PPTGen: A First Step Towards Multimodal Compositional Generation

Anonymous ACL submission

Abstract

Current generative AI struggles to simultaneously generate and compose multimodal content into a structured whole, a capability inherent to human communication, which makes it highly effective for conveying complex ideas. To formalize this problem, we propose the Multimodal Compositional Generation (MCG) task: creating structured content through the integrated planning of information, modality, and layout. Further, we select presentation (PPT) generation as a representative MCG task to study as its rich modalities and challenging 2D free-form layout. To solve the MCG challenge in the PPT generation task, we propose PPTGen, a three-stage compositional generation framework: Our framework first considers content planning from the information view and then layout planning with visual imagination. Finally, a rendering module converts these plans into API calls to produce the PPT. To evaluate and analyze our approach, we construct MCG-PPTBench, a comprehensive benchmark assessing free generation, controllability, and editability of MCG. Experiments on MCG-PPTBench confirm PPTGen's effectiveness. We also provide a foundational analysis of pipeline-based versus end-to-end methods to guide future research in MCG.

1 Introduction

Despite the remarkable success of generative models, their capabilities are predominantly confined to unimodal generation (Achiam et al., 2023; Touvron et al., 2023a; Rombach et al., 2022; Ho et al., 2022; Wang et al., 2025a; Kong et al., 2024; Agostinelli et al., 2023; Liu et al., 2023). While initial efforts have expanded into dual-modality tasks, such as interleaved text-image (Sun et al., 2023; Liao et al., 2025) or joint audio-video generation (Liu et al., 2024a; Wang et al., 2025b), these approaches arrange different modalities in simple sequential or parallel formats. This highlights their fundamental inability to truly ***generate*** and ***compose*** multimodal information into a cohesive, synergistically structured whole.

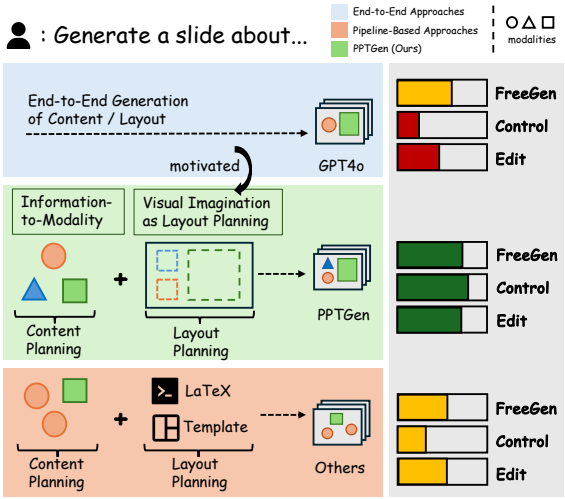

Figure 1: An illustration of end-to-end approaches, pipeline-based approaches and PPTGen (Ours). PPTGen adopts pipeline structure while borrow insights from end-to-end approaches for layout planning, thus achieving superior performance on free generation, controllability and editability.

In contrast, human communication is inherently *multimodal* and *compositional* (Galke et al., 2023; Donini et al., 2024), a capability that makes it highly effective for conveying complex ideas. Instead of presenting information as disconnected data points, humans create richly structured documents like presentations, web pages, and academic papers. The expressive power of these formats stems from two core principles:

(1) **Right Modality Clarifies Information**: Every idea has an optimal medium. We cannot fully capture the beauty of Van Gogh's Starry Night with text, just as we cannot represent the logic of a mathematical proof with images.

(2) **Good Layout Eases Understanding**: Putting related things together helps the reader understand. For example, a chart placed next to its explanation is easier to digest. If they were far

apart, the reader would have to work much harder to understand the connection.

To bridge this gap, we formulate the task of Multimodal Compositional Generation (MCG), a task that moves beyond simple unimodal generation to the holistic composition of structured multimodal content. At its core, MCG involves the integrated planning of three interconnected aspects for every content element: its information, its modality, and its spatial layout. In this work, we select presentation (PPT) generation as a representative MCG task due to (1) **Rich Multimodality**: PPTs inherently require the synthesis of diverse elements, including text (titles, content), visual media (images, videos), and data representations (charts, tables). (2) **Demanding Compositional Hierarchy**: PPT creation demands reasoning over a two-level structure: the free-form 2D spatial arrangement of shapes on a slide, and the 1D sequential ordering of slides into a coherent narrative. This makes it a challenging and direct benchmark for evaluating compositional capabilities.

To solve the MCG challenge in the PPT generation task, we propose **PPTGen**, a three-stage framework that includes a content planning, layout planning, and rendering module. Specifically, we employ an **Information-to-Modality (I2M)** process for content planning, which first plans a modality-agnostic information flow given a user query and then instantiates it into content of optimal modalities via specialized generative models. For layout planning, we propose **Visual Imagination as Layout Planning (VILP)**, a novel method that leverages a visual generative model to "imagine" the spatial layout. The module first generates a visual blueprint of shapes and their bounding boxes, from which the layout coordinates are extracted as the final plan. Finally, the Rendering Module translates the content and layout plans into a series of predefined API calls to render the final presentation.

While existing benchmarks for presentation generation (Ge et al., 2025; Zhang et al., 2024; Guo et al., 2023) primarily focus on high-level free generation, failing to measure critical abilities of MCG. To address this evaluation gap, we introduce **MCG-PPTBench**, a comprehensive benchmark that assesses three key dimensions: **Free Generation**, which tests macro-level planning of multimodal narratives from high-level prompts; **Controllability**, which measures precision in following detailed, fine-grained instructions; and **Editability**, which evaluates the ability to modify elements while preserving surrounding content. This holistic approach mirrors real-world scenarios that demand not only initial creation but also precise control and iterative refinement, offering a more robust measure of true compositional intelligence.

We validate PPTGen on MCG-PPTBench against both end-to-end and pipeline-based baselines. Our experiments show that PPTGen achieves state-of-the-art performance across all three dimensions. In free generation, PPTGen excels in logic coherence and modality richness, and thanks to the VILP module, it achieves layout quality comparable to leading end-to-end models. Crucially, PPTGen's key advantage lies in its flexible layout planning, which overcomes the rigid constraints of conventional compositional approaches to achieve vastly superior controllability and editability. Furthermore, our detailed error analysis reveals the fundamental limitations of end-to-end methods in following fine-grained instructions, underscoring the core challenges in MCG.

In summary, our contributions are four-fold:

(1) We formalize the task of Multimodal Compositional Generation (MCG) and establish presentation generation as its canonical benchmark.

(2) We propose PPTGen, a novel framework for MCG featuring an Information-to-Modality (I2M) content planner and a Visual Imagination as Layout Planning (VILP) layout planner.

(3) We introduce MCG-PPTBench, the first benchmark to evaluate MCG across three critical dimensions: free generation, controllability, and editability.

(4) We provide extensive experiments that validate PPTGen's state-of-the-art performance and reveal the core challenges of MCG for existing generative paradigms.

2 Related Work

2.1 Presentation Generation

Approaches to automated presentation generation can be broadly categorized into two paradigms: compositional and end-to-end.

Pipeline-based approaches (Fu et al., 2022; Zheng et al., 2025; Shi et al., 2025; Jung et al., 2025; Ge et al., 2025; Xu et al., 2025; Maniyar et al., 2025) adopt a pipeline architecture that decouples content generation from layout generation. For content, methods either extract information from source documents (Fu et al., 2022; Zheng

et al., 2025) or use LLMs to generate it from high-level prompts (Ge et al., 2025). Our work builds on the latter, but introduces an Information-to-Modality (I2M) module to select the optimal modality for each piece of information, thereby enriching the output. For layout, these methods are typically either markup-based, relying on rendering engines like LaTeX with limited customizability (Xu et al., 2025), or template-based, populating fixed templates that constrain the final design (Zheng et al., 2025; Maniyar et al., 2025). In contrast, our Visual Imagination as Layout Planning (VILP) module, inspired by visual planning(Feng et al., 2023), can generate bespoke layouts from scratch, supporting content-driven layout planning(Jiang et al., 2025).

The end-to-end paradigm (Ge et al., 2025) is an emerging direction that leverages visual generative models to treat slide creation as a holistic image generation task. While this approach is promising, it often struggles with the fine-grained controllability and editability required for complex documents.

2.2 Multimodal Generative Models

While the field of generative AI is advancing rapidly, most models remain unimodal in their output, producing either text (Achiam et al., 2023; Touvron et al., 2023a), images (Rombach et al., 2022; Ho et al., 2022), video (Wang et al., 2025a; Kong et al., 2024), or audio (Agostinelli et al., 2023; Liu et al., 2023). Recent research has explored dual-modality generation, such as interleaved text-image synthesis (Sun et al., 2023; Liao et al., 2025) or joint audio-video production (Liu et al., 2024a; Wang et al., 2025b).

However, these systems are limited in the variety of modalities they can handle and typically arrange them in simple, non-hierarchical structures (e.g., sequentially or in parallel). This reveals a gap in true Multimodal Compositional Generation (MCG): the integrated generation of diverse modalities within a coherent hierarchy. In this work, we formalize MCG and use presentation generation as a representative task.

3 Preliminaries

This section establishes the context for our work. We begin by situating Multimodal Compositional Generation (MCG) within the broader landscape of generative AI. Next, we justify using presentation generation as our representative task. Finally, we provide a formal definition of MCG.

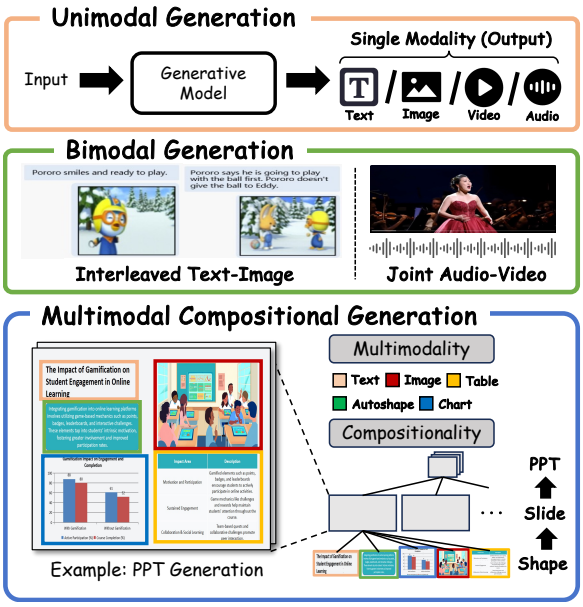

Figure 2: An illustration of the landscape of generative AI. PPT generation differs from other generative tasks due to (1) its rich modality (2) its need for a compositional hierarchy (Shape $\xrightarrow{\text{2D Layout}}$ Slide $\xrightarrow{\text{1D Layout}}$ PPT).

3.1 The Generative AI Landscape

The landscape of generative AI has evolved significantly, as illustrated in Figure 2. Early models focused on unimodal generation, producing single-modality artifacts like text or images with no compositional structure. The next step was bimodal generation, which combines two modalities in simple, fixed arrangements, such as sequential text-image streams or parallel video-audio streams.

This progression culminates in Multimodal Compositional Generation (MCG). MCG is an advanced task that generates and composes multiple modalities into a cohesive, structured whole, mirroring the complexity of human communication. We provide a more detailed analysis in Appendix A.

3.2 PPT Generation as a Representative Task

We select PPT generation as a representative task for studying MCG for two primary reasons:

Rich Multimodality Presentations inherently synthesize diverse modalities including textual elements (titles, textboxes), visual elements (images, videos), auditory elements (audio), and data representations (tables, charts).

Demanding Compositional Hierarchy Presentation composition involves a dual hierarchy: the 1D sequential ordering of slides and the 2D free-form spatial layout of elements on each slide. This differs from other MCG tasks like generating web

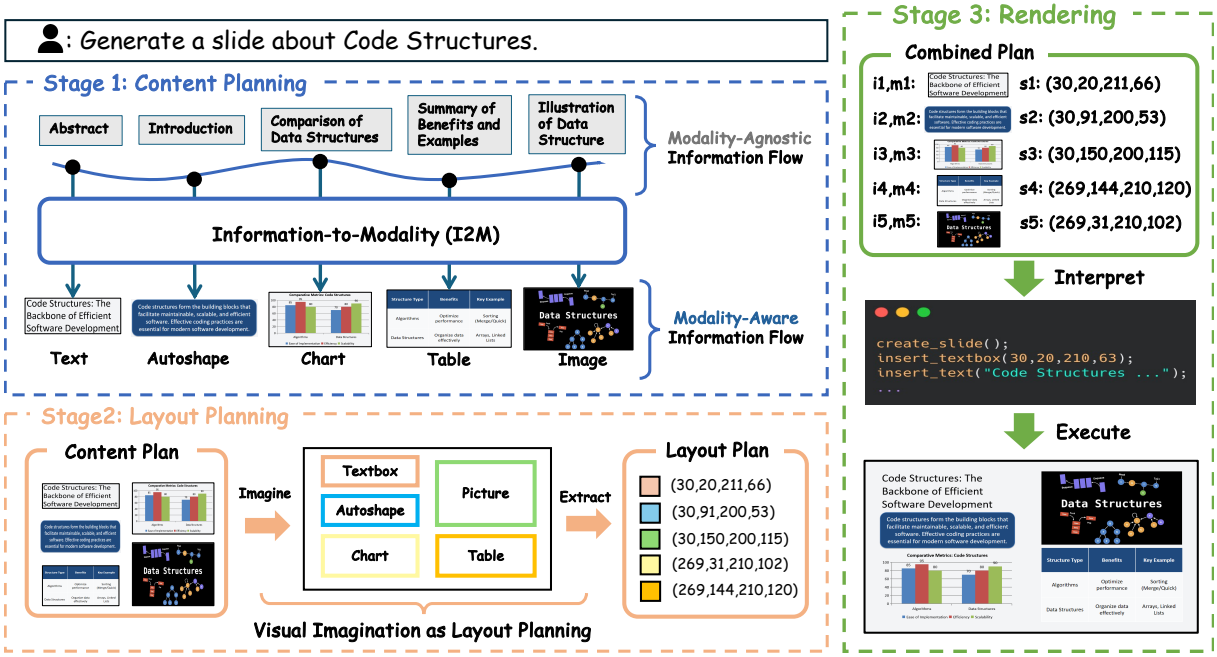

Figure 3: **Overview of the PPTGen framework.** Given a user query, PPTGen generates a slide in three stages: (1) **Content Planning** determines what information to present and its modality (text, chart, image); (2) **Layout Planning** predicts the spatial arrangement of elements; (3) **Rendering** executes API calls to produce the final slide.

pages or academic papers, whose layout is often auto-rendered from markup languages like HTML or LaTeX. In contrast, presentation generation requires the model to explicitly define the spatial arrangement of every element, making it a more direct test of compositional planning.

3.3 Formal Definition

We further formalize Multimodal Compositional Generation (MCG) as follows: Given a user query q, the task requires joint generation of content and structure. The objective is to produce a set of elements $E = e_k = (i_k, m_k, s_k)_{k=1}^{N}$. Each element e_k consists of modality-agnostic information (i_k), a chosen modality (m_k), and spatial structure (s_k). The spatial structure defines the element's placement within the compositional hierarchy (e.g., a bounding box of a shape).

4 The PPTGen Framework

Generating a presentation requires a holistic plan that jointly requires a holistic plan that jointly considers the information to convey, the best modality for each piece of information, and the overall structure. To address this, we propose PPTGen, a structured three-stage framework that decomposes this challenge, as illustrated in Figure 3.

Formally, given a user query q, our goal is to generate a set of slide elements $E = \{(i_k, m_k, s_k)\}_{k=1}^{N}$. Let $I = \{i_k\}$, $M = \{m_k\}$, $S = \{s_k\}$ denote the sets of information, modality, and structure to be planned, respectively. We factorize the joint probability into two sequential stages: content planning and layout planning:

$$P(I, M, S \mid q) = \underbrace{P(I, M \mid q)}_{\text{Content Planning}} \cdot \underbrace{P(S \mid I, M, q)}_{\text{Layout Planning}}$$

This factorization allows us to first handle semantic content creation and then visual composition, followed by a final rendering stage.

4.1 Stage 1: Content Planning via Information-to-Modality (I2M)

This stage is motivated by the principle that *Right Modality Clarifies Information*. While individual concepts information can have multiple representations (e.g. text, image), they must collectively form a coherent narrative. We therefore design the Information-to-Modality (I2M) module to first structure this narrative and then select the optimal modality for each part:

1. Modality-Agnostic Information Planning. First, we prompt a Large Language Model (LLM) with the user query q to generate a modality-agnostic information flow, $I = \{i_k\}$. This serves as a structured outline of the presentation's core

narrative, ensuring logical coherence before any visual or modal decisions are made.

2. Modality Instantiation. With the information flow I established, a second LLM pass determines the optimal modality m_k for each information chunk i_k. For example, it might assign text for a definition or a chart for statistical data. Finally, we use specialized generative models (e.g., a text-to-image model) to instantiate the final content for each planned element.

4.2 Stage 2: Visual Imagination as Layout Planning (VILP)

This stage is motivated by the principle that *Good Layout Eases Understanding*. While auto-rendered layouts are often rigid and lack aesthetic appeal, modern image generation models excel at creating visually pleasing spatial arrangements. To harness this capability, we introduce Visual Imagination as Layout Planning (VILP), which uses an image model for slide design.

1. Visual Blueprint Generation. We provide the image generation model with prompt that details all the content elements from Stage 1. The prompt instructs the model to "imagine" an aesthetic slide layout. It represents each element as a bounding box with a unique, specified color (e.g., red for the title's bounding box). The output is a single blueprint image of a well-composed slide.

2. Layout Coordinate Extraction. The generated blueprint image is then programmatically parsed. By detecting the uniquely colored boxes, we extract their pixel coordinates and dimensions (x, y, w, h). These geometric properties form the final layout plan $S = \{s_k\}$. This approach effectively transforms the abstract challenge of aesthetic design into a concrete visual generation problem.

4.3 Stage 3: Rendering via API Calls

The final stage converts the complete abstract plan $E = \{(i_k, m_k, s_k)\}$ into a presentation file. We achieve this by translating each element into a sequence of predefined API calls, $A = \{a_1, a_2, \ldots, a_n\}$, that programmatically construct the slide. For instance, rendering a textbox at a specific location is done via:

```
insert_textbox(x, y, w, h);
insert_text("Your text content here...");
```

For complete set of APIs, refer to Appendix B.

5 MCG-PPTBench: A Benchmark for Multimodal Compositional Generation

To address the gap in standardized evaluation for MCG, we introduce MCG-PPTBench. Unlike existing benchmarks that focus only on free generation quality, our benchmark provides a holistic evaluation framework. It assesses three key capabilities needed for real-world applications: Free Generation, Controllability, and Editability.

Free Generation tests a model's ability to create complete, multi-page presentations from high-level user prompts. This evaluates holistic planning of contents and structures. We evaluate the output based on *Logic Coherence*, *Layout Quality*, and *Modality Richness*.

Controllability measures how well a model follows detailed instructions to create a single slide with specific content and layouts. This tests its ability to generate individual components and adhere to structural constraints.

Editability assesses a model's capacity to understand and modify an existing slide based on a follow-up command, while maintaining compositional integrity. This tests its grasp of the relationships between different elements.

MCG-PPTBench is built on the principle of comprehensive diversity, including diversity of task type, task domain and task complexity (Appendix C). The construction process is detailed in Appendix C. We conducted strict quality control on data, including a manual review for each constructed data.

For evaluation, we employ a Multimodal Large Language Model equipped with a shape-level Chain-of-Thought (CoT) process to assess performance across all tasks. For Free Generation, we report the score on three dimensions: Logic Coherence, Layout Quality, and Modality Richness. For Controllability and Editability, we report both the shape-level accuracy and slide-level accuracy of performing controlled generation or editing. Further details are in Appendix C.

6 Experimental Setup

6.1 Compared Baselines

We evaluate our proposed method against representative baselines from the two primary paradigms for Multimodal Compositional Generation (MCG): End-to-End (E2E) and Pipeline-based approaches.

6.1.1 End-to-End Paradigm

GPT-4o (Hurst et al., 2024) represents End-to-End paradigm, which frames slide generation as a single image generation task. It produces the final slide layout and content simultaneously at the pixel level, without decomposed planning. We refer to this baseline as **End2End**.

6.1.2 Pipeline-Based Paradigm

PPTAgent (Zheng et al., 2025) is a pipeline-based approach that relies on templates for layout planning. It uses a predefined library of layout templates. After generating content, the method selects a suitable template and populates its placeholders. We refer to this baseline as **Pipeline-Template**.

PreGenie (Xu et al., 2025) is a pipeline-based approach that uses markup languages for layout planning. It generates the slide content and translates the content into a markup language like LaTeX, which automatically handles the final layout rendering, which we refer to as **Pipeline-Markup**.

PPTGen (Ours) also uses a pipeline structure. However, we integrate the layout planning strengths of end-to-end approaches. Specifically, our VILP module leverages E2E models for layout design. We refer to this baseline as **Pipeline-E2E**.

Detailed implementation specifications for all methods are provided in Appendix D.

7 Experimental Results

7.1 Quantitative Results and Analysis

In this section, we analyze the results for Free Generation, Controllability, and Editability.

7.1.1 Analysis of Free Generation

The results in Table 4 show that our method, PPTGen, achieves the best overall performance. It scores highest in Logic Coherence and Modality Richness, a result we attribute to the Information-to-Modality (I2M) module. By first planning a modality-agnostic information flow and then instantiating it with different modalities, the I2M module ensures logical coherence while producing a presentation rich in modalities. In contrast, the template-based method PPTAgent is limited by fixed templates, resulting in low modality diversity.

Notably, GPT4o also deliver strong results, particularly in layout quality, which highlights the potential of image-based slide generation. This strength motivated our proposed Visual Imagination as Layout Planning (VILP) module, which

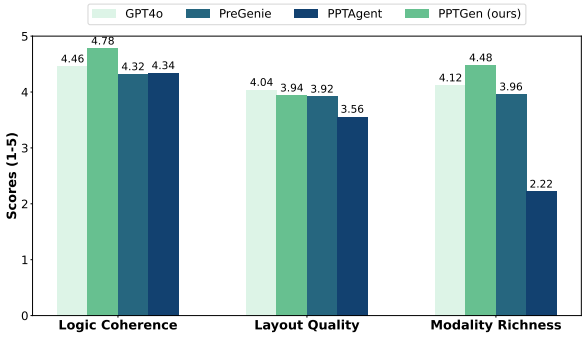

Figure 4: Results of compared baselins on the Free Generation task of MCG-PPTBench.

successfully integrates the spatial planning capabilities of End-to-End methods. Other pipeline-based approaches suffer from layout issues, such as content overflow or rigid templates.

> **Takeaway 7.1.1**
>
> PPTGen's superior performance in *Logic Coherence* and *Modality Richness* validates the effectiveness of our I2M module.
>
> GPT4o's leading *Layout Quality* confirms the motivation of VILP module, highlighting the potential of end-to-end approaches.

7.1.2 Analysis of Controllability

As shown in Table 1, PPTGen achieves the highest controllability accuracies at both shape and slide levels. In contrast, other baselines exhibit limitations in following detailed and fine-grained instructions. GPT4o exhibit low controllability, particularly for text and data-driven shapes, due to the inherent failure modes of image generation models (Section 7.2). Pipeline-based methods are also limited by the inflexibility of layouts. For example, PreGenie (Pipeline-Markup) struggles with spatial instructions like "move title to top right", while PPTAgent (Pipeline-Template) cannot handle structural changes like "convert this table to bar chart".

PPTGen's success stems from its hybrid design, which integrates the strengths of both pipeline and end-to-end paradigms. It leverages a pipeline structure to ensure the factuality and precision of individual components, while incorporating a flexible, visually-grounded layout planner inspired by End-to-End approaches. This combination enables the superior controllability that other approaches lack.

Table 1: Results of both shape-level and slide-level accuracy on the Controllability task.

Baseline	Paradigm	Shape-Level Acc.				Slide-Level Acc.
		Text	Picture	Chart	Table	
GPT4o	End2End	0.33	0.68	0.54	0.41	0.2105
PreGenie	Pipeline-Markup	0.63	0.62	0.66	0.67	0.3553
PPTAgent	Pipeline-Template	0.11	0.30	0.00	0.26	0.0000
PPTGen (ours)	Pipeline-E2E	**0.93**	**0.92**	**0.89**	**0.85**	**0.7895**

Table 2: Results of both shape-level and slide-level accuracy on the Editability task. PPTAgent is unable to perform edit due to its input constraints and template limitations, further detailed in Appendix D.

Baseline	Paradigm	Shape-Level Acc.				Slide-Level Acc.
		Text	Picture	Chart	Table	
GPT4o	End2End	0.794	**0.792**	0.567	0.615	0.4400
PreGenie	Pipeline-Markup	0.727	0.792	0.200	0.462	0.5200
PPTAgent	Pipeline-Template	–	–	–	–	–
PPTGen (ours)	Pipeline-E2E	**0.894**	0.667	**0.767**	**0.795**	**0.7400**

Takeaway 7.1.2

PPTGen combines the strength of pipeline-based and end-to-end approaches, thus excels in controllability.

7.1.3 Analysis of Editability

As shown in Table 2, PPTGen achieves the highest editability accuracy, demonstrating its practical utility for iterative refinement.

The baseline methods all struggle, primarily due to failures in content preservation. GPT4o suffers from the consistency issues common in image models, often altering unedited regions when a target is modified (Section 7.2). PreGenie relies on an automatic renderer, where a local edit can trigger a global reflow, unintentionally shifting other elements. PPTAgent is largely incapable of performing edits, as detailed in Appendix D. PPTGen succeeds through its API-based rendering process. Edits are executed by targeting specific element APIs, ensuring a "surgical" modification that preserves the properties and positions of all other content, maintaining the slide's compositional integrity.

Takeaway 7.1.3

PPTGen's pipeline ensures perfect content preservation, making it excel in editability.

7.2 Challenges of E2E-based Approaches

End-to-End approaches like GPT4o are a promising paradigm for MCG, but they face critical challenges due to inherent limitations in image generation and edit models:

Deficiency in Text Rendering Image generation models often render text inaccurately, especially with long English paragraphs or non-Latin scripts like Chinese. This issue worsens at smaller font sizes, compromising legibility and professionalism.

Hallucination in Controlled Generation When generating data-driven visuals like charts or tables, image generation models frequently fabricate or misrepresent data. A pie chart, for instance, might visually misrepresent the proportions of its data, making the output unreliable for fact-based tasks.

Element Inconsistency in Editing Image generation models fail to preserve element during edits. Modifying one element (e.g., a text box) can cause unintended changes to the layout or other elements, making iterative refinement impractical.

Takeaway 7.2

End-to-end approaches face the following limitations: (1) Deficiency in Text Rendering (2) Hallucination in Controlled Generation (3) Element Inconsistenct in Editing.

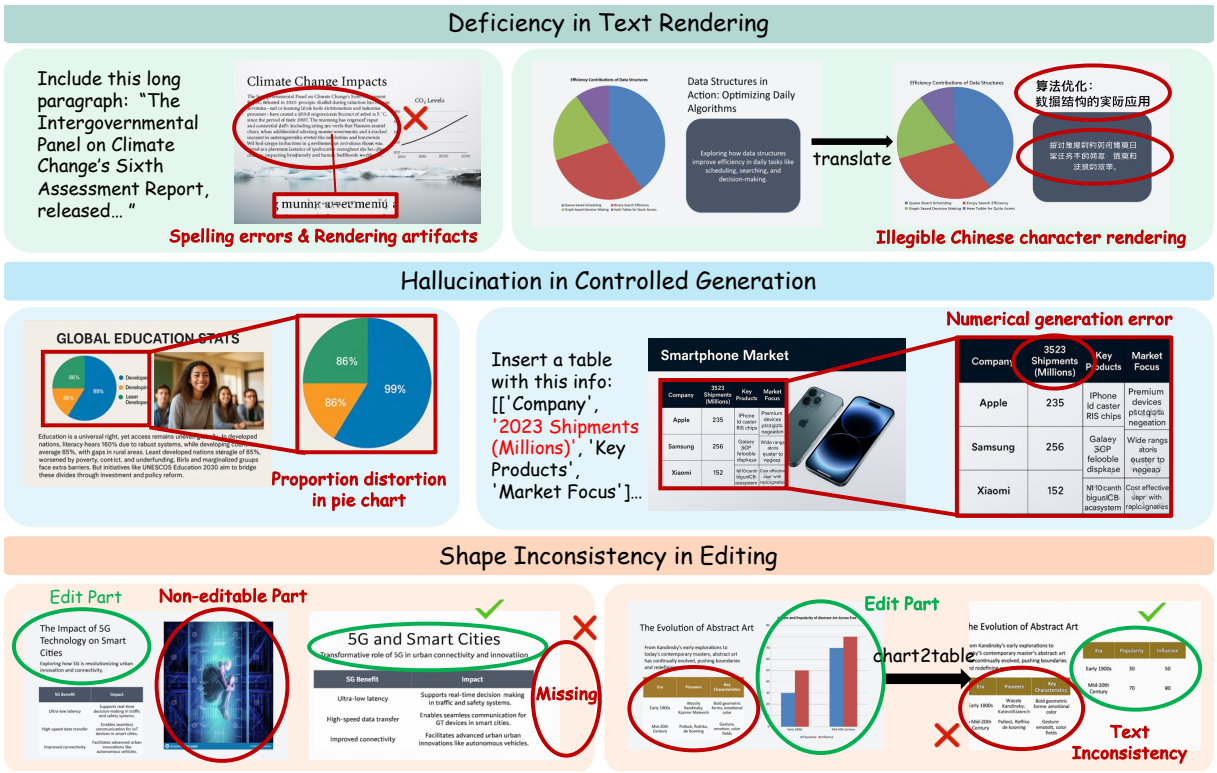

Figure 5: Challenges of End-to-End approaches. For each case, we label the mistake type in red.

7.3 Qualitative Case Study

A comparative case study in Appendix E illustrates the distinct behaviors and trade-offs of each paradigm.

The End-to-End approach GPT4o excels at generating well-designed, modally rich slides. However, its reliability is limited by critical challenges (detailed in Section 7.2). The pipeline-based baselines have their own constraints. The Pipeline-Template method PPTAgent is restricted by its fixed layouts, limiting the diversity of modalities and the scope of tasks it can perform. The Pipeline-Markup method PreGenie handles simple content well, but its layout structure often breaks with lengthy inputs, causing content to overflow.

In contrast, PPTGen combines the strengths of both paradigms. It inherits the flexible, aesthetic layouts of End-to-End approaches. At the same time, its pipeline architecture ensures the precision needed for high-fidelity control and editing.

8 Conclusion

In this paper, we formalize Multimodal Compositional Generation (MCG), a crucial yet underexplored task that is central to effective human communication. We introduced PPTGen, a novel framework that tackles this challenge by strategically decomposing it into content and layout planning. Our key innovations include the Information-to-Modality (I2M) process for content planning and Visual Imagination as Layout Planning (VILP) for layout planning. To validate our approach, we developed MCG-PPTBench, the first benchmark to holistically evaluate free generation, controllability, and editability. Experiments show that PPTGen synergizes the precision of pipeline methods with the layout flexibility of end-to-end models, excelling in all three dimensions. We also provide a detailed analysis of the current challenges of each generative paradigm. By defining the task, proposing a solution, creating a comprehensive benchmark, and detailedly analyzing the core challenges, we have laid a solid foundation for future research into this important frontier of generative AI.

9 Limitations

Our work has two primary limitations. First, our framework and benchmark are focused on presentation generation; extending these methods to other MCG domains, such as web pages and academic papers, remains an important next step. Second, we do not compare with methods that generate PPT through markup languages like XML. While a markup-based approach could offer more granular layout control compared with LaTeX, we found it is highly error-prone for current models due to the verbose and strict syntax of such formats.

References

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

Andrea Agostinelli, Timo I Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, et al. 2023. Musiclm: Generating music from text. *arXiv preprint arXiv:2301.11325*.

Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, et al. 2023. Audiolm: a language modeling approach to audio generation. *IEEE/ACM transactions on audio, speech, and language processing*, 31:2523–2533.

Haoyu Chen, Xiaojie Xu, Wenbo Li, Jingjing Ren, Tian Ye, Songhua Liu, Ying-Cong Chen, Lei Zhu, and Xinchao Wang. 2025. Posta: A go-to framework for customized artistic poster generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 28694–28704.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. 2023. Simple and controllable music generation. *Advances in Neural Information Processing Systems*, 36:47704–47720.

DAMO-NLP-SG. 2024. Webdesignagent: Towards effortless website creation. https://github.com/DAMO-NLP-SG/WebDesignAgent. Accessed: 2025-10-04.

Massimo Donini, Cristina Gena, and Alessandro Mazzei. 2024. Multimodal strategies for robot-to-human communication. *Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*.

Weixi Feng, Wanrong Zhu, Tsu-jui Fu, Varun Jampani, Arjun Akula, Xuehai He, Sugato Basu, Xin Eric Wang, and William Yang Wang. 2023. Layoutgpt: Compositional visual planning and generation with large language models. *Advances in Neural Information Processing Systems*, 36:18225–18250.

Tsu-Jui Fu, William Yang Wang, Daniel McDuff, and Yale Song. 2022. Doc2ppt: Automatic presentation slides generation from scientific documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 634–642.

Lukas Galke, Yoav Ram, and Limor Raviv. 2023. Deep neural networks and humans both benefit from compositional language structure. *Nature Communications*, 15.

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

Google. 2025. Veo 3. https://deepmind.google/models/veo/. Accessed: 2025-10-04.

Yiduo Guo, Zekai Zhang, Yaobo Liang, Dongyan Zhao, and Duan Nan. 2023. Pptc benchmark: Evaluating large language models for powerpoint task completion. In *Annual Meeting of the Association for Computational Linguistics*.

Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. 2022. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*.

OpenAI Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mkadry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alexander Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alexandre Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoochian, Amin Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, Andrew Galu, Andrew Kondrich, Andrew Tulloch, An drey Mishchenko, Angela Baek, Angela Jiang, An toine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, B. Ghorbani, Ben Leimberger, Ben Rossen, Benjamin Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogo Giertler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll L. Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Chris Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, Dane Sherburn, Daniel Kappler, Daniel Levin, Daniel Levy, David Carr, David Farhi, David Mély, David Robinson, David Sasaki, Denny Jin, Dev Valladares, Dimitris Tsipras, Doug Li, Phong Duc Nguyen, Duncan Findlay, Edede Oiwoh, Edmund Wong, Ehsan Asdar, Elizabeth Proehl, Elizabeth Yang, Eric Antonow, Eric Kramer, Eric Peterson, Eric Sigler, Eric Wallace, Eugene Brevdo, Evan Mays, Farzad Khorasani, Felipe Petroski Such, Filippo Raso, Francis Zhang, Fred von Lohmann, Freddie Sulit, Gabriel Goh, Gene Oden, Geoff Salmon, Giulio Starace, Greg Brockman, Hadi Salman, Hai-Biao Bao, Haitang

Hu, Hannah Wong, Haoyu Wang, Heather Schmidt, Heather Whitney, Hee woo Jun, Hendrik Kirchner, Henrique Pondé de Oliveira Pinto, Hongyu Ren, Huiwen Chang, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian Osband, Ian Silber, Ian Sohl, Ibrahim Okuyucu, Ikai Lan, Ilya Kostrikov, Ilya Sutskever, Ingmar Kanitscheider, Ishaan Gulrajani, Jacob Coxon, Jacob Menick, Jakub W. Pachocki, James Aung, James Betker, James Crooks, James Lennon, Jamie Ryan Kiros, Jan Leike, Jane Park, Jason Kwon, Jason Phang, Jason Teplitz, Jason Wei, Jason Wolfe, Jay Chen, Jeff Harris, Jenia Varavva, Jessica Gan Lee, Jessica Shieh, Ji Lin, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joanne Jang, Joaquin Quiñonero Candela, Joe Beutler, Joe Landers, Joel Parish, Johannes Heidecke, John Schulman, Jonathan Lachman, Jonathan McKay, Jonathan Uesato, Jonathan Ward, Jong Wook Kim, Joost Huizinga, Jordan Sitkin, Jos Kraaijeveld, Joshua Gross, Josh Kaplan, Josh Snyder, Joshua Achiam, Joy Jiao, Joyce Lee, Juntang Zhuang, Justyn Harriman, Kai Fricke, Kai Hayashi, Karan Singhal, Katy Shi, Kavin Karthik, Kayla Wood, Kendra Rimbach, Kenny Hsu, Kenny Nguyen, Keren Gu-Lemberg, Kevin Button, Kevin Liu, Kiel Howe, Krithika Muthukumar, Kyle Luther, Lama Ahmad, Larry Kai, Lauren Itow, Lauren Workman, Leher Pathak, Leo Chen, Li Jing, Lia Guy, Liam Fedus, Liang Zhou, Lien Mamitsuka, Lilian Weng, Lindsay McCallum, Lindsey Held, Ouyang Long, Louis Feuvrier, Lu Zhang, Lukasz Kondraciuk, Lukasz Kaiser, Luke Hewitt, Luke Metz, Lyric Doshi, Mada Aflak, Maddie Simens, Made laine Boyd, Madeleine Thompson, Marat Dukhan, Mark Chen, Mark Gray, Mark Hudnall, Marvin Zhang, Marwan Aljubeh, Ma teusz Litwin, Matthew Zeng, Max Johnson, Maya Shetty, Mayank Gupta, Meghan Shah, Mehmet Ali Yatbaz, Mengxue Yang, Mengchao Zhong, Mia Glaese, Mianna Chen, Michael Janner, Michael Lampe, Michael Petrov, Michael Wu, Michele Wang, Michelle Fradin, Michelle Pokrass, Miguel Castro, Miguel Castro, Mikhail Pavlov, Miles Brundage, Miles Wang, Mina Khan, Mira Murati, Mo Bavarian, Molly Lin, Murat Yesildal, Nacho Soto, Natalia Gimelshein, Na talie Cone, Natalie Staudacher, Natalie Summers, Natan LaFontaine, Neil Chowdhury, Nick Ryder, Nick Stathas, Nick Turley, Nikolas A. Tezak, Niko Felix, Nithanth Kudige, Nitish Shirish Keskar, Noah Deutsch, Noel Bundick, Nora Puckett, Ofir Nachum, Ola Okelola, Oleg Boiko, Oleg Murk, Oliver Jaffe, Olivia Watkins, Olivier Godement, Owen Campbell-Moore, Patrick Chao, Paul McMillan, Pavel Belov, Peng Su, Peter Bak, Peter Bakkum, Peter Deng, Peter Dolan, Peter Hoeschele, Peter Welinder, Phil Tillet, Philip Pronin, Phil Tillet, Prafulla Dhariwal, Qim ing Yuan, Rachel Dias, Rachel Lim, Rahul Arora, Rajan Troll, Randall Lin, Raphael Gontijo Lopes, Raul Puri, Reah Miyara, Reimar H. Leike, Renaud Gaubert, Reza Zamani, Ricky Wang, Rob Donnelly, Rob Honsby, Rocky Smith, Rohan Sahai, Rohit Ramchandani, Romain Huet, Rory Carmichael, Rowan Zellers, Roy Chen, Ruby Chen, Ruslan Ramilevich Nigmatullin, Ryan Cheu, Saachi Jain, Sam Altman, Sam Schoenholz, Sam Toizer, Samuel Miserendino, Sandhini Agarwal, Sara Culver, Scott Ethersmith, Scott Gray, Sean Grove, Sean Metzger, Shamez Hermani, Shantanu Jain, Shengjia Zhao, Sherwin Wu, Shino Jomoto, Shirong Wu, Shuaiqi Xia, Sonia Phene, Spencer Papay, Srinivas Narayanan, Steve Coffey, Steve Lee, Stewart Hall, Suchir Balaji, Tal Broda, Tal Stramer, Tao Xu, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Cunninghman, Thomas Degry, Thomas Dimson, Thomas Raoux, Thomas Shadwell, Tianhao Zheng, Todd Underwood, Todor Markov, Toki Sherbakov, Tom Rubin, Tom Stasi, Tomer Kaftan, Tristan Heywood, Troy Peterson, Tyce Walters, Tyna Eloundou, Valerie Qi, Veit Moeller, Vinnie Monaco, Vishal Kuo, Vlad Fomenko, Wayne Chang, Weiyi Zheng, Wenda Zhou, Wesam Manassra, Will Sheu, Wojciech Zaremba, Yash Patil, Yilei Qian, Yongjik Kim, Youlong Cheng, Yu Zhang, Yuchen He, Yuchen Zhang, Yujia Jin, Yunxing Dai, and Yury Malkov. 2024. Gpt-4o system card. *ArXiv*, abs/2410.21276.

Zi-Han Jiang, Chien-Wei Lin, Wei-Hua Li, Hsuan-Tung Liu, Yi-Ren Yeh, and Chu-Song Chen. 2025. Relation-rich visual document generator for visual information extraction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 14449–14459.

Kyudan Jung, Hojun Cho, Jooyeol Yun, Soyoung Yang, Jaehyeok Jang, and Jaegul Choo. 2025. Talk to your slides: Language-driven agents for efficient slide editing. *Preprint*, arXiv:2505.11604.

Kling AI. 2024. Kling ai. https://klingai.com/. Accessed: 2025-10-04.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jia-Liang Xiong, Xin Li, Bo Wu, Jianwei Zhang, Kathrina Wu, Qin Lin, Junkun Yuan, Yanxin Long, Aladdin Wang, Andong Wang, Changlin Li, Duojun Huang, Fan Yang, Hao Tan, Hongmei Wang, Jacob Song, Jiawang Bai, Jianbing Wu, Jinbao Xue, Joey Wang, Kai Wang, Mengyang Liu, Peng-Yu Li, Shuai Li, Weiyan Wang, Wenqing Yu, Xi Deng, Yang Li, Yi Chen, Yutao Cui, Yuanbo Peng, Zhen Yu, Zhiyu He, Zhiyong Xu, Zixiang Zhou, Z. Xu, Yang-Dan Tao, Qinglin Lu, Songtao Liu, Daquan Zhou, Hongfa Wang, Yong Yang, Di Wang, Yuhong Liu, Jie Jiang, and Caesar Zhong. 2024. Hunyuanvideo: A systematic framework for large video generative models. *ArXiv*, abs/2412.03603.

Chao Liao, Liyang Liu, Xun Wang, Zhengxiong Luo, Xinyu Zhang, Wenliang Zhao, Jie Wu, Liang Li, Zhi Tian, and Weilin Huang. 2025. Mogao: An omni foundation model for interleaved multi-modal generation. *arXiv preprint arXiv:2505.05472*.

Jinpeng Lin, Min Zhou, Ye Ma, Yifan Gao, Chenxi Fei, Yangjian Chen, Zhang Yu, and Tiezheng Ge. 2023. Autoposter: A highly automatic and content-aware design system for advertising poster generation. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 1250–1260.

Haohe Liu, Zehua Chen, Yiitan Yuan, Xinhao Mei, Xubo Liu, Danilo P. Mandic, Wenwu Wang, and Mark D. Plumbley. 2023. Audioldm: Text-to-audio generation with latent diffusion models. In *International Conference on Machine Learning*.

Haohe Liu, Gael Le Lan, Xinhao Mei, Zhaoheng Ni, Anurag Kumar, Varun Nagaraja, Wenwu Wang, Mark D Plumbley, Yangyang Shi, and Vikas Chandra. 2024a. Syncflow: Toward temporally aligned joint audio-video generation from text. *arXiv preprint arXiv:2412.15220*.

Yixin Liu, Kai Zhang, Yuan Li, Zhiling Yan, Chujie Gao, Ruoxi Chen, Zhengqing Yuan, Yue Huang, Hanchi Sun, Jianfeng Gao, et al. 2024b. Sora: A review on background, technology, limitations, and opportunities of large vision models. *arXiv preprint arXiv:2402.17177*.

Suyash Maniyar, Vishvesh Trivedi, Ajoy Mondal, Anand Mishra, and C. V. Jawahar. 2025. Ai-generated lecture slides for improving slide element detection and retrieval. *Preprint*, arXiv:2506.23605.

OpenAI. 2023. Dall·e 3. https://openai.com/dall-e-3/. Accessed: 2025-10-04.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Bjrn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494.

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

Quan Sun, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, Yueze Wang, Hongcheng Gao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. 2023. Emu: Generative pretraining in multimodality. *arXiv preprint arXiv:2307.05222*.

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

Changyao Tian, Xizhou Zhu, Yuwen Xiong, Weiyun Wang, Zhe Chen, Wenhai Wang, Yuntao Chen, Lewei Lu, Tong Lu, Jie Zhou, et al. 2024. Mm-interleaved: Interleaved image-text generative modeling via multi-modal feature synchronizer. *arXiv preprint arXiv:2401.10208*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aur'elien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models. *ArXiv*, abs/2302.13971.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023b. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Xiaofeng Meng, Ningying Zhang, Pandeng Li, Ping Wu, Ruihang Chu, Rui Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wen-Chao Zhou, Wente Wang, Wen Shen, Wenyuan Yu, Xianzhong Shi, Xiaomin Huang, Xin Xu, Yan Kou, Yan-Mei Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhengbin Han, Zhigang Wu, and Ziyu Liu. 2025a. Wan: Open and advanced large-scale video generative models. *ArXiv*, abs/2503.20314.

Duomin Wang, Wei Zuo, Aojie Li, Ling-Hao Chen, Xinyao Liao, Deyu Zhou, Zixin Yin, Xili Dai, Daxin Jiang, and Gang Yu. 2025b. Universe-1: Unified audio-video generation via stitching of experts. *arXiv preprint arXiv:2509.06155*.

Yixuan Weng, Minjun Zhu, Guangsheng Bao, Hongbo Zhang, Jindong Wang, Yue Zhang, and Linyi Yang. 2024. Cycleresearcher: Improving automated research via automated review. *arXiv preprint arXiv:2411.00816*.

Xiaojie Xu, Xinli Xu, Sirui Chen, Haoyu Chen, Fan Zhang, and Ying-Cong Chen. 2025. Pregenie: An agentic framework for high-quality visual presentation generation. *Preprint*, arXiv:2505.21660.

Zekai Zhang, Yiduo Guo, Yaobo Liang, Dongyan Zhao, and Nan Duan. 2024. Pptc-r benchmark: Towards evaluating the robustness of large language models for powerpoint task completion. *ArXiv*, abs/2403.03788.

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

A Landscape of Generative AI

In this section, we present a comprehensive landscape of **Generative AI models**, systematically categorized by their *output modality* and *composition schema*, as summarized in Table 3. The taxonomy spans three primary categories—**Unimodal Generation**, **Bimodal Generation**, and **Multimodal Compositional Generation**—each reflecting distinct levels of representational complexity and integration across modalities.

Category 1: Unimodal Generation. This category includes models that generate content within a *single* modality, such as text, image, video, or audio, without explicit compositional reasoning. Representative text generation models include GPT-4 (Achiam et al., 2023) and Llama3 (Touvron et al., 2023b), while image generation is exemplified by Imagen (Saharia et al., 2022) and DALL-E 3 (OpenAI, 2023). For video generation, Imagen Video (Ho et al., 2022) and Sora (Liu et al., 2024b) demonstrate advanced frame-level synthesis capabilities. In the audio domain, AudioLM (Borsos et al., 2023) and MusicGen (Copet et al., 2023) showcase high-fidelity single-modality generation. These unimodal systems are powerful within their respective domains but lack cross-modal or compositional reasoning.

Category 2: Bimodal Generation. This category integrates two modalities, allowing either interleaved or joint generation schemes. For example, Gemini 2.5 Pro (Team et al., 2023) and Mm-interleaved (Tian et al., 2024) produce interleaved text–image outputs through fixed sequential schemas. Meanwhile, Veo 3 (Google, 2025) and Kling (Kling AI, 2024) enable joint audio–video generation via parallel composition, aligning multimodal signals in time and semantics. These models represent early progress toward modality alignment and synchronized generative reasoning.

Category 3: Multimodal Compositional Generation. The most advanced class focuses on structured, layout-aware integration across multiple modalities. WebDesignAgent (DAMO-NLP-SG, 2024) autonomously composes HTML-based webpages by integrating textual, visual, and structural content. CycleResearcher (Weng et al., 2024) extends this paradigm to scientific document generation, producing LaTeX-formatted academic papers with coherent structure and multimodal content. Finally, our proposed system generates *presentations* that combine text, images, charts, and layouts across slides, leveraging both sequential and spatial composition. These systems embody **compositional intelligence**, where multimodal elements are not merely co-generated but hierarchically organized within spatial and semantic frameworks.

Overall, the landscape reveals a clear progression from *single-modality generation* toward *structured multimodal composition*, marking a paradigm shift from raw content synthesis to reasoning-driven, layout-aware generative intelligence.

B Details of PPTGen

B.1 Defined API calls

Our framework provides a comprehensive suite of APIs for fine-grained presentation generation. These APIs can be broadly categorized into several groups: (1) **Slide Operations** for creating, navigating, and modifying slides; (2) **Selection Methods** for targeting specific shapes, textboxes, or other elements; (3) **Element Operations** for manipulating the properties of selected elements, such as text, pictures, tables, and charts, including their content, formatting, and positioning. A comprehensive list of all supported APIs and their parameters is provided in Table 4.

C Details of MCG-PPTBench

This section details the design principles, data construction process, and evaluation methodology for MCG-PPTBench.

C.1 Design Principles

We constructed MCG-PPTBench based on the principle of comprehensive diversity across three critical dimensions:

Diversity of Task Type The benchmark covers the three core capabilities of Multimodal Compositional Generation (MCG): free generation, controllability, and editability.

Diversity of Task Domain We ensure broad coverage of topics and modalities.

For **Free Generation**, we sourced topics from 20 distinct domains (e.g., Business, Technology, Education) found in SlidesCarnival [1].

For **Controllability** and **Editability**, the tasks cover a wide range of atomic operations on different modalities, including text, images, tables,

[1] https://slidescarnival.com

Table 3: The landscape of Generative AI, categorized by their output modality and composition schema.

Generative Task	Representative Model	Output Modality	Composition Schema
Category 1: Unimodal Generation			
Text-Only	GPT-4(Achiam et al., 2023), Llama3(Touvron et al., 2023b)	Single	No Composition
Image-Only	Imagen(Saharia et al., 2022), DALL-E 3(OpenAI, 2023)	Single	No Composition
Video-Only	Imagen Video(Ho et al., 2022), Sora(Liu et al., 2024b)	Single	No Composition
Audio-Only	AudioLM(Borsos et al., 2023), MusicGen(Copet et al., 2023)	Single	No Composition
Category 2: Bimodal Generation			
Interleaved Text-Image	Gemini2.5 Pro(Team et al., 2023), Mm-interleaved(Tian et al., 2024)	Dual	Sequential (Fixed)
Joint Audio-Video	Veo3(Google, 2025), Kling(Kling AI, 2024)	Dual	Parallel (Fixed)
Category 3: Multimodal Compositional Generation			
Web Page	WebDesignAgent(DAMO-NLP-SG, 2024)	Multiple	Auto-Rendered (HTML)
Academic Paper	CycleResearcher(Weng et al., 2024)	Multiple	Auto-Rendered (LaTeX)
Poster	AutoPoster(Lin et al., 2023), POSTA(Chen et al., 2025)	Multiple	2D Canvas
Presentation	Ours	Multiple	2D Canvas + Sequential

charts, and autoshapes. These operations are detailed in Table 5 and Table 6, with statistics shown in Figure 6.

Diversity of Task Complexity We incorporated varying levels of difficulty across all tasks. For free generation, presentations range from 3 to 7 pages. For controllability and editability, instructions combine 1 to 3 atomic operations to test the handling of complex compositional requirements.

C.2 Data Construction

Free Generation The input is a high-level, open-ended query. We generate each query by sampling a topic and a target length (3 to 7 pages), then using an LLM to synthesize them into a natural language prompt.

Controllability The input is a detailed instruction for creating a single slide. We construct each instruction by sampling 1 to 3 atomic commands. An LLM then composes these into a coherent instruction and generates corresponding evaluation

API Name	Parameters	Description
	Slide Operations	
create_slide	-	Creates a new blank slide and sets it as the current slide.
move_to_slide	idx	Sets the slide at the specified index as the current slide.
set_background_color	color	Sets the background color of the current slide using a hex code.
	Selection Methods	
choose_shape_by_type	shape_type, index	Selects a shape of a specific type and index on the current slide.
choose_textbox	-	Selects the first textbox on the current slide.
choose_picture	-	Selects the first picture on the current slide.
choose_table	index	Selects the table at the specified index on the current slide.
choose_chart	-	Selects the first chart on the current slide.
choose_autoshape	-	Selects the first autoshape on the current slide.
choose_cell	row, col	Selects a cell in the currently selected table.
	Textbox Operations	
insert_textbox	left, top, width, height	Inserts a new textbox at the specified position and dimensions.
insert_text	text	Inserts or replaces text in the selected shape or cell.
append_text	text	Appends text to the selected shape.
delete_text	-	Deletes all text from the selected shape or cell.
set_font_size	size	Sets the font size for the text in the selected shape.
set_font_color	color	Sets the font color for the text in the selected shape.
set_font_bold	is_bold	Toggles bold formatting for the text in the selected shape.
set_font_italic	is_italic	Toggles italic formatting for the text in the selected shape.
set_font_underline	is_underline	Toggles underline formatting for the text in the selected shape.
set_font_style	name	Sets the font name (e.g., 'Arial') for the text in the selected shape.
	Picture Operations	
insert_picture	path, left, top, width, height	Inserts a picture from a file path at the specified position and dimensions.
	Generic Shape Operations	
set_height	height	Sets the height of the selected shape.
set_width	width	Sets the width of the selected shape.
set_left	left	Sets the horizontal position (left edge) of the selected shape.
set_top	top	Sets the vertical position (top edge) of the selected shape.
set_fill	color	Sets the fill color of the selected shape.
	Table Operations	
insert_table	rows, cols, left, top, width, height	Inserts a new table with specified rows, columns, position, and dimensions.
set_cell_text	row, col, text	Sets the text of a specific cell in the selected table.
set_cell_fill	row, col, color	Sets the background color of a specific cell.
set_table_font_size	size	Sets the font size for all cells in the selected table.
set_cell_font_size	row, col, size	Sets the font size for a specific cell.
set_table_fill	color	Sets the background color for all cells in the selected table.
set_cell_alignment	row, col, alignment	Sets the text alignment for a specific cell.
	Chart Operations	
insert_chart	chart_type, left, top, width, height, data_json	Inserts a chart of a specified type, position, dimensions, and with given data.
set_chart_title	title	Sets the title of the selected chart.
set_chart_legend	visible	Shows or hides the legend of the selected chart.
set_chart_data_labels	visible	Shows or hides the data labels of the selected chart.

Table 4: A comprehensive list of defined APIs for presentation generation and manipulation.

criteria. Finally, all cases are validated by humans.

Editability The input consists of a pre-existing slide and an edit instruction. We first create an instruction by sampling 1 to 3 atomic commands. Then, we programmatically generate a valid starting slide that contains the required elements for the edit to be applicable.

C.3 Evaluation

All evaluations in MCG-PPTBench adopt an MLLM-as-Judge paradigm. Concretely, we use OpenAI's GPT-4o (vision-capable) as the judge model. The judge is queried with temperature set to 0 and receives task-specific prompts together with rendered slide images; it produces structured reasoning and final decisions required by each task.

Free Generation The MLLM assesses the full presentation against the original query. It provides scores on a 1-5 scale for three dimensions: *Logic Coherence*, *Layout Quality*, and *Modality Richness*. Concise rationales are recorded with the scores.

Controllability The MLLM evaluates the generated slide against the given instruction and its criteria (e.g., specified chart, table, textbox, picture). Using Chain-of-Thought (CoT) reasoning, it assigns a binary score of 1 (satisfied) or 0 (not satisfied)for each criterion.This process yields two metrics: a **shape-level score** (the proportion of satisfied criteria for the targeted elements,) and a **slide-level score** (a binary pass/fail for the entire instruction).

Editability The MLLM compares the original and edited slides against the edit instruction and its criteria. Using CoT reasoning, it evaluates two conditions: (1) correct implementation of the intended changes for the targeted shapes, measured as a **changed shape-level score** by criterion-hit ratio; and (2) perfect preservation of all unrelated shapes, measured as an **unchanged score** (0/1). The **slide-level score** is 1 only if both conditions are fully satisfied.

For reproducibility, all evaluations are implemented in our `PPTGen/eval` scripts with deterministic settings (temperature 0) and use batched processing with timeouts and retries to ensure robustness.

C.4 Evaluation Prompts

In this section, we illustrate the prompts used for MLLM-based evaluation. For Free Generation, see Figure 8. For Controllability, see Figure 9. For Editability, see Figure 10.

D Implementation Details of Baselines

This section details the implementation methodology for each baseline model, ensuring a fair comparison by utilizing the same underlying models where applicable.

D.1 Model Implementation Details

The model usage in PPTGen is distributed across several modules, each responsible for a specific task. For generating the textual content and overall structure of the presentation, we use the GPT-4.1 model as content planner. For planning the visual layout of slides, we employ the gpt-image-1 model as layout planner. For image generation, the system primarily uses the wanx2.1 model, supplemented by a web-based image retrieval module as a fallback. Finally, to interpret user instructions and modify the slide accordingly, we utilize the GPT-4.1 as slide editor, enabling natural language–driven editing operations.

D.2 Model Implementation Details

GPT4o This approach treats slide generation as a visual creation task. For free-form generation, the image generation model is sequentially conditioned on a rendering of the previous slide and the user's instruction to produce the next slide image. Controlled generation is implemented as an instructed image generation task, where user specifications directly guide the visual output, while editability is handled as an image editing task by modifying the original slide rendered as an image.

PreGenie This method implements the core idea of the PreGenie framework using LaTeX, as the original code is not available. The process involves using an LLM to convert user content into a structured LaTeX document, which is then compiled to render the final slide without explicit layout planning. For both controllability and editability tasks, user instructions are processed by the LLM to modify the underlying LaTeX markup before recompilation.

PPTAgent This method follows the approach of PPTAgent by leveraging a library of pre-designed templates. For free-form generation, the system iterates over a set of 30 open-source templates, using an LLM to generate content that is populated into

selected layouts. In the controllability task, user prompts are first compiled into a Markdown file that is processed by the Markdown-to-slide function using the same templates. However, PPTAgent does not support true editability, as it cannot directly modify existing slides or update specific elements based on user instructions. Since PPTagent is built upon a template-based approach, it is inherently constrained by the predefined element configurations within those templates. Consequently, it cannot arbitrarily add or remove element types, nor can it perform type transformations (e.g., converting a chart into a table). In essence, PPTagent functions more as a document-to-PPT converter rather than an autonomous editing agent capable of dynamically manipulating existing slides. Therefore, from a functional design perspective, it lacks the ability to execute automated PPT editing in response to natural language instructions.

E Case Study

In this section, we present the comparative case study in Figure 7.

Control Type	Description	Examples
Text	Adding brief textual content to slides.	Insert titles, labels, short descriptions, or captions.
Long English	Inserting extended English text passages.	Add detailed paragraphs, comprehensive explanations, or lengthy content blocks.
Chinese	Adding Chinese language text content.	Insert Chinese text, poetry, cultural content, or translations.
Image	Incorporating visual elements into presentations.	Add pictures, logos, illustrations, or background images from files.
Table	Creating structured data presentations.	Insert tables with headers, rows, and organized data content.
Bar Chart	Generating categorical data visualizations.	Create bar charts with categories, values, and comparative data.
Line Chart	Creating trend and time-series visualizations.	Generate line charts showing data progression over time or categories.
Pie Chart	Producing proportional data representations.	Create pie charts displaying percentage distributions and market shares.

Table 5: Table of PPT Control Types

Edit Type	Description	Examples
Textbox Revision	Modifying the content or properties of a textbox.	Replace text, add/delete text, translate content, change font/size.
Image Manipulation	Altering images or pictures on a slide.	Replace image, delete image, rotate, zoom in/out, crop.
Table Revision	Making changes to a table's structure or content.	Replace entire table, add/delete rows or columns, edit cell content.
Modality Conversion	Transforming content from one format (modality) to another.	Text -> AutoShape, Table -> Chart, Chart -> Table, Chart -> Text description.

Table 6: Table of PPT Edit Types

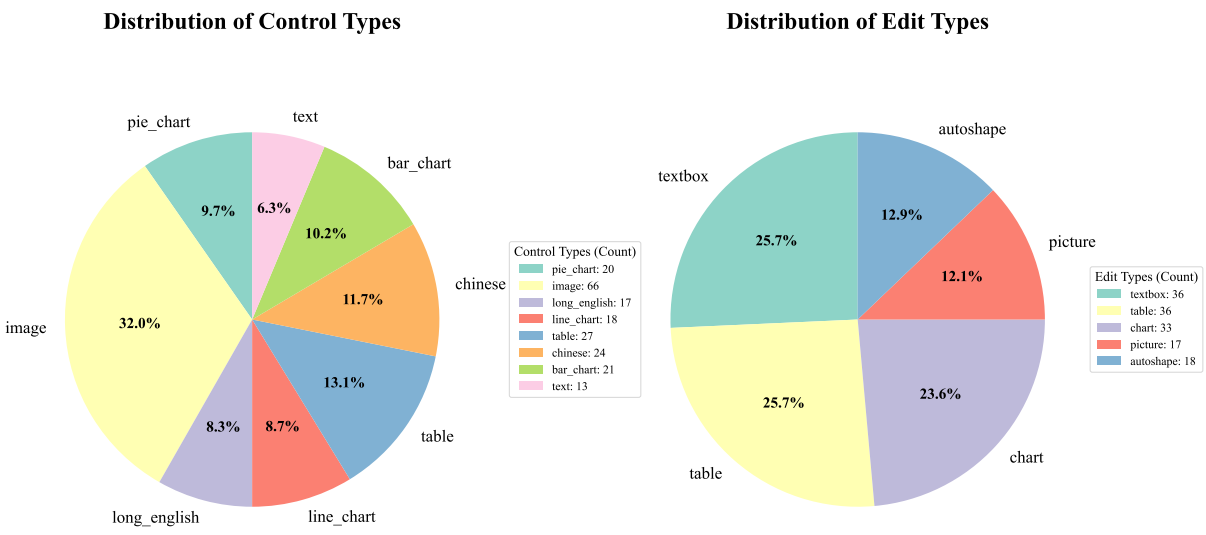

Figure 6: **Overview of data preparation.**

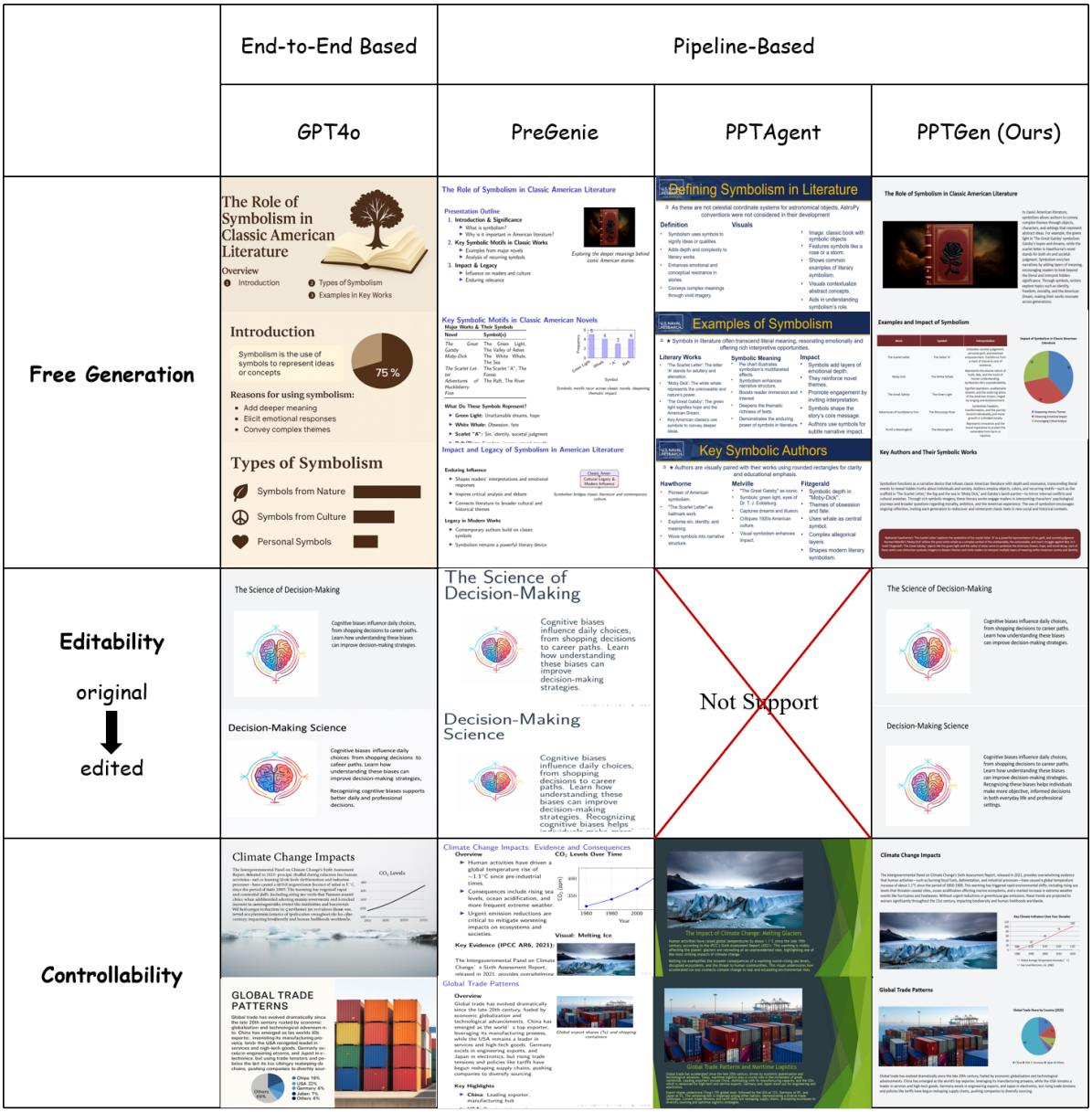

Figure 7: **Case study**

Evaluation Prompt for Free Generation

You are an expert presentation evaluator. Please analyze this batch of len(encoded_images) slides and evaluate them on three dimensions. Each dimension should be scored from 1-5 Points (Excellent).
Evaluation Dimensions:
1. Thematic Relevance and Logical Connectivity:
1 Point (Poor): Slides exhibit completely disconnected themes with no identifiable logical sequence, random topic jumps, and absence of any coherent narrative thread throughout the presentation.
2 Points (Below Average): Slides show minimal thematic connection with weak logical transitions, significant content gaps between slides, and unclear progression that confuses the overall message.
3 Points (Average): Slides maintain moderate thematic relevance with basic logical flow, some unclear transitions between key points, but overall direction is discernible to the audience.
4 Points (Good): Slides demonstrate strong thematic coherence with clear logical progression, well-structured transitions between concepts, and minor inconsistencies in narrative flow.
5 Points (Excellent): Slides exhibit exceptional thematic unity with seamless logical connectivity, perfect narrative structure, smooth transitions, and compelling storyline that engages throughout.
2. Layout Rationality and Visual Aesthetics:
1 Point (Poor): Slides contain severe layout issues including overlapping elements, content extending beyond slide boundaries, excessive blank spaces, and fundamentally broken spatial organization.
2 Points (Below Average): Slides show basic layout attempts but suffer from significant spacing problems, poor element distribution, some boundary violations, and unbalanced visual composition.
3 Points (Average): Slides maintain acceptable layout rationality with elements properly contained within boundaries, reasonable space utilization, but suboptimal visual balance and aesthetic appeal.
4 Points (Good): Slides demonstrate good layout rationality with well-distributed elements, effective space utilization, proper boundaries, and appealing visual design with minor aesthetic improvements needed.
5 Points (Excellent): Slides exhibit exceptional layout rationality with perfect element distribution, optimal space utilization, flawless boundaries, stunning visual aesthetics, and professional design quality.
3. Multimodal Information Richness:
1 Point (Poor): Slides contain exclusively plain text content with no visual elements, charts, images, tables, shapes, or any form of multimodal information presentation.
2 Points (Below Average): Slides include basic text formatting and one additional modality (either simple images or basic shapes), but lack diversity in information presentation methods.
3 Points (Average): Slides incorporate 2-3 distinct information modalities (such as text + images + basic charts), showing moderate variety in content presentation approaches.
4 Points (Good): Slides feature 4-5 different information modalities (text, images, charts, tables, shapes) with good integration and appropriate usage for content enhancement.
5 Points (Excellent): Slides demonstrate exceptional multimodal richness with 6+ diverse modalities (textboxes, tables, images, charts, autoshapes, diagrams) seamlessly integrated for maximum information density.
IMPORTANT: You must respond with ONLY a valid JSON object. Do not include any markdown formatting, code blocks, or additional text. The response should start with and end with .
JSON format:
{
"thematic_relevance_connectivity": "reason": "detailed explanation", "score": number,
"layout_rationality_aesthetics": "reason": "detailed explanation", "score": number,
"multimodal_information_richness": "reason": "detailed explanation", "score": number
}

Figure 8: Evaluation prompt for free generation

Evaluation Prompt for Controllability

Your task is to judge whether the generated ppt image follows instruction.
Instruction
instruction
Rule
- Check whether the generated image meets the following criterion.
The criterion is:
{criterion}
Output Format
first output the judging process, finally output bbox{1} if the generated picture follows instruction else {0}.

Figure 9: Evaluation prompt for controllability

Evaluation Prompt for Editability

Your task is to judge whether the edited ppt image follows the intruction and matches eval criteria.
The first image is the original ppt image, while the second is the edited ppt image.
Instruction
instruction
Rule
- Only the shape mentioned in the criterion needs edition.
The critetion is:
{criterion}
- For all the shape that needs edition, carefully check whether they meet the corresponding criterion.
- For all the shape that does not need edition, carefully check they are EXACTLY THE SAME before and after edition, including the content, data, layout, position or appearance.
- You should enumerate all the shapes on the ppt image, tell me whether it needs edition, and whether it meets criterion or remain unchanged.
Output Format
first output the judging process, finally output
bbox{1} if the edited follows instruction
else bbox{0}.

Figure 10: Evaluation prompt for editability