UniData: Universal Multimodal Instruction Generation Pipeline

Jiaqi Tang[1]*, Yi-Feng Wu[2]*, Yuting Zhang[3]*, Hao Lu[3], Bowen Fu[4], Qing-Guo Chen[2], Xiaogang Xu[5], Yuwei Hu[2], Shiyin Lu[2], Wei Wei[4], Lei Zhang[4], Zhao Xu[2], Weihua Luo[2], Qifeng Chen[1]†, Ying-Cong Chen[3]†

[1]The Hong Kong University of Science and Technology [2]Alibaba Group
[3]The Hong Kong University of Science and Technology (Guangzhou)
[4]Northwestern Polytechnical University [5]Zhejiang University
{jtang092, cqf}@ust.hk, yingcongchen@hkust-gz.edu.cn

Abstract

Multimodal Large Language Models (MLLMs) are increasingly being applied in a wider range of real-world scenarios. However, due to the substantial labor cost, creating high-quality multimodal instruction datasets for MLLMs remains a significant challenge. Although some methods propose to generate instruction data, they often face limitations in modality support and struggle with generating multi-round instructions. To address these problems, we introduce UniData, a universal instruction generation pipeline, to transform simple user requirements into multi-round, multimodal instructions. Specifically, UniData first expands user requirements into multiple diverse events. Using these events, UniData then integrates an any-to-any large model for multimodal instruction generation. Finally, UniData enhances data quality by correcting irrelevant and redundant flows during inference, leveraging correlations between instruction rounds. To train this pipeline, we also build UniDataset, a dataset comprising $20,000$ entries across nine modalities for improved multimodal generation. Our experiments demonstrate that UniData achieves SOTA performance in data quality and can also enhance the understanding and generation capabilities of other multimodal models.

1 Introduction

"In God we trust;
all others must bring data."

— W. Edwards Deming

Recently, Multimodal Large Language Models (MLLMs) have been extensively developed across various fields (Cui et al., 2024; Li et al., 2024a). However, to propel MLLMs toward fulfilling real-world requirements, data scarcity has gradually become a significant issue (Liu et al., 2023; Müller and Unay, 2017; Yin et al., 2023; Wang et al., 2024a, 2023).

To acquire high-quality data, most previous methods have relied on manual collection and annotation (Li et al., 2017), which consequently demands substantial human resources (Tang et al., 2024; Liu et al., 2024b; Fang et al., 2024). While some approaches (Li et al., 2023b; Liu et al., 2023) have started utilizing GPT (OpenAI, 2024) for automated or semi-automated data generation to alleviate labor costs, most of them are designed for one specific scenario and cannot generalize to broader applications.

In this paper, we propose UniData, a universal multimodal instruction generation pipeline. As shown in Figure 1, by simply inputting basic requirements, users can finally obtain large-scale multimodal instruction datasets.

While some solutions have attempted to generate instructions in a universal setting, such as Self-Instruct (Wang et al., 2023) and VIGC (Wang et al., 2024a), these methods are primarily restricted to language modality (as shown in Figure 2 (A)(B)). Multimodal Self-Instruct (Zhang et al., 2024) attempts to generate instructions with corresponding abstract images, but its capabilities are limited to this (as shown in Figure 2 (C)). Furthermore, all of these methods provide limited support for multi-round instruction generation.

To address the aforementioned problems, we first focus on enhancing data diversity by expanding user keyword requirements into a series of potential instructional events, as described in Section 4.1. Next, to improve multimodal support, we propose an any-to-any large model that assists users

*Equal contribution.
†Corresponding authors.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 25793–25813
October 24-29, 2026 ©2026 Association for Computational Linguistics

Figure 1: Usage of UniData. **(I)** Users only input one simple requirement, combined with random emotions and occasions to create diverse events. **(II)** Each diverse event will generate one multimodal instruction. **(III)** The generated multimodal instructions include various modalities.

in generating multi-round, multimodal instructions iteratively, as described in Section 4.2. Finally, to ensure data quality in a multi-round context, we correct the instruction flow by assessing the correlation between instructions from previous and current rounds. This process effectively minimizes redundancy and irrelevance during inference (Section 4.3).

To address the limitations in quantity, quality, and copyright issues of existing data (in Section 3), we also construct UniDataset to support the training of UniData. This dataset comprises 20,000 entries and incorporates nine distinct modalities on both the input and output sides. These modalities include `<language>`, `<image>`, `<music>`, `<emoji>`, `<code>`, `<map>`, `<link>`, `<math>`, and `<QR code>` (see Section 3.1). The inclusion of these diverse modalities is intended to enhance our data generation pipeline, enabling the creation of more comprehensive multimodal instructions and thereby improving overall versatility.

Experimental results demonstrate that the quality of our generated data surpasses that of other generation pipelines on all four dimensions we evaluate:

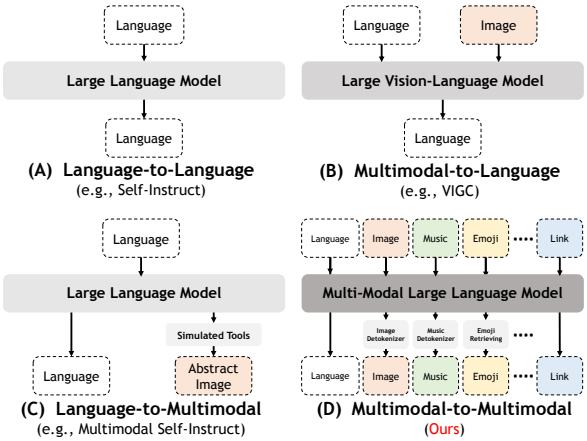


Figure 2: Comparison of different instruction generation frameworks. **(A)** Self-Instruct (Wang et al., 2023) can only support language modality. **(B)** VIGC (Wang et al., 2024a) integrates visual modality into its generation. **(C)** Multimodal Self-Instruct (Zhang et al., 2024) can output abstract images. **(D)** Ours (UniData) can freely understand and generate different modalities in instructions.

reasonableness, clarity, detail, and relevance. Furthermore, our generated data can also enhance the understanding and generation capabilities of other multimodal large language models. Our contribution is shown as follows:

- We propose UniData, a universal multimodal instruction generation pipeline that can create multi-round multimodal instructions from one simple requirement.
- We propose UniDataset, a multimodal instruction dataset comprising 20,000 instructions with nine distinct modalities in both input and output. UniDataset addresses existing limitations in data quantity, quality, and copyright status, and provides a robust data foundation for training UniData or other any-to-any models.
- Through comprehensive experiments, we demonstrate that UniData surpasses existing pipelines in data quality, achieving SOTA performance in reasonableness, clarity, detail, and relevance. Additionally, the generated data can improve the understanding and generation capabilities of other multimodal models.

2 Related Work

Multimodal Large Language Models Multimodal Large Language Models can understand multimodal information (Liang et al., 2024) by integrating various modalities, including but not limited to text, images, audio, and video. Due to

their versatility, they have been widely applied in diverse fields, such as embodied intelligence (Li et al., 2024a), video understanding (Tang et al., 2024), and autonomous driving (Cui et al., 2024).

Most previous multimodal models (Liu et al., 2023; Li et al., 2023b; Lu et al., 2024b) were designed to understand multimodal information, yet their output is limited to language only. Recently, SEED-X (Ge et al., 2024) and Show-o (Xie et al., 2025) have begun to explore the integrated capabilities of image understanding and generation. Additionally, NExT-GPT (Wu et al., 2024) has advanced by integrating more modalities, including images, audio, and video, for unified understanding and generation.

Nonetheless, training these models encounters substantial challenges, particularly regarding data availability. For example, NExT-GPT (Wu et al., 2024) is unable to provide training data due to copyright constraints. Furthermore, Show-o (Xie et al., 2025) is limited to training in an end-to-end way due to the lack of an appropriate dataset. Motivated by these challenges, we propose UniData, which is designed to generate multimodal instructions to facilitate the model training process and potentially support further applications.

Instruction Generation The scarcity of training data impacts the performance of Multimodal Large Language Models, leading to increased interest in how to generate high-quality instruction data. Self-Instruct (Wang et al., 2023) pioneered the generation of diverse instructions using a small set of seed tasks, but it focuses only on the language modality (Figure 2 (A)). To overcome these limitations, VIGC (Wang et al., 2024a) integrates visual understanding into the instruction generation process (Figure 2 (B)). Besides, Multimodal Self-Instruct (Zhang et al., 2024) addresses the problem of abstract image synthesis in instruction generation, utilizing synthetic tools to create both diagrams and corresponding text (Figure 2 (C)). However, these methods still support limited multimodal capabilities.

To address these constraints, we propose UniData, a universal multimodal instruction generation pipeline designed to both understand and generate nine types of multimodal information (Figure 2 (D)). This approach broadens the applicability of multimodal instruction generation compared to previous research.

3 Dataset Construction

To train UniData, a dataset containing both multimodal inputs and outputs is essential. However, datasets meeting these criteria are scarce. As summarised in Table 10 (Appendix A.3), the majority of existing datasets are confined to a single modality, especially for the output. While NExT-GPT (MosIT) (Wu et al., 2024) and GenHowTo (Souček et al., 2024) have developed some initial datasets encompassing multimodal inputs and outputs, they are hindered by a restricted range of modalities, limited instruction rounds, and a small data size. Furthermore, copyright concerns limit their availability.

To enrich our data resources, we have also developed a multimodal dataset named UniDataset (the **Ours** row in Table 10). This dataset supports inputs and outputs across nine modalities, thereby facilitating the training process of UniData.

3.1 Procedure of Data Construction

Stage (I): Preparing Diverse Events To accommodate data diversity, it is crucial to generate varied `<events>` (tasks) for each multi-round instruction. Drawing inspiration from Self-Instruct (Wang et al., 2023), we first create a *Topic Pool* with 90 `<topics>` spanning multiple domains, such as e-commerce, lifestyle, and sports. Then, we randomly pair two topics and input them into GPT-4o (OpenAI, 2024) to generate diverse `<events>`, as shown in Figure 3 (I). Compared to a single `<topic>`, this approach introduces greater randomness ($\binom{90}{2} = 4{,}005$ unique combinations), thereby facilitating data diversity.

Stage (II): Generating Multi-Round Instructions After obtaining `<events>` (tasks), we utilize them to generate multi-round instructions. Considering other attributes of instructions, we also randomly choose different `<rounds>` (ranging from 5 to 30), `<emotion>` (random proportions of seven distinct emotions), and `<occasion>` (formal or informal) to collectively guide GPT-4o (OpenAI, 2024) in generating diverse instructions, as shown in Figure 3 (II). Ultimately, we generate $20,000$ multi-round instructions.

Stage (III): Forecasting Multimodal Integration Our current multi-round instructions are restricted to a single modality (`<language>`). To generate multimodal content, we utilize GPT-4o (OpenAI, 2024) to predict potential multimodal content and

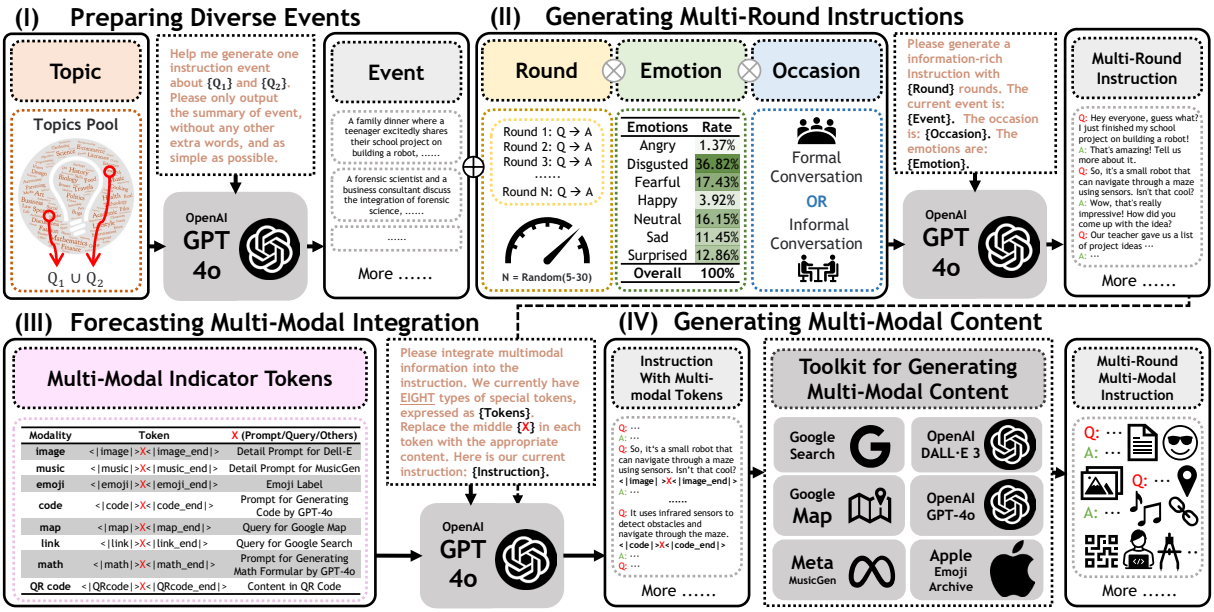


Figure 3: Procedure of data construction. **Stage (I)** prepares diverse events by random topic pairs. **Stage (II)** leverages these events to produce multi-round instructions. **Stage (III)** forecasts eight types of multimodal indicator tokens and integrates these into instructions. **Stage (IV)** uses a toolkit to generate multimodal content.

incorporate the relevant indicators.

Beyond the base <language> modality, we forecast indicators for eight modalities: <image>, <music>, <emoji>, <code>, <map>, <link>, <math>, and <QR code>. Additionally, we generate prompts that describe the corresponding multimodal content, as shown in Figure 3 (III). For instance, we generate prompts for GPT-4o (OpenAI, 2024) for modalities such as <image>, <code>, and <math>. For <map> and <link>, we develop search queries intended for use with Google Search and Google Maps.

Stage (IV): Generating Multimodal Content In the final stage, we develop an advanced Toolkit for Generating Multimodal Content. This toolkit includes GPT-4o (OpenAI, 2024), Google Search, Google Maps, DALL·E 3, MusicGen (Copet et al., 2023), and the Apple Emoji Archive to assist in producing multimodal content. We leverage the generated prompts/content/queries to drive this toolkit, enabling the creation of multimodal content.

3.2 Data Format and Statistics

Each UniDataset sample is generated by mapping a contextual triple to a multimodal **instruction** that contains nine typed slots, one per modality (Eq. (1)). Slots are filled by dedicated generators (Appendix A.2) conditioned on a modality-specific cue (a <prompt>, <label>, <query>, or <content> string), while the <language> slot carries the dialogue text itself.

$$(\texttt{<event>}, \texttt{<emotion>}, \texttt{<occasion>}) \xrightarrow{\text{Generate}} \textbf{instruction} \begin{cases} \rightarrow \texttt{<language>} \\ \texttt{<prompt>} \rightarrow \texttt{<image>} \\ \texttt{<prompt>} \rightarrow \texttt{<music>} \\ \texttt{<label>} \rightarrow \texttt{<emoji>} \\ \texttt{<prompt>} \rightarrow \texttt{<code>} \\ \texttt{<query>} \rightarrow \texttt{<map>} \\ \texttt{<query>} \rightarrow \texttt{<link>} \\ \texttt{<prompt>} \rightarrow \texttt{<math>} \\ \texttt{<content>} \rightarrow \texttt{<QR code>}. \end{cases} \tag{1}$$

4 Methodology

Previous works, such as Multimodal Self-Instruct (Zhang et al., 2024) and VIGC (Wang et al., 2024a), typically support only limited multimodal generation or understanding capabilities, which do not satisfy users' practical needs. In real-world applications, users often prefer to provide only one or a few simple requirements to generate large volumes of high-quality data.

Overview To address the aforementioned issues, we propose UniData, a pipeline for generating multimodal instructions. **Initially**, based on user-provided requirements, we develop an event generator that expands on the limited inputs to create diverse events in Section 4.1. **Next**, to generate multimodal instructions, we construct a large model for multimodal understanding and generation in Section 4.2. **Finally**, to mitigate potential biases during long-term multi-round instruction genera-

tion, we correct the instruction flow to filter out unreasonable data in Section 4.3.

4.1 Expanding Diverse Events

Typically, users seek to generate one high-quality instruction dataset by providing only a few keyword requirements (K), such that ($\mathrm{K} \rightarrow \sum_{n=1}^{N} \mathrm{I}_n$). However, if we treat one K as a single <event> (T), it becomes challenging to produce a diverse set of instructions.

Therefore, we first need to expand the variety of T, so we build the Diverse Event Generator (**DEG**) to expand diverse events, as shown in Figure 4 and Eq. (2), as

$$\sum_{n=1}^{N} \mathrm{I}_n \leftarrow \{\mathrm{T}_1, \mathrm{T}_2, \ldots, \mathrm{T}_N\} = \mathbf{DEG}(\mathrm{K} \oplus \mathrm{S}), \quad (2)$$

where $\oplus$ is the concatenation operator. The set $\{\mathrm{T}_1, \mathrm{T}_2, \ldots, \mathrm{T}_N\}$ denotes the collection of <events>, each corresponding to a distinct instruction (I_n). The term S represents stochastic factors, such as random emotion proportions, random occasions, and other random seeds, which collectively enhance the capability of $\mathbf{DEG}(\cdot)$ to generate a broader spectrum of diverse events.

4.2 Generating Multimodal Instructions

After obtaining diverse <events>, the next step is to generate multimodal instructions. Inspired by previous any-to-any models (Wu et al., 2024; Xie et al., 2025), we develop the Multimodal Instruction Generator (**MIG**), a text-centred generator coupled with modality-specific realisation tools that can both understand and generate multimodal content, as shown in Figure 5 and Eq. (3),

$$\mathrm{I}_n^M = \mathbf{MIG}(\mathrm{T}_n \oplus \sum_{m=1}^{M-1} \mathrm{I}_n^m), \quad (3)$$

where $\mathbf{MIG}(\cdot)$ takes as input one event T_n and previously generated multimodal instructions ($\sum_{m=1}^{M-1} \mathrm{I}_n^m$), and predicts the next-round multimodal instruction (I_n^M).

Multimodal Tokenization To generate the next-round instructions (I_n^M), **MIG** must understand the multimodal instruction from previous rounds ($\sum_{m=1}^{M-1} \mathrm{I}_n^m$). Thus, it is necessary to convert the previous multimodal instructions into token embeddings within a unified feature space.

Given that our training dataset, UniDataset, comprises nine distinct modalities, our tokenizing strategy ($\mathbf{F}(\cdot)$) is shown as Eq. (4),

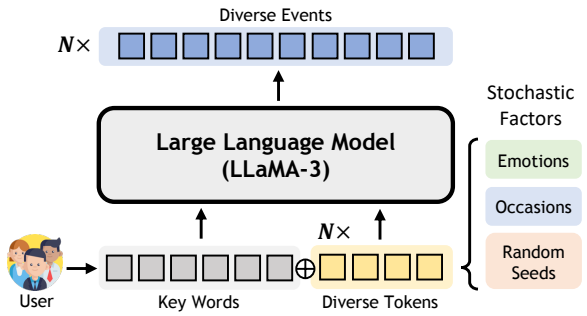


Figure 4: Pipeline of Diverse Event Generator (**DEG**). Users can generate diverse events by inputting their needs.

$$\mathbf{F}(\mathrm{I}) = \begin{bmatrix} \mathbf{f}_{\text{text}}[\mathrm{I}_\mathrm{L}, \mathbf{C}(\mathrm{I}_\mathrm{E} \oplus \mathrm{I}_\mathrm{C} \oplus \mathrm{I}_\mathrm{MP} \oplus \mathrm{I}_\mathrm{LK} \oplus \mathrm{I}_\mathrm{MT} \oplus \mathrm{I}_\mathrm{Q})] \\ \mathbf{f}_{\text{img}}(\mathrm{I}_\mathrm{I}) \\ \mathbf{f}_{\text{mus}}(\mathrm{I}_\mathrm{M}) \end{bmatrix}, \quad (4)$$

where $\mathbf{f}_x(\cdot), x \in \{\text{text}, \text{img}, \text{mus}\}$ represent the tokenizer and embedding layer/projection for different modalities ($\mathrm{I}_x, x \in \{\mathrm{L}, \mathrm{I}, \mathrm{M}, \mathrm{E}, \mathrm{C}, \mathrm{MP}, \mathrm{LK}, \mathrm{MT}, \mathrm{Q}\}$).

For modalities that are not inherently linguistic but can be represented as text ($\mathrm{I}_x, x \in \{\mathrm{E}, \mathrm{C}, \mathrm{MP}, \mathrm{LK}, \mathrm{MT}, \mathrm{Q}\}$), we can easily convert these into textual representations (prompt/label/query/content) by the formatter ($\mathbf{C}(\cdot)$). Then, these modalities can be combined with the language modality (I_L) and tokenized directly by using the default LLaMA Tokenizer (Grattafiori et al., 2024) ($\mathbf{f}_{\text{text}}(\cdot)$). The tokenized outputs are subsequently passed through a pre-trained embedding layer to generate token embeddings.

For <image> (I_I), we utilize the visual tokenizer and projection ($\mathbf{f}_{\text{img}}(\cdot)$) from Ovis-1.5 (Lu et al., 2024b) to convert images into visual embeddings. For <music> (I_M), drawing on previous research (Gong et al., 2023; Chu et al., 2023; Du et al., 2023; Verma, 2024), we integrate a pre-trained Whisper (Radford et al., 2023) as the music tokenizer, followed by a Multi-Layer Perceptron (MLP) that projects into the desired embedding space.

Multimodal Indicator To distinguish different modalities, we introduce new special tokens to indicate the beginning and end of each modality. As shown in Figure 5, the start of a modality is indicated by <|modality|>, where

$$\texttt{modality} \in \{\texttt{image}, \texttt{music}, \texttt{emoji}, \texttt{code}, \texttt{map}, \texttt{link}, \texttt{math}, \texttt{qrcode}\}, \quad (5)$$

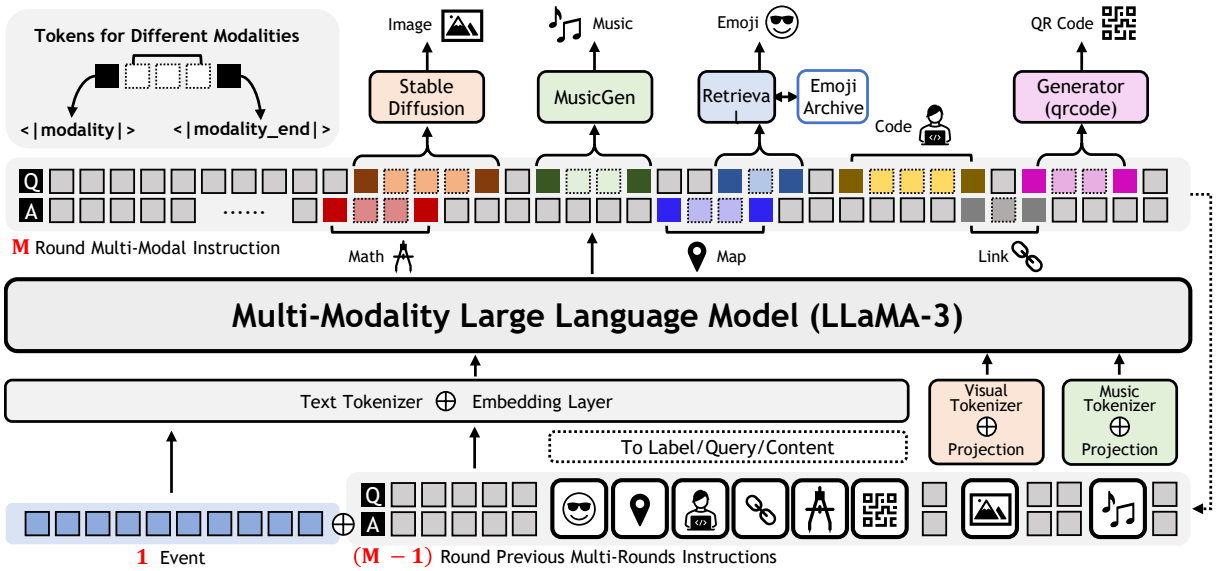


Figure 5: Pipeline of Multimodal Instruction Generator (**MIG**). With just **one** event provided by the user, **MIG** can iteratively generate **multi-round multimodal** instructions based on the context.

and the end of the modality is indicated by `<|modality_end|>`. This strategy allows the model to identify the position of multimodal insertion.

Architecture of Large Language Model After aligning the multimodal signals into a unified embedding space, we select LLaMA-3 (Grattafiori et al., 2024) as the main backbone of our pipeline. We also use the causal attention mechanism to sequentially predict the next token.

Multimodal Generation To generate multimodal instructions, we first need to establish multimodal token labels during training. Although most modalities can be represented as text during tokenization, the image and music modalities lack corresponding text representations. However, we find that during data construction (as described in Eq. (1)), we obtain `<prompts>` ($\mathrm{P_I}$ and $\mathrm{P_M}$) for generating images and music. Therefore, we can use these prompts as tokens to represent these two modalities discretely. Eq. (6) shows the ground truth label ($\mathbf{V}$),

$$\begin{aligned}\mathbf{V} =& \{\mathrm{V_L}, (\mathrm{V_E}, \mathrm{V_C}, \mathrm{V_{MP}}, \mathrm{V_{LK}}, \mathrm{V_{MT}}, \mathrm{V_Q}), \mathrm{V_I}, \mathrm{V_M}\}, \\ \text{where} \quad & \mathrm{V_I} = \mathbf{f}_{\text{text}}(\mathrm{P_I}), \quad \mathrm{V_M} = \mathbf{f}_{\text{text}}(\mathrm{P_M}),\end{aligned} \tag{6}$$

and $\mathbf{f}_{\text{text}}(\cdot)$ is the tokenizer of LLaMA-3 (Grattafiori et al., 2024) and $\mathrm{V}_x, x \in \{\mathrm{L, I, M, E, C, MP, LK, MT, Q}\}$ are the labels of different modalities. Our goal is to predict the next token ($\mathbf{V}^k$) given the previous tokens ($\{\mathbf{V}^i\}_{i=1}^{k-1}$) and the model parameters (Θ), as Eq. (7),

$$\mathcal{L} = -\sum_k \log p(\mathbf{V}^k | \{\mathbf{V}^i\}_{i=1}^{k-1}; \Theta). \tag{7}$$

Multimodal Detokenization We use the default LLaMA-3 text detokenizer (Grattafiori et al., 2024) to convert all modalities into their text representations. For `<code>`, `<map>`, `<link>`, and `<math>`, they can be directly detokenized into their original information. For `<image>`, we use Stable Diffusion 3 (Esser et al., 2024) to transform the text into the corresponding image. For `<music>`, we employ MusicGen (Copet et al., 2023) to convert the text into the corresponding music. For `<emoji>`, we perform a similarity search of the output text to retrieve the corresponding emoji from the emoji archive. For `<QR code>`, we can directly convert its content into a QR code by qrcode[1].

4.3 Correcting Instruction Flow

Although now we can generate multimodal instructions by **MIG**, during inference, it may also encounter common issues like repetitive generation or hallucinations (Xu et al., 2022; Huang et al., 2024). Therefore, cleaning and filtering data during long-term inference is crucial for maintaining data quality. We design an inference chain in Figure 6. We compare the newly generated one-round instruction (I_n^M) against the previous k-round instructions ($\sum_{m=M-k}^{M-1} \mathrm{I}_n^m$) and apply the following three rules:

1. **Repetitive Generation**: If the new instruction is close to the previous ones, increase the REPETITION PENALTY (↑) to reduce repetition.
2. **Irrelevant Generation**: If the new instruction

[1] https://pypi.org/project/qrcode/

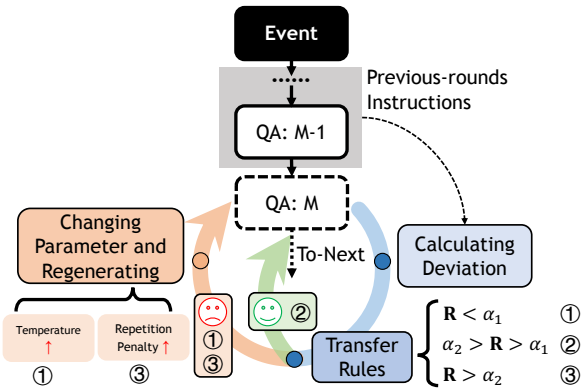


Figure 6: Inference chain for correcting instruction flow.

is unrelated to the previous ones, raise the TEMPERATURE (↑) to introduce randomness, thereby sampling more alternatives.

3. **High-Quality Generation**: If the instruction is neither repetitive nor irrelevant, we regard it as high-quality and proceed to the generation of the next round.

To assess deviation, we use cosine similarity, as Eq. (8),

$$\mathbf{R} = \frac{1}{k} \sum_{m=M-k}^{M-1} \texttt{cos_s}(\mathbf{I}_n^M, \mathbf{I}_n^m) = \frac{1}{k} \sum_{m=M-k}^{M-1} \frac{\mathbf{I}_n^M \cdot \mathbf{I}_n^m}{|\mathbf{I}_n^M|\,|\mathbf{I}_n^m|}. \tag{8}$$

If $\mathbf{R} < \alpha_1$, it indicates "Irrelevant Generation". If $\mathbf{R} > \alpha_2$, it indicates "Repetitive Generation". If $\alpha_2 > \mathbf{R} > \alpha_1$, it indicates "High-Quality Generation". Empirically, we currently set $\alpha_1 = 0.05$ and $\alpha_2 = 0.4$ to minimize errors. Users can also adjust these values dynamically to maintain instruction quality and relevance based on their needs.

5 Experiments

In this section, we compare our instruction generation pipeline with others to evaluate instruction quality (see Section 5.1). Besides, we assess whether the generated instructions can improve multimodal generation and understanding performance after fine-tuning (see Section 5.2). Finally, we demonstrate the effectiveness of expanding events and correcting instruction flow (see Section 5.3).

Furthermore, we conduct a user study to assess the quality of the generated instructions, demonstrate the adaptability and balance of multimodal content to support different needs, validate performance on out-of-domain data, and provide additional qualitative results in the Appendix.

Training & Testing During the pre-training phase, we adhere to the training strategy and dataset outlined in Ovis-1.5 (Lu et al., 2024b), enabling the model to acquire initial multimodal understanding ability. In the fine-tuning phase, we keep the parameters of the visual and music tokenizers fixed and randomly select 90% of UniDataset for model fine-tuning, reserving the remaining 10% for testing.

5.1 Instruction Quality Evaluation

Baselines We select two representative instruction generation frameworks, Self-Instruct (Wang et al., 2023) and VIGC (Wang et al., 2024a), for comparison on our test data. In Self-Instruct (Wang et al., 2023), due to the unavailability of GPT-3, we replace it with the latest GPT-4 (OpenAI, 2024). A discussion of more recent instruction-data generation methods and how they compare in scope is provided in Appendix C.7.

Evaluation Metrics Following previous work (Li et al., 2023b; Tang et al., 2024), we use GPT-Guided metrics to evaluate the quality differences between generated results and references. Our goal is not to precisely match the ground truth but to produce high-quality instructions. We focus on four key metrics: ① **Reasonableness** - assesses if the instruction is logical and appropriate; ② **Clarity** - ensures the instructions are clear and unambiguous; ③ **Detail** - evaluates whether the instructions include more detailed information; ④ **Relevance** - checks that the content is related to the desired outcome.

Performance Analysis Table 1 presents the quantitative results of our data quality. UniData surpasses both baselines on all four dimensions, with the largest margin on Clarity and Relevance. Furthermore, UniData supports more rounds of instruction generation and can produce rich multimodal content within instructions. Figure 7 shows the qualitative performance in data quality. Compared to other baselines that are limited to generating textual content, UniData can generate high-quality multimodal instructions.

5.2 Post Fine-Tuning Assessment

Referring to previous research (Wang et al., 2024a, 2023), we also evaluate **the effectiveness of the generated multimodal instructions** (+ $\mathcal{I}_{\texttt{UniData}}$) by fine-tuning other multimodal models.

Table 1: Quantitative performance in data quality. Red indicates the best performance. (#Others includes all other modalities (#Emoji, #Code, #Math, #Map, #Link, #QR Code)).

Method	Backbones	GPT-Guided Metrics (↑) (OpenAI, 2024)				Avg. #Multimodalities(↑)			Avg. #Rounds
		Reasonableness	Clarity	Detail	Relevance	#Image	#Music	#Others	
Self-Instruct (Wang et al., 2023)	GPT-4 (OpenAI, 2024)	0.624	0.444	0.594	0.520	-	-	-	≈ 3
VIGC (Wang et al., 2024a)	Vicuna 7B (Chiang et al., 2023)	0.460	0.359	0.534	0.569	-	-	-	≈ 1
Ours (UniData)	LLaMA-3 7B (Grattafiori et al., 2024)	0.661	0.527	0.667	0.682	1.92	0.74	7.37	≈ 17.5

Table 2: Improvement of music understanding.

	BLEU (↑)	BLEU-4 (↑)	METEOR (↑)	ROUGE (↑)	BERT (↑)
Baseline	0.2675	0.2019	0.3232	0.3376	0.8932
Baseline + $\mathcal{I}_{\texttt{UniData}}$	0.2681	0.2076	0.3330	0.3496	0.8936

Table 3: Music generation.

	FAD (↓)	CLAP (↑)	KL (↓)
Baseline	16.69	0.3138	2.1358
Baseline + $\mathcal{I}_{\texttt{UniData}}$	16.51	0.3192	2.1038

Table 4: Improvement on different modalities.

Modality	Code (↑)	Math (↑)	Text-based (↑)	Reasoning (↑)
Benchmark	MBPP	MathVista	MMLU-Pro	MathVision
Baseline	20.8	59.4	20.3	12.8
Baseline + $\mathcal{I}_{\texttt{UniData}}$	21.3	60.1	21.9	13.0

Table 5: Image understanding/generation.

	Image Understanding			Image Generation (↑)
	Single (↑)	Multi (↑)	Interleaved (↑)	
Baseline	26.19%	37.73%	19.42%	24.43%
Baseline + $\mathcal{I}_{\texttt{UniData}}$	27.15%	39.09%	35.25%	25.90%

Table 6: Improvement on visual understanding.

Benchmark	MMMU (↑)	ChartQA (↑)	TextVQA (↑)
Baseline	35.8	17.4	58.9
Baseline + $\mathcal{I}_{\texttt{UniData}}$	36.4	17.9	60.1

Table 7: Quantitative performance in ablation study.

	Reasonableness (↑)	Clarity (↑)	Detail (↑)	Relevance (↑)	#Ins.
w/o Expansion	0.557	0.518	0.607	0.550	k
w/o Correction	0.648	0.489	0.610	0.654	$N \times k$
Ours	0.661	0.527	0.667	0.682	$N \times k$

Visual Understanding We utilize a suite of benchmarks (MMMU (Yue et al., 2024), ChartQA (Masry et al., 2022), and TextVQA (Singh et al., 2019)) to evaluate the effectiveness of our generated data in visual reasoning tasks. As detailed in Table 6, the integration of our data improves the performance of Ovis2-1B (Lu et al., 2024b) in these benchmarks.

Image Understanding and Generation We evaluate image understanding and generation capabilities using the SEED-X model (Ge et al., 2024) and the SEED-Bench-2 benchmark (Li et al., 2023a). Our data improves both Image-to-Text and Text-to-Image tasks. As shown in Table 5, the gain is largest on the interleaved setting (19.42% → 35.25%), with smaller gains on the single- and multi-image settings. Additionally, in the domain of image generation, our dataset helps the model produce relevant images based on textual inputs.

Music Understanding and Generation We utilize the MuMu-LLaMA model (Liu et al., 2024a) to assess advancements in music understanding and generation. As illustrated in Tables 2 and 3, our generated instructions improve both domains, although the margins are small. In the realm of music understanding (Table 2), all five metrics move in the expected direction, with the clearest gains on METEOR and ROUGE. In music generation (Table 3), the fine-tuned model exhibits decreased FAD and KL scores alongside an elevated CLAP score, reflecting more coherent and high-quality music synthesis from textual prompts.

Code Generation We evaluate the impact of our generated data on code modality by enhancing the code generation capabilities of a large language model, LLaMA 2 (Touvron et al., 2023), using the Mostly Basic Python Programming (MBPP) Benchmark (Austin et al., 2021). As presented in Table 4, our instructions improve performance.

Mathematical Problem To evaluate the efficacy of our generated instructions in enhancing complex reasoning and mathematical problem-solving capabilities, we benchmark the Ovis2-1B (Lu et al., 2024b) on the MathVista (Lu et al., 2024a) and MathVision (Wang et al., 2024b) datasets. In Table 4, our instructions improve performance on both mathematical benchmarks. The gains are consistent in direction across the two, though modest in magnitude.

Other Text-based Modalities To confirm the versatility of our data across various text-based modalities, including language, QR code, link, and map, we assessed their performance using a general language-based benchmark, MMLU-Pro (Wang et al., 2024c). As shown in Table 4, adding our data raises MMLU-Pro accuracy from 20.3 to 21.9, indicating that the text-based modalities also transfer to a general language benchmark.

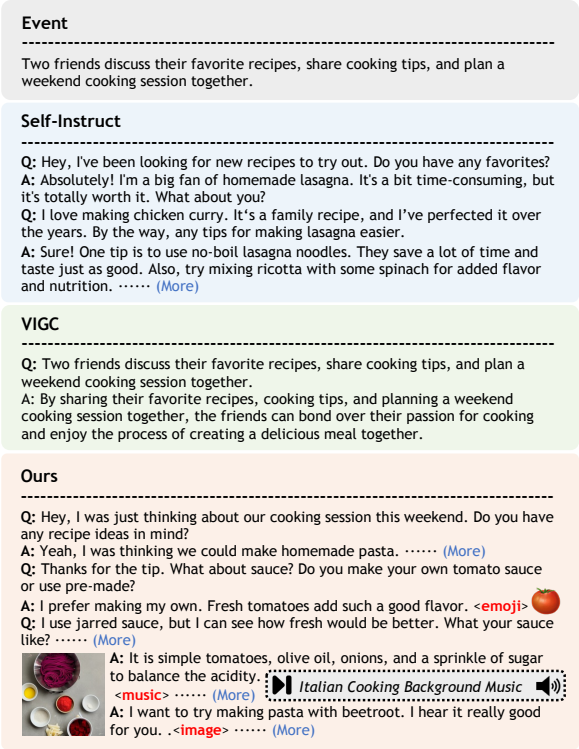


Figure 7: Qualitative comparison in data quality.

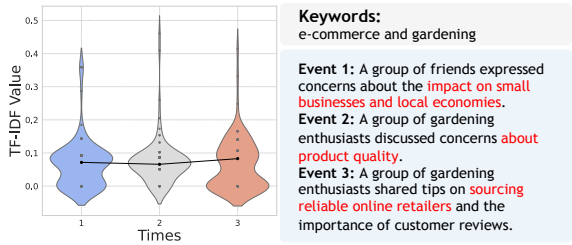


Figure 8: (**left**) Difference distribution of the events in three random generations. (**right**) Example of diverse events based on the same keywords.

5.3 Ablation Study

Effectiveness of Expanding Diverse Events To demonstrate the effectiveness of expanding events, we randomly expand three events for each keyword on the test data to observe whether these events differ from one another (diversity). As shown in Figure 8 (**left**), the TF-IDF (Ramos et al., 2003) feature distributions across three generation runs (on a random subset) show considerable diversity. In Figure 8 (**right**), we also provide an example showing that different events exhibit distinct content (in highlighted text). In the **w/o Expansion** setting of Table 7, the generation of instructions relies solely on the few keywords provided by the user. Thus, the generation pipeline lacks the necessary enriched prompt information, which leads to a decline in the quality of generated instructions.

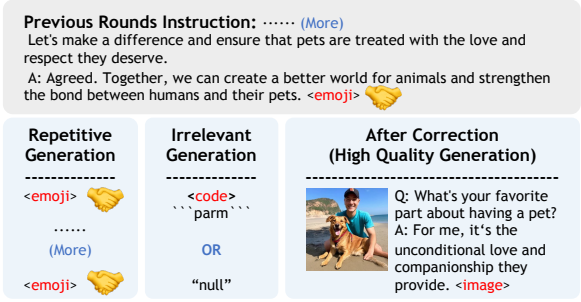


Figure 9: Example of flow error correction.

Additionally, the number of instructions that can be generated is substantially reduced.

Effectiveness of Correcting Instruction Flow To demonstrate the effectiveness of correcting instruction flow, we run inference twice, *with* and *without* the correction, as shown in Table 7 and Figure 9. Table 7 shows that after correction, our data quality improves on all four dimensions. Figure 9 provides an example of the repetitive generation problem (same `<emojis>`) and irrelevant generation problem (`<code>` or "null") in multimodal content. After correction, instruction generation can effectively avoid these errors.

6 Conclusion

This paper presents UniData, a universal pipeline for multimodal instruction generation. As a fundamental contribution, UniData establishes a framework for scalable data acquisition critical to MLLMs. By delivering high-quality, diverse, and cost-effective instruction sets, UniData accelerates progress in multimodal learning paradigms and their downstream applications.

Limitations

UniData is a text-centered orchestration pipeline rather than a native any-to-any model, so per-modality quality is bounded by the external tools that realize each modality. Long-horizon structured reasoning is handled only by simple correction rules, and the nine-modality coverage still omits video, 3D, motion, tables, biological data, and spoken input. Finally, UniData inherits biases from its LLM backbones, search APIs, and modality generators; these may enter UniDataset and be amplified downstream, and systematic bias audits remain future work.

References

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and 1 others. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing GPT-4 with 90%* ChatGPT quality. https://lmsys.org/blog/2023-03-30-vicuna/.

Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. 2023. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models. *Preprint*, arXiv:2311.07919.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. 2023. Simple and controllable music generation. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Can Cui, Yunsheng Ma, Xu Cao, Wenqian Ye, Yang Zhou, Kaizhao Liang, Jintai Chen, Juanwu Lu, Zichong Yang, Kuei-Da Liao, and 1 others. 2024. A survey on multimodal large language models for autonomous driving. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 958–979.

Shengyuan Ding, Shenxi Wu, Xiangyu Zhao, Yuhang Zang, Haodong Duan, Xiaoyi Dong, Pan Zhang, Yuhang Cao, Dahua Lin, and Jiaqi Wang. 2025. MM-IFEngine: Towards multimodal instruction following. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1099–1109.

Zhihao Du, Jiaming Wang, Qian Chen, Yunfei Chu, Zhifu Gao, Zerui Li, Kai Hu, Xiaohuan Zhou, Jin Xu, Ziyang Ma, Wen Wang, Siqi Zheng, Chang Zhou, Zhijie Yan, and Shiliang Zhang. 2023. Lauragpt: Listen, attend, understand, and regenerate audio with gpt. *Preprint*, arXiv:2310.04673.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. 2024. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 12606–12633. PMLR.

Xinyu Fang, Kangrui Mao, Haodong Duan, Xiangyu Zhao, Yining Li, Dahua Lin, and Kai Chen. 2024. MMBench-Video: A long-form multi-shot benchmark for holistic video understanding. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*.

Yukang Feng, Jianwen Sun, Chuanhao Li, Zizhen Li, Jiaxin Ai, Fanrui Zhang, Yifan Chang, Sizhuo Zhou, Shenglin Zhang, Yu Dai, and Kaipeng Zhang. 2026. A high quality dataset and reliable evaluation for interleaved image-text generation. In *International Conference on Learning Representations (ICLR)*.

Yuying Ge, Sijie Zhao, Jinguo Zhu, Yixiao Ge, Kun Yi, Lin Song, Chen Li, Xiaohan Ding, and Ying Shan. 2024. Seed-x: Multimodal models with unified multi-granularity comprehension and generation. *arXiv preprint arXiv:2404.14396*.

Yuan Gong, Alexander H. Liu, Hongyin Luo, Leonid Karlinsky, and James Glass. 2023. Joint audio and speech understanding. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Jacob Hansen, Wei Lin, Junmo Kang, Muhammad Jehanzeb Mirza, Hongyin Luo, Rogerio Feris, Alan Ritter, James Glass, and Leonid Karlinsky. 2025. Instructify: Demystifying metadata to visual instruction tuning data conversion. *arXiv preprint arXiv:2505.18115*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2024. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*.

Bohao Li, Yuying Ge, Yixiao Ge, Guangzhi Wang, Rui Wang, Ruimao Zhang, and Ying Shan. 2023a. Seed-bench-2: Benchmarking multimodal large language models. *arXiv preprint arXiv:2311.17092*.

Guoliang Li, Yudian Zheng, Ju Fan, Jiannan Wang, and Reynold Cheng. 2017. Crowdsourced data management: Overview and challenges. In *Proceedings of the 2017 ACM international conference on Management of Data*, pages 1711–1716.

KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. 2023b. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*.

Luxi Li, Yuchen Li, Xiaotong Zhang, Yuhang He, Jianjian Yang, Bin Tian, Yunfeng Ai, Lingxi Li, Andreas Nüchter, and Zhe Xuanyuan. 2024a. Embodied intelligence in mining: Leveraging multi-modal large language model for autonomous driving in mines. *IEEE Transactions on Intelligent Vehicles*.

Pengxiang Li, Zhi Gao, Bofei Zhang, Tao Yuan, Yuwei Wu, Mehrtash Harandi, Yunde Jia, Song-Chun Zhu, and Qing Li. 2024b. FIRE: A dataset for feedback

integration and refinement evaluation of multimodal models. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Zijing Liang, Yanjie Xu, Yifan Hong, Penghui Shang, Qi Wang, Qiang Fu, and Ke Liu. 2024. A survey of multimodel large language models. In *Proceedings of the 3rd International Conference on Computer, Artificial Intelligence and Control Engineering*, CAICE '24, page 405–409, New York, NY, USA. Association for Computing Machinery.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36.

Shansong Liu, Atin Sakkeer Hussain, Qilong Wu, Chenshuo Sun, and Ying Shan. 2024a. Mumu-llama: Multi-modal music understanding and generation via large language models. *arXiv preprint arXiv:2412.06660*.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. 2024b. MMBench: Is your multi-modal model an all-around player? In *Computer Vision – ECCV 2024*, pages 216–233. Springer.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2024a. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations (ICLR)*.

Shiyin Lu, Yang Li, Qing-Guo Chen, Zhao Xu, Weihua Luo, Kaifu Zhang, and Han-Jia Ye. 2024b. Ovis: Structural embedding alignment for multimodal large language model. *arXiv:2405.20797*.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279. Association for Computational Linguistics.

Henning Müller and Devrim Unay. 2017. Retrieval from and understanding of large-scale multi-modal medical datasets: a review. *IEEE transactions on multimedia*, 19(9):2093–2104.

OpenAI. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 28492–28518. PMLR.

Juan Ramos and 1 others. 2003. Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, volume 242, pages 29–48. Citeseer.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. 2019. Towards VQA models that can read. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8317–8326.

Tomáš Souček, Dima Damen, Michael Wray, Ivan Laptev, and Josef Sivic. 2024. Genhowto: Learning to generate actions and state transformations from instructional videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jiaqi Tang, Hao Lu, Ruizheng Wu, Xiaogang Xu, Ke Ma, Cheng Fang, Bin Guo, Jiangbo Lu, Qifeng Chen, and Ying-Cong Chen. 2024. HAWK: Learning to understand open-world video anomalies. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pages 139751–139785.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, and 49 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Prateek Verma. 2024. Whisper-gpt: A hybrid representation audio large language model. *Preprint*, arXiv:2412.11449.

Bin Wang, Fan Wu, Xiao Han, Jiahui Peng, Huaping Zhong, Pan Zhang, Xiaoyi Dong, Weijia Li, Wei Li, Jiaqi Wang, and Conghui He. 2024a. VIGC: Visual instruction generation and correction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 5309–5317.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Mingjie Zhan, and Hongsheng Li. 2024b. Measuring multimodal mathematical reasoning with MATH-Vision dataset. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508. Association for Computational Linguistics.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, and 1 others. 2024c. Mmlu-pro: A more robust and challenging

multi-task language understanding benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. 2024. Next-gpt: Any-to-any multimodal llm. In *Proceedings of the International Conference on Machine Learning*, pages 53366–53397.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. 2025. Show-o: One single transformer to unify multimodal understanding and generation. In *International Conference on Learning Representations (ICLR)*.

Jin Xu, Xiaojiang Liu, Jianhao Yan, Deng Cai, Huayang Li, and Jian Li. 2022. Learning to break the loop: Analyzing and mitigating repetitions for neural text generation. In *Advances in Neural Information Processing Systems*.

Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Yuhao Dan, Chenlin Zhao, Guohai Xu, Chenliang Li, Junfeng Tian, Qian Qi, Ji Zhang, and Fei Huang. 2023. mplug-docowl: Modularized multimodal large language model for document understanding. *Preprint*, arXiv:2307.02499.

Zhenfei Yin, Jiong Wang, Jianjian Cao, Zhelun Shi, Dingning Liu, Mukai Li, Xiaoshui Huang, Zhiyong Wang, Lu Sheng, Lei Bai, Jing Shao, and Wanli Ouyang. 2023. LAMM: Language-assisted multimodal instruction-tuning dataset, framework, and benchmark. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, and 1 others. 2024. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567.

Letian Zhang, Quan Cui, Bingchen Zhao, and Cheng Yang. 2025. Oasis: One image is all you need for multimodal instruction data synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3542–3551.

Wenqi Zhang, Zhenglin Cheng, Yuanyu He, Mengna Wang, Yongliang Shen, Zeqi Tan, Guiyang Hou, Mingqian He, Yanna Ma, Weiming Lu, and Yueting Zhuang. 2024. Multimodal self-instruct: Synthetic abstract image and visual reasoning instruction using language model. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 19228–19252. Association for Computational Linguistics.

Overview of the Appendix

This appendix is organized as follows:

- **Dataset construction** (Appendix A): the topic pool that seeds generation, the external multimodal toolkit, dataset diversity, bias-mitigation procedures, and modality extensibility.
- **Methodology** (Appendix B): a worked example of the instruction-flow correction step and an analysis of why UniData improves over a vanilla GPT-4o pipeline.
- **Experiments** (Appendix C): training and inference recipes, evaluation prompts, a user study, out-of-distribution results, and ablations on multimodal balance and user-conditioned generation.
- **Additional analyses**: a scope comparison with recent instruction-data pipelines (Appendix C.7).
- **Future work** (Appendix D) and additional qualitative examples (Figures 10–13).

A Dataset Construction

A.1 Topic Pool

We curate a **topic pool** of 90 domains spanning everyday life, science and engineering, business, the arts, and the humanities (Table 8). The breadth of this pool is the primary driver of UniDataset's content diversity, as it determines the universe of scenarios from which subsequent events are drawn.

For each instruction sample we draw two topics Q_1, Q_2 uniformly at random and merge them into a composite theme,

$$\langle topic \rangle \;=\; Q_1 \cap Q_2. \tag{9}$$

Intersecting unrelated domains (for instance, *cooking* $\cap$ *chemistry* or *music* $\cap$ *marketing*) introduces controlled stochasticity into the pipeline and yields interdisciplinary prompts that are difficult to obtain from any single-domain source.

A.2 Multimodal Toolkit

To realize the any-to-any generation interface described in the main paper, UniData orchestrates a suite of off-the-shelf tools, each responsible for a single modality. Table 9 summarizes every component, its target modality, its source, and the specific role it plays in the pipeline. We deliberately favor publicly available APIs and open-source models so that the pipeline can be reproduced and extended by the community.

A.3 Data Diversity

Table 10 compares UniDataset with widely-used multimodal instruction datasets along three orthogonal axes: (i) the number of dialogue rounds per sample, (ii) the count of distinct input/output modalities, and (iii) the overall corpus size. UniDataset supports nine modalities on both the input and the output side, the largest modality set in this comparison, and provides on average 17.5 rounds per dialogue (vs. at most 4.8 for prior work).

A.4 Human Oversight for Bias Mitigation

To reduce the bias inherited from any single LLM, we insert lightweight human oversight at two carefully chosen points in the pipeline.

Topic curation. The 90 seed topics are hand-selected to balance category coverage and cultural breadth, preventing the corpus from skewing toward a narrow set of domains.

Event filtering. After event expansion, annotators discard implausible events and remove near-duplicates. Cleaner events produce better downstream prompts and, importantly, mitigate the propagation of LLM-specific stylistic biases into the final instructions.

A.5 Modality Extensibility

The pipeline is designed to absorb new modalities without architectural changes.

Stage 3 (raw-instruction generation). The planner emits modality-agnostic slots that are filled at realization time, so adding a modality beyond the current nine does not require changing the planner.

Stage 4 (multimodal realization). New APIs (such as Text-to-Video or Text-to-3D models) can be plugged in via a thin adapter that maps modality slots to API calls, with no change to upstream components.

B Methodology Details

B.1 Instruction-Flow Correction in Practice

Correction is performed *before* multimodal realization. At this stage every multimodal slot is still represented by its textual prompt rather than the rendered media, so the corrector can reason about content, dependencies, and redundancy purely at the language level. A representative snippet is shown below.

Table 8: The 90-domain topic pool used to seed UniDataset generation. At sampling time two domains are intersected (Eq. 9) to form a composite theme.

e-commerce	academic discussions	lifestyle	sports	mathematics	business
technology	travels	health	entertainment	art	history
cooking	parenting	fashion	finance	politics	literature
gardening	astronomy	music	education	computer science	programming
design	research	internship	food	beauty	entrepreneurship
startups	algorithm	bugs	scholar	IT	vision
supervision	classroom	assignment	game	psychology	social media
film	theater	architecture	photography	environment	agriculture
real estate	marketing	economics	philosophy	law	medicine
engineering	physics	biology	robotics	cybersecurity	artificial intelligence
family	space exploration	climate change	urban planning	transportation	hospitality
tourism	fitness	relationships	holidays	charity	DIY projects
survival skills	life	cartoon	pets	wildlife	marine biology
genetics	neuroscience	anthropology	sociology	linguistics	culture
mythology	archaeology	forensic science	criminology	military	diplomacy

Table 9: External tools used by UniData during multimodal realization. Each tool is responsible for one modality, allowing the pipeline to be extended modularly.

Tool	Modality	Source	Role in the pipeline
GPT-4o (OpenAI, 2024)	<language>	OpenAI API	Orchestrator and language backbone; expands events, drafts dialogue, and validates outputs.
Google Search	<link>	https://www.google.com	Provides up-to-date factual snippets and reference URLs.
Google Maps	<map>	https://www.google.com/maps	Supplies geographical references and routing information.
DALL·E 3	<image>	https://openai.com/index/dall-e-3/	Synthesizes photo-realistic and stylized images from text prompts.
MusicGen (Copet et al., 2023)	<music>	Open-source model	Generates instrumental clips from textual descriptions.
Apple Emoji Archive	<emoji>	https://www.apple.com/	Provides culturally familiar emoji symbols for affective cues.

Table 10: Comparison with current multimodal instruction datasets. *Num* is the number of distinct modality types; *Class* lists the modalities themselves. Legend: L:<language>; I:<image>; V:<video>; 3D:<point cloud>; LK:<link>; A:<audio>; M:<music>; E:<emoji>; C:<code>; MP:<map>; MT:<math>; Q:<QR code>.

Dataset	Input Modality		Output Modality		Avg. #Rounds	#Instruction
	Num	Class	Num	Class		
LLaVA (Liu et al., 2023)	2	L/I	1	L	≈ 1.0	5K
VideoChat (Li et al., 2023b)	2	L/V	1	L	≈ 1.8	11K
FIRE (Li et al., 2024b)	2	L/I	1	L	≈ 1.0	100K
LAMM (Yin et al., 2023)	3	L/I/3D	1	L	≈ 3.3	196K
mPLUG-DocOwl (Ye et al., 2023)	4	L/I/T/L	1	L	-	-
GenHowTo (Souček et al., 2024)	2	L/I	2	L/I	-	-
MosIT (Wu et al., 2024)	4	L/I/A/V	4	L/I/A/V	≈ 4.8	5K
Ours (UniDataset)	9	L/I/M/E/C/MP/LK/MT/Q	9	L/I/M/E/C/MP/LK/MT/Q	≈ 17.5	20K

Example: textualized multimodal slot during correction

Q: …
A: This further stimulates economic growth by expanding markets and reducing food scarcity issues.
`<|image|>` *Prompt: farmland used for marketing campaigns.* `<|image_end|>`

Because the textual prompt is already available at this stage, the corrector can detect redundancy or topical drift without inspecting the final media, which keeps correction fast and cheap. A tighter feedback loop that conditions the corrector on the rendered media itself is left to future work.

B.2 Why UniData Outperforms a Vanilla Pipeline

Richer multimodal toolkit. UniData couples GPT-4o with an external toolkit (DALL·E for images, MusicGen for audio, search APIs for facts), extending the effective modality range far beyond what a single language model can produce natively.

Pipeline design. In Table 12, Self-Instruct runs on a GPT-4 backbone while UniData runs on LLaMA-3 7B, yet UniData scores higher on all four GPT-guided metrics. Under this comparison the gain is attributable to event expansion and instruction-flow correction rather than to backbone capacity.

C Experimental Details

C.1 Training and Inference

Training of DEG. The Diverse Event Generator (DEG) is fine-tuned on top of LLaMA-3 using $\langle topic \rangle \rightarrow \langle event \rangle$ pairs harvested from UniDataset. Fine-tuning takes roughly 40 hours on a single NVIDIA A100 (80 GB).

Training of MIG. The Multimodal Instruction Generator (MIG) is fine-tuned on four NVIDIA A100 (80 GB) GPUs for roughly 40 hours.

C.2 Evaluation Prompt and Metrics

We define four GPT-4o-judged metrics (*Reasonableness*, *Clarity*, *Detail*, and *Relevance*) and query the judge with the unified prompt shown below.

Unified GPT-4o evaluation prompt

Evaluate the `{metric}` of the multimodal instruction **{generated}**, compared to the ground truth **{reference}**. Return a single score in $[0, 1]$.
Example output: `{"score": 0.675}`.
Definition of `{metric}`: **{explanation}**.

The four metrics are operationalized as follows:

- **Reasonableness**: logical consistency and factual accuracy with respect to the reference.
- **Clarity**: readability, well-formedness, and unambiguity of the generated text.
- **Detail**: richness of information conveyed in each turn.
- **Relevance**: topical adherence to the input context across the full dialogue.

The evaluator is asked to score each metric independently so that absolute and relative comparisons can be made across the four dimensions.

C.3 User Study

To complement the automatic GPT-4o scores, we run a forced-choice user study. We recruit 15 evaluators from a mix of academic and industrial backgrounds, all with prior experience with large language models. Each evaluator inspects 50 samples and, for every one of the four evaluation dimensions, selects which method among Self-Instruct (Wang et al., 2023), VIGC (Wang et al., 2024a), and UniData produced the best output. The results, reported in Table 11, show that UniData is preferred by a substantial margin on every dimension.

Table 11: User study: percentage of samples (%) on which each method is selected as the best by 15 human evaluators across 50 samples (per-dimension forced choice). Red indicates the best performance.

Method	Reason. (↑)	Clarity (↑)	Detail (↑)	Relev. (↑)
Self-Instruct (Wang et al., 2023)	18.5	9.7	9.8	8.5
VIGC (Wang et al., 2024a)	10.3	5.3	14.4	10.9
Ours (UniData)	71.2	85.0	75.8	80.6

C.4 Out-of-Distribution Performance

To probe generalization, we construct an out-of-distribution (OOD) test set of 100 events drawn from domains outside the seed topic pool (collected via web scraping followed by manual screening). Table 12 shows that UniData remains the strongest method on every metric while simultaneously producing visibly richer multimodal content (more images, music, and other modalities per dialogue) and longer interactions.

C.5 Multimodal Distribution across Dialog Position

A common failure mode of existing pipelines is concentrating multimodal content in the opening turns of a dialogue. Table 13 shows that UniDataset

Table 12: Quality of generated instructions on the out-of-distribution (OOD) set. **Bold** marks the best score in each column. *#Others* aggregates the emoji, code, math, map, link, and QR-code modalities.

Method	Backbones	GPT-Guided Metrics (↑)				Avg. #Multimodalities(↑)			Avg. #Rounds
		Reasonableness	Clarity	Detail	Relevance	#Image	#Music	#Others	
Self-Instruct (Wang et al., 2023)	GPT-4 (OpenAI, 2024)	0.735	0.475	0.608	0.580	-	-	-	≈ 3
VIGC (Wang et al., 2024a)	Vicuna 7B (Chiang et al., 2023)	0.714	0.507	0.732	0.737	-	-	-	≈ 1
Ours (UniData)	LLaMA-3 7B (Grattafiori et al., 2024)	0.752	0.701	0.808	0.845	1.04	0.27	3.82	≈ 10

instead distributes multimodal content nearly uniformly across the entire dialogue (the five normalized buckets each carry roughly 20 % of the multimodal mass). The qualitative examples in Figures 11–13 corroborate this: modalities appear consistently in early, middle, and late rounds.

Table 13: Fraction of multimodal content (%) as a function of the normalized position within a dialogue. Position is binned into five equal-width buckets.

Position in dialogue	Multimodal frequency (%)
0 – 20 %	20.5
20 – 40 %	19.4
40 – 60 %	19.5
60 – 80 %	21.3
80 – 100 %	19.3

C.6 Adaptation to User-Specified Modalities

When the user steers the pipeline toward a particular modality, the generated content shifts accordingly (Table 14). For example, an image-focused input yields roughly 15× more image content than the general setting, while a music-focused input boosts the music count by a similar factor. The orchestration layer therefore responds to the requested modality rather than defaulting to a fixed mix.

Table 14: Average count of each modality per dialogue when the user specifies a desired modality at input time. Red indicates the targeted modality in each row.

Input type	#Image	#Music	#Others
General	1.92	0.74	7.37
Music-related	1.10	12.30	2.10
Image-related	14.90	0.10	5.20

C.7 Comparison with Recent Instruction-Data Pipelines

The two baselines in Table 1 (Self-Instruct and VIGC) are chosen because they represent the two dominant styles of universal instruction-data generation: single-modality seed-based expansion and vision-conditioned instruction generation. For completeness, we position UniData against several recent instruction-data pipelines that appeared alongside or after this work.

- **Oasis** (Zhang et al., 2025) generates language-side instruction–response pairs from a single conditioning image; assistant-side output is text-only.
- **Instructify** (Hansen et al., 2025) iteratively generates multi-turn visual instruction conversations by aggregating over 40 source datasets; assistant-side output is again text-only.
- **MM-IFEngine** (Ding et al., 2025) produces constraint-rich image–instruction–answer triples for SFT and DPO; assistant-side output is text-only.
- **InterSyn** (Feng et al., 2026) synthesizes single- and multi-turn interleaved image–text dialogues, with assistant-side output covering language and image.

These pipelines target different source data, tasks, and evaluation protocols, so their paper-reported scores cannot be placed in a common "quality" column with those in Table 1 without rerunning every method under an identical protocol. We therefore treat them as scope comparisons rather than head-to-head numeric competitors. The distinguishing feature of UniData relative to these pipelines is the combination of *symmetric* nine-modality input/output coverage (rather than a language-only or language+image assistant side) and long multi-round interactions (17.5 rounds on average). Among the recent works listed, InterSyn (Feng et al., 2026) is closest in spirit but still limits assistant-side output to language and image.

C.8 Additional Qualitative Results

Figure 10 visualizes the output diversity of **DEG**: three independently sampled events for the same keyword pair differ substantially in content, tone, and target modality.

Figures 11–13 present three end-to-end examples produced by **MIG**. Each example interleaves several modalities (image, music, code, emoji) across a multi-round dialogue, illustrating both the

modality breadth and the long-horizon coherence achieved by UniData.

D Future Work

We view UniData as a foundation for the automated production of multimodal training data, and identify three concrete directions for follow-up work.

Scaling backbones. Replace the LLaMA-3 backbone of **MIG** with more recent and larger open-source models to characterize how multimodal instruction quality scales with backbone capacity.

Expanding modalities. Plug in tools for video, 3D, motion, and tabular generation so that UniDataset can extend beyond the current nine modalities, increasing applicability to embodied and scientific domains.

User-conditioned generation. Tighten the control loop so that UniData behaves as an interactive data factory: an end user provides their downstream specification, and UniData produces purpose-built data aligned with that deployment scenario.

Keywords

cartoon and supervision

Events

Two friends discuss their favorite cartoons and how they use supervision to ensure their kids watch age-appropriate shows.

Keywords

sports and education

Events

A group of students and a teacher discuss the impact of sports on academic performance. Students share personal experiences, highlighting how playing sports has improved their focus, teamwork skills, and stress levels. The teacher emphasizes the importance of balancing academics and sports, agreeing that physical activity can enhance overall learning outcomes.

Keywords

pets and anthropology

Events

Two friends discuss how studying the relationship between humans and their pets can provide insights into cultural practices, social structures, and human evolution.

Keywords

social media and film

Events

Two friends discuss how social media influences film promotion, noting that viral trends can boost box office sales, while also considering the downside of online criticism and the pressure on filmmakers to create content constantly.

Keywords

parenting and biology

Events

Two parents discuss how understanding their children's emotions can improve their parenting skills.

Keywords

survival skills and family

Events

A father teaches his children how to build a shelter and start a fire in the woods, emphasizing the importance of these skills for survival and family bonding.

Figure 10: Diverse event generation by **DEG**. For the same keyword pair, three independently sampled events differ substantially in content, scenario, and intended modality, illustrating the diversity that drives downstream instruction variety.

Event

Two parents discuss the benefits of supervising their children while watching cartoons, agreeing that it ensures content appropriateness and provides opportunities for educational conversations.

Ours

Q: Good afternoon, how was your day today?
A: Good afternoon. It was quite eventful, thank you. I heard you wanted to discuss something about our children and their TV time?
Q: Yes, I was thinking about how important it is for us to supervise what they are watching. What are your thoughts on this?
A: I completely agree. It's crucial to ensure that the content is appropriate for their age. There are so many shows that aren't suitable for children.
Q: Absolutely. I find it quite disheartening when they come across something they should not have seen. Have you had any similar experiences?
A: Yes, I was quite surprised and upset the first time it happened. It can be quite distressing.
Q: Indeed. It makes me fearful that they might be exposed to harmful content. How do you feel about the potential risks?
A: I share your concerns. The internet is vast, and it's easy to come across inappropriate material. We need to be vigilant.
Q: That true. I also feel a bit sad that they might miss out on educational content. How can we balance supervision with allowing them to explore safely?
A: It's a delicate balance. We can set specific times for supervised viewing and also introduce them to a variety of educational shows. <**link**>

Search: List of recommended educational TV shows for children
Result:
1. What's on these kids TV shows? Any recommendations? Movies...
2. Can you recommend me a good TV series that is suitable for... - Reddit
3. TV Reviews, Kids TV shows | Common Sense Media

Q: That sounds like a good plan. I was disgusted to hear that some parents are so lax. Do you think stricter supervision is necessary?
A: Yes, definitely. Stricter supervision ensures that we can control the content they watch and monitor their viewing habits.
Q: I felt a bit of fear thinking about the long-term consequences if we're not vigilant. Do you think it will help in their development?
A: Absolutely. It provides a safe environment for them to learn and grow without being exposed to negative influences. <**emoji**>
Q: I'm glad we are on the same page. It makes me happy to know that we can make a positive impact. How do you think we should start implementing this?
A: We could start by creating a list of approved shows and setting up parental controls on their devices. <**code**>

…… (More)
approved_shows =
[{"show": "Sesame Street", "age_group": "3-5"}, {"show": "Dora the Explorer", "age_group": "3-5"}, {"show": "Blue's Clues", "age_group": "3-5"},
{"show": "The Magic School Bus", "age_group": "5-8"}, {"show": "Rugrats", "age_group": "5-8"}, {"show": "Hey Arnold!", "age_group": "5-8"},
{"show": "Recess", "age_group": "5-8"}, {"show": "The Simpsons", "age_group": "8-12"}, {"show": "Batman: The Animated Series", "age_group": "8-12"},
{"show": "Talespin", "age_group": "8-12"}]
…… (More)
setup_parental_controls('parental_controls.json', approved_shows)

Q: That a practical approach. Do you think we should also discuss the shows with them to understand their preferences better?
A: Yes, I think that's essential. It will give us a better understanding of what content they are interested in and what we should avoid. <**emoji**>

Q: I felt a bit sad thinking about the time commitment this will require. Do you think it will be manageable?
A: It will be challenging, but I believe it worth it. We can take turns supervising or even set up a shared schedule. .<**image**>
Q: I was surprised by how much thought this requires. Do you think it will be effective in keeping them engaged while still being educational?
A: Yes, I do. It will keep them engaged while we ensure they are watching content that is beneficial for their development.
Q: I feel a bit fearful about the potential pushback they might give. What should we do if they resist? .<**music**>

Calming Background for Education

A: We need to remain consistent and communicate clearly why it's important for their well-being. It might take some time, but persistence is key.
Q: I was disgusted by the idea that some parents might prioritize their own convenience over their children's safety. How do we ensure we stay committed?
A: We should remind ourselves of our primary goal is keeping them safe and educated. It a team effort, and we need to support each other. <**emoji**>
Q: I felt a bit sad thinking about the potential for missed opportunities if we're too strict. How do we find the right balance?
A: We should be flexible and open to adjustments based on what works best for our family. It's not a one-size-fits-all solution. <**math**>
$$
$ = \\frac{T}{F} \\times (1 - \\frac{D}{100}) *How to balance screen time
$$
Q: I was happy to hear that. Do you think it will also help in teaching them valuable life lessons?
A: Definitely. We can use TV time as an opportunity to discuss various topics and teach them critical thinking skills. <**link**>

Search: Effective ways to teach children through TV shows
Result:
1. Better Not TV: What Parents Should Know and Do About TV for Kids
2. How to Effectively Integrate TV into Your Kids' Curriculum - LinkedIn
3. Show Your Child the Difference Between TV Shows & Real Life00

Q: Certainly. It more than just watching cartoons; it about creating a positive and educational experience. <**emoji**>
A: Agreed. By supervising their viewing habits, we are giving them the best possible start in life.

Q: I feel a bit relieved knowing we have a plan. Shall we start implementing it this evening?
A: Yes, let do that. We can create the list of approved shows and set up parental controls. It will be a good start. <**map**>
Location: Our home address
Address: 1815 Avenue M, New York, NY 100090

…… (More)

Figure 11: End-to-end qualitative example (I): a multi-round, multimodal instruction produced by UniData. Image, music, and text are interleaved into a coherent narrative.

Event

A panel of experts discussed the impact of technological advancements in agriculture on global economic growth, highlighting increased crop yields, reduced costs, and potential job displacement.

Ours

Q: Thank you all for joining this important discussion on the impact of technological advancements in agriculture on global economic growth. To start, can you explain the benefits of increased crop yields that technology brings?
A: Certainly. Increased crop yields due to technological advancements result in higher food production, which can lead to lower food prices and increased consumer incomes. This further stimulates economic growth by expanding markets and reducing food scarcity issues. <**image**>
Q: That's a positive outlook. However, you mentioned potential job displacement. Can you elaborate on that?
A: The integration of technology such as automated machinery and precision farming systems can lead to job displacement, particularly for manual labor jobs. This shift requires workers to upgrade their skills beyond merely agricultural tasks, which can be both challenging and costly. <**link**>

Search: Research articles on job displacement due to automation
Result:
1. Understanding how automation affects workers, jobs, and wages
2. AI's impact on jobs and work is a choice between two futures
3. (PDF) The Unintended Consequences of Automation: Separating…

Q: Given the fear this might instill in workers, what measures can be taken to mitigate these risks?
A: Education and training programs are crucial. Governments and private sectors must invest in retraining workers to adapt to higher-tech roles, such as data analysis, machinery maintenance, and administrative tasks. This transition period can be supported through subsidies and workforce development initiatives. <**link**>

Search: linking to resources for workforce development programs
Result:
1. Workforce Development Services - Employers' Training Resources
2. U.S. Department of Labor - WorkSource Texas
3. Workforce Resources - Region 18 Workforce Development Board

---

Q: What about the environmental impact? Can technological advancements in agriculture help address sustainability concerns?
A: Absolutely. Technologies like precision agriculture can reduce waste and conserve resources by optimizing crop management. This not only benefits the environment by minimizing excess fertilizer and pesticide use but also improves crop quality and yields sustainably. <**map**>
Location: Location of a sustainable farming project using precision technology
Title: Mightybytes technology
Address: 35 11th Ave, New York, NY 10011

Q: Finally, considering these mixed effects, do you believe the overall economic growth justified the investment in these technologies, or should there be a more balanced approach?
A: While the potential economic growth from increased yields and efficiency is significant, it's essential to weigh these benefits against the social costs of job displacement and the need for robust support systems for affected workers. A balanced approach that includes investment in education, social safety nets, and sustainable practices would be ideal. <**math**>
$$
\max \\left(\\alpha \\cdot G(T) - \\beta \\cdot C(T) \\right)
$$
Q: Thank you for your insights. This has been a thought-provoking discussion on the multifaceted impact of technological advancements in agriculture.
A: Thank you for having us. It's crucial to continue these conversations to ensure that technological progress benefits all sectors of society, not just the economic ones. <**emoji**> 👏

Figure 12: End-to-end qualitative example (II): a different topic pair, showing variation in dialogue length and modality mix while preserving topical coherence.

Event

A panel of experts discussed the integration of artificial intelligence in robotics, focusing on advancements in machine learning, ethical considerations, and potential impacts on various industries. They highlighted the importance of collaboration between technologists and ethicists to ensure responsible development and deployment.

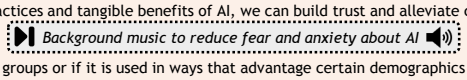
Ours

Q: Good evening, everyone. Let's begin our discussion on the integration of artificial intelligence in robotics. What do you think are the recent advancements in machine learning that are most impactful?
A: One of the most impactful advancements is in deep learning and reinforcement learning. These technologies have significantly enhanced the ability of robots to learn and adapt to complex environments autonomously.
Q: That sounds promising, but what about the ethical considerations? How do we ensure these advancements are used responsibly?

A: Ethical considerations include ensuring transparency in AI decision-making processes and preventing biases. Collaboration with ethicists is crucial to create frameworks that guide responsible development and deployment of robotic systems.
Q: Can you elaborate on the potential impacts of AI in various industries?
A: AI can revolutionize industries by enhancing efficiency and decision-making accuracy. For example, in healthcare, AI-driven robots can assist in surgeries, providing precision and reducing risks. In the automotive industry, AI can improve automation and safety features in vehicles, reducing human error. Additionally, in the financial sector, AI can facilitate fraud detection, thus safeguarding customers.<image>
Q: Those examples are compelling, but they also raise concerns. What are some of the risks associated with AI on a broader scale?
A: Risks include job displacement due to automation and cybersecurity threats as AI systems become more sophisticated. It's essential to address these challenges through robust policies and continuous innovation in security measures.
Q: Collaboration between technologists and ethicists is often mentioned as a solution. How can they effectively work together to address these challenges?
A: Effective collaboration can be fostered through interdisciplinary teams, mutual respect, and open communication channels. It‘s important to integrate ethical considerations from the initial design phase to ensure sustainable and responsible AI applications. <link>

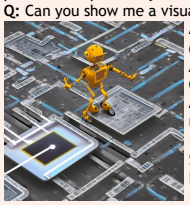
Search: Best practices for interdisciplinary collaboration in AI and ethics
Result:
1. A framework for motion towards human AI-Ethics at scale - ScienceDirect.com
2. Embedding worst-case scenarios into responsible artificial intelligence...
3. [PDF] Responsible Cost Ag_share WEB downloadLevels of abstraction...

Q: Given all these developments, what role should government regulation play in the integration of AI in robotics?
A: Government regulation should ensure that ethical guidelines are adhered to and that accountability mechanisms exist for misuse or unintended consequences. Regulations can also promote innovation by setting standards while adapting as technology evolves.
Q: Are there any examples where government regulation has successfully balanced innovation and safety?
A: The European Union's General Data Protection Regulation (GDPR) is one example. It has balanced privacy rights and innovation by implementing stringent data protection measures, which has largely been met with positive reception.
Q: How do we address the concerns of those who are fearful or distrustful of AI?
A: Addressing fears requires transparency and education. By demonstrating ethical practices and tangible benefits of AI, we can build trust and alleviate concerns. It‘s also important to involve stakeholders in decision-making processes. <music>
Q: What about the potential for AI to perpetuate inequalities if not handled correctly?



A: AI can perpetuate inequalities if access to technology remains limited to privileged groups or if it is used in ways that advantage certain demographics over others. Policies should ensure equitable access and unbiased applications.
Q: It's a complex issue with no easy solutions. How can we stay committed to ethical AI development while rapidly advancing technology?
A: Committing to ethical AI development requires ongoing vigilance, continuous education, and the integration of ethical principles into all stages of technology development. It also requires a shift in mindset where ethical implications are given equal priority with technical innovation.

Q: In conclusion, what are the key takeaways from this discussion on AI integration in robotics?
A: Key takeaways include the importance of machine learning advancements, the need for ethical considerations, the potential positive impacts across various industries, and the necessity for collaboration between technologists and ethicists to ensure responsible development and deployment of AI in robotics. <QR Code>
Q: Thank you all for your insights. As we continue to work towards a responsible integration of AI in robotics, what final thoughts would you like to share with our audience?
A: It‘s an exciting journey with many possibilities. We must remain vigilant, collaborate openly, and prioritize ethical considerations. Together, we can harness AI's potential to positively benefit society while minimizing risks.
Q: Can you show me a visual representation of AI in robotics?



A: Sure, here‘s an image of a robot using AI to navigate a complex environment. <image>
Q: That's fascinating. Can you summarize the three main ethical concerns discussed?
A: Yes, they focused on transparency, prevention of biases, and accountability.
Q: What are some potential impacts of AI on the job market?
A: AI can automate certain tasks, leading to job displacement in some sectors, but it can also create new opportunities in tech and AI-related fields. <link>

Search: AI impact on job market
Result:
1. AI Will Transform the Global Economy. Let's Make Sure It Benefits...
2. The Impact of Artificial Intelligence on Employment: Trends and...
3. The Impact of AI on the Job Market: Trends and Future Projections

--- ····· (More)

Q: What was the response to the GDPR regulation in the tech industry?
A: Responses were mixed; while some welcomed it for protecting privacy, others expressed concerns about its potential to stifle innovation. <emoji> !
Q: Can you write a code snippet for educating AI ethics?
A: Sure, here's a basic code snippet for educational purposes. <code>

```
import random
class AI_Ethics_Quiz:
    def __init__(self):
        self.questions = [ {"question": "What are some ethical concerns regarding AI development?", "options": ["Bias in AI", "Job displacement", "Autonomous weapons", "Risk of misuse"],
"answer": "Bias in AI"}, {"question": "Which of the following is a risk associated with AI but not with human actions?", "options": ["Increased surveillance", "Improved efficiency",
"Enhanced creativity", "Job displacement"], "answer": "Increased surveillance"}, {"question": "What is the most important principle for developing AI?", "options": ["Maximize profit",
"Ensure transparency", "Ensure fairness", "Ensure safety"], "answer": "Ensure fairness"} ]
    def shuffle_questions(self):
        random.shuffle(self.questions)
    def ask_question(self):
        print("Welcome to the AI Ethics Quiz!") for i, question in enumerate(self.questions): print(f"{i+1}: {question['question']}")
        for j, option in enumerate(question['options']):
            print(f"{j+1}. {option}") answer = input("Your answer: ") print("Correct Answer:", question['answer']) print()
    def play_quiz():
        quiz = AI_Ethics_Quiz() quiz.shuffle_questions() quiz.ask_question() play_quiz()
```

····· (More)

Figure 13: End-to-end qualitative example (III): a long-horizon dialogue that interleaves text, image, code, and emoji content, illustrating UniData's behavior on extended interactions.