OMNIPRESENT: GENERATING COHERENT PRESENTATION SUITES WITH SELF-IMPROVING PLANNERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Transforming static research papers into dynamic media such as posters, slides, and videos is essential for effective dissemination but remains a labor-intensive challenge. Existing automated approaches often treat these formats in isolation and consequently fail to maintain semantic consistency across the entire presentation suite. We address this fragmentation by formalizing the task of unified presentation suite generation and proposing **OmniPresent** to orchestrate the creation of coherent deliverables. Our framework adopts a renderable HTML representation to enable centralized content planning and a self-correcting verify-and-repair loop that actively resolves conflicts across modalities. It further learns from verification experience across papers, converting recurring, evidence-grounded failures into reusable updates to format-specific planners for future papers. We further facilitate scalable research in this domain by releasing **OmniPreBench**, a comprehensive dataset comprising over one thousand papers with paired artifacts, and establishing a rigorous VLM-based evaluation protocol. Empirical results confirm that our method generates high-quality and faithful presentation suites that greatly surpass strong baselines in both accuracy and visual appeal. Code and data will be released.

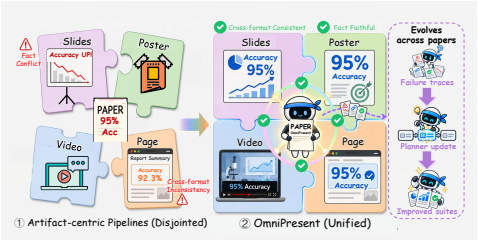
1 INTRODUCTION

Publishing a paper is often just the first step in the research communication journey. In practice, authors engage with audiences through various additional formats. Posters offer high-density overviews. Slide decks support oral presentations. Videos and interactive project pages provide narrative depth and resource access. However, manually crafting these diverse artifacts is a time-consuming endeavor that places a heavy burden on researchers. Driven by the need for efficiency, recent advancements in large language models have begun to automate these individual formats through systems dedicated to video production (Liu et al., 2025b; Zhu et al., 2025b; Shi et al., 2025) or slide generation (Liu et al., 2025a; Zheng et al., 2025; Liang et al., 2025; Xiao et al., 2026).

However, these existing approaches typically operate in an artifact-centric manner as they treat each format as an independent task (Fig. 1a and 1b). This fragmentation forces authors to coordinate multiple isolated systems and ultimately fails to address the complex interplay between different media forms. Moreover, such a disjointed approach is particularly risky because maintaining suite-level coherence is inherently difficult under heterogeneous format constraints. For instance, a poster requires spatial condensation while a video demands temporal flow, forcing the same underlying content to be repeatedly compressed and re-expressed. When performed by disjointed systems or manual adaptation, this process frequently leads to inconsistent terminology and mismatched quantitative results. Furthermore, critical nuances such as ablation studies are often inadvertently omitted during these transfers. These discrepancies do more than confuse the audience, as they fundamentally undermine scientific rigor and hinder reproducibility. Such errors can reflect recurring weaknesses in content planning. Correcting an individual output leaves the underlying planner unchanged, motivating a mechanism that *learns reusable planning rules from experience*.

Despite the urgency of this challenge, progress is currently limited by the lack of a common evaluation foundation. Specifically, the field lacks a unified framework to assess the generation of a coherent multi-format suite from a single source paper. Without a dataset of paired papers and presentation suites, researchers cannot systematically compare methods or diagnose failure modes regarding cross-format consistency. Consequently, it remains difficult to develop mechanisms that enforce strict

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Artifact-centric *v.s.* unified suite generation.

Method	Poster	Slides	Video	WebPage
Paper2Poster (Pang et al., 2025)	✔			
PPTAgent (Zheng et al., 2025)		✔		
EvoPresent (Ge et al., 2025)		✔	✔	
PresentAgent (Shi et al., 2025)			✔	
AutoPage (Ma et al., 2026b)				✔
PaperX (Yu et al., 2026)	✔	✔		
OmniPresent	✔	✔	✔	✔

(b) Capability coverage across formats.

Figure 1: **From artifact-centric pipelines to unified suite generation. (a)** Independently generating different artifacts can cause cross-format inconsistencies, such as mismatched claims, numbers, or terminology. OmniPresent unifies generation with cross-artifact checking to improve factual faithfulness and suite-level consistency. **(b)** Compared with prior systems that cover only a subset of formats, OmniPresent supports all four formats within one framework.

alignment in claims and numbers. This absence of scalable evaluation protocols creates a significant barrier to improving the accessibility and faithfulness of automated research dissemination.

To bridge this gap, we formalize the task of *paper to presentation suite generation*. Given a research paper, the goal is to synthesize a comprehensive set of deliverables spanning multiple formats, including a poster, slides, video, and page, as shown in Fig. 1a. This generation process is governed by four essential requirements: (i) **faithfulness**, ensuring that salient claims and quantitative results are strictly grounded in evidence from the source paper such as linked sections or figures. (ii) **coverage**, capturing the central contributions, methodology, and main experimental findings. (iii) **cross-format consistency**, maintaining strict alignment in terminology, core claims, and numbers across all artifacts, as shown in Fig. 1a. and (iv) **format compliance**, adhering to the unique structural and communicative constraints of each medium.

To tackle this complex generation task while satisfying the rigorous constraints outlined above, we propose **OmniPresent**, a unified framework designed to make suite-level coherence enforceable. A key design decision in our framework is the adoption of renderable HTML as the universal underlying representation for all deliverables. We choose HTML because it offers a flexible, structured medium capable of rendering both static layouts (for posters and slides) and dynamic flows (for videos and project pages). Crucially, unlike pixel-based formats, the structured nature of the DOM tree enables precise, code-level verification, directly addressing the need for automated consistency checking. Built upon this foundation, OmniPresent first performs shared parsing to construct a canonical content plan and evidence map. It then generates each artifact under format constraints while referencing this shared plan, and applies a cross-artifact verify-and-repair loop to detect and resolve conflicts across outputs, such as mismatched numbers, terminology drift, and unsupported claims. To learn from this verification experience, an offline outer loop aggregates evidence-grounded failures across papers and proposes instruction updates to individual planner components. Paired evaluation selects updates that reduce initial planning errors while preserving final content quality.

To support principled and scalable evaluation of this suite-level task, we introduce **OmniPreBench**, a benchmark of paired multi-format presentation artifacts curated from recent top-tier AI conference papers. The benchmark comprises 1,036 papers with associated posters, slides, videos, and project pages. To operationalize the assessment, we further propose a scalable LLM-as-judge rubric that evaluates two critical dimensions: content fidelity (via Fact-Score, Content Accuracy, and QA) and presentation quality (via Readability, Visual Aesthetics, and Layout).

Our contributions are summarized as follows: ♣ **Task.** We formalize *unified presentation suite generation*, a task requiring the synthesis of coherent poster, slides, video, and project page from a single paper. ♦ **Method.** We propose OmniPresent, a multi-agent framework that enforces suite-level coherence via centralized content planning and a novel verify-and-repair loop to resolve cross-format conflicts. OmniPresent also learns reusable planner rules from cross-paper verification experience to improve planning for future papers. ♥ **Data&Bench.** We introduce OmniPreBench, a large-scale benchmark of paired multi-format data, accompanied by a scalable VLM-based evaluation protocol for quality and consistency. ♠ **Results.** Extensive experiments demonstrate that OmniPresent produces faithful and high-quality artifacts, significantly outperforming strong baselines.

2 Related Works

LLM-based Scientific Agents. Large Language Models (LLMs) have progressed from standalone text generators to *agentic* systems embedded in collaborative frameworks (Wang et al., 2024b; Xi et al., 2025; Yao et al., 2023; Chase, 2022; Wu et al., 2024). This shift equips them with the ability to plan (Yao et al., 2023), invoke tools (Qu et al., 2025; Li et al., 2025), and carry out multi-step reasoning (Fu et al., 2023) in pursuit of higher-level goals. In scientific workflows, such agents show strong potential to automate the research lifecycle, spanning literature surveys (Wang et al., 2024c), scholarly writing (Weng et al., 2025; Zhao et al., 2026), experimental reproducibility (Seo et al., 2025; Miao et al., 2025), and even in peer-review process (Ma et al., 2026a; Zhu et al., 2025a). Paper2Agent (Miao et al., 2025) further advances this direction by turning static papers into interactive agents for knowledge retrieval. More recently, this agentic paradigm has also been applied to the *dissemination* stage, where agents generate presentation artifacts to communicate completed research across different media formats.

Automated Research Dissemination. Turning a finished paper into compelling dissemination artifacts is essential for research impact. Early template-driven pipelines were brittle and struggled to translate dense scholarly content into coherent narratives (Hu & Wan, 2013; Paramita & Khodra, 2016; Sun et al., 2021; Xu & Wan, 2022). Recent progress in LLMs has enabled agentic systems that can both generate content and reason about visual structure, substantially advancing automated presentation generation. This progress spans multiple formats. For slide decks, PPTAgent (Zheng et al., 2025) and SlideSpawn (Kumar & Chowdary, 2024) distill papers into modular slides, while PreGenie (Xu et al., 2025) and AUTOPRESENT (Ge et al., 2025) emphasize narrative structure and visual design; EvoPresent (Liu et al., 2025a) and Talk to Your Slides (Jung et al., 2025) add refinement through editing. For posters, Paper2Poster (Pang et al., 2025), P2P (Sun et al., 2025), and PosterGen (Zhang et al., 2025) adopt layout-aware multi-agent workflows, often guided by VLM feedback. For video, Paper2Video (Zhu et al., 2025b), Preacher (Liu et al., 2025b), PresentAgent (Shi et al., 2025), and AutoPR (Chen et al., 2025) generate video abstracts and narration for broader outreach. AutoPage (Ma et al., 2026b) builds project webpages as interactive entry points. Yet, most efforts still advance *one artifact at a time*. In this work, we propose the task of unified presentation suite generation.

3 OmniPresent

We propose **OmniPresent**, a unified framework that transforms a static scientific paper into a coherent presentation suite. As illustrated in Fig. 2, OmniPresent follows a *Plan-Verify-Render* paradigm. Unlike traditional frameworks that generate artifacts independently, we introduce a shared *Knowledge Stream* to ground all content and a *Cross-Artifact Verification* loop to ensure suite-level consistency. We further introduce **Experiential Planner Learning (EPL)**, which turns verification experience across papers into reusable updates to format-specific planners.

3.1 Task Formulation and Overview

Given a paper x, we generate a suite $\mathcal{H}$ containing slides, a poster, a project page, and a video. We distinguish these four outputs from the three content-planning formats $\mathcal{P} = \{\textsc{Slides}, \textsc{Poster}, \textsc{Page}, \textsc{Video}\}$. Each $f \in \mathcal{P}$ produces renderable HTML; video is derived from the rendered slides and their synchronized speech script.

The paper is represented by structured resources $\mathcal{R}$, an evidence index $\mathcal{E}$, and a shared Knowledge Stream $\mathcal{K}$. For each planning format, we generate an initial plan:

$$p_f^{(0)} = \pi_f(\mathcal{K}; \theta_f), \qquad f \in \mathcal{P}. \tag{1}$$

Here $\theta_f = \{\theta_{f,q}\}_q$ contains the instructions and constraints for the components q of planner f, and $\Theta = \{\theta_f\}_{f\in\mathcal{P}}$ denotes the complete planner configuration. Verification revises the current plans $\mathbf{p} = \{p_f\}_{f\in\mathcal{P}}$; offline learning revises Θ for future papers. Algorithm 1 summarizes generation with a fixed configuration.

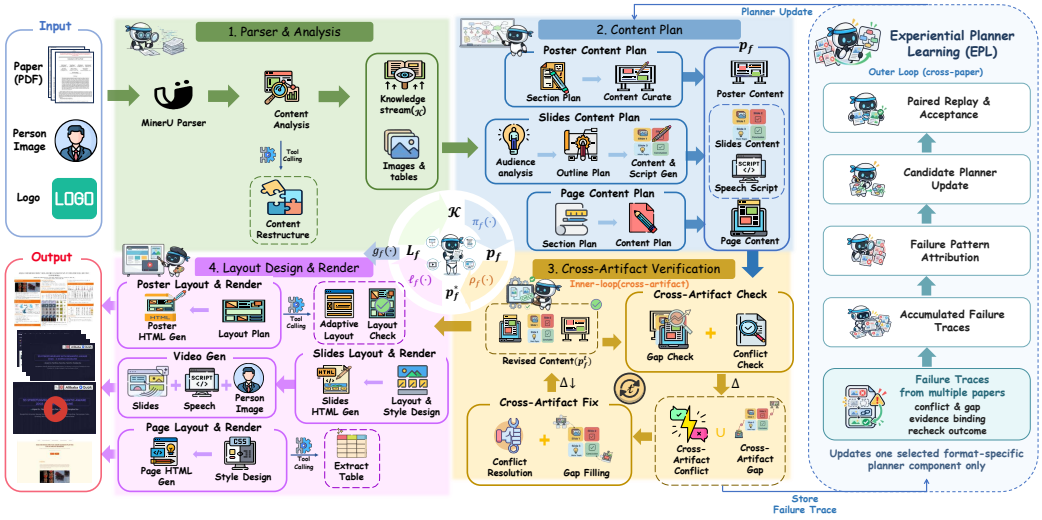


Figure 2: **Overview of OmniPresent.** Given a paper PDF, the parser and analysis module extracts structured resources and a shared knowledge stream. Format-specific planners draft content for the project page, poster, and slides, while a Cross-Artifact Verification loop detects and fixes conflicts across outputs. Layout and style agents then render each deliverable as executable HTML. An offline EPL aggregates the resulting evidence-grounded failure traces and updates the planner component.

3.2 Stage 1&2: Evidence-Grounded Content Planning

Shared evidence representation. MinerU (Wang et al., 2024a) extracts text, figures, tables, and references into $\mathcal{R}$. A Content Analysis Agent groups related resources, such as an ablation discussion and its table, into semantic clusters. The evidence index $\mathcal{E}$ links atomic claims to source spans. The resulting Knowledge Stream $\mathcal{K}$ consists of units (c, e, t): a statement c, an evidence pointer $e \in \mathcal{E}$, and a semantic type t such as contribution, method, or result (schema in App. I). All planners use these shared units, allowing their claims to be checked against the same evidence.

Poster Planner (π_{POSTER}) optimizes for *spatial information density*. The *Section Planning Agent* first establishes the logical framework by decomposing the paper into distinct thematic sections. Subsequently, the *Content Curation Agent* curates content for each section by identifying and synthesizing the most critical findings into fact-grounded bullet points and selecting the most visually compelling figures to ensure a high-impact and engaging presentation.

Slides Planner (π_{SLIDES}) optimizes for *adaptive narrative flow*. Unlike rigid summarization templates, *Audience Analysis Agent* synthesizes a high-level storyline by jointly conditioning on the paper genre and user constraints. This capability allows OmniPresent to generalize across diverse scientific domains and distinct target audiences. Guided by this narrative arc, the *Outline Planning Agent* sequences the logical beats, which drive the *Content & Script Gen Agent* to synthesize synchronized visuals and speech commentary strictly aligned with the specific presentation context.

Page Planner (π_{PAGE}) optimizes for *hierarchical engagement*. The *Section Planning Agent* performs adaptive restructuring based on the paper genre, consolidating fragmented chapters into a compact navigation structure. Subsequently, *Content Planning Agent* populates this schema by extracting the essence of technical descriptions and embedding figures/tables as inline visual tokens based on semantic relevance, ensuring a coherent linear reading flow.

3.3 Stage 3: Cross-Artifact Verification and Repair

Cross-Artifact Verification (CV) checks the plans jointly before rendering, using the source evidence to identify factual conflicts and missing coverage. Let $\mathcal{C}_f^{(t)}$ denote the semantic claims extracted from plan $p_f^{(t)}$ at repair iteration t.

Conflict check. The checker identifies unsupported claims and contradictions against $\mathcal{E}$ or other formats, producing $\Delta^{(t)}_{\text{conflict}}$. For example, different accuracy values must be checked against the corresponding source entities, metrics, and experimental conditions before they are classified as conflicting.

Coverage check. We adopt a salience-coverage constraint in which Slides form the narrative backbone and cover the highlights selected for Poster and Page:

$$\mathcal{I}(\textsc{Poster}) \cup \mathcal{I}(\textsc{Page}) \subseteq \mathcal{I}(\textsc{Slides}) \subseteq \mathcal{I}(\textsc{Paper}). \tag{2}$$

Here $\mathcal{I}$ denotes salient semantic units. The corresponding Gap check identifies claims missing from Slides:

$$\Delta^{(t)}_{\text{gap}} = (\mathcal{C}^{(t)}_{\textsc{Poster}} \cup \mathcal{C}^{(t)}_{\textsc{Page}}) \setminus \mathcal{C}^{(t)}_{\textsc{Slides}}. \tag{3}$$

Evidence-grounded repair. The fixer uses the diagnosed issues and their evidence to revise the affected plans:

$$p_f^{(t+1)} = \rho_f\Big(\mathbf{p}^{(t)}, \Delta^{(t)}_{\text{conflict}} \cup \Delta^{(t)}_{\text{gap}}, \mathcal{E}\Big). \tag{4}$$

Unsupported claims are corrected, missing coverage is added to Slides, and changes to the Slides plan trigger synchronized script updates. The cycle stops when no issues remain or after at most T repairs. A final check follows the last repair; any residual issues or invalid checks remain recorded. We denote the resulting plans by $\mathbf{p}^*$.

Recording verification experience. Each run records the paper and planner version, claim and evidence locations, issue type, severity, associated component, repair action, and recheck outcome in $\mathcal{T}_x$. These traces connect a diagnosed error to its source and attempted correction, providing the experience used by offline planner learning.

3.4 Stage 4: Layout and Rendering

The verified plans are converted into layout specifications L_f and then HTML artifacts:

$$L_f = \ell_f(p_f^*), \qquad h_f = g_f(L_f), \qquad f \in \mathcal{P}. \tag{5}$$

The Poster stream packs section content and checks overflow; the Slides stream applies a shared style and per-slide layouts; the Page stream organizes inline tables and figures within the navigation structure. A Video Generation module combines rendered slides, synchronized speech scripts, and a presenter image. Separating content verification from rendering allows semantic repairs before layout generation. Format-specific rendering details are provided in App. D.1.

3.5 Learning to Plan from Verification Experience

EPL converts recurring verification failures into reusable planner instructions through offline proposal and selection. Algorithm 2 in the appendix summarizes the procedure.

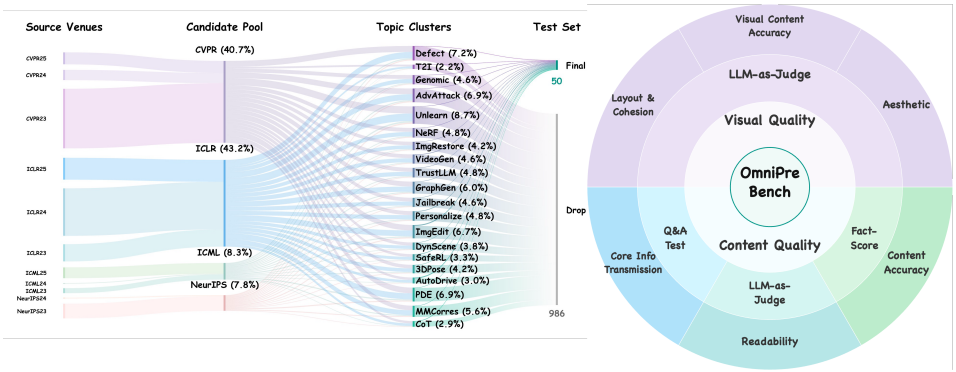
Experience aggregation and attribution. We group Meta-Train traces F_{train} by semantic role, format, and the first planner component responsible for an error or omission. Only evidence-resolved, attributable failures inform updates; shared-extraction errors and uncertain cases remain unattributed.

Localized instruction updates. For component (f, q), we propose revised instructions or constraints from its current instructions and attributed failure traces:

$$\theta'_{f,q} = \text{Propose}(\theta_{f,q}, F^{f,q}_{\text{train}}). \tag{6}$$

The candidate Θ' changes only this component, encoding a reusable rule while leaving paper-specific facts grounded in each paper's Knowledge Stream.

Paired selection and reuse. We compare candidate and retained configurations on the same Meta-Dev papers, fixing all other pipeline components, decoding settings, repair budget, and early-stop rule. Acceptance requires a lower mean pre-CV Conflict-plus-Gap count, and no decrease in final FactScore or QA for any format on paired development means. Rejected candidates leave the configuration unchanged; accepted updates are retained. The selected configuration is frozen before Final-Test and reused for future papers. Details are provided in App. B.



(a) Dataset Construction **(b) Evaluation Suite**

Figure 3: **Overview of OmniPreBench.** (**a**) Dataset construction and testset curation. Papers are collected from multiple venue-year sources, filtered into a qualified candidate pool, clustered by topic, and then stratified to form the final testset. (**b**) Evaluation suite design. The inner ring defines two top-level dimensions, *Content Quality* and *Visual Quality*. The middle ring shows the evaluation mechanisms. The outer ring lists the corresponding fine-grained criteria for each dimension.

4 OMNIPREBENCH

4.1 EVALUATION DATASET

Data Source. We construct OmniPreBench by crawling AI conference papers spanning 2023 to 2025, and retaining only entries with a complete suite of publicly available dissemination artifacts, as illustrated in Fig. 3a. Each benchmark sample includes the paper PDF together with its associated project page, poster, slide deck, and presentation video. After filtering by this resource-completeness criterion, we obtain 1,036 qualified papers from a curated set of major venues and years, covering a broad range of topics and modalities. Full venue-year coverage, filtering rules, and source statistics are provided in Appendix E.1.

Testset Selection. From the qualified pool, we curate a 50-paper testset that prioritizes both diversity and impact. We first compute paper embeddings using SPECTER2 and perform clustering to obtain a semantics-aware partition of topics. We then select papers to ensure broad cluster coverage and venue-year balance, while favoring higher-impact papers measured by citation counts. Detailed sampling procedures and hyperparameters are reported in Appendix E.1. For EPL, this fixed 50-paper test set is Final-Test. We sample disjoint 200-paper Meta-Train and 100-paper Meta-Dev subsets from the other 986 papers, stratified by topic cluster and venue–year.

4.2 EVALUATION METRICS

We evaluate generated artifacts using a multi-dimensional framework (Fig. 3b) with two top-level dimensions, *Content Quality* and *Visual Quality*. We report six metrics, computed with scalable LLM-assisted protocols, including fact extraction and verification, Q&A-based testing, and rubric-based judging. Details of metrics are reported in Appendix E.2.

Content Quality. We assess factual correctness, information preservation, and writing quality along three axes. ***(i)*** **Fact-Score** (Min et al., 2023) measures whether salient paper facts are preserved in the generated artifact. Using Gemini-3-Flash, we first extract a set of atomic facts $\mathcal{F}_{src}$ from the source paper, and then verify for each $f \in \mathcal{F}_{src}$ whether it is entailed by the generated content $\mathcal{G}$ (see Eq. 7 in App. E) ***(ii)*** **Q&A Test** evaluates core information transmission by generating question-answer pairs from the source paper and testing whether the questions can be answered correctly using only the generated artifact as context; we report accuracy over the question set. ***(iii)*** **Readability** is scored by an LLM-based rubric that assesses clarity, coherence, and fluency.

Visual Quality. We evaluate visual fidelity and presentation quality with VLM-assisted protocols. ***(iv)*** **Visual Content Accuracy** checks whether visual elements are correctly rendered and semantically aligned with surrounding text, including faithful formula rendering and relevance between figures/diagrams and adjacent descriptions. ***(v)*** **Layout & Cohesion** assesses structural organization,

such as grouping and hierarchy, balance of spatial distribution, and whether the layout supports a natural reading order. ***(vi)*** **Aesthetics** rates overall visual appeal, including typography, color harmony, and professional look-and-feel.

Interaction Validity. For project pages, this metric measures the proportion of functional links that correctly navigate to intended destinations. We use an automated browsing agent to simulate and verify success via VLM-based visual feedback (see App. E.2 for implementation details.)

5 EXPERIMENTS

We evaluate OmniPresent on OmniPreBench with a focus on suite-level quality, including evidence-grounded faithfulness, coverage of salient contributions, cross-format consistency, and format compliance. We compare against two competitive baseline families: *(i) strong closed-source models* that generate all artifacts end to end in a single pass, and *(ii) format-specialized agentic systems* designed for individual artifacts. In addition to overall performance, we conduct controlled comparisons that isolate the effect of unified parsing and planning, as well as the cross-artifact verify-and-repair loop. We further evaluate the offline EPL, asking whether recurring verification failures can be converted into localized planner updates that reduce pre-verification errors without changing the inner generation pipeline. To move beyond aggregate metrics, we also conduct qualitative case studies to visualize the suite-level advantages of OmniPresent. Due to the content limitation, detailed analyses are provided in App. L.

5.1 EXPERIMENTAL SETUP

Baselines and Implementation details. We compare OmniPresent against two categories of baselines. First, we evaluate strong end-to-end generation using closed-source models that directly produce all artifacts in a single pass, without explicit suite-level planning or verification. We include Gemini3-Flash (Team et al., 2023), GPT-5.2-mini (OpenAI, 2025), Grok4-Fast (xGr) and the latest GPT-6-Sol (OpenAI, 2026) as representative high-capability proprietary models. Second, we compare against existing format-specialized agentic systems that are designed for individual artifacts rather than a unified suite, including Paper2Poster (Pang et al., 2025) for posters, EvoPresent (Liu et al., 2025a) for slide decks and videos, and AutoPage (Ma et al., 2026b) for project pages. These methods provide strong, domain-specific pipelines tailored to each format. We further include PaperX Yu et al. (2026), a concurrent unified framework that uses a Scholar DAG as the intermediate representation to generate multi-format artifacts via adaptive graph traversal. Additionally, we include *Human-authored* artifacts as a competitive reference to evaluate the gap between automated generation and human standards. Further implementation details are provided in Appendix F

5.2 MAIN RESULTS

Tab. 1 reports overall results on slides, posters, and project pages under our content and visual metrics. Overall, the gains are systematic rather than incidental: OmniPresent improves both content faithfulness and visual presentation across formats while holding the backbone model fixed, indicating that the improvements come from the *suite-level OmniPresent pipeline* rather than model capacity.

***Findings 1*: OmniPresent outperforms strong specialized and partially unified baselines without sacrificing per-format quality.** Compared with format-specialized systems and the concurrent partially unified baseline PaperX (only poster and slides), OmniPresent achieves stronger per-format performance under the same Gemini-3-Flash backbone (Tab. 1). In slide generation, OmniPresent improves over EvoPresent and PaperX in *FactScore* (0.816 *v.s.* 0.666/0.652), *QA Accuracy* (0.751 *v.s.* 0.713/0.725), and *Aesthetics* (4.53 *v.s.* 3.38/3.98). In poster generation, it also surpasses Paper2Poster and PaperX, *e.g.*, in *FactScore* (0.755 *v.s.* 0.470/0.526) and *Visual Content Accuracy* (3.16 *v.s.* 2.88/2.03). This supports our core hypothesis in Sec. 1 that sharing a centralized *Knowledge Stream* (Sec. 3.2) and performing *Cross-Artifact Verification* (Sec. 3.3) does not just enforce coherence but actually improves individual artifact quality by grounding them in a unified evidence index.

***Findings 2*: OmniPresent ensures superior functional integrity and navigational reliability.** Beyond static visual quality, we evaluate the interactive functionality of project pages. As shown in Tab. 1, OmniPresent achieves a significantly higher *Interaction Success* rate compared to baselines.

Table 1: **Evaluation results across our full suite of OmniPreBench.** We report content quality (FactScore, Readability, QA), visual quality (visual content accuracy, aesthetics, layout), interaction success rate (Page only), and compression rate (higher indicates less redundancy). For each proprietary backbone, we compare end-to-end generation with OmniPresent instantiated using the same backbone; values in parentheses denote the absolute gains of OmniPresent over its matched end-to-end baseline.

Method	Content			Visual			Interaction	Compression
	FactScore↑	Readability↑	QA↑	Content acc↑	Aesthetics↑	Layout↑	Success↑	Compression Rate
Slides								
Human-authored	0.716	3.26	0.711	2.22	3.53	3.10	-	12.62
EvoPresent-Gemini-3-flash (Liu et al., 2025a)	0.666	3.38	0.713	1.74	3.38	2.60	-	24.96
PaperX-Gemini-3-flash (Yu et al., 2026)	0.652	3.39	0.725	2.63	3.98	3.09	-	9.35
GPT-5-mini	0.751	3.00	0.716	2.27	3.00	2.85	-	6.92
OmniPresent-GPT-5-mini	**0.830** (+0.079)	**3.20** (+0.20)	**0.724** (+0.008)	**2.84** (+0.57)	**3.89** (+0.89)	**3.49** (+0.64)	-	6.74
Gemini-3-flash	0.585	3.18	0.707	1.94	2.39	2.33	-	15.18
OmniPresent-Gemini-3-flash	**0.816** (+0.231)	**3.40** (+0.22)	**0.751** (+0.044)	**2.75** (+0.81)	**4.53** (+2.14)	**3.54** (+1.21)	-	8.55
Grok-4.1-fast	0.691	2.98	0.705	2.04	2.00	2.29	-	11.95
OmniPresent-Grok-4.1-fast	**0.827** (+0.136)	**3.08** (+0.10)	**0.723** (+0.018)	**2.12** (+0.08)	**3.08** (+1.08)	**2.33** (+0.04)	-	8.55
GPT-6-sol	0.762	3.19	0.720	2.33	3.14	2.99	-	6.68
OmniPresent-GPT-6-sol	0.801 (+0.039)	3.38 (+0.19)	0.756 (+0.036)	2.90 (+0.57)	3.93 (+0.79)	3.58 (+0.59)	-	11.37
Poster								
Human-authored	0.733	3.43	0.710	3.05	3.50	3.02	-	15.34
Paper2Poster-Gemini-3-flash (Pang et al., 2025)	0.470	3.50	0.659	2.88	2.05	3.00	-	5.33
PaperX-Gemini-3-flash (Yu et al., 2026)	0.526	3.51	0.664	2.03	3.02	2.68	-	23.62
GPT-5-mini	0.766	3.00	0.730	2.85	2.27	2.85	-	5.35
OmniPresent-GPT-5-mini	**0.779** (+0.013)	**3.46** (+0.46)	**0.746** (+0.016)	**3.26** (+0.41)	**3.79** (+1.52)	**2.96** (+0.11)	-	3.30
Gemini-3-flash	0.585	3.44	0.695	2.43	3.63	3.02	-	20.29
OmniPresent-Gemini-3-flash	**0.755** (+0.170)	**3.60** (+0.16)	**0.716** (+0.021)	**3.16** (+0.73)	**3.76** (+0.13)	**3.20** (+0.18)	-	3.75
Grok-4.1-fast	0.702	3.18	0.713	2.06	2.59	2.55	-	12.82
OmniPresent-Grok-4.1-fast	**0.754** (+0.052)	**3.22** (+0.04)	**0.717** (+0.004)	**3.10** (+1.04)	**3.73** (+1.14)	**3.04** (+0.49)	-	3.21
GPT-6-sol	0.771	3.13	0.736	3.18	3.22	2.91	-	3.97
OmniPresent-GPT-6-sol	0.782 (+0.011)	3.44 (+0.31)	0.749 (+0.013)	3.30 (+0.12)	3.82 (+0.60)	3.11 (+0.20)	-	12.55
Page								
Human-authored	0.721	3.12	0.706	3.40	3.20	3.46	1.00	10.89
AutoPage-Gemini-3-flash (Ma et al., 2026b)	0.705	3.10	0.701	2.90	2.07	3.35	0.66	3.28
GPT-5-mini	0.741	3.06	0.710	1.82	3.25	3.56	0.04	4.78
OmniPresent-GPT-5-mini	**0.801** (+0.060)	**4.48** (+1.42)	**0.714** (+0.004)	**3.28** (+1.46)	**3.45** (+0.20)	**3.66** (+0.10)	0.76 (+0.72)	8.84
Gemini-3-flash	0.485	3.08	0.699	1.14	3.46	3.02	0.02	13.13
OmniPresent-Gemini-3-flash	**0.782** (+0.297)	**3.62** (+0.54)	**0.712** (+0.013)	**3.12** (+1.98)	**3.59** (+0.13)	**3.96** (+0.94)	0.92 (+0.90)	6.98
Grok-4.1-fast	0.666	2.78	0.703	1.40	3.22	3.60	0.04	9.60
OmniPresent-Grok-4.1-fast	**0.791** (+0.125)	**3.66** (+0.88)	**0.747** (+0.044)	**3.42** (+2.02)	**3.63** (+0.41)	**3.94** (+0.34)	0.72 (+0.68)	12.38
GPT-6-sol	0.753	3.40	0.722	3.22	3.29	3.61	0.50	5.38
OmniPresent-GPT-6-sol	0.812 (+0.282)	4.22 (+0.82)	0.750 (+0.28)	3.29 (+0.07)	3.53 (+0.24)	3.99 (+0.38)	0.90 (+0.40)	9.96

This success is a direct result of *Evidence Indexing (Sec. 3.2):* the former ensures that every external link (*e.g.*, GitHub repository) is grounded in the source paper's metadata, while the latter precisely maps these resources to the correct HTML code during the rendering process.

***Findings 3*: Synchronous content-script planning enhances video faithfulness and information accessibility.** Tab. 2 compares OmniPresent against EvoPresent in the video task. Even when using the same backbone, our method consistently outperforms the baseline, particularly in *FactScore* and *QA Accuracy*, indicating that our unified knowledge stream provides more reliable grounding for temporal content in video. Furthermore, an inter-table analysis between Tab. 1 (see *slides*) and Tab. 2 (video) reveals an interesting *"readability boost"*: the Readability score for videos is consistently higher than that of the corresponding slides. While standard sequential pipelines derive narration only after slide content is fixed, OmniPresent generates *speaker notes* in tandem with the slide content planning phase. This integrated approach, as designed in our *Content Planning Agent* in Sec. 3.2, ensures a higher degree of alignment between visual bullet points and verbal explanations, which directly translates to improved readability.

5.3 Ablation Study

Ablation setting. To isolate the effect of our *Cross-Artifact Verification* (CV, Sec. 3.3), we conduct a controlled ablation by removing the CV module while keeping all other components, prompts, and decoding settings identical. We evaluate both configurations using Gemini-3-Flash. As shown in Tab. 3, we assess per-format performance across webpages, slides, and posters, focusing on *Content Quality* and *Compression* (Com), the latter of which measures the efficiency of information density.

***Findings 4*: CV is critical for suite-level faithfulness and information-dense research suite.** As reported in Tab. 3, removing the CV leads to a consistent decline in both *FactScore* and *QA Accuracy*. *This result directly validates the "Verify-and-Repair" loop* described in Sec. 3.3. By cross-referencing generated claims against the *Evidence Index* in Sec. 3.2, the CV identifies and rectifies hallucination drifts that often occur during independent format planning. The boost in QA scores further indicates that CV ensures higher information coverage, as the *Chief Editor* (see Sec. 3.3) proactively checks for missing salient contributions across the suite. Notably, while *Readability* remains stable, which suggests that the backbone's linguistic fluency is unaffected, the *Com* metric improves with CV.

Table 2: **Evaluation results of video.** We report content metrics (FactScore, Readability, QA) and compression for video artifacts. For each backbone, the improvements of OmniPresent in parentheses are computed relative to EvoPresent.

Method	FactScore↑	Readability↑	QA↑
Human-authored	0.849	3.51	0.759
EvoPresent-Gemini-3-flash	0.687	3.50	0.732
OmniPresent-Gemini-3-flash	0.865 (+0.179)	3.53 (+0.03)	0.773 (+0.041)
OmniPresent-GPT-5-mini	0.856 (+0.170)	3.39 (-0.11)	0.757 (+0.025)
OmniPresent-Grok-4.1-fast	0.813 (+0.126)	3.24 (-0.26)	0.746 (+0.014)

Table 3: **Ablation study on CV.** We ablate the CV on OmniPresent and report the absolute change w.r.t. w/o CV in parentheses.

Setting	Method	Content Quality			Compression
		FactScore↑	Readability↑	QA↑	Rate
Page	w/o CV	0.782	3.44	0.698	3.67
	w/ CV	0.810 (+0.028)	3.42 (-0.02)	0.712 (+0.014)	3.75
Slides	w/o CV	0.774	3.48	0.718	6.58
	w/ CV	0.816 (+0.042)	3.48 (+0.00)	0.750 (+0.032)	6.98
Poster	w/o CV	0.732	3.58	0.700	8.31
	w/ CV	0.755 (+0.023)	3.60 (+0.02)	0.716 (+0.016)	8.55
Video	w/o CV	0.793	3.52	0.732	6.81
	w/ CV	0.865 (+0.072)	3.53 (+0.01)	0.773 (+0.041)	6.03

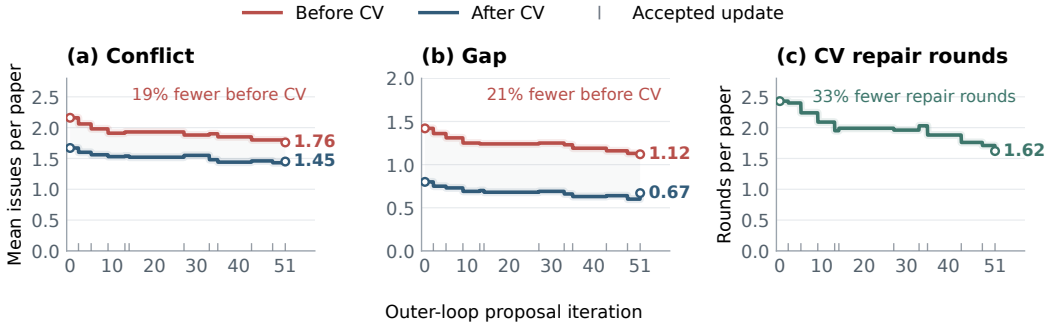


Figure 4: **Interaction of EPL and CV across proposal iterations** on 100 Meta-Dev papers with Gemini-3-Flash. Panels (a) and (b) track mean Conflict and Gap counts before and after CV; panel (c) reports mean executed repair rounds per paper. Iteration zero uses the original planners. Markers identify accepted updates. Both loops operate throughout adaptation: EPL improves initial planning, while CV repairs each generated suite.

This confirms that our verification process *not only filters inaccuracies but also prunes redundant or repetitive content, leading to a more concise and professional suite*.

5.4 Effectiveness of Experiential Planner Learning

We compare original and EPL-selected planners with CV enabled, holding all other pipeline components, decoding settings, repair budgets, and stopping rules fixed. Fig. 4 tracks development on 100 Meta-Dev papers with Gemini-3-Flash; selected planners are evaluated separately on 50 Test papers.

Finding 5: EPL improves initial planning and reduces repair effort. Across 51 proposal iterations, pre-CV Conflict and Gap counts decrease by 19% and 21%, respectively, showing that improvements arise before repair. Executed repair rounds decrease by 33% to 1.62 per paper, while final post-CV errors remain below their iteration-zero levels. These trends support the value of converting verification experience into reusable planner updates. On Final-Test, both Gemini-3-Flash and GPT-5-mini require approximately 31% fewer repair rounds and achieve higher mean final FactScore and QA (Tab. 4 in the appendix). Together, these results support improved planning on held-out papers with reduced subsequent repair effort. Full protocols and results are provided in App. C.

6 Conclusion

In this work, we introduce *OmniPresent*, a unified system that generates a coherent presentation suite from scientific papers in executable HTML, including project webpages, posters, slides, and videos. Shared evidence-grounded planning and cross-artifact verification align content across formats. Experiential Planner Learning turns verification experience across papers into reusable planner updates, improving initial plans and reducing subsequent repair effort. To support principled, large-scale evaluation of this suite-level task, we build OmniPreBench, a new benchmark of paired multi-format artifacts with a scalable evaluation protocol. Experiments demonstrate stronger content fidelity and presentation quality than competitive baselines. Together, these contributions advance reliable scientific communication through coherent generation and learning from verification experience.

AI USE STATEMENT

In this work, generative AI models are integral components of both the proposed OmniPresent framework and its evaluation pipeline. Their roles, configurations, and evaluation procedures are described in the main text and appendix. We also used generative AI tools for assistance with interpreting experimental results and debugging code used in our research implementation. We do not use generative AI tools to generate synthetic benchmark datasets, develop the method, design experiments, or derive theoretical or mathematical claims. Additionally, we used generative AI tools to identify and summarize related literature, revise manuscript text, and develop and refine plotting scripts and scientific figures. We have reviewed all AI-assisted work. Specifically, we reviewed the manuscript text, references, code changes, figures, and interpretations of the experimental results, and retained responsibility for all research decisions and conclusions. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies automated systems for scientific communication by generating coherent presentation suites, including project pages, posters, slides, and videos, from research papers. OmniPreBench is constructed from publicly available scientific papers and associated dissemination artifacts. The primary risks of such systems arise from inaccurate or misleading scientific communication. Generated artifacts may contain hallucinated or unsupported claims, misplaced emphasis, or cross-format inconsistencies, which could propagate scientific inaccuracies if released without careful review. Automated generation may also be misused to produce large volumes of low-quality or deceptive scientific materials. OmniPresent mitigates factual risks through evidence-grounded generation and cross-artifact verification designed to detect unsupported, contradictory, or missing content. Nevertheless, these mechanisms cannot guarantee error-free outputs. We therefore regard expert human review as necessary before public dissemination and encourage transparent disclosure of AI-generated content and compliance with the licensing terms of source materials.

REPRODUCIBILITY STATEMENT

The paper documents the benchmark and evaluation procedures needed to reproduce our study. Section 3 and Algorithms 1–2 describe the generation pipeline and planner-learning procedure, with additional details in Appendices I, D.1, and B. Section 4 and Appendix E document benchmark construction, data splits, and evaluation metrics. Section 5.1 describes the baselines and backbone controls; Appendix K provides the human evaluation protocol. The paired evaluation and statistical procedures for EPL are described in Section 5.4 and Appendix C. We plan to publicly release the benchmark data and code to support independent reproduction and further analysis.

REFERENCES

Grok 4 | xAI — x.ai. https://x.ai/news/grok-4. [Accessed 15-10-2025].

Harrison Chase. LangChain, October 2022. URL https://github.com/langchain-ai/langchain.

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!, 2025. URL https://arxiv.org/abs/2510.09558.

Yao Fu, Hao Peng, Tushar Khot, and Mirella Lapata. Improving language model negotiation with self-play and in-context learning from ai feedback. *arXiv preprint arXiv:2305.10142*, 2023.

Jiaxin Ge, Zora Zhiruo Wang, Xuhui Zhou, Yi-Hao Peng, Sanjay Subramanian, Qinyue Tan, Maarten Sap, Alane Suhr, Daniel Fried, Graham Neubig, et al. Autopresent: Designing structured visuals from scratch. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 2902–2911, 2025.

Yue Hu and Xiaojun Wan. Ppsgen: Learning to generate presentation slides for academic papers. In *IJCAI*, pp. 2099–2105, 2013.

Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, Qianli Ma, Nattapol Chanpaisit, Chenyang Si, Yuming Jiang, Yaohui Wang, Xinyuan Chen, Ying-Cong Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench++: Comprehensive and versatile benchmark suite for video generative models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. doi: 10.1109/TPAMI.2025.3633890.

Kyudan Jung, Hojun Cho, Jooyeol Yun, Soyoung Yang, Jaehyeok Jang, and Jaegul Choo. Talk to your slides: Language-driven agents for efficient slide editing, 2025. URL https://arxiv.org/abs/2505.11604.

Keshav Kumar and Ravindranath Chowdary. Slidespawn: An automatic slides generation system for research publications. *arXiv preprint arXiv:2411.17719*, 2024.

Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. Search-o1: Agentic search-enhanced large reasoning models. *arXiv preprint arXiv:2501.05366*, 2025.

Xin Liang, Xiang Zhang, Yiwei Xu, Siqi Sun, and Chenyu You. Slidegen: Collaborative multimodal agents for scientific slide generation. *arXiv preprint arXiv:2512.04529*, 2025.

Yen-Ting Lin and Yun-Nung Chen. Llm-eval: Unified multi-dimensional automatic evaluation for open-domain conversations with large language models. *arXiv preprint arXiv:2305.13711*, 2023.

Chengzhi Liu, Yuzhe Yang, Kaiwen Zhou, Zhen Zhang, Yue Fan, Yanan Xie, Peng Qi, and Xin Eric Wang. Presenting a paper is an art: Self-improvement aesthetic agents for academic presentations, 2025a. URL https://arxiv.org/abs/2510.05571.

Jingwei Liu, Ling Yang, Hao Luo, Fan Wang, Hongyan Li, and Mengdi Wang. Preacher: Paper-to-video agentic system, 2025b. URL https://arxiv.org/abs/2508.09632.

Qianli Ma, Chang Guo, Zhiheng Tian, Siyu Wang, Jipeng Xiao, Yuanhao Yue, and Zhipeng Zhang. Paper2rebuttal: A multi-agent framework for transparent author response assistance. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 46134–46166, 2026a.

Qianli Ma, Siyu Wang, Chen Yilin, Yinhao Tang, Yixiang Yang, Chang Guo, Bingjie Gao, Zhening Xing, Yanan Sun, and Zhipeng Zhang. Human-agent collaborative paper-to-page crafting. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 39963–39999, 2026b.

Jiacheng Miao, Joe R. Davis, Yaohui Zhang, Jonathan K. Pritchard, and James Zou. Paper2agent: Reimagining research papers as interactive and reliable ai agents, 2025. URL https://arxiv.org/abs/2509.06917.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12076–12100, 2023.

OpenAI. Gpt-5 system card. Technical report, August 2025. URL https://openai.com/index/gpt-5-system-card. Available at: https://cdn.openai.com/gpt-5-system-card.pdf.

OpenAI. Introducing GPT-6 Sol and Luna. OpenAI, September 2026. URL https://openai.com/index/introducing-gpt-6-sol-and-luna/.

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.

Kanya Paramita and Masayu Leylia Khodra. Tailored summary for automatic poster generator. In *2016 International Conference On Advanced Informatics: Concepts, Theory And Application (ICAICTA)*, pp. 1–6. IEEE, 2016.

Changle Qu, Sunhao Dai, Xiaochi Wei, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, Jun Xu, and Ji-Rong Wen. Tool learning with large language models: A survey. *Frontiers of Computer Science*, 19(8):198343, 2025.

Minju Seo, Jinheon Baek, Seongyun Lee, and Sung Ju Hwang. Paper2code: Automating code generation from scientific papers in machine learning. *arXiv preprint arXiv:2504.17192*, 2025.

Jingwei Shi, Zeyu Zhang, Biao Wu, Yanjie Liang, Meng Fang, Ling Chen, and Yang Zhao. Presentagent: Multimodal agent for presentation video generation. *arXiv preprint arXiv:2507.04036*, 2025.

Edward Sun, Yufang Hou, Dakuo Wang, Yunfeng Zhang, and Nancy XR Wang. D2s: Document-to-slide generation via query-based text summarization. *arXiv preprint arXiv:2105.03664*, 2021.

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.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Bin Wang, Chao Xu, Xiaomeng Zhao, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Rui Xu, Kaiwen Liu, Yuan Qu, Fukai Shang, Bo Zhang, Liqun Wei, Zhihao Sui, Wei Li, Botian Shi, Yu Qiao, Dahua Lin, and Conghui He. Mineru: An open-source solution for precise document content extraction, 2024a. URL https://arxiv.org/abs/2409.18839.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, et al. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345, 2024b.

Yidong Wang, Qi Guo, Wenjin Yao, Hongbo Zhang, Xin Zhang, Zhen Wu, Meishan Zhang, Xinyu Dai, Qingsong Wen, Wei Ye, et al. Autosurvey: Large language models can automatically write surveys. *Advances in neural information processing systems*, 37:115119–115145, 2024c.

Yixuan Weng, Minjun Zhu, Guangsheng Bao, Hongbo Zhang, Jindong Wang, Yue Zhang, and Linyi Yang. Cycleresearcher: Improving automated research via automated review. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=bjcsVLoHYs.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, et al. Autogen: Enabling next-gen llm applications via multi-agent conversations. In *First Conference on Language Modeling*, 2024.

Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, et al. The rise and potential of large language model based agents: A survey. *Science China Information Sciences*, 68(2):121101, 2025.

Lingao Xiao, Yalun Dai, Yangyu Huang, Qihao Zhao, Wenshan Wu, Hugo He, Ruishuo Chen, Jin Jiang, Qianli Ma, Jiahuan Zhang, Xin Zhang, Ying Xin, Yang Ou, Yan Xia, Scarlett Li, Longbo Huang, Zhipeng Zhang, Yang He, Yap Kim Hui, and Yan Lu. Researchstudio-reel: Automate the last mile of research from paper to poster, video, and blog. *arXiv preprint arXiv:2607.04438*, 2026. URL https://arxiv.org/abs/2607.04438.

Sheng Xu and Xiaojun Wan. Posterbot: A system for generating posters of scientific papers with neural models. 2022.

Xiaojie Xu, Xinli Xu, Sirui Chen, Haoyu Chen, Fan Zhang, and Ying-Cong Chen. Pregenie: An agentic framework for high-quality visual presentation generation, 2025. URL https://arxiv.org/abs/2505.21660.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

Tao Yu, Minghui Zhang, Zhiqing Cui, Hao Wang, Zhongtian Luo, Shenghua Chai, Junhao Gong, Yuzhao Peng, Yuxuan Zhou, Yujia Yang, Zhenghao Zhang, Haopeng Jin, Xinming Wang, Yufei Xiong, Jiabing Yang, Jiahao Yuan, Hanqing Wang, Hongzhu Yi, Yan Huang, and Liang Wang. Paperx: A unified framework for multimodal academic presentation generation with scholar dag, 2026. URL https://arxiv.org/abs/2602.03866.

Zhilin Zhang, Xiang Zhang, Jiaqi Wei, Yiwei Xu, and Chenyu You. Postergen: Aesthetic-aware paper-to-poster generation via multi-agent llms, 2025. URL https://arxiv.org/abs/2508.17188.

Qihao Zhao, Yangyu Huang, Yalun Dai, Lingao Xiao, Jianjun Gao, Xin Zhang, Wenshan Wu, Scarlett Li, Yang He, Yan Lu, and Yap Kim Hui. Researchstudio-idea: An evidence-grounded research-ideation skill suite from ml conference outcomes. *arXiv preprint arXiv:2607.04439*, 2026. URL https://arxiv.org/abs/2607.04439.

Hao Zheng, Xinyan Guan, Hao Kong, Jia Zheng, Weixiang Zhou, Hongyu Lin, Yaojie Lu, Ben He, Xianpei Han, and Le Sun. Pptagent: Generating and evaluating presentations beyond text-to-slides. *arXiv preprint arXiv:2501.03936*, 2025.

Minjun Zhu, Yixuan Weng, Linyi Yang, and Yue Zhang. Deepreview: Improving llm-based paper review with human-like deep thinking process, 2025a. URL https://arxiv.org/abs/2503.08569.

Zeyu Zhu, Kevin Qinghong Lin, and Mike Zheng Shou. Paper2video: Automatic video generation from scientific papers, 2025b. URL https://arxiv.org/abs/2510.05096.

Algorithm 1 Single-Paper Generation

Require: Paper x; planner configuration Θ; maximum repair rounds T; presenter image a
Ensure: Presentation suite $\mathcal{H}$ (or failure status) and verification trace $\mathcal{T}_x$
$(\mathcal{R}, \mathcal{E}) \leftarrow \mathrm{Parser}(x)$; $\mathcal{K} \leftarrow \mathrm{Distill}(\mathcal{R}, \mathcal{E})$
$\mathcal{P} \leftarrow \{\textsc{Slides}, \textsc{Poster}, \textsc{Page}\}$; $\mathcal{T}_x \leftarrow \mathrm{InitTrace}(x, \Theta)$
for each $f \in \mathcal{P}$ **do**
  $p_f^{(0)} \leftarrow \pi_f(\mathcal{K}; \theta_f)$
  ▷ Slides plan includes its speech script
end for
for $t = 0, \ldots, T$ **do**
  $\mathbf{p}^{(t)} \leftarrow \{p_f^{(t)}\}_{f \in \mathcal{P}}$
  $\mathcal{C}^{(t)} \leftarrow \{\mathrm{ExtractClaims}(p_f^{(t)})\}_{f \in \mathcal{P}}$
  $\Delta_C^{(t)} \leftarrow \mathrm{ConflictCheck}(\mathcal{C}^{(t)}, \mathcal{E})$
  $\Delta_G^{(t)} \leftarrow (\mathcal{C}^{(t)}_{\textsc{Poster}} \cup \mathcal{C}^{(t)}_{\textsc{Page}}) \setminus \mathcal{C}^{(t)}_{\textsc{Slides}}$
  Append this check, claim/evidence locations, and recheck outcomes to $\mathcal{T}_x$
  if the check is invalid or times out **then**
    return failure status and $\mathcal{T}_x$
  ▷ Not a zero-error suite
  end if
  if $\Delta_C^{(t)} \cup \Delta_G^{(t)} = \varnothing$ **or** $t = T$ **then**
    break
  ▷ The last repair is always rechecked
  end if
  $\Delta^{(t)} \leftarrow \Delta_C^{(t)} \cup \Delta_G^{(t)}$
  $\mathbf{p}^{(t+1)} \leftarrow \mathrm{Fix}(\mathbf{p}^{(t)}, \Delta^{(t)}, \mathcal{E})$
  Synchronize the speech script with the repaired Slides plan
  Append repair actions and affected component locations to $\mathcal{T}_x$
end for
$\mathbf{p}^* \leftarrow \mathbf{p}^{(t)}$
  ▷ Residual issues remain recorded if the budget is exhausted
for each $f \in \mathcal{P}$ **do**
  $L_f \leftarrow \ell_f(p_f^*)$; $h_f \leftarrow g_f(L_f)$
end for
$s^* \leftarrow \mathrm{Script}(p^*_{\textsc{Slides}})$
$h_{\textsc{Video}} \leftarrow \mathrm{VideoGen}(h_{\textsc{Slides}}, s^*, a)$
return $\mathcal{H} = \{h_f\}_{f \in \mathcal{P}} \cup \{h_{\textsc{Video}}\}$ and $\mathcal{T}_x$

Algorithm 2 Learning from Experience (EPL)

Require: Initial planner configuration Θ_0; Meta-Train traces F_{train}; Meta-Dev papers $\mathcal{D}_{\mathrm{dev}}$; proposal budget B; maximum repair rounds T
Ensure: Selected planner configuration Θ^*
$\Theta \leftarrow \Theta_0$
$\mathcal{U} \leftarrow \mathrm{AttributeAndGroup}(F_{\mathrm{train}})$
  ▷ Group evidence-resolved failures by semantic role, format, and component
  ▷ Exclude shared-extraction errors and unresolved attributions from proposals
$M \leftarrow \mathrm{Evaluate}(\Theta, \mathcal{D}_{\mathrm{dev}}, T)$ using Algorithm 1
for $k = 1, \ldots, B$ **do**
  if $\mathcal{U} = \emptyset$ **then**
    break
  end if
  Select a component (f, q) and a target issue $j \in \{C, G\}$ from $\mathcal{U}$
  $\theta'_{f,q} \leftarrow \mathrm{Propose}(\theta_{f,q}, \mathcal{U}_{f,q,j})$
  $\Theta' \leftarrow \Theta$ with only component (f, q) replaced by $\theta'_{f,q}$
  $M' \leftarrow \mathrm{Evaluate}(\Theta', \mathcal{D}_{\mathrm{dev}}, T)$ using the same fixed pipeline
  Compare M' and M on papers valid in both conditions; account for all failed attempts
  if paired metrics are unavailable **then**
    Mark the proposal unevaluable and retain Θ
  else if $\overline{C+G}' < \overline{C+G}$ **then**
    if $\overline{\mathrm{FactScore}}'_g \geq \overline{\mathrm{FactScore}}_g$ and $\overline{\mathrm{QA}}'_g \geq \overline{\mathrm{QA}}_g$ for every $g \in \mathcal{P}$ **then**
      $\Theta \leftarrow \Theta'$; $M \leftarrow M'$
  ▷ Accept and retain this planner version
    end if
  end if
  Record the proposal and decision
  ▷ Rejected proposals leave Θ unchanged
end for
return $\Theta^* \leftarrow \Theta$
  ▷ Freeze before Final-Test; reuse for future papers

A Demo

We provide an **anonymous** project gallery in the supplementary material (see the `.zip` file), which contains an HTML page showcasing representative. Open the *html* file to see more cases, including posters, pages, slides, and a video generated by OmniPresent.

B Experiential Planner Learning Protocol

Each EPL rollout records the paper and planner version, claim and evidence locations, CV issue type, responsible component, repair action, and recheck outcome. Only evidence-resolved traces inform proposals; invalid checks enter the failure accounting.

B.1 Component attribution and localized proposals

We attribute an issue to the first component that contains or omits the relevant claim: Slides storyline and outline omissions map to Audience Analysis and Outline Planning; unsupported Poster claims map to Content Curation, and Page hierarchy gaps to Section Planning. Numeric conflicts follow the first broken source entity–metric–condition binding; shared-extraction errors and uncertain evidence remain unattributed. Each proposal edits one component and names its CV target.

B.2 PAIRED REPLAY AND ACCEPTANCE

Retained and candidate configurations replay on the same Meta-Dev papers with fixed resources, Knowledge Stream, model settings, maximum T, and early-stop rule; executed repair counts may differ. We compare pre- and post-CV C and G, repair rounds, and final per-format FactScore and QA (Sec. 5.4). The mean pre-CV Conflict-plus-Gap count is $\overline{C+G} = |\mathcal{D}_{\text{dev}}|^{-1} \sum_{x \in \mathcal{D}_{\text{dev}}} (C_x^{\text{pre}} + G_x^{\text{pre}})$ over the paired Meta-Dev papers. Acceptance requires this mean to decrease and no decrease in final FactScore or QA for any format on paired paper means. Rejection preserves the incumbent; Final-Test compares only the original and selected complete configurations.

Algorithmic specification. Algorithm 2 uses Meta-Train traces from Algorithm 1. $\mathcal{P}$ denotes the three planner formats; bars denote paired means over valid paper pairs, including the per-format final quality comparisons used for acceptance. Evaluation records retain per-paper pre-/post-CV counts, repair rounds, final per-format scores, and failures. Parsed resources, Knowledge Stream, backbone, checker, fixer, renderer, decoding settings, and T remain fixed. Recorded runs are reused for repeated paper–configuration pairs; new configurations start from fresh plans. Final-Test is used only after selection.

C ADDITIONAL EPL ANALYSES

This section details the evaluation protocol and results supporting the EPL analysis in Sec. 5.4. We distinguish the independent Final-Test comparison from the Meta-Dev trajectory used during planner selection.

C.1 EVALUATION SETUP

We compare original and EPL-selected planners on the same 50 Final-Test papers, with CV enabled in both conditions. EPL runs separately for Gemini-3-Flash and GPT-5-mini using the shared paper-level Meta-Train/Meta-Dev/Final-Test partition in Sec. 4.1. Meta-Train traces drive proposals, Meta-Dev supports selection, and Final-Test is evaluated only after the planner configuration is frozen.

Within each backbone, only planner instructions differ. Parsed resources, Knowledge Stream, generator, checker, fixer, renderer, decoding settings, maximum repair budget T, and early-stop rule remain fixed. Each paper–configuration pair is run once. Thus, the maximum repair budget is matched, while the number of executed repairs may differ.

C.2 METRICS AND RUN ACCOUNTING

Following Sec. 3.3, $C = |\Delta_{\text{conflict}}|$ counts unsupported or contradictory claims, and $G = |\Delta_{\text{gap}}|$ counts missing cross-artifact coverage. We report per-paper means before the first repair and after the final check, deduplicating issues within each check by claim and artifact location. A repair round comprises one fixer pass followed by rechecking. Initially issue-free suites execute zero repairs; suites reaching T repairs receive a final check to record residual issues. Final FactScore and QA are evaluated on rendered artifacts and averaged equally over Slides, Poster, and Page.

Across attempted paper-runs pooled over both backbones, invalid or timed-out checks and missing artifacts each occur in 0% of runs under both conditions. Budget exhaustion with nonzero C or G occurs in 21% of runs with the original planners and 12% with EPL. Quality and issue-count comparisons use papers valid in both conditions; failure rates cover all attempts. Invalid or missing outputs are not assigned zero error counts, and valid budget-exhausted runs retain their residual issues.

C.3 RESULTS ON HELD-OUT PAPERS

Table 4 compares the original and selected planners on Final-Test. Under both backbones, EPL lowers pre-CV Conflict and Gap counts, indicating improved initial planning. After CV, residual issue counts are also lower and mean final FactScore and QA are higher. Executed repair rounds decrease from 2.42 to 1.66 for Gemini-3-Flash and from 2.30 to 1.58 for GPT-5-mini, approximately 31% in

both cases. These comparisons evaluate whether planner updates selected on Meta-Dev transfer to held-out papers.

Table 4: Effect of EPL on the 50-paper Final-Test set. Within each backbone, the original and EPL-selected planners use the same maximum CV repair budget T and early-stop rule. Conflict, Gap, and executed repair rounds are single-run paper-level means; final FactScore and QA average Slides, Poster, and Page and use the same Gemini-3-Flash judge. Parentheses show EPLminus original under the matched backbone.

Setting	Pre-CV Conflict↓	Pre-CV Gap↓	Post-CV Conflict↓	Post-CV Gap↓	CV repair rounds↓	Final FactScore↑	Final QA↑
Gemini-3-Flash							
Original planners + CV	2.20	1.46	1.70	0.82	2.42	0.784	0.726
EPLplanners + CV	**1.82** (-0.38)	**1.16** (-0.30)	**1.54** (-0.16)	**0.68** (-0.14)	**1.66** (-0.76)	**0.806** (+0.022)	**0.739** (+0.013)
GPT-5-mini							
Original planners + CV	1.96	1.32	1.62	0.78	2.30	0.803	0.728
EPLplanners + CV	**1.70** (-0.26)	**1.04** (-0.28)	**1.42** (-0.20)	**0.72** (-0.06)	**1.58** (-0.72)	**0.810** (+0.007)	**0.743** (+0.015)

C.4 Paired-bootstrap uncertainty

For each backbone, we resample the 50 matched Final-Test paper pairs with replacement and recompute EPL-minus-original differences in Conflict, Gap, repair rounds, FactScore, and QA. The 95% intervals use the 2.5th and 97.5th percentiles of 10,000 paired bootstrap samples (Table 5). Pairing preserves paper difficulty across conditions. These intervals quantify variation across papers, not decoding variability from repeated runs.

Table 5: Paired-bootstrap 95% uncertainty intervals for the Final-Test EPLcomparison. Each entry is the EPL-minus-original difference; intervals are computed over the same 50 paper pairs used in Table 4.

Metric	Gemini-3-Flash Δ [95% CI]	GPT-5-mini Δ [95% CI]
Pre-CV Conflict	-0.38 [-0.66, -0.12]	-0.26 [-0.50, -0.04]
Pre-CV Gap	-0.30 [-0.50, -0.10]	-0.28 [-0.54, -0.06]
Post-CV Conflict	-0.16 [-0.38, $+0.04$]	-0.20 [-0.44, $+0.02$]
Post-CV Gap	-0.14 [-0.30, $+0.02$]	-0.06 [-0.24, $+0.14$]
CV repair rounds	-0.76 [-1.10, -0.44]	-0.72 [-1.00, -0.42]
Final FactScore	$+0.022$ [$+0.005$, $+0.040$]	$+0.007$ [-0.016, $+0.027$]
Final QA	$+0.013$ [-0.004, $+0.034$]	$+0.015$ [-0.007, $+0.032$]

C.5 Development trajectory and interaction with CV

For Gemini-3-Flash, Fig. 4 tracks the retained configuration over 51 proposal iterations on 100 Meta-Dev papers, starting from the original planners. Each new configuration generates fresh initial plans from the same parsed resources and executes the complete CV loop. Repaired plans are never carried into a subsequent configuration. Rejected proposals preserve the retained configuration, and repeated paper–configuration pairs reuse their recorded measurements.

Across this trajectory, pre-CV Conflict decreases from 2.16 to 1.76 and Gap from 1.42 to 1.12. Mean repair rounds decrease from 2.43 to 1.62, with local rebounds after some updates. Post-CV errors remain lower than pre-CV errors throughout. The two loops therefore contribute at different scopes: CV repairs the current suite, while EPL improves the planner supplying future initial plans. This trajectory describes development-time selection; independent generalization and final content quality are evaluated in Table 4.

C.6 FORMAT-SPECIFIC UPDATES AND SELECTION DIAGNOSTICS

Table 6 compares accepted Slides (H-S), Poster (H-P), and Page (H-G) configurations separately and jointly (H-All) against the original configuration H0 on Meta-Dev, using Gemini-3-Flash and fixed CV settings. FactScore and QA average the three formats; Inclusion averages the two pairwise scores defined in App. H. Table 7 summarizes proposal counts, acceptance rates, and changes in pre-CV errors. These diagnostics examine component-specific effects during development and complement the held-out comparison.

Table 6: Format-specific EPLupdate analysis on Meta-Dev with Gemini-3-Flash as the generation backbone. Conflict and Gap are mean pre-CV issues per paper; FactScore, QA, and Inclusion are evaluated after CV.

Setting	Conflict↓	Gap↓	FactScore↑	QA↑	Inclusion↑
H0 (original)	2.16	1.42	0.780	0.723	0.863
H-S (Slides)	1.96	1.27	0.790	0.729	0.873
H-P (Poster)	2.00	1.35	0.784	0.724	0.864
H-G (Page)	2.17	1.36	0.782	0.728	0.869
H-All (combined)	**1.76**	**1.12**	**0.803**	**0.740**	**0.879**

Table 7: Accepted component updates in the measured 51-proposal Meta-Dev trajectory with Gemini-3-Flash. Shares are independently rounded fractions of the 11 accepted updates; proposal rounds are those in the longitudinal curve.

Planner	Accepted updates	Share	First round	Last round
Slides	5	45.5%	5	51
Poster	3	27.3%	2	43
Page	3	27.3%	14	48
All accepted updates	11	100.0%	2	51

C.7 CROSS-FORMAT CONFLICT BREAKDOWN

Table 8 reports Poster–Slides and Page–Slides contradiction counts for Gemini-3-Flash on Final-Test, alongside the suite-level Gap counts from Table 4. Both pairwise contradiction counts decrease before CV and are further reduced after repair. The pairwise columns cover contradictions between the listed formats; the full Conflict metric also includes unsupported claims.

Table 8: Single-run paper-level means of pairwise contradiction and suite-level Gap counts on Final-Test with Gemini-3-Flash as the generation backbone. The pairwise counts use the existing Conflict Check; the full Conflict total also includes unsupported claims (Table 4).

Setting	Poster–Slides↓	Page–Slides↓	Gap↓
Original, before CV	0.48	0.66	1.46
EPL, before CV	0.40	0.54	1.16
Original, after CV	0.42	0.44	0.82
EPL, after CV	**0.34**	**0.38**	**0.68**

D DETAILS OF OMNIPRESENT

D.1 DETAILS OF STAGE 4: LAYOUT DESIGN AND RENDER

Stage 4 transforms the verified plan p_f^* into visual artifacts via a visualization layer decoupled from planning. Utilizing Adaptive Layout Module (ℓ_f) and Adaptive Render Module (g_f), this stage

accommodates verification-based modifications (Sec. 3.3) without breaking rigid templates. ℓ_f maps p_f^* to a flexible specification L_f through specialized streams:

Poster Stream. The *Layout Plan Agent* organizes section interiors by information density, then uses a Compact Packer tool to optimize visual block arrangement (masonry layout) for minimal area. A Layout Check tool then checks the layout, adjusting fonts or margins to fix overflows.

Slides Stream. Managed by the *Layout & Style Design Agent*, this stream ensures visual hierarchy and consistency. The agent first executes **Style Design** to unify aesthetics based on p_f^* and audience analysis (Sec. 3.2). It then performs **Layout Design** for each slide, structuring space according to narrative needs to optimize information delivery.

Page Stream. The Style Design Agent improves readability and supports personalized styles. It employs the "Table Extraction" tool to reconstruct tables into web-compliant formats before styling. Complex layout design is bypassed as HTML naturally adapts to webpage structures.

Finally, g_f generates executable HTML from L_f. A Video Gen module then synthesizes rendered slides with speech scripts (from p_f^*) and a presenter image into a synchronized audio-visual video.

E Details of OmniPreBench

E.1 Construction of the Dataset

Data Sources. We construct our benchmark by initially collecting a broad set of 31,521 candidate papers from 13 major AI conference (*e.g.*, AAAI, ACL, CVPR, NeurIPS, ICLR, ICML) tracks spanning 2023 to 2025, stored in our raw data repository. However, to enable a comprehensive multimodal evaluation, we enforce a strict resource completeness criterion: valid benchmark entries must feature not only the full paper PDF but also a suite of ground-truth multimedia assets, including a project webpage, a poster, a slide deck, and a presentation video.

Due to the varying availability of these open-access resources across different venues, we narrow our focus to a curated subset of 11 specific venue-year combinations: *CVPR* (2023-2025), *ICLR* (2023-2025), *ICML* (2023-2025), and *NeurIPS* (2023-2024). Unlike other venues such as AAAI or ACL, which were excluded from the final selection due to sparse multimedia hosting, these chosen conferences consistently utilize platforms (*e.g.*, YouTube, OpenReview, SlidesLive) that archive high-quality supplementary materials. This filtering process yields a high-quality candidate pool of 1,036 papers, ensuring that every sample in our benchmark possesses the necessary ground-truth data for cross-verifying generated content against multiple modalities.

Data Analysis. The selected candidate pool represents a diverse cross-section of the AI landscape, capturing distinct shifts in community focus over the past three years. In 2023, the field was dominated by foundational breakthroughs in *Reasoning* (*e.g.*, Chain-of-Thought, Cluster 20) and *Personalized Image Synthesis* (*e.g.*, DreamBooth, Cluster 12). This focus expanded in 2024 to address *Multimodal Safety & Robustness* (*e.g.*, Jailbreak Attacks, Cluster 11) and specialized scientific applications (*e.g.*, Genomics, Cluster 3). Most recently, in 2025, we observe a significant pivot towards *Temporal & Embodied Intelligence*, characterized by a surge in Video Generation and Robotic Manipulation research. This thematic progression ensures that agents are challenged to synthesize and visualize diverse technical narratives, ranging from abstract theoretical frameworks to dynamic visual demonstrations into coherent multimodal artifacts.

Testset Filtering. To establish a robust and representative benchmark, we curate a core test set of 50 papers from this qualified pool of 1,036 works. We employ a *semantics-aware stratified sampling strategy*, grounded in SPECTER2[1] embeddings and K-Means clustering. We set $K = 20$ to empirically balance the granularity of topic discovery with semantic coherence, ensuring a fine-grained coverage of distinct research sub-domains. To ensure topological diversity, we first enforce a strict coverage of all semantic clusters by selecting the highest-cited non-survey paper from each. This is complemented by a source consistency check, which mandates the inclusion of representative works from every selected venue and year to mitigate potential domain or temporal shifts. Finally, to align the dataset with community research trends without compromising balance, we fill the remaining slots using a round-robin mechanism across high-impact clusters (ranked by average

[1] https://huggingface.co/allenai/specter2

citation). This approach effectively prevents the over-concentration of dominant topics (*e.g.*, GenAI) while maintaining comprehensive semantic breadth.

Testset for EPL. For EPL, this fixed 50-paper test set is Final-Test. We sample disjoint 200-paper Meta-Train and 100-paper Meta-Dev subsets from the other 986 papers, stratified by topic cluster and venue–year. The remaining 686 papers are unused by EPL; each paper's artifacts remain in one split.

E.2 Design of the metrics

In this section, we report design protocols of the OmniPreBench metrics in detail:

Content Quality Given the multimodal nature of the output, we evaluate three content dimensions:

(I) **Content Accuracy:** To quantify the faithfulness and completeness of the generation, we propose a key point coverage metric adapted from **FactScore**. We utilize **Gemini-3-Flash** to perform a two-stage evaluation: (1) *Extraction*: The model extracts a set of critical arguments and atomic facts, denoted as $\mathcal{F}_{src}$, from the original academic paper. (2) *Verification*: The model verifies whether each extracted fact $f \in \mathcal{F}_{src}$ is adequately covered in the generated artifacts (slides, posters, or videos). The final accuracy score is calculated as the ratio of covered facts to total key facts:

$$\text{Fact-Score} = \frac{1}{|\mathcal{F}_{src}|} \sum_{f \in \mathcal{F}_{src}} \mathbb{I}(\text{entails}(\mathcal{G}, f)) \,, \tag{7}$$

where $\mathcal{G}$ represents the generated content (*e.g.*, slide, poster), and $\mathbb{I}(\cdot)$ is an indicator function that equals 1 if the fact f is entailed by $\mathcal{G}$, and 0 otherwise.

(II) **Core Info Transmission:** To measure whether key information is effectively preserved, we employ a **Q&A Test** pipeline. An LLM generates question-answer pairs based on the source text; we then evaluate if these questions can be correctly answered solely using the generated output.

(III) **Readability:** We leverage an **LLM-as-Judge** Huang et al. (2025); Lin & Chen (2023) to score the linguistic quality. This dimension evaluates the text's clarity, logical flow, and overall fluency, ensuring the output is coherent and easy to follow.

Visual Quality Given the multimodal nature of the output, we utilize Vision-Language Models (VLMs) as judges to evaluate three visual dimensions:

(I) **Visual Content Accuracy:** This metric evaluates technical rendering and semantic alignment. It specifically checks: (1) whether complex elements (*e.g.*, mathematical formulas) are rendered without artifacts; and (2) if the visual elements are semantically relevant to the adjacent text.

(II) **Layout & Cohesion:** We employ an VLM-as-a-Judge to assess the structural organization. This includes evaluating whether elements are grouped logically, if the spatial distribution is balanced, and if the layout facilitates a natural reading order.

(III) **Aesthetic:** We calculate an overall aesthetic score to simulate human design judgment. This dimension evaluates the artistic style, color harmony, and the professional look-and-feel of the final document.

Interaction Validity. For project pages, we assess the functional integrity of interactive elements, specifically the validity of external and navigational links. To evaluate the functional reliability of the generated project pages, we developed an automated pipeline using `browser-use`[2]. For each page, the agent identifies and clicks all external links and navigational buttons. We capture browser snapshots immediately before and after each click. We use *Gemini-3-Flash* model to analyze these pairs of snapshots to determine if the navigation was successful (*e.g.*, reaching a valid external repository, a paper PDF, or the correct internal anchor) or resulted in a dead link/error page. The *Interaction Success* score is the ratio of successful navigations to total interactive elements.

[2]https://github.com/browser-use/browser-use

F IMPLEMENTATION DETAILS.

To ensure fair comparisons, we control for the underlying foundation model whenever possible. When comparing with proprietary end-to-end LLM baselines, we instantiate OmniPresent using the *same* proprietary model as its backbone, so that gains can be attributed to our unified parsing, planning, and cross-artifact verification rather than model capacity. When comparing with prior format-specialized systems, we also align the backbone model used in their implementations with the one used in OmniPresent, ensuring that improvements primarily reflect differences in system design and suite-level verification rather than differences in base model choice. For EPL, the parser, Knowledge Stream, verifier, renderer, and decoding settings are held fixed across planner versions; only the selected format-specific component is changed.

G MORE FINDINGS

***Findings 6*: OmniPresent consistently improves both content and visual quality over end-to-end baselines.** Across all three formats and evaluated backbones, OmniPresent yields systematic gains in factuality (FactScore), information transmission (QA), and visual presentation. As shown in Tab. 1, when using GPT-5-mini as the backbone for slides, our pipeline increases the *FactScore from 0.751 to 0.830* ($+0.079$) and significantly boosts *Aesthetics from 3.00 to 3.89* ($+0.89$). These improvements are especially pronounced in visual metrics (Content Accuracy and Aesthetics), validating that our *Adaptive Layout Agent* effectively translates shared knowledge into format-compliant HTML, reducing the hallucination under layout pressure common in end-to-end generation.

***Findings 7*: Gains are larger when the base generator is weaker, supporting OmniPresent as a backbone-agnostic system-level improvement.** The relative improvements of OmniPresent are most striking when paired with less capable models like Gemini-3-Flash. For slides, OmniPresent elevates Gemini's FactScore from *0.585 to 0.816* (a massive $+0.231$ gain) and its Aesthetics from *2.39 to 4.53*. While stronger backbones like GPT-5-mini also benefit, the margins are narrower, indicating that our "Plan-Verify-Render" architecture serves as a robust constraint framework that bounds the stochastic nature of LLMs through explicit structural planning and verification loops. It compensates for the inherent reasoning gaps in weaker models by providing explicit structural constraints and a "Chief Editor" loop to fix omissions and inconsistencies.

***Findings 8*: OmniPresent achieves professional-grade quality, occasionally surpassing human-authored standards in rigor and accuracy.** A key highlight of our evaluation is the comparison between OmniPresent and *human-authored* artifacts. As shown in Tab. 1, our method yields a *higher FactScore* than human artifacts (*e.g.*, 0.830 *v.s.* 0.795 in Slides). This suggests that the systematic grounding provided by our *Evidence Index* effectively mitigates the oversights and "memory drifts" often present in human-curated content. Furthermore, OmniPresent consistently outperforms GT in *Visual Quality* metrics across several formats. For instance, in Project Pages, our *Adaptive Layout Agent* ensures more consistent structural rigor and precise alignment than manual creation, which can suffer from irregular spacing or typographical inconsistencies. These results demonstrate that by decoupling content planning from rendering and enforcing cross-artifact verification, OmniPresent not only emulates human expertise but provides a level of technical precision that is difficult to maintain manually across a complex suite.

H DISCUSSION

Choice of HTML as Output Format. A natural question is why OmniPresent generates HTML instead of traditional static formats like LaTeX, Markdown, or PPTX. We justify this choice through three lenses: **(i) Universal Representation:** HTML serves as a highly flexible intermediate representation (IR) capable of unifying disparate modalities. It can natively encode the structured text of a PAGE, the spatial layouts of a POSTER, the sequential transitions of SLIDES, and the dynamic animations required for VIDEO. **(ii) Precise Layout Control:** Unlike Markdown, HTML paired with CSS allows our *Adaptive Layout Agent* to exert pixel-level control over asset scaling and positioning, which is critical for preventing the visual overflows identified in Finding 6. **(iii) Interactive & Dynamic Potential:** HTML is inherently "alive"; it enables the interactive link verification and responsive design that static formats cannot support. By targeting HTML, OmniPresent ensures

Table 9: **Cross-Artifact Verification (CV) improves suite-level inclusion.** We report Inclusion Score between *Poster/Slides* and *Page/Slides* (higher is better), under identical backbones. Enabling CV consistently increases inclusion across all models, indicating that CV corrects missing cross-artifact content and strengthens suite-level consistency.

Setting	Method	Inclusion Score		
		Poster/Slides	Page/Slides	Avg
Gemini-3-flash	w/o CV	0.818	0.882	0.850
	w/ CV	0.839 (+0.021)	0.903 (+0.021)	0.871 (+0.021)
Grok-4.1-fast	w/o CV	0.807	0.856	0.832
	w/ CV	0.817 (+0.010)	0.897 (+0.041)	0.857 (+0.026)
GPT-5-mini	w/o CV	0.811	0.874	0.843
	w/ CV	0.842 (+0.031)	0.893 (+0.019)	0.867 (+0.025)

Table 10: **Evaluation results on 10 non-AI research papers using OmniPreBench.** We collect 10 papers from diverse non-AI domains, including chemistry, biology, and medicine, generate different artifacts from them, and evaluate all methods on OmniPreBench.

Method	Content			Visual			Interaction	Compression
	FactScore↑	Readability↑	QA↑	Content acc↑	Aesthetics↑	Layout↑	Success↑	Compression Rate
Slides								
EvoPresent-Gemini-3-flash (Liu et al., 2025a)	0.636	3.37	0.678	2.12	3.24	3.05	-	11.69
Gemini-3-flash	0.596	3.15	0.696	1.99	2.35	2.18	-	14.57
OmniPresent-Gemini-3-flash	0.785	3.52	0.761	2.83	4.38	2.58	-	8.97
Poster								
Paper2Poster-Gemini-3-flash (Pang et al., 2025)	0.511	3.62	0.682	2.78	2.16	3.01	-	22.86
Gemini-3-flash	0.592	3.38	0.700	2.46	2.33	3.15	-	22.31
OmniPresent-Gemini-3-flash	0.716	3.68	0.725	2.82	4.22	3.41	-	4.23
Page								
AutoPage-Gemini-3-flash (Ma et al., 2026b)	0.682	3.05	0.699	3.01	2.02	3.37	0.40	3.95
Gemini-3-flash	0.533	2.95	0.668	2.39	3.27	3.18	0.10	12.78
OmniPresent-Gemini-3-flash	0.788	3.74	0.782	3.40	3.79	4.05	0.90	8.31

that the generated suite is not only renderable for immediate viewing but also easily editable and extensible for future digital dissemination.

GAP experiments. Table 9 evaluates whether the Cross-Artifact Verification (CV) module in our Cross-Artifact Verification stage improves the intended inclusion relationship across artifacts, i.e., salient content expressed in the Page/Poster should be covered by the Slides. Concretely, let C_f denote the set of extracted salient semantic units (claims) from artifact f; we compute the pairwise Inclusion Score as $\mathrm{Inc}(A, \mathrm{Slides}) = \frac{|C_A \cap C_{\mathrm{Slides}}|}{|C_A|} = 1 - \frac{|C_A \setminus C_{\mathrm{Slides}}|}{|C_A|}$, for $A \in \{\mathrm{Poster}, \mathrm{Page}\}$. Across three backbones (Gemini-3-flash, Grok-4.1-fast, and GPT-5-mini), enabling CV consistently increases Inclusion Score for both Poster/Slides and Page/Slides, yielding higher average scores in all settings. These gains suggest that generating artifacts independently tends to omit salient claims or takeaways, while CV's suite-level check-and-fix loop identifies missing items and prompts targeted fixes, thereby improving cross-modality completeness and overall suite coherence.

Out-of-domain Generalization on Non-AI Papers. To examine whether OmniPresent generalizes beyond AI-domain inputs, we additionally curate a small out-of-domain test set consisting of 10 research papers from non-AI disciplines, including chemistry, biology, and medicine. This setting is challenging because these papers often contain domain-specific terminology, chemical or biomedical visualizations, and highly specialized experimental protocols that differ substantially from the AI conference papers used in OmniPreBench. As shown in Table 10, OmniPresent remains consistently effective across slides, posters, and project pages. Compared with format-specialized baselines Ma et al. (2026b); Liu et al. (2025a) using the same Gemini-3-Flash backbone, OmniPresent achieves higher content faithfulness and information transmission in all three formats, improving FactScore from 0.636 to 0.785 for slides, from 0.511 to 0.716 for posters, and from 0.682 to 0.788 for pages. The gains are also reflected in visual quality: OmniPresent improves visual content accuracy, aesthetics, and layout over the corresponding specialized systems, indicating that the *adaptive HTML-based rendering pipeline is not limited to AI-style figures or presentation conventions*. Notably,

Table 11: **Conflict Reduction Rate under Cross-Artifact Verification (CV).** Higher indicates stronger reduction of cross-artifact conflicts between *Poster/Slides* and *Page/Slides*. CV reduces conflicts consistently across backbones, improving suite-level consistency.

Setting	Conflict Reduction Rate		
	Poster & Slides	Page & Slides	Avg
Gemini-3-flash	12.5%	33.3%	22.9%
Grok-4.1-fast	16.7%	31.2%	24.0%
GPT-5-mini	18.3%	28.4%	23.4%

the improvement on project pages is particularly large in visual content accuracy and interaction success, suggesting that the evidence-grounded planning and verification mechanism helps preserve domain-specific resources and generate more reliable navigational structures. These results provide preliminary evidence that OmniPresent functions as a domain-agnostic presentation-suite generation framework. While the non-AI set is intentionally small and should not be interpreted as a comprehensive cross-domain benchmark, the consistent gains indicate that the core design of shared knowledge extraction, format-aware planning, and cross-artifact verification transfers well to scientific papers outside the AI domain.

Effective of Cross-Artifact Verification Beyond improving inclusion, we further evaluate whether CV reduces cross-artifact inconsistencies. Table 11 reports the Conflict Reduction Rate between *Poster/Slides* and *Page/Slides*. Across all three backbones, CV achieves substantial conflict reductions (about 23–24% on average), with particularly large gains on Page–Slides where longer-form pages are more prone to drift from slide phrasing. These results support that CV not only fills missing content, but also actively resolves suite-level contradictions by aligning claims and supporting evidence across modalities.

I Detailed Schema of Knowledge Stream $\mathcal{K}$

The Knowledge Stream $\mathcal{K}$ acts as the central, modality-agnostic source of truth for all downstream planning agents.

Source and Construction. As described in **Stage 1** (Sec 3.2), the construction of $\mathcal{K}$ begins after the **MinerU Parser** extracts raw resources $\mathcal{R}$ (text blocks, formulas, figures). The **Content Analysis Agent** processes these raw resources, invoking a *restructure tool* to merge disjoint spans into coherent semantic clusters.

Structure Definition. Formally, we define each knowledge unit $k \in \mathcal{K}$ as a tuple (c, e, t), where c represents the *semantic statement* (claim), e denotes the *supporting evidence* (references, raw data, or figure pointers), and t is the *topical cluster* (e.g., Method, Experiment).

To bridge this formal definition with Large Language Model (LLM) prompting, we employ a **contextual linearization strategy**. Rather than passing disjoint JSON objects—which often degrades the model's ability to associate claims with proofs across long contexts—we serialize each tuple into a unified natural language string. This format strictly binds the high-level argument (c) with its low-level substantiation (e) under a specific scope (t):

"[Topic Tag t] <Logical Argument c> + <Specific Supporting Evidence e>"

Specifically:

- **Topic Tag (t):** Acts as a *semantic anchor* (e.g., `[Stage 2]`), allowing the planner to quickly retrieve relevant blocks via keyword matching or attention heads.
- **Logical Argument (c):** The high-level narrative or atomic claim derived from the text.
- **Specific Evidence (e):** The concrete data, parameters, formulas, or figure paths that ground the argument.

By fusing c and e into a single textual unit (e.g., *"...achieved SOTA performance (c) with 95% accuracy on ImageNet (e)..."*), we enforce **in-context grounding**. This ensures that whenever an agent retrieves a claim, the supporting evidence is inevitably included in the immediate context window, significantly reducing hallucination during the generation of downstream artifacts.

Example: Knowledge Stream ($\mathcal{K}$) Data Structure (JSON)

- **[Poster Planner]** Optimizes for **spatial information density** (Argument). The 'Section Planning Agent' establishes thematic sections, while the 'Content Curation Agent' selects visually compelling figures (Evidence/Mechanism).
- **[Slides Planner]** Optimizes for **adaptive narrative flow** (Argument). Unlike rigid templates, it uses an 'Audience Analysis Agent' to synthesize a storyline conditioning on paper genre and user constraints (Specific Mechanism).
- **[Page Planner]** Optimizes for **hierarchical engagement** (Argument). The 'Content Planning Agent' embeds figures/tables as inline visual tokens based on semantic relevance (Specific Evidence).
- **[Framework Visualization]** Figure 2(Path: images/Figure_2.jpg)Figure 2 depicts the pipeline (Evidence). In Stage 2, it explicitly shows the branching into Format-Specific Planners (Poster, Slides, Page) derived from the Knowledge Stream (Description).

J Human Preference Study

Study design. We conduct a user study to (i) validate that OmniPresent improves human-perceived visual quality over competitive baselines and (ii) assess the alignment between our VLM-based visual rubric and human judgment. Participants evaluate generated suites on a 5-point Likert scale across three visual quality dimensions. By comparing these human scores with our automated metrics, we evaluate the reliability of our evaluation framework. Detailed settings are provided in Appendix K.

***Findings 9*: Human preference supports both method effectiveness and rubric validity.** Experimental results in Tab. 12 indicate a strong human preference for OmniPresent over both end-to-end baselines and specialized systems. Participants consistently award OmniPresent the highest mean ratings in *Visual Aesthetics* and *Layout* across all formats, underscoring its capability to produce professional-grade presentation materials. Crucially, we observe a high correlation between human ratings and our VLM-based scores; the preference ordering produced by humans perfectly mirrors the rankings generated by our automated rubric. This alignment suggests that our OmniPreBench effectively captures the nuanced visual attributes valued by users.

K Details of Human Preference Study

In this section, we provide the user guidance for the human preference study.

Study protocol. We recruited 13 PhD-level participants to assess whether the *visual quality* judgments produced by our VLM-based evaluator align with human preferences. Each participant was presented with presentation suites generated by OmniPresent and by baseline systems, including posters, project pages, and slide decks. Participants rated each suite using a 5-point Likert scale, focusing on visual aspects specified in the questionnaire, and were instructed to judge presentation quality rather than the scientific contribution of the underlying paper. All participants were compensated for their time.

User Guidance. The user guidance is listed below:

Table 12: **User study results for visual quality across artifacts.** We report Mean ± Var (higher is better) for visual accuracy (V-Acc.), aesthetics (Aes.), and layout quality (Lay.) on *Slides*, *Poster*, and *Page*. Our OmniPresent consistently achieves the best visual scores compared with artifact-specific and backbone baselines.

Method	V-Acc.↑	Aes.↑	Lay.↑
Slides			
EvoPresent-Gemini-3-flash (Liu et al., 2025a)	3.25 ± 0.34	3.43 ± 0.39	2.57 ± 0.41
gpt-5-mini	3.34 ± 0.50	2.97 ± 0.63	2.78 ± 0.48
grok-4-fast	3.04 ± 0.51	2.06 ± 0.55	2.29 ± 0.51
gemini-3-flash	2.95 ± 0.33	2.43 ± 0.42	2.32 ± 0.39
OmniPresent-Gemini3-Flash	**3.78** ± 0.23	**4.54** ± 0.27	**3.57** ± 0.29
Poster			
Paper2Poster-Gemini-3-flash (Pang et al., 2025)	2.88 ± 0.33	2.76 ± 0.43	3.03 ± 0.34
gpt-5-mini	2.87 ± 0.48	2.26 ± 0.59	2.96 ± 0.52
grok-4-fast	2.02 ± 0.40	2.63 ± 0.58	2.60 ± 0.48
gemini-3-flash	2.36 ± 0.33	3.65 ± 0.33	3.06 ± 0.33
OmniPresent-Gemini3-Flash	**3.55** ± 0.26	**3.77** ± 0.24	**3.24** ± 0.27
Page			
AutoPage-Gemini-3-flash (Ma et al., 2026b)	3.23 ± 0.30	2.59 ± 0.39	3.33 ± 0.37
gpt-5-mini	2.79 ± 0.43	3.32 ± 0.57	3.59 ± 0.45
grok-4-fast	2.43 ± 0.43	3.18 ± 0.56	3.63 ± 0.51
gemini-3-flash	2.17 ± 0.30	3.47 ± 0.39	3.04 ± 0.31
OmniPresent-Gemini3-Flash	**3.56** ± 0.26	**3.63** ± 0.28	**3.98** ± 0.24

User Guidance for Human Preference Study

Thank you for participating in this study!
This user study aims to evaluate the quality of presentation suites (slides, posters, and webpages) automatically generated from academic papers. During the task, you will examine generated presentation suites and rate them based on the visual quality dimension. Your responses will help us assess the effectiveness of webpage-generation methods and improve future systems.

You will be shown several examples of presentation suites produced by different models. For each presentation suite, you will provide **Likert-style ratings** across multiple dimensions. There are no correct answers, please rate based on your genuine impressions of the presentation suite itself.

Example Questionnaire

Each question presents a statement such as:

> *"The webpage features a harmonious layout."*

You will then select a score from **1 to 5**, based on your agreement with the statement.

Score	Meaning
5	Strongly agree
4	Agree
3	Neutral / unsure
2	Disagree
1	Strongly disagree

Guidelines:
- Rate each dimension independently.
- Do not revise earlier responses based on later presentation suites.

- Focus on the *presentation suite*, not the scientific contribution of the paper.
- Provide your best judgment based on what you see.
- If a dimension is difficult to evaluate, select **3 (Neutral)**.

Evaluation Dimensions

Please evaluate each presentation suite along the following dimensions based on the criteria below:

1. **Content Quality**
   a. **Factual Fidelity**
      - **Accuracy:** The generated content accurately reflects the facts and claims from the source paper without distortion.
      - **No Hallucination:** There are no fabricated data, made-up citations, or false statements not present in the original text.
   b. **Information Completeness**
      - **Core Coverage:** The presentation covers the paper's core contributions and key findings (corresponding to the ability to answer key questions).
      - **Self-containment:** The content is self-explanatory; you can understand the main ideas without needing to constantly refer back to the source paper.
   c. **Readability & Clarity**
      - **Fluency:** The text is coherent, grammatically correct, and easy to follow.
      - **Presentation Style:** The language is concise and appropriate for a presentation context (avoiding excessive verbosity or raw copy-pasting).
2. **Visual Quality**
   a. **Visual Content Accuracy**
      - Mathematical formulas are rendered faithfully and are legible.
      - Figures and diagrams are relevant to the adjacent text descriptions.
   b. **Layout & Cohesion**
      - The layout exhibits clear grouping and hierarchy.
      - The structure supports a natural reading order.
   c. **Aesthetics**
      - The design has a professional look-and-feel.
      - Typography is readable and color usage is harmonious.

Thank you!

Your feedback is extremely valuable and will contribute to improving automated presentation suites generation systems. We greatly appreciate your time and participation.

L CASE STUDY

To move beyond aggregate metrics, we conduct qualitative case studies to visualize the suite-level advantages of OmniPresent.

This section provides additional qualitative comparisons to complement the main quantitative results. We include representative cases for **Page** (Fig. 5), **Poster** (Fig. 6), **Slides** (Fig. 7), as well as direct comparisons against **end-to-end baselines** (Fig. 8). Our qualitative analysis reveals that artifact-centric baselines often suffer from significant *visual rendering failures*.

Across these examples, our system consistently produces a more **coherent presentation suite**: (i) key claims and takeaways remain aligned across modalities, (ii) visual elements (figures/tables) are selected and placed to support the surrounding text rather than occupying empty space, and (iii) the overall layout maintains a readable hierarchy (titles, section blocks, and emphasis regions) with fewer artifacts.

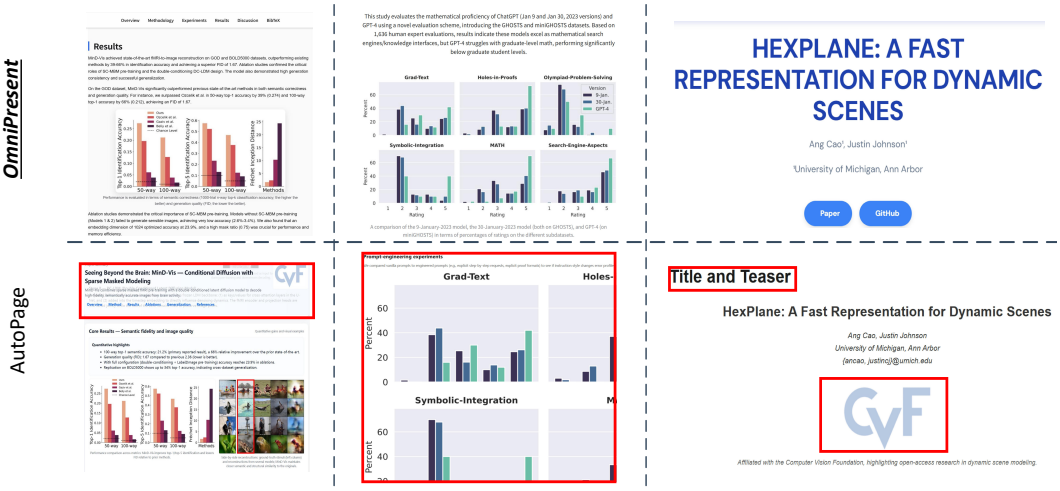


Figure 5: **Qualitative comparison between our system (top) and a prior system (bottom).** On the same paper, our method produces a more coherent presentation suite (page/poster/slides) with stronger cross-modal alignment in key regions (e.g., *Title & Teaser*, plots, and overall layout), whereas the baseline more often exhibits missing content, layout imbalance, or disproportionate visual elements. Red boxes highlight representative differences.



Figure 6: **Qualitative comparison between our system (top) and a prior system (bottom) across multiple papers.** Our method generates more structured and visually consistent posters with better content coverage and layout balance, while the baseline often suffers from sparse organization, uneven whitespace usage, and reduced cross-section coherence. The dashed grid indicates matched paper cases for side-by-side comparison.

In contrast, baseline systems frequently exhibit **content drop** (missing important components), **layout inefficiency** (large unused regions or overcrowded blocks), and **cross-modality inconsistency** (misaligned emphasis between page/poster/slides). We also observe recurring *visual rendering failures*, including disproportionate asset scaling (e.g., icons rendered excessively large or small; Fig. 5), text–image overflows and collisions (Fig. 8), and misplaced structural elements that break the intended reading flow (Fig. 7). The highlighted regions in the figures further illustrate typical failure modes such as duplicated or truncated visuals, incorrect cropping, and weakened correspondence between textual claims and supporting visuals.

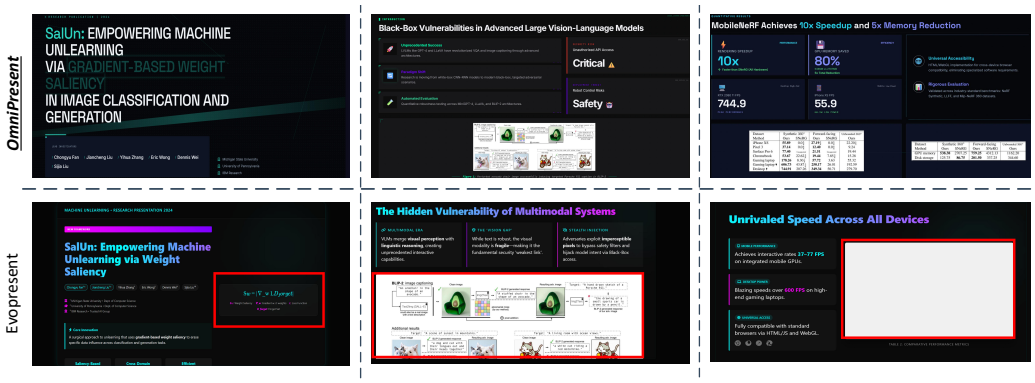


Figure 7: **Qualitative comparison of slide generation between our system (top) and EvoPresent (bottom).** Across multiple papers, our method produces cleaner, more information-dense slides with consistent typography, alignment, and visual hierarchy, while EvoPresent often leaves large unused regions (highlighted in red) or exhibits weaker content-to-layout utilization. Dashed separators indicate matched slide cases for side-by-side comparison.

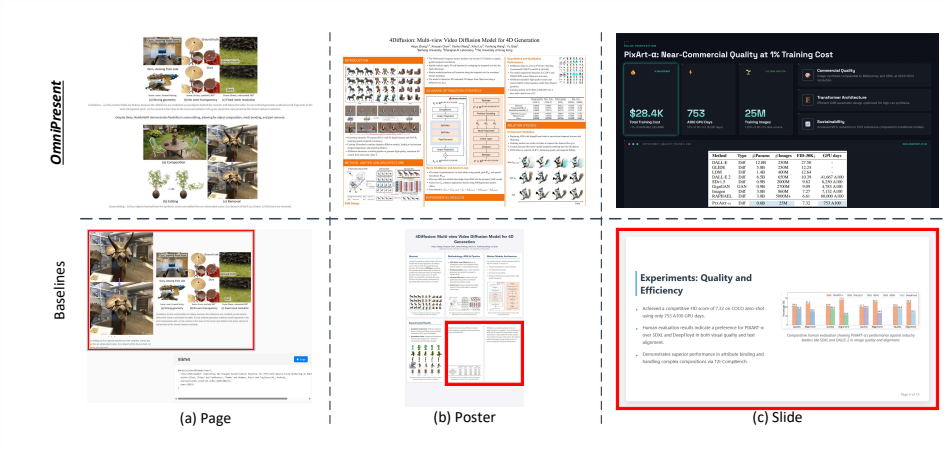


Figure 8: **Qualitative comparison with end-to-end baselines.** Top: our system (OmniPresent) generates coherent *page–poster–slide* suites with better content density, consistent visual hierarchy, and stronger cross-asset alignment. Bottom: end-to-end baselines often waste large regions, introduce layout inefficiencies, or lose key information (highlighted in red). (a–c) show representative examples for Page, Poster, and Slide. Dashed separators indicate matched cases across methods.

By contrast, OmniPresent produces professional-grade layouts with a more precise visual hierarchy. Leveraging our *Adaptive Layout Agent* (Sec. 3.4), OmniPresent renders visual assets with appropriate dimensions and spatial positioning, reducing the cluttered or "broken" appearance often seen in end-to-end HTML generation. Overall, these additional cases reinforce that our approach improves not only single-asset quality, but also the **suite-level consistency** required for practical scientific communication.

M Limitations and Impact Statement

This work advances machine learning systems for scientific communication by automating the generation of coherent dissemination suites (project pages, posters, slides, and videos) from research papers. By introducing explicit mechanisms for *faithfulness* and *cross-format consistency*, our approach lowers the technical and temporal barriers to producing high-quality research presentations. This has a direct positive societal impact: it democratizes the accessibility of scientific knowledge, enabling

researchers, educators, and the general public to more easily digest complex findings. However, automated dissemination carries inherent risks. If generated artifacts exhibit hallucinations or misplaced emphasis, they may propagate scientific inaccuracies more rapidly than manual curation. There is also potential for misuse, such as the mass production of low-fidelity or deceptive materials, which could strain the integrity of scientific discourse. Furthermore, inherent model biases may inadvertently skew content framing, and the application of such tools to sensitive or unpublished manuscripts raises significant confidentiality and copyright concerns. We mitigate these risks through our *evidence-grounded* generation and *cross-artifact verification* loop, which are specifically designed to catch unsupported or conflicting claims. Nevertheless, we emphasize that human-in-the-loop oversight remains indispensable for high-stakes dissemination. We advocate for responsible deployment, including clear disclosure of AI-generated content, adherence to original licensing, and rigorous expert review before public release.