table 1, we evaluate website completeness and connectivity. arXiv-HTML attains high rule-based connectivity but receives 64% lower human ratings, as it indiscriminately converts every citation into links, inflating metric scores while degrading user experience. alphaXiv shows balanced connectivity by selectively surfacing important links. For completeness, arXiv-HTML preserves verbose text with few images, scoring well with LLM and human judges but poorly on rule-based metrics. In contrast, our PWAGENT achieves 2% higher LLM-judged completeness than ground truth, demonstrating superior content condensation and balanced layout of text, images, and links. These findings reveal that code-based metrics miss real user experience, motivating our user-centered evaluation next.

Holistic Evaluation. As shown in the right half of Table 1, our PWAGENT achieves highest scores across all dimensions. While alphaXiv performs well in completeness and connectivity, it lacks interactive components, scoring 37% lower than our method in interactivity. Template-based methods effectively guide layout but constrain interactive element generation. Overall, PWAGENT outperforms all generation methods, achieving 91% of ground truth quality in aesthetics and 94% in informativeness, with a 59% improvement in interactivity over alphaXiv.

PaperQuiz. As shown in Table 2, we observe: (1) Without the conciseness penalty, arXiv-HTML scores strongly; once applied, both arXiv-HTML and end-to-end GPT-4o receive large deductions, highlighting the value of concise, engineered sites and supporting website generation as effective context compression. (2) Gemini and Qwen are strong and generally outperform GPT-4o and DeepSeek; templates lift all models—DeepSeek-Template nears Gemini-Template, and Qwen-Template approaches the ground-truth site. (3) Across methods, open-source reader models con-

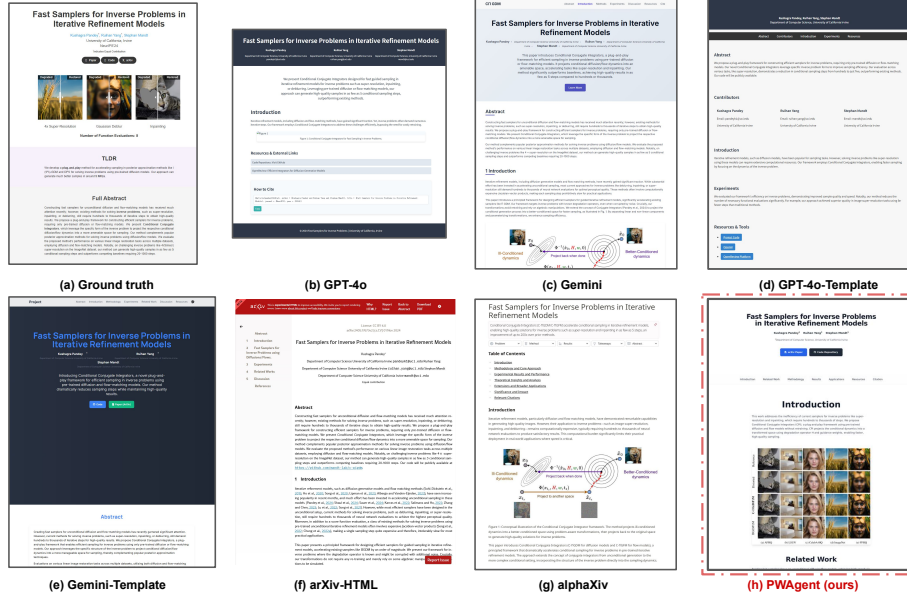


Figure 9: Illustration of website variants for the paper generated by different methods. GPT-4o fails to cover all components of a paper and amounts to only a simple paradigm; even with template, the sections remain incomplet. The arXiv-HTML is content-rich but is essentially a direct transfer of the original. The alphaXiv method is complete and concise in content, but it lacks a layout paradigm and visual aesthetic quality. Our PWAGENT show interactive and rich multimedia content to enrich presentation quality.

sistently beat closed-source ones, indicating some open-source MLLMs (e.g., Qwen) can match or exceed closed models on certain visual tasks. (4) PWAGENT achieves best or near-best results across tasks and models, with total information coverage rivaling arXiv-HTML; after the penalty, it still attains the highest overall score. (5) PWAGENT’s penalty remains nontrivial, and the ground-truth site scores lower than expected, likely because it includes many videos and animations; in practice, authors can start from PWAGENT and add multimedia to reach the most desirable design.

5.3 IN-DEPTH ANALYSIS

Efficiency Analysis. Figure 3 presents the average token cost per website. Our PWAGENT is highly token-efficient, requiring only \$0.025 to produce a high-quality academic page. By contrast, end-to-end methods are costlier: GPT-4o is about \$0.141 and Gemini about \$0.054 per website. This yields 82% and 54% cost reductions, respectively, while maintaining strong page quality and usability. Even template-aided open models around \$0.069 remain $2.8\times$ more expensive, yet offer no clear advantage. Overall, PWAGENT delivers *state-of-the-art* cost efficiency with high presentation quality.

Case Study. In Figure 9 and 10, we present a qualitative comparison of different website baselines for a paper. GPT-4o evidently struggles to generate a structurally coherent HTML webpage from the source PDF, and its content completeness remains poor even when provided with a template. In contrast, the website generated by Gemini appears content-rich at first glance, and its internal structure is significantly improved with a template. However, it suffers from an unbalanced image-to-text ratio with very few visuals, which hinders the reader’s ability to systematically understand the project. The official arXiv-HTML page, while comprehensive, is overly verbose. Although the alphaXiv website is well-illustrated with both images and text, its design is monotonous and lacks aesthetic appeal. In contrast, our PWAGENT not only preserves the structural integrity of the original paper but also achieves a well-balanced image-to-text ratio. Furthermore, it offers versatile styling and superior aesthetic quality. However, there is still room for improvement when compared to the human-designed version.

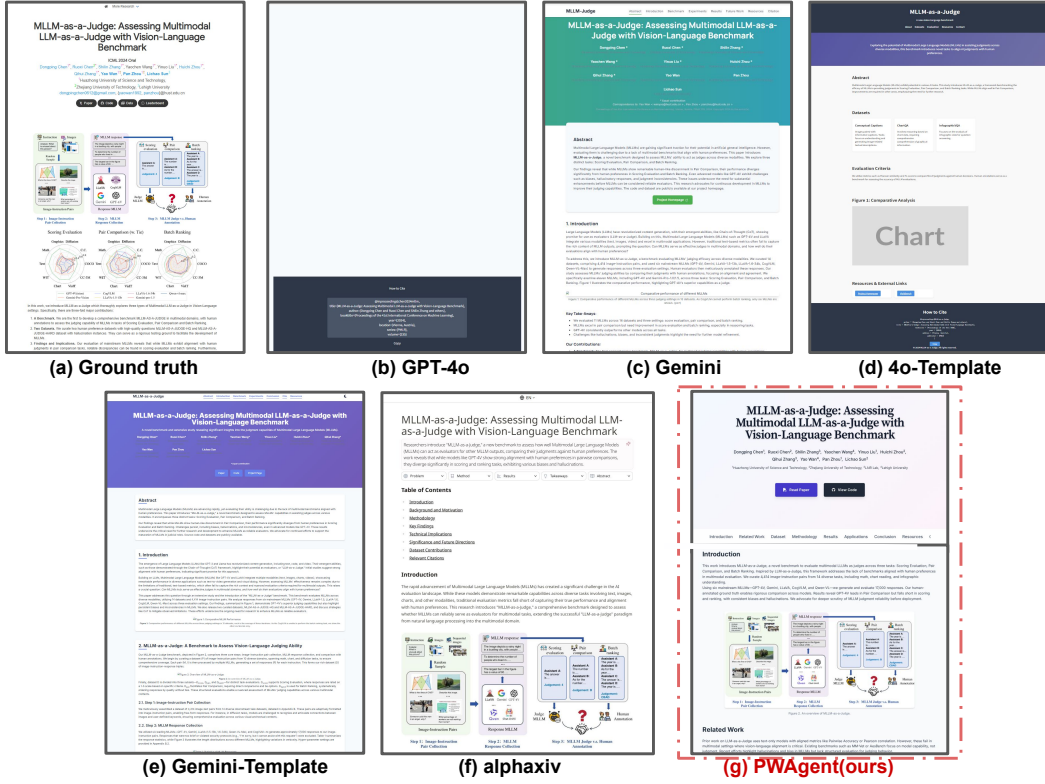


Figure 10: Illustration of website variants for the paper “MLLM-as-a-Judge: Assessing Multimodal LLM-as-a-Judge with Vision-Language Benchmark”³ generated by different methods.

6 RELATED WORK

HTML Code Generation. The field of automated front-end development has seen significant progress, with a primary focus on generating HTML from diverse inputs like screenshots, design prototypes, and natural language descriptions. This research has led to the establishment of several key benchmarks, including Design2Code (Si et al., 2024; Yang et al., 2025), Websight (Laurençon et al., 2024) and WebCode2M (Gui et al., 2025a). A variety of code generation strategies have been explored, ranging from direct translation to more structured approaches, such as the divide-and-conquer strategy of DCGen (Wan et al., 2024b) and the hierarchical generation process used by UICopilot (Gui et al., 2025b). These technologies have been applied to create mobile UIs (Xiao et al., 2024a; Zhou et al., 2024), multi-page websites (Wan et al., 2024a), and enhance web design (Xiao et al., 2024b; Li et al., 2024; Zhang et al., 2024), with performance often improved through model fine-tuning (Liang et al., 2024). More recently, multi-agent systems are being increasingly adopted for complex development tasks (Han et al., 2024; Liu et al., 2024a). For example, agentic workflows are now used to convert designs into functional code (Islam et al., 2024; Ding et al., 2025), and some systems assign distinct agents to specific sub-tasks, refining their output through iterative human feedback (Wang et al., 2024b).

Automated Processing of Scholarly Articles. Early methods for generating derivative content from academic papers primarily relied on template-based (Xu & Wan, 2021; Qiang et al., 2019; Cheng et al., 2024) or rule-driven models (Huang et al., 2022; Lin et al., 2023). Recently, with the maturation of AI agent technology, a substantial body of work has emerged for academic poster generation. A series of methods and benchmarks, including P2P (Sun et al., 2025), Paper-to-Poster (Pang et al., 2025), PosterGen (Zhang et al., 2025c), CreatiDesign (Zhang et al., 2025a), PosterCraft (Chen et al., 2025b), and DreamPoster (Hu et al., 2025), have explored pipelines for automatically converting papers into posters. These studies demonstrate that through well-designed multi-agent collaboration, the generated posters can achieve high fidelity with human-designed counterparts in terms of layout,

³<https://mllm-judge.github.io/>

content summarization, and visual aesthetics. Similarly, notable progress has been made in presentation slide generation. PresentAgent (Shi et al., 2025), Preacher (Liu et al., 2025), SciGA (Kawada et al., 2025), and SlideCode (Tang et al., 2025) introduce specialized datasets, benchmarks, and methodologies. The trend in these task-specific applications is gradually evolving towards broader automated visual design, as exemplified by systems like BannerAgency (Wang et al., 2025) for banner creation and VideoAgent (Wang et al., 2024a; Fan et al., 2024; Soni et al., 2024) for video production. With the advent of the MCP, researchers have begun to utilize MCP to empower agents for more sophisticated tasks. A prominent example is Paper2Agent (Miao et al., 2025), which underscores the potent capabilities of advanced agent systems in handling complex, unstructured academic information.

7 CONCLUSION AND DISCUSSION

We introduce PAPER2WEB, a novel task and benchmark for generating project homepages from academic papers, and identify key challenges faced by current generative models and automated methods in handling long-context and layout-sensitive tasks. Our framework, PWAGENT narrows the gap between machine- and human-designed webpages and sets a new efficiency standard for web-based scholarly communication, offering a practical and scalable solution.

While our work represents an initial step toward transforming static papers into exploratory web pages, it primarily aims to define the scope and standards of this emerging area rather than offer a definitive solution. We also propose simple yet multi-dimensional evaluation criteria that lay the groundwork for richer future assessments. Nonetheless, evaluating how multimedia elements contribute to effective academic communication remains an open challenge, which we plan to address through more robust agentic workflows and comprehensive evaluation methods in future work. We call for continued research on integrating multi-agent reasoning and multimodal understanding to advance the transformation of scholarly communication beyond static formats.

ACKNOWLEDGMENT

We thank Gui Yi and Jianuo Huang from Huazhong University of Science and Technology for their valuable input and feedback.

REFERENCES

- Qiguang Chen, Zheng Yan, Mingda Yang, Libo Qin, Yixin Yuan, Hanjing Li, Jinhao Liu, Yiyan Ji, Dengyun Peng, Jiannan Guan, Mengkang Hu, Yantao Du, and Wanxiang Che. Autopr: Let’s automate your academic promotion!, 2025a.
- SiXiang Chen, Jianyu Lai, Jialin Gao, Tian Ye, Haoyu Chen, Hengyu Shi, Shitong Shao, Yunlong Lin, Song Fei, Zhaoxu Xing, et al. Postercraft: Rethinking high-quality aesthetic poster generation in a unified framework. *arXiv preprint arXiv:2506.10741*, 2025b.
- Xianfu Cheng, Weixiao Zhou, Xiang Li, Jian Yang, Hang Zhang, Tao Sun, Wei Zhang, Yuying Mai, Tongliang Li, Xiaoming Chen, et al. Svipt: Fast and efficient scene text recognition with vision permutable extractor. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pp. 365–373, 2024.
- Christopher Clark and Santosh Divvala. Pdffigures 2.0: Mining figures from research papers, 2016. URL <https://pdffigures2.allenai.org/>.
- Zijian Ding, Qinshi Zhang, Mohan Chi, and Ziyi Wang. Frontend diffusion: Empowering self-representation of junior researchers and designers through agentic workflows. *arXiv preprint arXiv:2502.03788*, 2025.
- Abul Ehtesham, Aditi Singh, Gaurav Kumar Gupta, and Saket Kumar. A survey of agent interoperability protocols: Model context protocol (mcp), agent communication protocol (acp), agent-to-agent protocol (a2a), and agent network protocol (anp). *arXiv preprint arXiv:2505.02279*, 2025.

- Yue Fan, Xiaojian Ma, Rujie Wu, Yuntao Du, Jiaqi Li, Zhi Gao, and Qing Li. Videoagent: A memory-augmented multimodal agent for video understanding. In *European Conference on Computer Vision*, pp. 75–92. Springer, 2024.
- Baruch Fischhoff. The sciences of science communication. *Proceedings of the National Academy of Sciences*, 110(supplement_3):14033–14039, 2013.
- Charles Frankston, Jonathan Godfrey, Shamsi Brinn, Alison Hofer, and Mark Nazzaro. Html papers on arxiv—why it is important, and how we made it happen. *arXiv preprint arXiv:2402.08954*, 2024.
- Government Digital Service and Central Digital and Data Office. Publishing accessible documents, 2024. URL <https://www.gov.uk/guidance/publishing-accessible-documents>. Last updated 2024-08-14.
- Yi Gui, Zhen Li, Yao Wan, Yemin Shi, Hongyu Zhang, Bohua Chen, Yi Su, Dongping Chen, Siyuan Wu, Xing Zhou, et al. Webcode2m: A real-world dataset for code generation from webpage designs. In *Proceedings of the ACM on Web Conference 2025*, pp. 1834–1845, 2025a.
- Yi Gui, Yao Wan, Zhen Li, Zhongyi Zhang, Dongping Chen, Hongyu Zhang, Yi Su, Bohua Chen, Xing Zhou, Wenbin Jiang, et al. Uicopilot: Automating ui synthesis via hierarchical code generation from webpage designs. In *Proceedings of the ACM on Web Conference 2025*, pp. 1846–1855, 2025b.
- Shanshan Han, Qifan Zhang, Yuhang Yao, Weizhao Jin, and Zhaozhuo Xu. Llm multi-agent systems: Challenges and open problems. *arXiv preprint arXiv:2402.03578*, 2024.
- Xinyi Hou, Yanjie Zhao, Shenao Wang, and Haoyu Wang. Model context protocol (mcp): Landscape, security threats, and future research directions. *arXiv preprint arXiv:2503.23278*, 2025.
- Cheng-Ping Hsieh, Simeng Sun, Samuel Krizan, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. Ruler: What’s the real context size of your long-context language models? *arXiv preprint arXiv:2404.06654*, 2024. doi: 10.48550/arXiv.2404.06654. URL <https://arxiv.org/abs/2404.06654>. COLM 2024.
- Xiwei Hu, Haokun Chen, Zhongqi Qi, Hui Zhang, Dexiang Hong, Jie Shao, and Xinglong Wu. Dreamposter: A unified framework for image-conditioned generative poster design. *arXiv preprint arXiv:2507.04218*, 2025.
- Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. Layoutlmv3: Pre-training for document ai with unified text and image masking. In *Proceedings of the 30th ACM international conference on multimedia*, pp. 4083–4091, 2022.
- Md Ashraful Islam, Mohammed Eunus Ali, and Md Rizwan Parvez. Mapcoder: Multi-agent code generation for competitive problem solving. *arXiv preprint arXiv:2405.11403*, 2024.
- Takuro Kawada, Shunsuke Kitada, Sota Nemoto, and Hitoshi Iyatomi. Sciga: A comprehensive dataset for designing graphical abstracts in academic papers. *arXiv preprint arXiv:2507.02212*, 2025.
- Naveen Krishnan. Advancing multi-agent systems through model context protocol: Architecture, implementation, and applications. *arXiv preprint arXiv:2504.21030*, 2025.
- Anukriti Kumar and Lucy Lu Wang. Uncovering the new accessibility crisis in scholarly pdfs. In *Proceedings of the 26th ACM SIGACCESS Conference on Computers and Accessibility (ASSETS ’24)*, 2024. doi: 10.1145/3663548.3675634. URL <https://arxiv.org/abs/2410.03022>.
- Hugo Laurençon, Léo Tronchon, and Victor Sanh. Unlocking the conversion of web screenshots into html code with the websight dataset. *arXiv preprint arXiv:2403.09029*, 2024.
- Minghao Li, Yiheng Xu, Lei Cui, Shaohan Huang, Furu Wei, Zhoujun Li, and Ming Zhou. Docbank: A benchmark dataset for document layout analysis. *arXiv preprint arXiv:2006.01038*, 2020.

- Ryan Li, Yanzhe Zhang, and Diyi Yang. Sketch2code: Evaluating vision-language models for interactive web design prototyping. *arXiv preprint arXiv:2410.16232*, 2024.
- Shanchao Liang, Nan Jiang, Shangshu Qian, and Lin Tan. Waffle: Multi-modal model for automated front-end development. *arXiv preprint arXiv:2410.18362*, 2024.
- Jiawei Lin, Jiaqi Guo, Shizhao Sun, Zijiang Yang, Jian-Guang Lou, and Dongmei Zhang. Layout-prompter: Awaken the design ability of large language models. *Advances in Neural Information Processing Systems*, 36:43852–43879, 2023.
- Jiaheng Liu, Zehao Ni, Haoran Que, Sun Sun, Noah Wang, Jian Yang, Hongcheng Guo, Zhongyuan Peng, Ge Zhang, Jiayi Tian, et al. Roleagent: Building, interacting, and benchmarking high-quality role-playing agents from scripts. *Advances in Neural Information Processing Systems*, 37:49403–49428, 2024a.
- Jingwei Liu, Ling Yang, Hao Luo, Fan Wang Hongyan Li, and Mengdi Wang. Preacher: Paper-to-video agentic system. *arXiv preprint arXiv:2508.09632*, 2025.
- Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024b. doi: 10.1162/tacl.a_00638. URL <https://aclanthology.org/2024.tacl-1.9/>.
- Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel S. Weld. S2orc: The semantic scholar open research corpus. In *Proceedings of ACL 2020*, 2020. URL <https://aclanthology.org/2020.acl-main.447/>.
- Zimu Lu, Yunqiao Yang, Houxing Ren, Haotian Hou, Han Xiao, Ke Wang, Weikang Shi, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. Webgen-bench: Evaluating llms on generating interactive and functional websites from scratch. *arXiv preprint arXiv:2505.03733*, 2025.
- Reuben A Luera, Ryan Rossi, Franck Dernoncourt, Samyadeep Basu, Sungchul Kim, Subhojyoti Mukherjee, Puneet Mathur, Ruiyi Zhang, Jihyung Kil, Nedim Lipka, et al. Mllm as a ui judge: Benchmarking multimodal llms for predicting human perception of user interfaces. *arXiv preprint arXiv:2510.08783*, 2025.
- Jiacheng Miao, Joe R Davis, Jonathan K Pritchard, and James Zou. Paper2agent: Reimagining research papers as interactive and reliable ai agents. *arXiv preprint arXiv:2509.06917*, 2025.
- NHS Digital. Pdfs and other non-html documents — nhs digital service manual, 2025. URL <https://service-manual.nhs.uk/content/pdfs-and-other-non-html-documents>. Updated 2025-02.
- Wei Pang, Kevin Qinghong Lin, Xiangru Jian, Xi He, and Philip Torr. Paper2poster: Towards multimodal poster automation from scientific papers. *arXiv preprint arXiv:2505.21497*, 2025.
- Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5865–5874, 2021.
- Yu-Ting Qiang, Yan-Wei Fu, Xiao Yu, Yan-Wen Guo, Zhi-Hua Zhou, and Leonid Sigal. Learning to generate posters of scientific papers by probabilistic graphical models. *Journal of Computer Science and Technology*, 34(1):155–169, 2019.
- Jingwei Shi, Zeyu Zhang, Biao Wu, Yanjie Liang, Meng Fang, Ling Chen, and Yang Zhao. Present-agent: Multimodal agent for presentation video generation. *arXiv preprint arXiv:2507.04036*, 2025.
- Chenglei Si, Yanzhe Zhang, Ryan Li, Zhengyuan Yang, Ruibo Liu, and Diyi Yang. Design2code: Benchmarking multimodal code generation for automated front-end engineering. *arXiv preprint arXiv:2403.03163*, 2024.

- Achint Soni, Sreyas Venkataraman, Abhranil Chandra, Sebastian Fischmeister, Percy Liang, Bo Dai, and Sherry Yang. Videoagent: Self-improving video generation. *arXiv preprint arXiv:2410.10076*, 2024.
- Tao Sun, Enhao Pan, Zhengkai Yang, Kaixin Sui, Jiajun Shi, Xianfu Cheng, Tongliang Li, Wenhao Huang, Ge Zhang, Jian Yang, et al. P2p: Automated paper-to-poster generation and fine-grained benchmark. *arXiv preprint arXiv:2505.17104*, 2025.
- Wenxin Tang, Jingyu Xiao, Wenxuan Jiang, Xi Xiao, Yuhang Wang, Xuxin Tang, Qing Li, Yuehe Ma, Junliang Liu, Shisong Tang, et al. Slidecoder: Layout-aware rag-enhanced hierarchical slide generation from design. *arXiv preprint arXiv:2506.07964*, 2025.
- Lisbeth Thorlacius.] the role of aesthetics in web design. *Nordicom Review*, 28(1), 2007.
- Dominika Tkaczyk, Paweł Szostek, Mateusz Fedoryszak, Piotr Jan Dendek, and Łukasz Bolikowski. Cermine: automatic extraction of structured metadata from scientific literature. *International Journal on Document Analysis and Recognition (IJDAR)*, 18(4):317–335, 2015.
- Edward R Tufte and Peter R Graves-Morris. *The visual display of quantitative information*, volume 2. Graphics press Cheshire, CT, 1983.
- W3C Web Accessibility Initiative. Understanding success criterion 1.4.10: Reflow (wcag 2.x), 2018. URL <https://www.w3.org/WAI/WCAG21/Understanding/reflow.html>. Accessed 2025-10-06.
- Yuxuan Wan, Yi Dong, Jingyu Xiao, Yintong Huo, Wenxuan Wang, and Michael R Lyu. Mrweb: An exploration of generating multi-page resource-aware web code from ui designs. *arXiv preprint arXiv:2412.15310*, 2024a.
- Yuxuan Wan, Chaozheng Wang, Yi Dong, Wenxuan Wang, Shuqing Li, Yintong Huo, and Michael R Lyu. Automatically generating ui code from screenshot: A divide-and-conquer-based approach. *arXiv preprint arXiv:2406.16386*, 2024b.
- Heng Wang, Yotaro Shimose, and Shingo Takamatsu. Banneragency: Advertising banner design with multimodal llm agents. *arXiv preprint arXiv:2503.11060*, 2025.
- Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. Videoagent: Long-form video understanding with large language model as agent. In *European Conference on Computer Vision*, pp. 58–76. Springer, 2024a.
- Zheng Wang, Bingzheng Gan, and Wei Shi. Multimodal query suggestion with multi-agent reinforcement learning from human feedback. In *Proceedings of the ACM Web Conference 2024*, pp. 1374–1385, 2024b.
- Jingyu Xiao, Yuxuan Wan, Yintong Huo, Zhiyao Xu, and Michael R Lyu. Interaction2code: How far are we from automatic interactive webpage generation? *arXiv e-prints*, pp. arXiv–2411, 2024a.
- Shuhong Xiao, Yunnong Chen, Jiazhi Li, Liuqing Chen, Lingyun Sun, and Tingting Zhou. Prototype2code: End-to-end front-end code generation from ui design prototypes. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 88353, pp. V02BT02A038. American Society of Mechanical Engineers, 2024b.
- Sheng Xu and Xiaojun Wan. Neural content extraction for poster generation of scientific papers. *arXiv preprint arXiv:2112.08550*, 2021.
- Jian Yang, Wei Zhang, Jiaxi Yang, Yibo Miao, Shanghaoran Quan, Zhenhe Wu, Qiyao Peng, Liqun Yang, Tianyu Liu, Zeyu Cui, et al. Multi-agent collaboration for multilingual code instruction tuning. *arXiv preprint arXiv:2502.07487*, 2025.
- Hui Zhang, Dexiang Hong, Maoke Yang, Yutao Cheng, Zhao Zhang, Jie Shao, Xinglong Wu, Zuxuan Wu, and Yu-Gang Jiang. Creatidesign: A unified multi-conditional diffusion transformer for creative graphic design. *arXiv preprint arXiv:2505.19114*, 2025a.

- Tao Zhang, Yige Wang, ZhuHangyu ZhuHangyu, Li Xin, Chen Xiang, Tian Hua Zhou, and Jin Ma. Webquality: A large-scale multi-modal web page quality assessment dataset with multiple scoring dimensions. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 583–596, 2025b.
- Tianhao Zhang, Fu Peiguo, Jie Liu, Yihe Zhang, and Xingmei Chen. Nldesign: A ui design tool for natural language interfaces. In *Proceedings of the ACM Turing Award Celebration Conference-China 2024*, pp. 153–158, 2024.
- Zhilin Zhang, Xiang Zhang, Jiaqi Wei, Yiwei Xu, and Chenyu You. Postergen: Aesthetic-aware paper-to-poster generation via multi-agent llms. *arXiv preprint arXiv:2508.17188*, 2025c.
- Ting Zhou, Yanjie Zhao, Xinyi Hou, Xiaoyu Sun, Kai Chen, and Haoyu Wang. Bridging design and development with automated declarative ui code generation. *arXiv preprint arXiv:2409.11667*, 2024.
- Zeyu Zhu, Kevin Qinghong Lin, and Mike Zheng Shou. Paper2video: Automatic video generation from scientific papers, 2025. URL <https://arxiv.org/abs/2510.05096>.

Appendix

Table of Contents

A Detailed of rule-base metric	18
A.1 Rule-based Metric for Connectivity	18
A.2 Rule-based Metric for Completeness	18
B Human Annotation and Verification Details	18
B.1 Interactivity	19
B.2 Aesthetic	20
B.3 Informative	20
B.4 Completeness	21
B.5 Connectivity	21
C PaperQuiz	21
C.1 QA Dataset Curation.	21
C.2 Evaluation Workflow.	23
C.3 Case Study for PaperQuiz	25
D Prompt Template	26
D.1 Baseline Template	26
D.2 Parsing Template	29
D.3 Orchestrating Template	31
E More Examples of Case Study	33

A DETAILED OF RULE-BASE METRIC

A.1 RULE-BASED METRIC FOR CONNECTIVITY

Connectivity in a web-based academic project can be divided into external links and internal navigations. To quantify this aspect, we first parse the HTML structure of the generated webpage to identify relevant syntactic patterns. Specifically, external links are represented by `<a href="...">` elements pointing to URLs outside the current domain, while internal navigations are defined by anchor links of the form `href="#section-id"`, which reference local sections within the same document.

We record the number of detected external and internal links as S_{external} and S_{internal} , respectively. For external links, we further employ a URL parser to verify the *validity*, *relevance*, and *accessibility* of each link. Only those URLs that are reachable and contextually relevant to the webpage content are counted toward S_{external} .

The overall rule-based connectivity score S_{Con} is defined as:

$$S = \frac{S_{\text{external}} + S_{\text{internal}}}{2} \quad (3)$$

A.2 RULE-BASED METRIC FOR COMPLETENESS

Image–Text Balance Prior. The Image–Text Balance Prior encodes a heuristic rule: an effective academic project webpage should maintain an approximate balance between visual and textual content, avoiding extremes such as image-only pages or text-dense “wall-of-text” layouts. Concretely, we compute the *image–text ratio* of a generated webpage as follows:

1. When rendering the full page in a standard viewport, we first measure the area of all containers on the page and calculate the proportion of each container’s area occupied by image elements. The image areas are weighted according to the container size.
2. Text content is treated as the remaining area within each container (excluding images) and is weighted in the same manner by container area proportion.

Finally, the weighted image–text ratios of all containers are aggregated according to the relative area of each container within the entire page, yielding the overall page-level image–text ratio. This approach ensures that a few large images (e.g., full-width banners) and many small icons are appropriately distinguished based on their actual proportions, while the text proportion remains consistent with both container and overall page layout.

Information Efficiency Prior. The Information Efficiency Prior rewards concise and information-dense presentations by comparing the generated text length L with the median human-authored length W for comparable sections. In the main text, we introduced the ratio $r = L/W$ together with a scaling factor β . The median is chosen because human-designed webpages often favor short text supplemented by multimedia, leading to large standard deviations in length; the median better reflects typical requirements while mitigating the influence of extreme cases. The hyperparameter β controls the decay rate of the efficiency reward when $L > W$: smaller β values impose a stricter penalty on overly verbose text. The overall rule-based connectivity score S_{Con} is defined as:

$$S = \frac{S_{\text{img-txt}} + p(r)}{2} \quad (4)$$

B HUMAN ANNOTATION AND VERIFICATION DETAILS

The annotation is conducted by 6 authors of this paper independently. The diversity of annotators plays a crucial role in reducing bias and enhancing the reliability of the benchmark. These annotators have knowledge in this domain, with different genders, ages, and educational backgrounds. To ensure the annotators can proficiently mark the data, we provide them with detailed tutorials, teaching them how to evaluate model responses more objectively. Specifically, they are required to

Connectivity (1-5)

☐ Code links (GitHub/GitLab)
 ☐ Lab homepage
 ☐ Paper link (arXiv/official)
 ☐ Project/Dataset links
 ☐ Author homepages
 ☐ Related work links

0 criteria met, score: 1

Scores auto-saved

Completeness (1-5)

☐ Title
 ☐ Setup
 ☐ Authors
 ☐ Results
 ☐ Abstract
 ☐ Citation Format
 ☐ Contributions
 ☐ Video/Demo

0 criteria met, score: 1

Scores auto-saved

IAI (Interactive · Aesthetic · Informative) — 1-5

Interactive

Interactivity
 Discoverability & Learnability
 Accessibility & Reachability

Aesthetic

Layout & Hierarchy
 Typographic Readability
 Color & Contrast

Informative

Completeness & Depth
 Information Architecture & Findability
 Scannability

Figure 11: Human annotation instruction

give judgments without considering answer lengths, and certain names or positions of the response. Furthermore, we implement cross-validation between different annotators and conduct continuous monitoring to ensure they are maintaining objectivity and fairness. We rate paper webpages with 1–5 integer scores on five indicators: Interactivity, Aesthetic, Informative, Completeness, and Connectivity. Raters inspect the same rendering variant in the tool and score each indicator independently.

B.1 INTERACTIVITY

This metric evaluates the quality and responsiveness of interactions. It also considers discoverability and learnability, where key actions should be obvious and controls self-explanatory. Furthermore, it assesses accessibility and reachability, including keyboard navigation, screen-reader cues, and responsive/mobile usability.

To systematically assess this, we evaluate interactivity across four key areas:

- **Basic Interactions:** This criterion covers fundamental dynamic elements that enhance readability. Raters check for common interactions like over effects, expand/collapse sections for content, and clickable tabs.
- **Interactive Visualizations:** This assesses dynamic presentations of results. Raters look for interactive charts, comparison sliders, or other elements that allow users to actively explore model outputs.

- **Live Demo:** This evaluates the presence of hands-on experiences. Raters check for an embedded online demo or a video that allows users to observe the model’s performance directly.
- **Navigation Aids:** This focuses on features that improve browsing on long pages. Raters look for tools such as a floating table of contents, quick jump-to-section links, or a back-to-top button.

B.2 AESTHETIC

This dimension focuses on a clear layout and visual hierarchy that guide the user’s attention. The WebQuality (Zhang et al., 2025b) benchmark emphasizes that a well-structured design, which avoids hindering a user’s information acquisition, is a cornerstone of quality assessment. This evaluation includes typographic readability, ensured by appropriate font sizes, line height, and stable styling. Color and contrast are also evaluated for being harmonious, accessible, and providing sufficient distinction for all text and interface elements.

We evaluate the aesthetic quality based on four criteria adapted from Paper2Poster (Pang et al., 2025):

- **Element Quality:** This criterion assesses the individual visual components on the page. Raters evaluate the clarity and resolution of images, the design quality of illustrations or figures, and whether charts and tables are not only easy to understand but also thoughtfully designed.
- **Layout Balance:** This criterion focuses on the overall spatial organization and structure. Raters check for consistent alignment of all elements, reasonable sizing of images, and appropriate spacing between different sections to ensure a clean and flexible layout.
- **Engagement and Style:** This criterion evaluates the overall artistic and sensory appeal. Raters assess the consistency and harmony of the color scheme, the readability and appropriateness of the typography, and the creativity of the overall style and its effectiveness in engaging the user.
- **Clarity:** Inspired by MLLM as a UI judge (Luera et al., 2025), this criterion evaluates how clear and uncluttered the interface appears, encompassing both textual legibility and overall visual design. A clear interface avoids overwhelming the user with too many UI elements. This extends to the fundamental readability of the text, which should feature a clear, discernible font and sufficient contrast between the text and its background to ensure a comfortable reading experience without strain.

B.3 INFORMATIVE

This indicator measures the completeness and depth. It also assesses the information architecture and findability, which are supported by a logical structure, clear labels, and cross-links or search functionality. Scannability, achieved through effective use of headings, bullet points, callouts, and summaries, is another key aspect.

To formalize this assessment, we evaluate the informative quality based on the following three dimensions:

- **Logical Flow and Coherence:** Drawing from the criterion in Paper2Poster (Pang et al., 2025), this dimension evaluates the overall structure and narrative of the webpage. A high-quality page should present information in a logical sequence that mirrors the research process. The content should be coherent, guiding the reader through the project’s story without confusion.
- **Depth of Content:** This dimension assesses whether the core sections are explained with sufficient detail and insight. The webpage should provide a substantive discussion of key concepts, methodologies, and findings, demonstrating a thorough understanding and presentation of the research.
- **Scannability and Readability:** Raters assess whether the page structures content with visually distinct section titles, applies text formatting like bolding to emphasize key terms, and organizes parallel items or sequential steps into clear lists.

B.4 COMPLETENESS

This metric assesses whether the essential elements of a research webpage are present and sufficiently developed. Recent benchmarks like WebGen-Bench (Lu et al., 2025) have highlighted the importance of moving beyond static content to evaluate the generation of truly functional websites. Accordingly, our definition of completeness encompasses not only the presence of core content but also its operational integrity. Raters consider the coverage of core content, the adequacy and coherence of accompanying text and media, and whether the information feels up-to-date and self-contained.

Raters evaluate the presence and thoroughness of items within each dimension.

- **Element Completeness:** This dimension evaluates whether the webpage effectively summarizes the fundamental components of the research paper. Drawing from principles in the WebQuality (Zhang et al., 2025b) dataset, raters assess the presence of essential elements, such as the foundational metadata, a summary of core contributions, descriptions of the experimental setup, and a presentation of key results.
- **Rich Media and Artifacts:** This dimension assesses the inclusion of supplementary materials that go beyond static text to enhance understanding and demonstrate practical applications. This includes elements like an embedded video presentation or demo and any interactive visualizations that allow users to explore the data or results.
- **Scholarly Utility:** This dimension evaluates features that provide direct practical value to other researchers and facilitate the work’s dissemination. This primarily involves tools like an easy-to-copy citation format and clearly labeled links to official resources such as the paper’s PDF, source code, or datasets.

B.5 CONNECTIVITY

This metric evaluates the richness, relevance, and reliability of outward links. High-quality pages feature working links to code and reproducible artifacts, official paper pages, author or lab websites, datasets, and pertinent related work. Links should be contextually introduced with clear anchor text, free of dead or circular references, and should help readers navigate to deeper resources without friction.

Building upon the work on MRWeb (Wan et al., 2024a), we evaluate connectivity across three dimensions:

- **Resource Connectivity:** This dimension assesses the linkage to core research assets that enable reproducibility and deeper engagement. Raters check for direct, functional links to the source code repository, the official paper, and any associated project or data resources.
- **Scholarly Context Connectivity:** This dimension measures how well the webpage connects the research to the wider academic landscape. This is primarily evaluated by the presence and quality of links to related work.
- **Internal Navigation and Linking:** This dimension evaluates how effectively the webpage facilitates smooth and intuitive movement between its internal sections. Raters assess the presence and clarity of navigational elements—such as anchored headings, menus, or in-page links—that allow users to easily access key content areas without losing contextual flow.

C PAPERQUIZ

C.1 QA DATASET CURATION.

Each paper PDF is converted to markdown via our PDF parser. We then prompt o3 to generate 50 multiple-choice questions per paper, where we have 25 verbatim and 25 interpretive questions as follows:

- **Verbatim questions (25):** directly answerable from the paper text, covering 13 orthogonal content aspects (e.g., objectives, methodology, key results).

- **Interpretive questions (25):** requires high-level comprehension beyond verbatim text, spanning 10 conceptual dimensions (e.g., motivation, contribution synthesis, implication analysis).

The following is a prompt example of 25 Verbatim questions and 25 Interpretive questions, generated by GPT-o3.

Prompt: Generated Verbatim Questions

System prompt:

You are a highly precise Question-Generation agent for academic project websites. Your task is to read the supplied Markdown text and produce a structured set of exactly 25 multiplechoice QA items. Your primary goal is to strictly adhere to a mandatory Question Distribution Plan and a set of critical formatting rules. Failure to follow these rules precisely will result in an invalid output. The answers to your questions must be located verbatim or almost verbatim in the provided text. The questions must be suitable for website visitors: avoid deep theoretical proofs, reference lists, or citation minutiae.

Instructions:

You **MUST** generate questions according to the following to ensure all aspects are covered and the total is exactly 25 questions.

- **Locate Fact:** Find a specific, clear factual statement, number, or detail in the 'document_markdown'.
- **Classify Aspect:** Critically determine which single aspect (from A-M) this fact *most accurately* represents. Be extremely strict in your classification.
- **Formulate Question:** Based **ONLY** on the located fact, create a clear, answerable-from-text question.
- **Create Options:** Write the correct answer and three high-quality distractors as defined in the rules below.

Aspect Definitions & Special Instructions:

You will generate questions for the following aspects:

- A. Research domain & background context.
- B. Central problem / motivation / research gap
- C. Primary goal, hypothesis, or research question
- D. Key contributions or novelty statements
- E. Overall methodology or workflow (summarized)
- F. Qualitative insights or illustrative examples

.....

MANDATORY Formatting Rules:

Each question object **MUST** have exactly four options, labelled "A. ", "B. ", "C. ", and "D. ". Do not generate more or fewer than four. The "aspect" key is required and must contain a single letter from the list above.

Final Pre-Output Check: Before providing the final JSON, mentally perform this check:

- Is the total number of questions **EXACTLY** 25?
- Is the Question Distribution Plan followed perfectly?
- Does **EVERY** single question have **EXACTLY** four options?
- Is every question accurately classified with its `aspect` and do they follow all special instructions?
- If any check fails, you must restart and correct the errors.

document_markdown:

```
{{ document_markdown }}
```

Prompt: Generated Interpretive Questions**System_prompt:**

You are a highly precise QuestionGeneration agent for academic project websites. Your task is to read the supplied Markdown text and produce a structured set of exactly 25 multiple-choice QA items. Your primary goal is to strictly adhere to a mandatory Question Distribution Plan and a set of critical formatting rules. The answers to your questions must be located verbatim or almost verbatim in the provided text. The questions must be suitable for website visitors.

Instructions:

For each question you generate, you **MUST** follow these mental steps:

- **Locate Fact:** Find a specific, clear factual statement, number, or detail in the document_markdown.
- **Classify Aspect:** Critically determine which single aspect (from A–M) this fact most accurately represents. Be extremely strict in your classification.
- **Formulate Question:** Based **ONLY** on the located fact, create a clear, answerable-from-text question.
- **Create Options:** Write the correct answer and three high-quality distractors as defined in the rules below.

Aspect Definitions & Special Instructions:

You will generate questions for the following aspects:

- A. Title & authorship (title, author names, affiliations, keywords): For questions about author names, the incorrect options (distractors) **MUST** be fabricated but plausible-sounding names. Do not use real names from other contexts.
- B. Motivation / problem statement / research gap
- C. Objectives or hypotheses
- D. Dataset(s) or experimental materials
- E. Methodology (algorithms, model architecture, workflow steps)
- F. Key parameters or hyper-parameters (values, settings)
-

MANDATORY Formatting Rules:

Each question object **MUST** have exactly four options, labelled "A. ", "B. ", "C. ", and "D. ". Do not generate more or fewer than four. The "aspect" key is required and must contain a single letter from the list above.

Final Pre-Output Check: Before providing the final JSON, mentally perform this check:

- Is the total number of questions **EXACTLY** 25?
- Is the Question Distribution Plan followed perfectly?
- Does **EVERY** single question have **EXACTLY** four options?
- Is every question accurately classified with its `aspect` and do they follow all special instructions?
- If any check fails, you must restart and correct the errors.

Adhere to Mandatory JSON Format:

document_markdown:
 {{ document_markdown }}

C.2 EVALUATION WORKFLOW.

For each website snapshot, we query six MLLMs reader models to answer curated questions. These models include three open-source models (LLaVA-OneVision-Qwen2-7B-ov-hf, DeepSeek-V3.2-

Exp, and Qwen3-Coder-480B-A35B) and three closed-source models (o1, Gemini 2.5 Flash, and Grok Code Fast 1). Their outputs are evaluated according to two enforced rules:

- **No external knowledge.** Models must base answers solely on information present in the website snapshot.
- **Visual citation.** Each answer must include a reference to the website region supporting it (e.g., “See the ‘Results’ section”); if no region contains the answer, the model responds “NA.”

Prompt: Answer Qusetions

System prompt:

You are an answering agent. You will be provided with:

- An image of a project website snapshot.
- A JSON object called “questions” which contains multiple questions. Each question has four possible answers: A, B, C, or D.

Your goal is to analyze the website snapshot thoroughly and answer each question based on the information it provides. You should **NOT** use any external knowledge or context beyond the website snapshot image. You must rely solely on the content of the website snapshot to answer the questions.

For each question:

- If you find enough evidence in the website snapshot to decide on a specific option (A, B, C, or D), then choose that option. Also include a brief reference to the part of the webpage that supports your answer (e.g., “Top-left text”, “Header section”, etc.).
- If the website snapshot does not offer sufficient information to confidently choose any of the options, respond with “NA” for both the answer and the reference.

Your final output must be returned as a JSON object. For each question, the structure should be:

```
"Question N": {
  "answer": "A" | "B" | "C" | "D" | "NA",
  "reference": "<short description or 'NA'>"
}
```

Template:

Follow these steps to create your response:

1. Study the website snapshot image along with the “questions” provided.
2. For each question:
 - Decide if the website snapshot clearly supports one of the four options (A, B, C, or D). If so, pick that answer.
 - Otherwise, if the website snapshot does not have adequate information, use “NA” for the answer.
3. Provide a brief reference indicating where on the webpage you found the answer. If no reference is available (i.e., your answer is “NA”), use “NA” for the reference too.
4. Format your output strictly as a JSON object with this pattern:

```
{
  "Question 1": {
    "answer": "X",
    "reference": "some reference or 'NA'"
  },
  "Question 2": {
    "answer": "X",
    "reference": "some reference or 'NA'"
  },
  ...
}
```

5. Do not include any explanations or extra keys beyond the specified structure.
6. You must provide an answer entry for all questions in the “questions” object.

Example Output:**Questions:**

```
{{questions}}
```

C.3 CASE STUDY FOR PAPERQUIZ

Here we provide a simple Q&A example of PaperQuiz.

PaperQuiz Example

```
{
  "questions": {
    "Question 1": {
      "question": "What is the full title of the paper discussed in
the document?",
      "options": [
        "A. Universal Audio-Video Diffusion Networks for Multimodal
Synthesis",
        "B. Multisensory Diffusion: A Joint Model for Sound and
Vision",
        "C. Cross-Modal Transformer: Unified Audio and Video
Generation",
        "D. A Versatile Diffusion Transformer with Mixture of Noise
Levels for Audiovisual Generation"
      ]
    },
    ...
    "Question 25": {
      "question": "In the context of this paper, what does the term
\"time-segment\" specifically refer to?",
      "options": [
        "A. An entire training epoch",
        "B. One complete diffusion timestep in noise addition",
        "C. A full audio clip of any length",
        "D. A single unit in the temporal dimension such as a video
frame"
      ]
    }
  },
  "answers": {
    "Question 1": "D. A Versatile Diffusion Transformer with Mixture
of Noise Levels for Audiovisual Generation",
    ...
    "Question 25": "D. A single unit in the temporal dimension such
as a video frame"
  },
  "aspects": {
    "Question 1": "A",
    ...
    "Question 25": "M"
  },
  "understanding": {
    "questions": {
      "Question 1": {
```

```

    "question": "What multimodal generation challenge is
    identified as still open in the paper's introduction?",
    "options": [
      "A. Inferring audio labels from isolated spectrogram
      snapshots",
      "B. Classifying large multimodal datasets into predefined
      categories",
      "C. Producing single high-resolution images from textual
      captions",
      "D. Generating sequences across multiple modalities such as
      video and audio"
    ]
  },
  ...
  "Question 25": {
    "question": "Which unified approach is claimed by the authors
    to enable a single model to generate and manipulate sequences
    across modalities and time?",
    "options": [
      "A. The mixture of noise levels strategy introduced in this
      paper",
      "B. An unsupervised text summarization algorithm",
      "C. A rule-based system for audio classification",
      "D. A curriculum learning schedule for GANs"
    ]
  }
},

"answers": {
  "Question 1": "D. Generating sequences across multiple
  modalities such as video and audio",
  ...
  "Question 25": "A. The mixture of noise levels strategy
  introduced in this paper"
},

"aspects": {
  "Question 1": "A",
  ...
  "Question 25": "J"
}
}

```

D PROMPT TEMPLATE

D.1 BASELINE TEMPLATE

We exhibit the prompt templates used to generate end-to-end model generation baselines. When incorporating the template from the popular Nerfies academic website (Park et al., 2021), you only need to include this template as part of the prompt.

Prompt: Baseline LLM Generation

System prompt:

You are a document-to-website generation agent and n expert full-stack web developer and UI/UX designer specializing in creating beautiful, modern, and interactive academic project websites. Your task is to generate a complete, production-ready website based on research paper content and visual asset allocations. Your task is to read the supplied Markdown text

and design a professional, visually appealing academic conference website by generating an HTML file. Follow the guidelines below precisely.

- Is visually stunning and modern with a professional, clean, and academic design.
- Has rich interactivity and smooth animations.
- Effectively presents research content in an engaging way.
- Integrates external links and resources strategically.
- Uses advanced CSS and JavaScript for enhanced user experience.

Instructions:

You are creating a complete, beautiful, and interactive website for an academic research project. This is NOT a simple static page - it should be a sophisticated, modern web application with rich interactivity. Your task is to read the supplied Markdown text and design a professional, visually appealing academic conference website by generating an HTML file.

- **Design Requirements**

- **Visual Design**

- * Modern, professional, academic aesthetic.
- * Sophisticated color scheme (dark/light themes with multiple color variations).
- * Professional typography with hierarchy and multiple font weights.
- * Smooth animations and transitions with multiple animation types.
- * Interactive elements and hover effects with complex state changes.
- * Professional spacing and layout with multiple breakpoints.
- * Advanced visual effects (shadows, gradients, transforms).
- * **Background Style:** Avoid background images (especially in hero section); prefer solid colors such as #2d3748 (dark gray) or #ffffff (white) or subtle gradients. Do not fetch images from external sources like Unsplash.

- **Layout Structure**

- * Hero section with project title, authors, and key highlights.
- * Multi-level navigation with smooth scrolling and active state indicators.
- * Content sections with dynamic layouts based on importance.
- * Interactive visualizations and image galleries with lightbox and carousel.
- * External resources section with categorized link placement.
- * Footer with information, social links, and contact details.
- * Sidebar navigation with quick links and progress indicators.
- * Multiple columns and grid layouts.
- * Card-based content presentation.

- **Interactivity Features**

- * Smooth scrolling navigation with progress bars and scroll indicators.
- * Interactive image galleries with lightbox, zoom, and slideshow.
- * Animated counters and number transitions.
- * Hover effects and micro-interactions.
- * Responsive navigation menu with hamburger and dropdowns.
- * Loading animations and skeleton screens.
- * Interactive charts and visualizations with tooltips.
- * Modal dialogs and popup windows.
- * Form validation and interactive feedback.
- * Search with autocomplete.
- * Dark/light theme toggle with transitions.

- **External Links Integration**

- * Place important links strategically within content.
- * Dedicated “Resources & Tools” section with categories.

- * Integrate links naturally in context.
- * Attractive buttons for external links with hover effects.
- * Provide descriptive context for each resource.
- **Technical Requirements**
 - **CSS Requirements**
 - * Advanced animations and transitions with varied timing.
 - * Responsive design for mobile, tablet, and desktop.
 - * CSS Grid and Flexbox layouts.
 - * CSS variables for theming.
 - * Advanced selectors and pseudo-elements.
 - * Center single and multiple images responsively (max 3 per row).
 - * Smooth scrolling and scroll animations.
 - * Hover effects and micro-interactions.
 - * Professional color schemes with multiple variations.
 - * Advanced typography with clear hierarchy.
 - **JavaScript Requirements**
 - * Modern ES6+ syntax with error handling.
 - * Interactive image galleries with lightbox.
 - * Smooth scrolling navigation and progress indicators.
 - * Mobile menu with animations.
 - * Intersection Observer for scroll animations.
 - * Local storage for user preferences.
 - * Form validation and interactive feedback.
 - * Performance optimization and error handling.
 - * Advanced image handling and gallery functionality.
 - **Critical JavaScript Best Practices (MUST FOLLOW)**
 - * **DOM Element Access Timing:** All DOM element access must occur within a `DOMContentLoaded` listener.
 - * **Intersection Observer Setup:**
 - Set up observer before adding classes.
 - Observe elements immediately after adding `fade-in` class.
 - Never query `.fade-in` elements before setup.
 - Example: `element.classList.add('fade-in');`
`observer.observe(element);`
 - * **Event Listener Safety:** Always verify element existence before adding listeners.
 - * **Animation Class Management:** Ensure fade-in classes start invisible (`opacity: 0`) and become visible (`opacity: 1`) when animated.
 - * **Function Organization:** Wrap DOM-dependent code in initialization functions triggered by `DOMContentLoaded`.
- **Final Checklist**
 - Header includes title, authors, and affiliations.
 - Images sized using responsive CSS (`width: 100%`).
 - Dedicated “Resources & External Links” section with clickable URLs.
 - Each URL accompanied by description.
 - Preserve all original text content.
 - Images fit properly within containers.
 - Lists rendered as responsive grids.
- **Critical Checks**
 - Consistent, professional typography using fonts like Inter or Manrope.
 - Prominent author display below title with affiliations.

- “How to Cite” section with BibTeX and “Copy” button.
- No fixed image sizes in HTML; control via CSS (`w-full`).
- Implement Scroll-Spy in navigation.
- Encourage interactive demos over static images.
- Add elegant hover and scaling effects to all buttons.

document_markdown:
 {{ document_markdown }}
jinja_args:
 - document_markdown

D.2 PARSING TEMPLATE

We present the prompt templates used for paper deconstruction: (1) the prompt for paper summarizing, and (2) the prompt for image and table filtering.

Prompt: Paper Summarizing

- You are the author of the paper, and you will create a comprehensive content summary for a project website. Your task is to extract and expand the key information from the research paper to create detailed, informative content for each section.
- **IMPORTANT REQUIREMENTS:**
 - **Dual Constraint Adherence:** Each section must strictly meet BOTH of the following constraints.
 - **Content Richness:** On the premise of ensuring the character and sentence counts are not exceeded, each section must be rich with substantial detail.
 - **Information Completeness:** Include comprehensive coverage of all paper content, not just summaries.
 - **Website Depth:** Provide enough detail for website visitors to fully understand the research without reading the paper.
 - **Technical Thoroughness:** Explain technical concepts, methods, and results in detail.
- **CONTENT STRUCTURE FOR EACH SECTION:**
 The constraints below apply to every section (Introduction, Related Work, etc.).
 - **Introduction:** Write sentences covering the research background, core motivation, challenges, main contributions, and a general overview.
 - **Related Work:** Write sentences covering existing approaches, their detailed limitations, and the specific gaps in current research.
 - **Dataset Overview:** Write sentences covering the dataset’s composition, key features, core statistics, comparisons with other datasets, and its detailed characteristics.
 - **Methodology/Approach:** Write sentences covering core technical details, key algorithms, the implementation process, and the specific methods used.
 - **Results/Evaluation:** Write sentences covering the experimental setup, detailed core results, analysis of the results, and comprehensive performance comparisons.
 - **Applications:** Write sentences covering specific use cases, benefits, practical application scenarios, and representative examples.
 - **Conclusion:** Write sentences covering a summary of the research, reiterating the contributions, pointing out limitations, and providing a detailed outlook on future work.

- **OUTPUT FORMAT:**

Generate a JSON object with the following structure:

- **CONTENT GUIDELINES:**

On the premise of ensuring the character and sentence counts are not exceeded, please adhere to the following as much as possible:

- **Expand information:** Provide comprehensive coverage of paper content.
- **Include specific numbers:** Use actual statistics, dimensions, and measurements from the paper.
- **Be thorough and detailed:** Explain concepts, methods, and results in depth.
- **Explain significance:** Why is this important? What problems does it solve?
- **Compare and contrast:** How does this compare to existing approaches?
- **Future implications:** What are the broader impacts and applications?
- **Provide examples:** Include concrete examples and use cases.
- **Maintain technical depth:** Do not oversimplify technical concepts.

- Paper content to analyze:

```
{{ markdown_document }}
```

Prompt: Image/Table Filtering

- You are an assistant that reviews a research paper's content (`json_content`), along with corresponding `image_information` and `table_information`. Your task is to filter out any image or table entries that are irrelevant to the content described in `json_content`, specifically for creating a project website.
- Specifically:
 - Read through the full research paper data described in `json_content`.
 - Examine each entry within `image_information` and `table_information`.
 - Decide if each entry is relevant for a project website based on its caption, path, or any other information provided.
 - * For example, if an image has a caption that obviously does not fit into any section or does not relate to the paper's content outline, deem it "unimportant."
 - * Consider which images/tables would be most valuable for a project website.
 - Keep all images/tables that are relevant to the project website (i.e., related to the topics, sections, or discussions mentioned in `json_content`).
 - Do not impose any artificial quantity limits—include every visual element that enhances understanding of the research.
 - Produce an output containing just two keys: "image_information" for the filtered images, and "table_information" for the filtered tables. Each of these keys should map to an array of filtered objects.
 - The user will provide JSON:
 - * "`json_content`": The content of the research paper (sections, text, etc.)
 - * "`image_information`": A dict of images (each with caption, path, size constraints)
 - * "`table_information`": A dict of tables (each with caption, path, size constraints)
 - Your task:
 - * Read the research paper outline (`json_content`).
 - * Filter `image_information` and `table_information` so that only entries relevant to the project website content remain.

- * Relevance is determined by matching or relating captions to the paper’s sections or content.
- * Consider which visual elements would be most valuable for a project website (e.g., methodology diagrams, result charts, data summaries).
- * If an image or table does not clearly match or support any content in `json_content`, remove it.
- * Keep all relevant visual elements—do not limit the quantity artificially.
- You must output valid JSON containing only:

```
{
  "image_information": {...},
  "table_information": {...}
}
```

- Template Instructions:
- Please provide only the JSON object as your final output.

```
json_content:
{{ json_content }}

image_information:
{{ image_information }}

table_information:
{{ table_information }}
```

- Jinja arguments:
 - `image_information`
 - `table_information`
 - `json_content`

D.3 ORCHESTRATING TEMPLATE

We introduce the prompt templates that guide the Agent-Driven Iterative Refinement procedure.

Prompt for MLLM as Orchestrator

System message:

You are an expert web developer and UI/UX designer with extensive experience in analyzing website layouts, visual design, and user experience. Your task is to analyze website screenshots and provide targeted recommendations for improvement.

Your mission is to first classify the type of web component shown in a screenshot, and then analyze and optimize it based on a deep understanding of modern design systems and principles, using a protocol tailored to that specific component type. You must provide surgically precise feedback to guide code fixes.

Core Mission

- **Protocol for "Navigator"**
 - Focus: Ensure clarity, usability, and responsiveness in navigation elements.
 - 1. Component Flow & Alignment
 - Diagnosis: Are navigation links properly aligned?
 - Action: Suggest adjusting flexbox/grid properties (justify-content, gap) or applying uniform margins.
 - 2. Typography & Readability

Diagnosis: Are the link labels easy to read?

Action: Recommend increasing font size, adjusting font weight, or modifying colors for contrast.

...

- **Protocol for "Header/Hero"**

Focus: Maximize visual impact, establish a clear hierarchy, and communicate the primary purpose.

1. Visual Hierarchy & Flow

Diagnosis: Is the main heading prominent?

Action: Adjust font sizes or positioning to create a clear focal point.

2. Image Dominance & Sizing

Diagnosis: Does the background image enhance or overwhelm content?

Action: Suggest constraining height or applying a semi-transparent overlay.

...

- **Protocol for "Content Block" & "Component/Card"**

Focus: Ensure logical structure, effortless readability, and visual consistency.

1. Component Flow & Layout

Diagnosis: Are grouped elements laid out logically?

Action: Suggest using CSS Flexbox or Grid for adaptive alignment.

2. Typography & Readability

Diagnosis: Is the text comfortable to read?

Action: Recommend adjusting line-height and ensuring adequate contrast.

...

Response format:

```
{
  "is_needed_to_fix": true/false,
  "category": "The identified category of the component: Navigator |
Header/Hero | Content Block | Component/Card",
  "fix_suggest": "Detailed analysis and suggestions"
}
```

E MORE EXAMPLES OF CASE STUDY

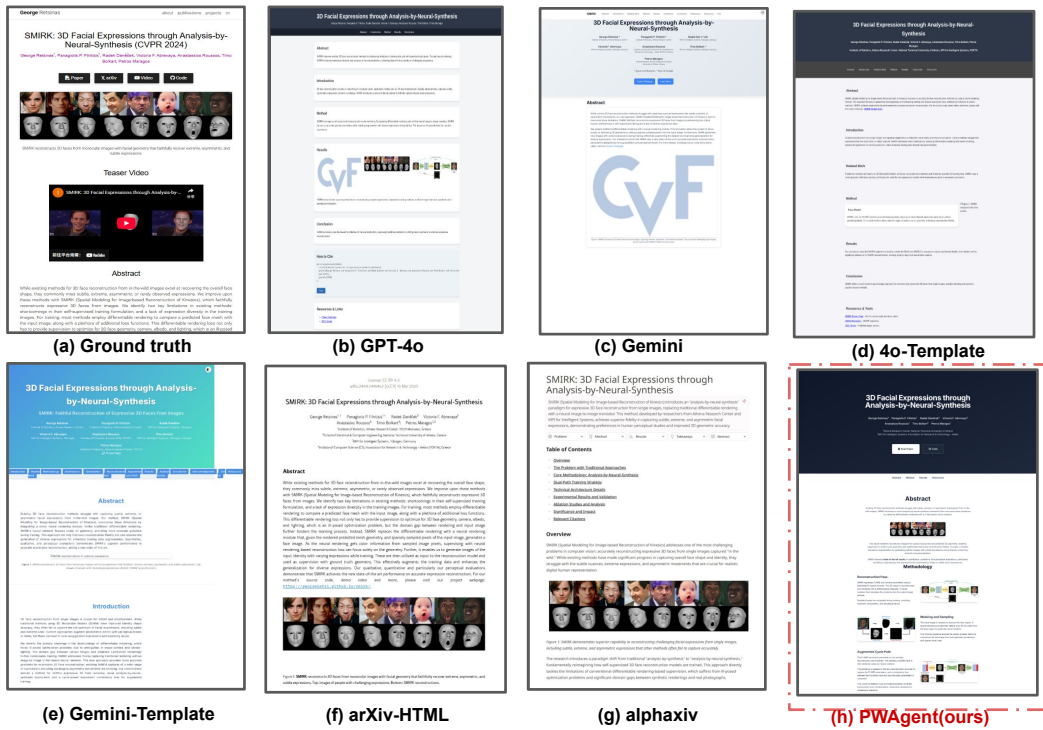


Figure 12: Illustration of website variants for the paper “SMIRK: 3D Facial Expressions through Analysis-by-Neural-Synthesis”⁴ generated by different methods.

⁴<https://georgeretsi.github.io/smirk/>



Figure 13: Illustration of website variants for the paper "Interactive3D: Create What You Want by Interactive 3D Generation"⁵ generated by different methods.

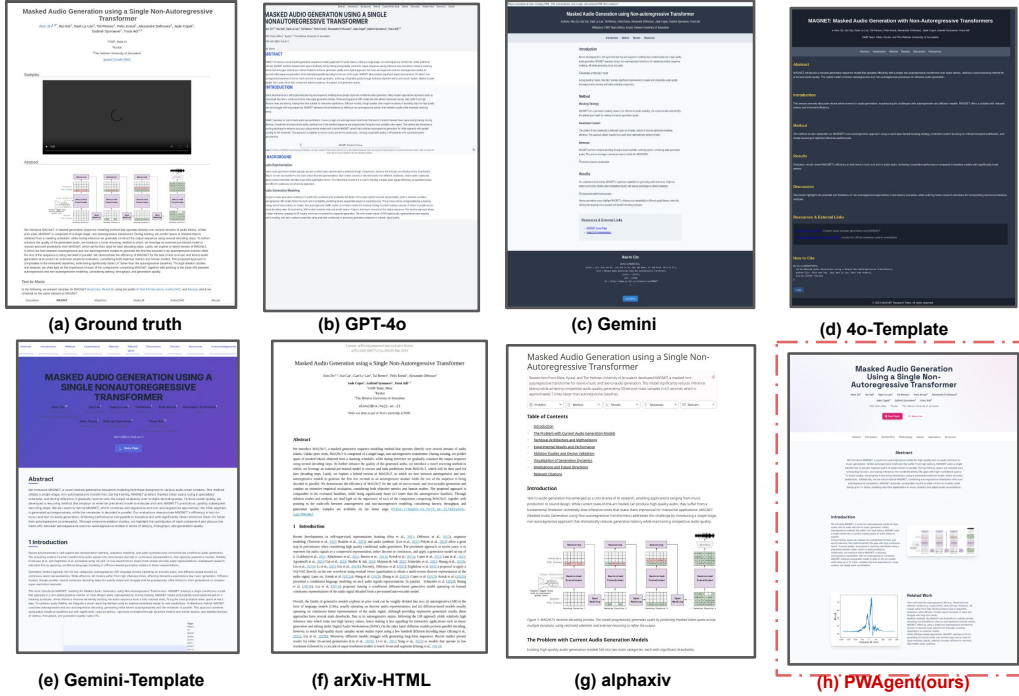


Figure 14: Illustration of website variants for the paper "Masked Audio Generation using a Single Non-Autoregressive Transformer"⁶ generated by different methods.

⁵<https://interactive-3d.github.io/>

⁶<https://pages.cs.huji.ac.il/adiyoss-lab/MAGNeT/>

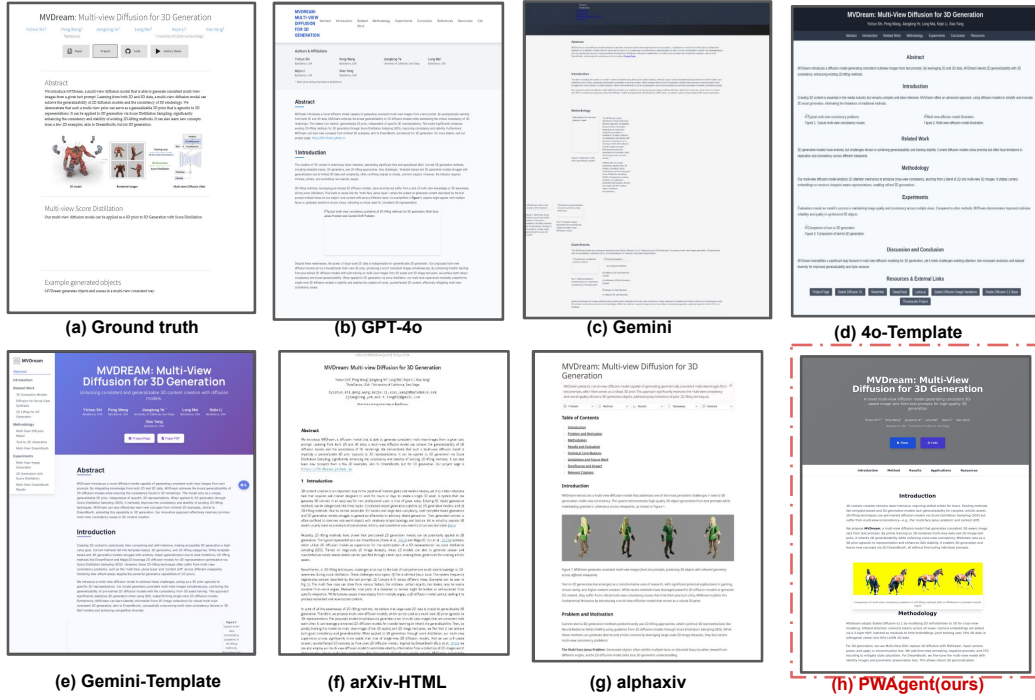


Figure 15: Illustration of website variants for the paper “MVDream: Multi-view Diffusion for 3D Generation”⁷ generated by different methods.

⁷<https://mv-dream.github.io/>