EVALUATING ANY-TO-ANY INSTRUCTION FOLLOWING WITH OMNIMODAL INSTRUCTION

Anonymous authors
Paper under double-blind review

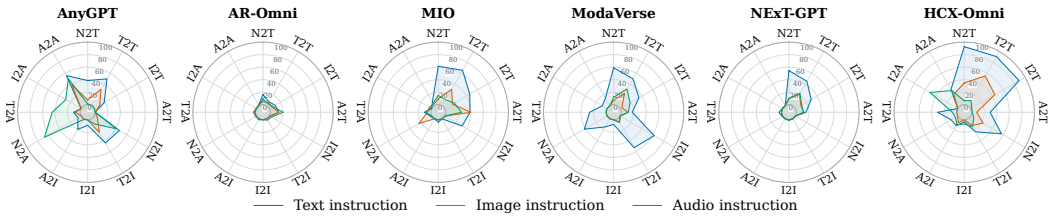


Figure 1: **Any-to-any models follow instructions poorly, and worse when the instruction is not text.** End-to-end instruction following score per context-to-output task (N/T/I/A: none/text/image/audio), with the instruction given as text, image, or speech. OMNI-INSTRUCT measures instruction following across any-to-any tasks and instruction modalities.

ABSTRACT

Any-to-any models are expected to follow user instructions and respond in multiple modalities. In this setting, instruction following is not limited to satisfying content or format constraints, but also requires understanding how the instruction is conveyed. To extensively assess this capability of any-to-any systems, we introduce a novel benchmark, OMNI-INSTRUCT, which evaluates instruction following in any-to-any models with various context, instruction and output modalities across text, image, and audio. Using our proposed framework, we extensively evaluate recent any-to-any models, decoupling the output-modality routing from instruction compliance. From our study, we discover that 1) routing failure is the dominant factor for most apparent performance drops, 2) even when correctly routed, models show distinct compliance failure patterns, and 3) models remain vulnerable to omnimodal instructions, struggling to understand and follow requests conveyed through images or speech.

1 INTRODUCTION

Real-world communication is inherently omnimodal; we speak, write a message, show an image, listen to audio, and often expect responses in different forms depending on the situation. This has motivated the development of any-to-any models (Lu et al., 2024; Wang et al., 2025; Wu et al., 2024; Inclusion AI et al., 2025; Tang et al., 2023) that aim to process inputs across text, image, and audio and produce outputs such as text, speech, images, or combinations of them. As these models operate beyond text-only interactions, their evaluation should also focus on measuring their capability to produce high-quality artifacts across multiple modalities.

Beyond artifact quality, however, a more basic capability must also be assessed: instruction following, which refers to meeting the requirements a user explicitly states (Ouyang et al., 2022; Zhou et al., 2023). For any-to-any models, instruction following becomes considerably more challenging than in conventional LLMs, since the model needs to precisely understand what the user intends across inputs that may span text, image, and audio and then deliver the response in the modality the user specifies. On the input side, the model should be able to distinguish the instruction from the context, which may itself be an image or audio, since instructions may not always be conveyed in text. They may be spoken in audio or embedded in an image in a form such as handwritten text. On the output side, if a user asks for a visual notice but the model returns a textual description, the response may be informative, but the model has still failed to perform the requested action. Such failures unnecessarily

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

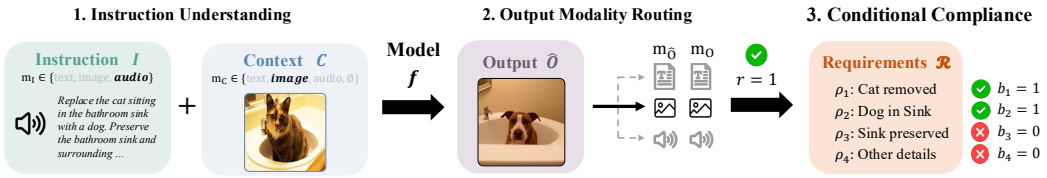


Figure 2: **Overview of OMNI-INSTRUCT.** A model receives an instruction I and a context C in any modality, must route its output to the requested modality, and, only if routed correctly, is scored on binary requirements ρ.

waste the user's time and effort in reformulating and repeating the request, as well as generation tokens spent on unused artifacts.

However, a systematic evaluation of instruction following capability in any-to-any models has been largely underexplored. Conventional benchmarks (Zhou et al., 2023; Jiang et al., 2024; Qin et al., 2024; Qian et al., 2025; Ding et al., 2025) typically assume a fixed input and response modality, focusing only on whether a model satisfies textual, structural, and perception-grounded constraints. Yet, as shown in Fig. 1, existing any-to-any models exhibit substantial variation in instruction following capabilities across diverse input-output modality settings. They even respond very differently depending on the instruction modality alone; for example, simply switching a text instruction to an image or speech form substantially shifts the performance.

To bridge this gap, we introduce OMNI-INSTRUCT, an any-to-any instruction following benchmark with omnimodal instruction, designed to evaluate this missing capability. We first decompose any-to-any instruction following into the three capabilities shown in Fig. 2: understanding the instruction from the given input, producing the output in the requested modality, and satisfying the fine-grained requirements in the instruction. In this benchmark, we independently vary the modality of the instruction, the context, and the requested output. This design allows us to test if a model can infer the intended output from realistic user instructions rather than only from an explicit text instruction.

For a clearer understanding of where a model fails, we separately evaluate the following two capabilities. First, we assess if the model routes to the requested output modality (*output-modality routing*). Then, conditioned on the produced output, we evaluate if the generated artifact satisfies the instruction with modality-specific requirements (*conditional compliance*). This separation is crucial, since a model can produce high-quality content in the wrong modality, or route to the correct modality without satisfying detailed constraints in the returned content. By separating these two, our OMNI-INSTRUCT systematically exposes failure modes of aggregated output-quality evaluation.

We conduct a comparative study on a variety of current any-to-any models, revealing several failure patterns. We discover that in fact many models fail at the early modality routing stage by falling back to text or omitting the requested modality. We also find that the modality used in the instruction itself significantly affects the performance; *e.g.*, replacing a text instruction with speech or image substantially deteriorates both routing accuracy and downstream compliance, even when the semantic request is unchanged. These results suggest that current any-to-any models remain far from robust instruction followers under any-to-any settings.

Our contributions are summarized as follows:

- We formulate any-to-any instruction following as three sequential capabilities—omnimodal instruction understanding, output-modality routing, and conditional compliance—treating the requested output modality as an explicit requirement of the instruction.
- We introduce OMNI-INSTRUCT, which independently varies instruction, context, and output modalities across diverse tasks, together with routing-aware metrics (RA, CC, E2E) that separate failures to deliver the requested modality from failures to satisfy its requirements.
- Leveraging our benchmark, we systematically diagnose current any-to-any models beyond aggregate scores, tracing where along the instruction following process their failures arise and highlighting directions for improvement.

2 RELATED WORK

Instruction Following Benchmarks. Text-only instruction following benchmarks (Zhou et al., 2023; Jiang et al., 2024; Qin et al., 2024; Wen et al., 2024) study verifiable, fine-grained, or decomposed

Table 1: **Comparison of representative instruction following benchmarks.** Instruction and context distinguish instruction delivery from task source material. Routing assesses requested output-modality fulfillment.

Benchmark	Instruction	Context	Output	Modality routing
IFEval (Zhou et al., 2023)	Text	Text	Text	×
FollowBench (Jiang et al., 2024)	Text	Text	Text	×
InFoBench (Qin et al., 2024)	Text	Text	Text	×
ComplexBench (Wen et al., 2024)	Text	Text	Text	×
VisIT-Bench (Bitton et al., 2023)	Text	Image	Text	×
MIA-Bench (Qian et al., 2025)	Text	Image	Text	×
MM-IFEval (Ding et al., 2025)	Text	Image	Text	×
Speech-IFEval (Lu et al., 2025)	Text	Audio	Text	×
IFEval-Audio (Gao et al., 2025)	Text	Audio	Text	×
Omni-Instruct (Ours)	**Text/Image/Audio**	**Text/Image/Audio**	**Text/Image/Audio**	○

constraints, including content, format, length, style, and structural requirements. They have established instruction following as a distinct evaluation target from general response quality, requiring the response not just to be fluent and useful but also to satisfy the explicitly given user constraints.

Extending to multimodal contexts, LLaVA-Bench (Liu et al., 2023) and VisIT-Bench (Bitton et al., 2023) evaluate instruction following for image-conditioned visual assistants, while MM-Vet (Yu et al., 2024) evaluates integrated vision-language capabilities with open-ended responses. More directly, MIA-Bench (Qian et al., 2025) evaluates complex instruction adherence over image-prompt pairs, and MM-IFEval (Ding et al., 2025) evaluates compose-level output constraints and perception-level constraints tied to image inputs. These benchmarks show that multimodal contexts affect instruction compliance, but they typically use multimodal inputs as source contexts while keeping the instruction primarily textual and the response channel predetermined.

Our benchmark extends instruction following evaluation to *any-to-any interactions*, as shown in Tab. 1. Following an instruction is not limited to satisfying constraints within a predetermined output channel; it also requires understanding the modality in which the instruction is given and selecting the output modality requested by the user. For this, we formulate instruction-modality understanding and output-modality routing as core capabilities of omnimodal instruction following, and design our evaluation focusing on the cases that prior instruction following benchmarks largely leave unmeasured.

Any-to-Any Models. Recent multimodal LLMs have evolved from vision-language models toward omnimodal models that can perceive and interact across text, image and audio. A first line of models (Xu et al., 2025; OpenBMB, 2025; Li et al., 2024; Fu et al., 2024; Liu et al., 2025; Fang et al., 2025) emphasizes omnimodal perception and interaction, typically producing text or speech responses from omnimodal inputs rather than arbitrary visual outputs. Meanwhile, any-to-any models (Tang et al., 2023; Wu et al., 2024; Lu et al., 2024; Zhan et al., 2024; Wang et al., 2025; Inclusion AI et al., 2025; Li et al., 2025b; Team, 2026; Cheng et al., 2026) aim to produce non-text or interleaved multimodal outputs, including images, speech or audio, and mixed responses.

Recent evaluation efforts such as OmniBench (Li et al., 2025a), OmniEval (Zhang et al., 2025), and Omni×R (Chen et al., 2025) evaluate integrated understanding and reasoning over visual, auditory, and textual inputs, typically through text or QA-style outputs. More recently, benchmarks such as MixEval-X (Ni et al., 2025), TaskAnything (Pu et al., 2025), and UniM (Li et al., 2026) broaden evaluation toward any-to-any or all-modality tasks, assessing real-world task mixtures, interleaved multimodal generation, output quality, preference, or judge behavior. In contrast to prior work, we formulate any-to-any interaction as a fine-grained instruction following problem, and extend this formulation to omnimodal instructions to evaluate whether models satisfy both the requested modality and task-specific constraints.

3 Any-to-Any Instruction Following

For any-to-any interaction, a user specifies not only what to do but also in which modality the result should be delivered, and the request itself may arrive in various forms. Following such a request requires the model to distinguish the instruction from the context, deliver its response in the requested output modality, and satisfy the detailed requirements of the request. We first define the components of any-to-any instruction following (Sec. 3.1), and describe its criteria and metrics (Sec. 3.2).

3.1 Definition of Context, Instruction, and Output

We distinguish two role-specific components of the model input: instruction and context. The instruction I specifies the requested operation, output form, and constraints, while the context C provides the evidence, source content, or editable artifact needed to perform the task specified in the instruction. This distinction is based on role, not modality; *e.g.*, an image may provide visual evidence or carry a task instruction, while an audio clip can also serve as source speech or a spoken directive. Given these inputs, a model f produces a response output $\hat{O} = f(I, C)$.

We denote the instruction, context, and requested output modalities by m_I, m_C, and m_O, respectively. We use $m_I, m_O \in \mathcal{M} = \{\text{text}, \text{image}, \text{audio}\}$ and $m_C \in \mathcal{M} \cup \{\varnothing\}$, where $\varnothing$ denotes the absence of context. Under the any-to-any setting, these axes can be arbitrarily combined; *i.e.*, (m_C, m_O) specifies a task family defined by modality, while m_I specifies how the instruction is conveyed.

3.2 Instruction Following Criteria and Metrics

Given an instruction and an optional context, any-to-any instruction following requires producing a response that performs the requested operation and satisfies its constraints in the requested modality. We decompose this process into three sequential elements: understanding the instruction, routing to the requested output modality, and complying with its detailed requirements.

Instruction Understanding. The model first needs to clearly separate the instruction from the context. That is, the instruction should be treated as a directive to execute, rather than as further material to describe, transcribe, or edit. This is often not trivial, since m_I and m_C vary independently and the same modality can play either role; *e.g.*, an image may be a picture to edit or a card stating how to edit, and a speech clip may be the utterance to transcribe or the spoken request itself. Unless the context and instruction are clearly distinguished, a model may perform an unintended task, *e.g.*, captioning an instruction image or transcribing a spoken directive. Therefore, we vary m_I to evaluate the robustness of omnimodal instruction following in any-to-any models.

Output-modality Routing. The model then needs to infer the output modality requested by the instruction and deliver its response in that modality. We call this *output-modality routing*. Let $m_{\hat{O}}$ denote the output modality of the generated response $\hat{O}$; we define the success of routing as an exact match between $m_{\hat{O}}$ and m_O:

$$r = \mathbb{I}\left[\, m_{\hat{O}} = m_O \,\right], \tag{1}$$

where $\mathbb{I}$ is the indicator function. We consider the modality match as a necessary condition for a correct answer. That is, if the answer is in a wrong modality, it is considered incorrect even if its content conveys correct information.

Conditional Compliance. Routing is necessary but not sufficient. Given a correctly routed response for an instance (denoted by $r = 1$), the model should perform the requested operation and satisfy its instruction-specific requirements, including context grounding, semantic and structural constraints, and modality-specific constraints such as textual format, visual layout, or speech duration and style. We denote these K requirements by $\mathcal{R} = \{\rho_j\}_{j=1}^{K}$, where each ρ_j specifies an atomic requirement. Let a binary rubric score $b_j \in \{0, 1\}$ indicate whether the model output $\hat{O}$ satisfies ρ_j. We define *conditional compliance* q as the fraction of satisfied requirements:

$$q = \mathbb{I}[r = 1] \cdot \frac{1}{K} \sum_{j=1}^{K} b_j. \tag{2}$$

Metrics. Over N test instances, we report the Routing Accuracy (RA), Conditional Compliance (CC), and End-to-end (E2E) scores:

$$\text{RA} = \frac{\sum_i r_i}{N}, \qquad \text{CC} = \frac{\sum_i r_i q_i}{\sum_i r_i}, \qquad \text{E2E} = \frac{\sum_i r_i q_i}{N}. \tag{3}$$

RA measures how often the requested output modality is correctly delivered, while CC measures mean instruction compliance among correctly routed responses. E2E evaluates instruction following over all requests, assigning zero credit to routing failures. Together, these metrics distinguish failures to deliver the requested output form from failures to satisfy requirements within that form, while capturing their combined effect.

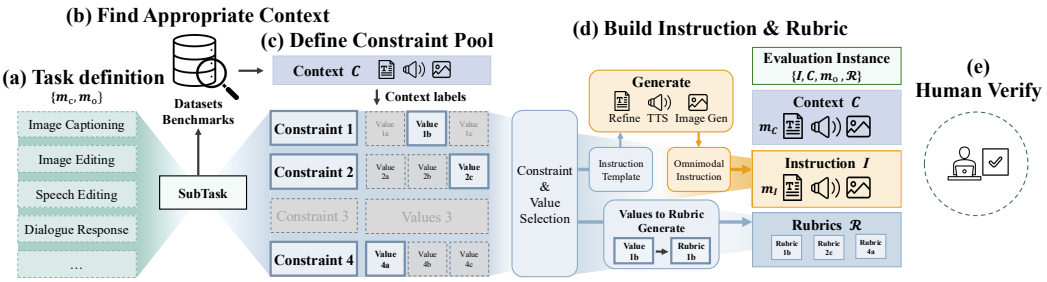


Figure 3: **Overview of our benchmark construction pipeline.**

4 The Omni-Instruct Benchmark

We present Omni-Instruct, a novel benchmark for any-to-any instruction following that instantiates the formulation in Sec. 3. As summarized in Fig. 3, we (a) define task families, (b) curate source contexts, (c) specify constraint pools, (d) compose instructions and rubrics from sampled constraints, and (e) verify each instance with human review. Each instance $\boldsymbol{x} = (I, C, m_O, \mathcal{R})$ consists of an instruction I delivered in modality m_I, an optional context C, a requested output modality m_O, and a set of binary rubrics $\mathcal{R}$. By construction, each rubric score b_j corresponds to a requirement $\rho_j \in \mathcal{R}$ stated in the instruction or required by grounding in the context.

Task Definition. Following Sec. 3.1, we exhaustively cover the cross product of four input context conditions (m_C) and three output modalities (m_O), yielding 12 any-to-any scenarios, illustrated in Fig. 3(a). Each task family, specified by a pair of modalities, covers various kinds of subtasks. For instance, the (image, text) family contains visual scene summarization, character recognition, and document or chart understanding, while the (audio, audio) family covers speech editing and spoken dialogue responses. In total, we cover 25 such subtasks, whose full list is provided in Tab. A1 in Sec. B.

Context Sourcing. We map each subtask to existing datasets and benchmarks that address a similar task and draw context assets from them, as illustrated in Fig. 3(b). Specifically, we incorporate dialogue and table corpora for text contexts (Gliwa et al., 2019; Nan et al., 2022; Parikh et al., 2020); natural images (Lin et al., 2014; Rösch et al., 2024), text-rich charts and documents (Masry et al., 2022; Singh et al., 2019; Mathew et al., 2021), and image-editing pairs (Zhang et al., 2023) for image contexts; and read speech (Panayotov et al., 2015) together with speech labeled for intent (Gerz et al., 2021) or emotion (Zhou et al., 2021) for audio contexts. The selection criteria for each source are twofold: 1) the asset must be meaningful for the target operation, and 2) it must carry labels, such as captions, transcripts, chart values, edit targets, or speaker and emotion labels, that can ground both the instruction and its judgment. We retain these labels as hidden reference metadata for the judge.

Constraint Pools. For each subtask, we define the constraints an instruction may impose, and each constraint specifies the type of values it admits, as seen in Fig. 3(c). These value pools are context-dependent; given a concrete context, each constraint is instantiated with candidate values grounded in the context. For image editing, for instance, the constraints specify which object to modify and how, which other content should be preserved, and whether the camera viewpoint should remain consistent. Across output modalities, text outputs are constrained in format, length, and sentence or bullet count; image outputs in objects, counts, colors, rendered text, and layout; and audio outputs in transcript content, duration, speaker gender, and emotion.

Instruction and Rubric Generation. For each instance, we sample a context and a corresponding set of constraint-value pairs, composed as an instruction and rubrics in Fig. 3(d). From this, an instruction template is first generated for each subtask. An LLM (OpenAI, 2025) is then instructed to naturally rephrase the template without adding or removing constraints. Finally, the instruction is delivered in three modalities: kept as text, rendered as an image by Gemini 3.1 Flash Image (Google DeepMind, 2026) in note, card, or message style, or synthesized as speech using GPT-4o-mini TTS (Hurst et al., 2024), mixed with mild real-world background noise (Snyder et al., 2015).

Human Verification. Every instance goes through a two-stage review. An omnimodal judge first screens it for task coherence, instruction-context and instruction-rubric alignment, clarity, rubric judgeability, source suitability, and answer leakage. Flagged instances, together with a sample of

Table 2: **Instruction following performance across modality axes.** Routing Accuracy (RA), Conditional Compliance (CC), and End-to-end (E2E) scores are on a 0–100 scale. Per column, **bold** and underlining mark the best and second-best distinct scores across the six models.

	Overall			Context modality											
	All			None			Text			Image			Audio		
Model	RA	CC	E2E	RA	CC	E2E	RA	CC	E2E	RA	CC	E2E	RA	CC	E2E
AnyGPT	44.33	42.04	18.64	52.67	45.28	23.85	**49.56**	50.29	24.92	38.22	19.24	7.35	36.89	**49.95**	**18.43**
AR-Omni	28.06	13.94	3.91	31.56	13.81	4.36	24.44	13.18	3.22	27.56	8.92	2.46	28.67	19.54	5.60
MIO	37.33	35.53	13.26	40.22	39.90	16.05	33.78	43.83	14.80	40.67	23.51	9.56	34.67	36.46	12.64
ModaVerse	**52.44**	27.39	14.37	**55.33**	36.97	20.46	49.33	44.36	21.89	**50.89**	15.47	7.87	**54.22**	13.37	7.25
NExT-GPT	33.50	21.20	7.10	34.22	24.34	8.33	33.78	31.58	10.67	33.78	16.66	5.63	32.22	11.74	3.78
HCX-Omni	43.06	**53.19**	**22.90**	50.89	**51.57**	**26.24**	43.56	**62.07**	**27.03**	38.22	**56.78**	**21.70**	39.56	42.05	16.63

	Instruction modality									Requested output modality								
	Text			Image			Audio			Text			Image			Audio		
Model	RA	CC	E2E	RA	CC	E2E	RA	CC	E2E	RA	CC	E2E	RA	CC	E2E	RA	CC	E2E
AnyGPT	56.50	43.46	24.56	34.17	30.97	10.58	**42.33**	**49.07**	**20.77**	33.50	39.17	13.12	**45.17**	37.20	**16.80**	**54.33**	47.82	**25.98**
AR-Omni	28.67	15.92	4.56	27.67	11.26	3.11	27.83	14.56	4.05	72.17	13.25	9.56	3.83	20.66	0.79	8.17	16.82	1.37
MIO	44.17	49.91	22.05	34.50	31.21	10.77	33.33	20.94	6.98	88.00	35.00	30.80	9.83	29.77	2.93	14.17	42.82	6.07
ModaVerse	**88.83**	36.46	32.39	34.17	12.11	4.14	34.33	19.15	6.57	**99.00**	23.10	22.87	30.00	**41.76**	12.53	28.33	27.19	7.70
NExT-GPT	34.17	36.32	12.41	33.17	13.15	4.36	33.17	13.68	4.54	98.83	21.06	20.81	0.33	20.83	0.07	1.33	31.70	0.42
HCX-Omni	58.17	**67.84**	**39.46**	**42.83**	**39.86**	**17.07**	28.17	43.22	12.17	62.17	**62.10**	**38.60**	40.17	36.32	14.59	26.83	**57.83**	15.52

passed ones, are then rewritten by human annotators. For image and audio instructions, annotators additionally verify that the carrier is legible and conveys exactly the refined instruction, without hints or added constraints. The resulting benchmark contains 1,800 instances. See Sec. A for details.

Evaluation Protocol. We evaluate each instance with the metrics defined in Sec. 3.2. We first check whether a valid, decodable artifact of the requested type is returned. A malformed or truncated media span is judged as absent. Routing succeeds only when every requested modality is delivered and no unrequested media is emitted. Note that we permit accompanying text alongside a requested image or audio as an exception, since text serves as the default output for the majority of models. For each correctly routed response ($r = 1$), an omnimodal judge, Gemini 3.5 Flash, takes the instruction, context, hidden reference metadata, response, and rubric, and assigns a binary rubric score $b_j \in \{0, 1\}$ to each rubric item $\rho_j \in \mathcal{R}$. These labels are aggregated into the compliance score q according to Eq. (2). On a held-out calibration set, comparison with the human judge majority yields Cohen's $\kappa = 0.766$ for rubric labels and $\mathrm{ICC}(A, 1) = 0.849$ for instance-level scores. Secs. D and E provide further details on human-alignment results and ablations on judge models.

5 Comparative Study on Omni-Instruct

We compare six models that both understand and generate text, images, and audio: AnyGPT (Zhan et al., 2024), AR-Omni (Cheng et al., 2026), MIO (Wang et al., 2025), ModaVerse (Wang et al., 2024), NExT-GPT (Wu et al., 2024), and HCX-Omni (Team, 2026). We examine output-modality routing (Sec. 5.2) and conditional compliance (Sec. 5.3) separately, then analyze how the instruction modality affects both (Sec. 5.4), and probe the source of this gap (Sec. 5.5).

5.1 Overall Comparison

Tab. 2 summarizes the overall performance of the competing models along the instruction, context, and requested-output axes. Overall, the current any-to-any models struggle with instruction following. Even the best model, HCX-Omni, reaches an overall E2E score of only 22.90, and the remaining models score between 3.91 and 18.64. Fig. 1 further breaks this down by context–output cell and instruction modality. For most models, text instructions form the outer envelope, while image and speech instructions shrink it substantially. AnyGPT is an exception; under speech instructions, its profile shifts toward audio-output cells, which we will analyze in Sec. 5.2.

Another conclusion we draw from this overall comparison is that routing and compliance are distinct capabilities, given that the ranking of models substantially differs between the two stages. ModaVerse achieves the highest routing accuracy (52.44) but ranks only fourth in conditional compliance (27.39).

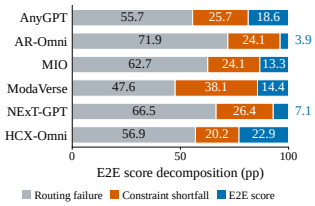


Figure 4: **End-to-end score decomposition.** Bars show routing failure, constraint shortfall after successful routing, and achieved E2E score, in percentage points.

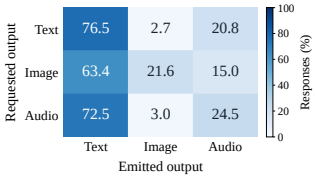


Figure 5: **Text fallback.** Models frequently collapse non-text output requests into text or no-media responses.

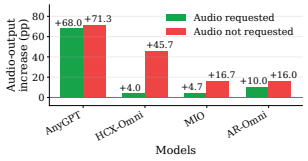


Figure 6: **Audio bias.** Bars show the increase in audio-output emissions when audio is present in the input, compared with matched non-audio-input cases.

Conversely, HCX-Omni ranks third in RA (43.06) while ranking first in CC (53.19). Selecting the requested output modality and fulfilling the request are thus distinct capabilities that a single aggregate score would conflate. We therefore analyze routing and compliance separately in Sec. 5.2-5.3.

5.2 Output-Modality Routing

Routing failure is the dominant source of score loss. Most models deliver the requested output modality for fewer than half of the requests (Tab. 2). Decomposing each model's E2E deficit (Fig. 4), we find that routing accounts for 55–75% of the loss. These failures concentrate on non-text outputs. For four of the models, RA exceeds 70 on text-output requests but is at most 30 on image- and audio-output requests.

Text fallback: requests answered in the wrong modality. Models fall back to text in 63.4% of image-output requests and 72.5% of audio-output requests (Fig. 5). These responses often engage with the request while ignoring the output form it specifies. Asked to replace the sheep in an attached image with a cow, AnyGPT replies "the sheep herd has been replaced by a cow", without producing any edited image. With an image instruction, ModaVerse returns a description of the instruction image itself. In each case, the model violates an explicit requirement of the instruction—the requested output modality—so routing failures are themselves failures of instruction following. These responses resemble habits of text-centric language models: a canned capability disclaimer, a caption of the input image, or a verbal claim of having acted. This may show that behaviors inherited from the language backbone can override the output form requested by the instruction. We analyze how these behaviors arise, and how they depend on the instruction modality, in Sec. 5.4.

Unrequested audio: an input-dependent bias. Unlike text fallback, which pulls outputs toward the backbone's default modality regardless of the input, audio output errors are triggered by the input itself. On instances with neither an audio context nor an audio target, delivering the instruction as speech sharply increases audio outputs, even when audio is not requested (Fig. 6). For HCX-Omni, for example, the audio-output rate rises from 9% to 54%, and AnyGPT exhibits the strongest bias, emitting audio for most such requests under speech instructions. In the affected models, audio input thus biases the output toward audio, even when the instruction explicitly requests another modality.

5.3 Conditional Compliance

Correct routing does not ensure compliance. Even among correctly routed responses, models satisfy only a fraction of the requested conditions. Conditional Compliance ranges from 13.9 to 53.2, and only HCX-Omni satisfies more than half of the conditions on average (Tab. 2). Unlike routing failures, compliance failures are not concentrated on non-text outputs. Once routed, HCX-Omni complies nearly as well on audio as on text (57.8 vs. 62.1), and ModaVerse complies better on images than on text (41.8 vs. 23.1). Instead, they concentrate on requests grounded in images. For most models, CC drops sharply with an image context compared with a text context (50.3 to 19.2 for AnyGPT), with HCX-Omni the only exception. Compliance also depends on the requested operation (Sec. F). HCX-Omni leads on summarizing or responding to a given input, whereas ModaVerse and AnyGPT lead on caption-to-image generation. To understand what these scores consist of, we next examine individual rubric items.

Rubrics enable fine-grained comparison but can reward misdirected responses. To examine CC in more detail, we categorize rubric items by the type of constraint they specify (Sec. A.2). This decomposition separates responses that routing alone cannot: in Fig. 10, all three models deliver the

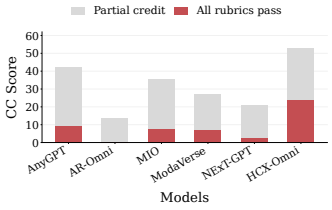


Figure 7: **Partial and strict CC score.** Strict scoring requires all rubrics for each instance to pass, which causes a dramatic decrease.

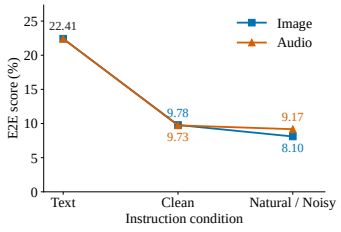


Figure 8: **Various instruction conditions.** Mean E2E scores across models on matched tasks with text, clean, and natural/noisy instructions.

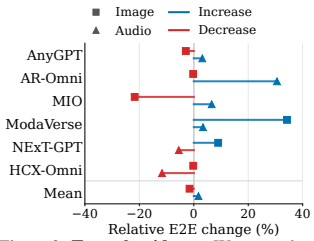


Figure 9: **Textual guidance.** We append an extra text cue that only localizes the instruction. Blue and red indicate a score increase and decrease after guidance.

requested image, yet satisfy different subsets of the rubrics. Across models, surface rubrics such as length are satisfied far more often than content rubrics (e.g., 84.3 vs. 34.0 for AnyGPT). Such rubrics can be met even without performing the operation; AR-Omni, for instance, earns partial credit for merely echoing the instruction. Partial credit thus supports fine-grained comparison among responses that attempt the task, but may overstate performance when the requested operation itself is not performed.

Strict scoring reveals how rarely requests are fulfilled. To remove such incidental credit, we additionally apply *strict* scoring, which credits a response only when every rubric item passes (Fig. 7). Scores drop for every model, but unevenly. HCX-Omni retains 45% of its partial CC, whereas AnyGPT and MIO, whose partial credit comes largely from length and preservation rubrics, retain little. Correct routing is thus far from sufficient: most correctly routed responses still miss at least one requested condition. The ranking of models is unchanged, but the gap widens, with HCX-Omni's lead over the next-best model growing from 1.3× to 2.5×. Strict scores are therefore important when comparing models whose partial credit comes mainly from surface rubrics, while partial scores remain useful for diagnosing which requirements fail. Full strict results are reported in Sec. F.

5.4 Understanding Omnimodal Instruction

The same request degrades when delivered through a non-text instruction. Having seen that failures arise at both routing and compliance, we now examine how both depend on the modality of the instruction. Every model obtains a lower E2E score with image and speech instructions than with text instructions (Tab. 2); for ModaVerse, E2E drops from 32.4 to 4.1 and 6.6, respectively. Breaking these scores down by context–output combination also reveals diverse degradation patterns (Fig. 1). To understand what underlies this drop, we next examine what models actually do when given a non-text instruction, by inspecting the observed behavior of each response.

Non-text instructions induce substitute behaviors even without context. To isolate the effect of the instruction carrier, we first examine instances without context, where the instruction is the only input and thus cannot be confused with task material. In this setting, non-text instructions elicit behaviors that are absent under text instructions. ModaVerse describes the instruction image in every image-instructed case instead of generating the requested artifact. MIO repeats the spoken instruction back in 66.7% of speech-instructed cases. AnyGPT often asks what it should do. The instruction thus appears to be processed as content to describe or transcribe rather than as a directive, a failure attributable to the carrier alone.

With context, substitution shifts toward the context. When a context is present, models often copy, answer, or describe the context instead of performing the requested operation. With a text context, image instructions more than double this context substitution relative to text instructions (12% to 28%), whereas speech instructions instead elicit echoes of the instruction itself in up to a third of responses. For example, asked through an image instruction to change the day in an attached utterance to "Wednesday", HCX-Omni simply reproduces the original "Monday". Since different context modalities involve different tasks, we read these contrasts as behavioral tendencies of each model rather than causal effects. Together, these behaviors suggest that, when an instruction arrives in a non-text modality, models often fail to distinguish it from the material it is meant to operate on: they treat the instruction itself as context, or act on the context in its place. We examine next where this failure arises.

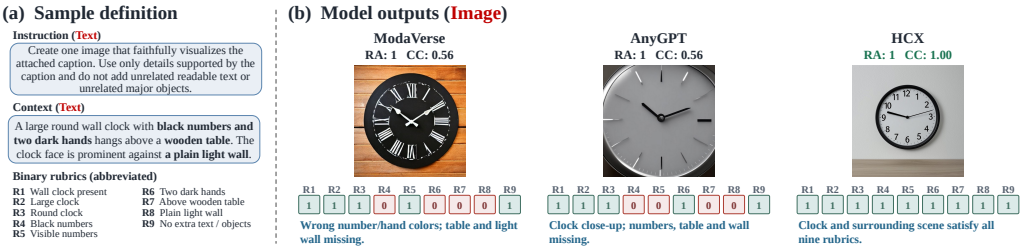


Figure 10: **Qualitative comparison.** All models successfully route to the requested image output (RA= 1), showing different levels of fine-grained conditional compliance across the nine rubrics.

5.5 Further Analysis with Additional Experiments

We consider three candidate explanations for this gap. Models may fail to *perceive* the instruction, fail to *locate* it among the inputs, or fail to *recognize and act on* it as a directive. We test each with additional experiments.

Simplifying the carrier leaves most of the gap intact. To test perception, we replace natural images with the same instruction text rendered on a white canvas, and noisy speech with its clean source (details in Sec. G.1). Mean E2E across models improves only slightly with the clean counterparts (Fig. 8), closing about 12% of the gap to text for images and 4% for speech. The gains are concentrated in a few models. HCX-Omni improves substantially with clean images (15.7 to 23.5), and AnyGPT modestly with clean speech, while the changes for the other models are not significant. This suggests that legibility helps only models that already recognize the instruction, whereas for the others the failure lies upstream of perception. Most of the gap thus persists even under the most legible carriers, so perceptual difficulty is not its primary source.

Indicating where the instruction is does not restore execution. We next test whether models simply fail to find the instruction among their inputs. To this end, we add short textual guidance that indicates where the instruction is, e.g., "Follow the instruction in the {image, audio}", without revealing its content. We then compare E2E scores with and without this guidance on the same image- and speech-instructed instances (Sec. G.2). As shown in Fig. 9, the mean change is negligible for both carriers, and no model-level effect survives multiple-comparison correction. Simply being told *where* the instruction is does not lead models to *act* on it.

Stating the requirements does not imply fulfilling them. Finally, to separate understanding the instruction from executing it, we ask the models, instead of performing the task, to describe in text what the instruction asks them to do, which output modality it requires, and which requirements the output must satisfy (details in Sec. G.3). A report counts as correct only if all three answers are correct. Averaged over models, the task, the output modality, and the requirements are correctly reported for 67%, 58%, and 44% of text instructions, but for only 11%, 14%, and 10% of image and speech instructions. Under non-text instructions, only HCX-Omni produces fully correct reports, for about a third of the requests. Yet correct reports seldom translate into execution: only 2.6% of these requests are strictly fulfilled when the original task is executed. Taken together, these results show that neither clearer carriers nor explicit pointers restore execution. The bottleneck instead lies in omnimodal instruction understanding: most models fail to grasp what a non-text instruction asks, and even correct understanding does not ensure the request is carried out. Probes of understanding therefore complement, but cannot replace, end-to-end evaluation of execution.

6 Conclusion

We introduce Omni-Instruct, an instruction following benchmark for any-to-any models with omnimodal instruction. We formulate any-to-any instruction following as delivering the requested output modality and satisfying task-specific requirements. Our experiments reveal that output-modality routing is a major bottleneck across the evaluated models, while conditional compliance remains limited even when routing succeeds. Moreover, conveying the same instructions through images or speech often substantially reduces end-to-end performance relative to text. By separating output-modality routing from conditional compliance, Omni-Instruct provides a diagnostic framework for identifying where any-to-any instruction following breaks down.

AI USE STATEMENT

Generative models were used as part of the benchmark pipeline: an LLM rewrote templated instructions into natural phrasing, and an image generation model and a TTS model generated the image and audio instructions, respectively (Sec. A). Gemini 3.5 Flash served as the judge model (Sec. A.3). Generated content such as omnimodal instructions, rubrics, and outputs was verified by human annotators, and the various judge models were validated against human majority labels (Secs. D and E). LLMs were also used to polish the writing of this paper.

REFERENCES

Yonatan Bitton, Hritik Bansal, Jack Hessel, Rulin Shao, Wanrong Zhu, Anas Awadalla, Joshua P Gardner, Rohan Taori, and Ludwig Schmidt. VisIT-bench: A dynamic benchmark for evaluating instruction-following vision-and-language models. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track (NeurIPS Datasets and Benchmarks)*, 2023.

Lichang Chen, Hexiang Hu, Mingda Zhang, Yiwen Chen, Zifeng Wang, Yandong Li, Pranav Shyam, Tianyi Zhou, Heng Huang, Ming-Hsuan Yang, and Boqing Gong. OmnixR: Evaluating omni-modality language models on reasoning across modalities. In *International Conference on Learning Representations (ICLR)*, 2025.

Dongjie Cheng, Ruifeng Yuan, Yongqi Li, Runyang You, Wenjie Wang, Liqiang Nie, Lei Zhang, and Wenjie Li. AR-Omni: A unified autoregressive model for any-to-any generation. *arXiv:2601.17761*, 2026.

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

Qingkai Fang, Shoutao Guo, Yan Zhou, Zhengrui Ma, Shaolei Zhang, and Yang Feng. LLaMA-Omni: Seamless speech interaction with large language models. In *International Conference on Learning Representations (ICLR)*, 2025.

Chaoyou Fu, Haojia Lin, Zuwei Long, Yunhang Shen, Meng Zhao, Yifan Zhang, Xiong Wang, Di Yin, Long Ma, Xiawu Zheng, Ran He, Rongrong Ji, Yunsheng Wu, Caifeng Shan, and Xing Sun. VITA: Towards open-source interactive omni multimodal llm. *arXiv:2408.05211*, 2024.

Yiming Gao, Bin Wang, Chengwei Wei, Shuo Sun, and AiTi Aw. IFEval-audio: Benchmarking instruction-following capability in audio-based large language models. In Kentaro Inui, Sakriani Sakti, Haofen Wang, Derek F. Wong, Pushpak Bhattacharyya, Biplab Banerjee, Asif Ekbal, Tanmoy Chakraborty, and Dhirendra Pratap Singh (eds.), *International Joint Conference on Natural Language Processing (IJCNLP)*, 2025.

Daniela Gerz, Pei-Hao Su, Razvan Kusztos, Avishek Mondal, Michał Lis, Eshan Singhal, Nikola Mrkšić, Tsung-Hsien Wen, and Ivan Vulić. Multilingual and cross-lingual intent detection from spoken data. In *Annual Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. SAMSum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Workshop on New Frontiers in Summarization*, 2019.

Google DeepMind. Gemini 3.1 Flash Image Model Card, 2026. URL `https://storage.googleapis.com/deepmind-media/Model-Cards/Gemini-3-1-Flash-Image-Model-Card.pdf`.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. GPT-4o system card. *arXiv:2410.21276*, 2024.

Inclusion AI, Biao Gong, Cheng Zou, Chuanyang Zheng, Chunluan Zhou, Canxiang Yan, Chunxiang Jin, Chunjie Shen, Dandan Zheng, Fudong Wang, et al. Ming-omni: A unified multimodal model for perception and generation. *arXiv:2506.09344*, 2025.

Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjun Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. FollowBench: A multi-level fine-grained constraints following benchmark for large language models. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

Yadong Li, Haoze Sun, Mingan Lin, Tianpeng Li, Guosheng Dong, Tao Zhang, Bowen Ding, Wei Song, Zhenglin Cheng, Yuqi Huo, Song Chen, Xu Li, Da Pan, Shusen Zhang, Xin Wu, Zheng Liang, Jun Liu, Tao Zhang, Keer Lu, Yaqi Zhao, Yanjun Shen, Fan Yang, Kaicheng Yu, Tao Lin, Jianhua Xu, Zenan Zhou, and Weipeng Chen. Baichuan-Omni technical report. *arXiv:2410.08565*, 2024.

Yanlin Li, Minghui Guo, Kaiwen Zhang, Shize Zhang, Yiran Zhao, Haodong Li, Congyue Zhou, Weijie Zheng, Yushen Yan, Shengqiong Wu, Wei Ji, Lei Cui, Furu Wei, Hao Fei, Mong-Li Lee, and Wynne Hsu. UniM: A unified any-to-any interleaved multimodal benchmark. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Yizhi Li, Yinghao Ma, Ge Zhang, Ruibin Yuan, King Zhu, Hangyu Guo, Yiming Liang, Jiaheng Liu, Zekun Moore Wang, Jian Yang, Siwei Wu, Xingwei Qu, Jinjie Shi, Xinyue Zhang, Zhenzhu Yang, Yidan WEN, Yanghai Wang, Shihao Li, Zhaoxiang Zhang, Ruibo Liu, Emmanouil Benetos, Wenhao Huang, and Chenghua Lin. OmniBench: Towards the future of universal omni-language models. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track (NeurIPS Datasets and Benchmarks)*, 2025a.

Yunxin Li, Xinyu Chen, Shenyuan Jiang, Haoyuan Shi, Zhenyu Liu, Xuanyu Zhang, Nanhao Deng, Zhenran Xu, Yicheng Ma, Meishan Zhang, et al. Uni-MoE-2.0-omni: Scaling language-centric omnimodal large model with advanced moe, training and data. *arXiv:2511.12609*, 2025b.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, 2014.

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

Zuyan Liu, Yuhao Dong, Jiahui Wang, Ziwei Liu, Winston Hu, Jiwen Lu, and Yongming Rao. Ola: Pushing the frontiers of omni-modal language model. *arXiv:2502.04328*, 2025.

Jiasen Lu, Christopher Clark, Sangho Lee, Zichen Zhang, Savya Khosla, Ryan Marten, Derek Hoiem, and Aniruddha Kembhavi. Unified-IO 2: Scaling autoregressive multimodal models with vision, language, audio, and action. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Ke-Han Lu, Chun-Yi Kuan, and Hung yi Lee. Speech-IFEval: Evaluating instruction-following and quantifying catastrophic forgetting in speech-aware language models. In *Interspeech*, pp. 2078–2082, 2025.

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

Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. DocVQA: A dataset for VQA on document images. In *Proceedings of Winter Conference on Applications of Computer Vision (WACV)*, 2021.

Linyong Nan, Chiachun Hsieh, Ziming Mao, Xi Victoria Lin, Neha Verma, Rui Zhang, Wojciech Kryściński, Hailey Schoelkopf, Riley Kong, Xiangru Tang, Mutethia Mutuma, Ben Rosand, Isabel Trindade, Renusree Bandaru, Jacob Cunningham, Caiming Xiong, and Dragomir Radev. FeTaQA: Free-form table question answering. *Transactions of the Association for Computational Linguistics (TACL)*, 10:35–49, 2022.

Jinjie Ni, Yifan Song, Deepanway Ghosal, Bo Li, David Junhao Zhang, Xiang Yue, Fuzhao Xue, Yuntian Deng, Zian Zheng, Kaichen Zhang, Mahir Shah, Kabir Jain, Yang You, and Michael Shieh. MixEval-X: Any-to-any evaluations from real-world data mixture. In *International Conference on Learning Representations (ICLR)*, 2025.

OpenAI. GPT-5 System Card, 2025. URL https://openai.com/index/gpt-5-system-card/.

OpenBMB. MiniCPM-o 2.6: A GPT-4o level MLLM for vision, speech, and multimodal live streaming on your phone, 2025. URL https://openbmb.notion.site/MiniCPM-o-2-6-A-GPT-4o-Level-MLLM-for-Vision-Speech-and-Multimodal-Live-Streaming-on-Your-Phone-185ede1b7a558042b5d5e45e6b237da9.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. LibriSpeech: An ASR corpus based on public domain audio books. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

Ankur Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. ToTTo: A controlled table-to-text generation dataset. In *Annual Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

Shu Pu, Yaochen Wang, Dongping Chen, Yuhang Chen, Guohao Wang, Qi Qin, Zhongyi Zhang, Zhiyuan Zhang, Zetong Zhou, Shuang Gong, Yi Gui, Yao Wan, and Philip S. Yu. Judge anything: MLLM as a judge across any modality. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2025.

Yusu Qian, Hanrong Ye, Jean-Philippe Fauconnier, Peter Grasch, Yinfei Yang, and Zhe Gan. MIA-bench: Towards better instruction following evaluation of multimodal LLMs. In *International Conference on Learning Representations (ICLR)*, 2025.

Yiwei Qin, Kaiqiang Song, Yebowen Hu, Wenlin Yao, Sangwoo Cho, Xiaoyang Wang, Xuansheng Wu, Fei Liu, Pengfei Liu, and Dong Yu. InFoBench: Evaluating instruction following ability in large language models. In *Findings of the Association for Computational Linguistics (ACL Findings)*, 2024.

Philipp J. Rösch, Norbert Oswald, Michaela Geierhos, and Jindřich Libovický. Enhancing conceptual understanding in multimodal contrastive learning through hard negative samples. In *ACL Workshop on Advances in Language and Vision Research (ALVR)*, 2024.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA models that can read. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

David Snyder, Guoguo Chen, and Daniel Povey. Musan: A music, speech, and noise corpus. *arXiv:1510.08484*, 2015.

Zineng Tang, Ziyi Yang, Chenguang Zhu, Michael Zeng, and Mohit Bansal. Any-to-any generation via composable diffusion. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

NAVER Cloud HyperCLOVA X Team. HyperCLOVA X 8B Omni. *arXiv:2601.01792*, 2026.

Xinyu Wang, Bohan Zhuang, and Qi Wu. ModaVerse: Efficiently transforming modalities with llms. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Zekun Moore Wang, King Zhu, Chunpu Xu, Wangchunshu Zhou, Jiaheng Liu, Yibo Zhang, Jessie Wang, Ning Shi, Siyu Li, Yizhi Li, Haoran Que, Zhaoxiang Zhang, Yuanxing Zhang, Ge Zhang, Ke Xu, Jie Fu, and Wenhao Huang. MIO: A foundation model on multimodal tokens. In *Annual Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2025.

Bosi Wen, Pei Ke, Xiaotao Gu, Lindong Wu, Hao Huang, Jinfeng Zhou, Wenchuang Li, Binxin Hu, Wendy Gao, Jiaxing Xu, Yiming Liu, Jie Tang, Hongning Wang, and Minlie Huang. Benchmarking complex instruction-following with multiple constraints composition. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track (NeurIPS Datasets and Benchmarks)*, 2024.

Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. NExT-GPT: Any-to-any multimodal LLM. In *International Conference on Machine Learning (ICML)*, 2024.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. Qwen2.5-omni technical report. *arXiv:2503.20215*, 2025.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. MM-Vet: Evaluating large multimodal models for integrated capabilities. In *International Conference on Machine Learning (ICML)*, 2024.

Jun Zhan, Junqi Dai, Jiasheng Ye, Yunhua Zhou, Dong Zhang, Zhigeng Liu, Xin Zhang, Ruibin Yuan, Ge Zhang, Linyang Li, Hang Yan, Jie Fu, Tao Gui, Tianxiang Sun, Yu-Gang Jiang, and Xipeng Qiu. AnyGPT: Unified multimodal LLM with discrete sequence modeling. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. MagicBrush: A manually annotated dataset for instruction-guided image editing. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track (NeurIPS Datasets and Benchmarks)*, 2023.

Yiman Zhang, Ziheng Luo, Qiangyu Yan, Wei He, Borui Jiang, Xinghao Chen, and Kai Han. OmniEval: A benchmark for evaluating omni-modal models with visual, auditory, and textual inputs. *arXiv:2506.20960*, 2025.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv:2311.07911*, 2023.

Kun Zhou, Berrak Sisman, Rui Liu, and Haizhou Li. Seen and unseen emotional style transfer for voice conversion with a new emotional speech dataset. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.

APPENDIX

This appendix documents benchmark construction, evaluation and judge validation, additional diagnostic analyses, and qualitative examples for the six models in the main evaluation. The qualitative examples connect the reported scores to observed responses.

A DETAILS FOR BENCHMARK CONSTRUCTION

The benchmark contains 600 base requests, each with a context when applicable, a canonical instruction, hidden reference metadata, and a binary rubric. Each request is evaluated under text, image, and audio instruction carriers. The task distribution appears in Sec. B.

A.1 CONTEXT ASSET SOURCES

We use existing datasets as context assets and hidden grounding metadata; the requested operations and constraints are authored for this benchmark. Context-free requests draw on scenario and constraint pools. Context-bearing requests use captions, transcripts, chart values, OCR/field annotations, and speaker or emotion labels to support binary judging. Audio contexts are restricted to speech. The source corpora are listed in Sec. 4.

Where possible, task families reuse semantic sources across context or output modalities. C11 uses TTS-derived spoken descriptions from sources also used in C05, while C07 and C09 reuse visual contexts. C05 and C11 are separate task collections: their final context wording and instructions are not identical in every case. Instruction-carrier comparisons instead match the same base request within a context–output cell.

A.2 INSTRUCTION AND RUBRIC DESIGN

The public instruction specifies the operation, output modality, and explicit constraints. Binary content atoms assess these requirements and, when a context is present, the required grounding in that context using hidden reference metadata. Text atoms cover content, preservation, format, length, and lexical constraints; image atoms cover objects, counts, attributes, relations, visible text, and exclusions; audio atoms cover speech content, duration, speaker attributes, emotion, and preservation. Requirements are separated into individually assessable atoms where possible. Constraints that cannot be judged from the output and reference metadata are removed. The analysis categories and counts appear in Sec. F.4.

A.3 TWO-STAGE QUALITY REVIEW

All 600 base items undergo MLLM-assisted screening after drafting and task-specific revision. Gemini 3.5 Flash checks task coherence, instruction clarity, judgeability, source suitability, and leakage, along with alignment between instruction and context, instruction and rubric, and rubric and context. It returns pass, minor-fix, major-fix, or reject, with suggested instruction, rubric, and metadata edits when needed.

Annotators review items, choosing to keep the original, use the model suggestion, or supply a custom revision. Interventions address hidden-answer leakage, overly restrictive rubrics, and mismatches between context assets and judging metadata. This human triage follows automatic screening; it is distinct from the separate human labeling study used to assess the judge in Sec. D.

A.4 INSTRUCTION-MODALITY CARRIER GENERATION

Carrier variants are derived after the two-stage review. For non-text carriers, the canonical instruction is conveyed through media rather than supplied as public text. The context, output target, and judging rubric remain fixed within each base request. Localization cues are added only in Sec. G.2.

Audio carriers. GPT-4o-mini TTS synthesizes the instructions, with seven audio-friendly wording adaptations across the full base set. Gemini 3.1 Flash Lite audits all 600 clips against their expected instruction text. Flagged clips undergo literal-reading retries or wording repairs and another audit,

with human checks of flagged and sampled clips. Noisy variants add MUSAN (Snyder et al., 2015) non-speech, non-music backgrounds to the repaired clean clips: 525 use free-sound sources at 24–30 dB SNR and 75 use sound-bible sources at 26–32 dB SNR.

Image carriers. Clean carriers render the instruction as left-aligned text on a 1280×900 white canvas. Natural carriers use Gemini 3.1 Flash Image to place instructions on scenes or interfaces such as notes, message windows, and work orders (Fig. A2). Generated carriers undergo OCR/semantic review, regeneration of failing items, human review of borderline cases, and a final OCR audit with Gemini 3.1 Flash Lite followed by targeted repairs. The diagnostic comparison changes these complete carrier variants; its matching and encoding details appear in Sec. G.1.

B Benchmark Taxonomy and Statistics

B.1 Task Taxonomy

The context–output axis crosses four context modalities (none, text, image, audio) with three requested output modalities (text, image, audio). Each of the 12 cells contains 50 base requests. Tab. A1 lists the 25 subtask families and their counts; C01–C03 each count as one no-context generation family. C07 has two families: visual scene summarization and visual text/chart/document summarization, with OCR-style reporting and chart/document understanding included in the latter. The table retains each cell's original family classification; shared source material across cells does not imply equal family counts.

B.2 Base and Carrier Counts

The 600 bases comprise 150 requests per context modality and 200 per output modality. Each base has five carrier variants, giving 600 instances per carrier and 3,000 in total. Main evaluation uses text-direct, image-natural, and audio-noisy (1,800 instances per model); image-clean and audio-clean provide the diagnostic counterparts. Switching the instruction carrier preserves the base request's context, target output, and rubric. The clean-carrier and guidance comparisons are described in Secs. G.1 and G.2.

C Evaluation and Model Implementation Details

C.1 Model Implementations

We evaluate AnyGPT, AR-Omni, MIO, ModaVerse, NExT-GPT, and HCX-Omni. Tab. A2 summarizes how each model's emitted representation is converted into a text, image, or audio artifact.

C.2 Artifact Validation and Binary Judging

Routing is evaluated through a deterministic artifact-delivery check. A requested image or audio artifact must be present and decodable; a text claim of completion does not establish media delivery (Fig. A1(a,b)). Routing requires the requested output modality and no unrequested image or audio; accompanying text is allowed for image or audio requests. Semantic compliance is assessed separately, so a decodable artifact can pass routing while failing the content requirements.

For each routed response, the judge receives the instruction, context, hidden references, actual output, and binary rubric in separate fields. Each content atom is credited only when clearly satisfied by the relevant output. Missing, empty, unreadable, or unlistenable material fails the affected requirements. The output contains one Boolean decision and a comment per atom:

```
{"rubric_results": [{"index": 1, "pass": true,
  "comment": "..."}], "notes": "..."}
```

Deterministic routing/default atoms are excluded from the content score.

Table A1: Context–output taxonomy and subtask distribution. The 12 cells contain 50 base requests each (600 total), spanning 25 subtask families. C01–C03 each count as one generation family; the remaining cells use the listed subtask counts. Each base request is delivered through text, image, and audio instructions in the main evaluation.

Cell	Subtask family / artifact coverage	Count
C01 (None→Text)	Controlled text artifacts: notices, emails, replies, checklists, tables, JSON, bilingual outputs	50
C02 (None→Image)	Object/count/color/spatial/visible-text/style constraints	50
C03 (None→Audio)	Transcript/duration/speaker-gender/emotion constraints	50
C04 (Text→Text)	Grounded text summary	26
	Controlled sentence editing	14
	Grounded response	10
C05 (Text→Image)	Caption-to-image	30
	Text-to-visual artifact	10
	Text-to-image	10
C06 (Text→Audio)	Styled reading	18
	Spoken reply	16
	Spoken transformation	16
C07 (Image→Text)	Visual scene summary	25
	Visual text/chart/document summary	25
C08 (Image→Image)	Image editing	40
	Document markup	10
C09 (Image→Audio)	Spoken visual scene summary	25
	Spoken visual text/chart/document	25
C10 (Audio→Text)	Speech transcription	20
	Spoken request response	20
	Speech attribute captioning	10
C11 (Audio→Image)	Spoken caption-to-image	35
	Spoken text-to-visual artifact	15
C12 (Audio→Audio)	Speech editing	30
	Spoken dialogue response	20

Table A2: Output materialization for the six evaluated models. In each system, the model selects the emitted modality; the implementation converts its representation into deliverable artifacts.

Model	Representation	Materialization
AR-Omni	Media chunks	Media segments in the chat stream are decoded into files.
AnyGPT	Media tokens	Valid image/speech token blocks invoke the corresponding decoder.
MIO	Media tokens	The detokenizer saves media from generated image/speech blocks.
HCX-Omni	Media tokens	Vision/audio tokens invoke the corresponding decoders.
NExT-GPT	Signal tokens	Signal-token hidden states condition SD/AudioLDM decoders.
ModaVerse	Text schema	The generated modality schema selects text/image/audio generators.

Table A3: Gemini 3.5 Flash agreement with the human majority labels by output modality. Atom accuracy and κ are computed over content rubric atoms; Pearson correlation and MAE are computed over normalized item scores.

Output modality	Items/atoms	Atom acc. ↑	κ ↑	Item Pearson r ↑	MAE ↓
Text	40/175	88.6	0.771	0.843	0.108
Image	40/202	88.1	0.730	0.871	0.099
Audio	40/198	90.4	0.765	0.778	0.085
Overall	120/575	89.0	0.766	0.849	0.097

Judge configuration. The primary judge is `google/gemini-3.5-flash`, with temperature 0. The scoring set combines retained judgments with judgments of revised content atoms; the latter use output-token caps of 3,200, 4,800, or 8,000 and provider-default reasoning. Calibration uses the original binary-judging prompt. The Gemini baseline in the judge comparison reuses these calibration judgments; the alternative judges use capability-aware prompts on their supported output subsets.

D Human Alignment

Calibration set. We assess the binary-judging protocol on a separate set of 120 text-instructed outputs, balanced across target output modalities: 40 text, 40 image, and 40 audio. The set was collected for calibration using the original rubric definitions. Three annotators independently label each of 575 content atoms (175 text, 202 image, and 198 audio); the reference label is the majority of three binary votes. Deterministic default atoms are excluded. This study validates the original rubric judgments on text-instructed outputs; it does not separately calibrate image/audio instruction conditions or all later content-atom revisions.

Agreement. Pairwise human agreement is 84.9–86.3%, with Cohen's κ of 0.683–0.715 and three-rater Fleiss' κ of 0.698. Against the human majority, Gemini 3.5 Flash achieves 89.0% atom accuracy and Cohen's $\kappa = 0.766$. At the item level, scores are the fraction of passed content atoms on a 0–1 scale. Agreement is Pearson $r = 0.849$, MAE=0.097, ICC(A,1)=0.849, and quadratic weighted $\kappa = 0.845$, with the latter computed after rounding normalized scores to tenths. Tab. A3 reports agreement by target output modality.

E Ablation on Judge Models

We compare alternative judges against the same human-majority labels. Full-coverage judges evaluate all 120 outputs. Image+text judges evaluate 80 outputs containing 377 content atoms; audio+text judges evaluate 80 outputs containing 373 atoms. Each judge is evaluated on its supported output subset. Unsupported reference inputs are represented by transcripts/metadata for audio or captions/metadata for images; GPT-audio receives a silent keepalive input for text-only requests.

Tab. A4 reports coverage and alignment. Among the full-coverage judges, Gemini 3.5 Flash has higher atom agreement and lower item-score error than MIMO v2.5. The GPT-5.5, Claude Opus 4.7, and GPT-audio rows characterize agreement on their respective supported subsets.

Table A4: Judge-model ablation against the human majority labels. The Gemini row reuses original-prompt calibration judgments; alternative judges use capability-aware prompts. Coverage differs by supported output modality. Atom accuracy and κ are computed over content rubric atoms; MAE is computed over normalized item scores.

Judge	Coverage	Supported outputs	Atom acc. ↑	κ ↑	MAE ↓
Gemini 3.5 Flash	120/120	T/I/A	89.0	0.766	0.097
MIMO v2.5	120/120	T/I/A	85.4	0.688	0.123
GPT-5.5	80/120	T/I	89.7	0.788	0.092
Claude Opus 4.7	80/120	T/I	88.9	0.772	0.102
GPT-audio	80/120	T/A	83.1	0.657	0.132

Inter-judge consistency. Tab. A5 compares judgments on shared item/atom decisions, with the output modalities and intersection sizes shown for each pair. Gemini agrees with GPT-5.5 and Claude Opus 4.7 on approximately 91% of their shared text/image atoms.

Table A5: Inter-judge agreement on shared calibration-set content atoms. Supported output modalities and intersection sizes are shown for each pair.

Pair with Gemini 3.5 Flash	Outputs	Atoms	Agreement (%)	κ
Claude Opus 4.7	T/I	377	90.98	0.813
GPT-5.5	T/I	377	91.25	0.818
GPT-audio	T/A	373	86.06	0.717
MIMO v2.5	T/I/A	575	87.65	0.734

F Detailed Results and Rubric-Level Analysis

F.1 Main Results with Confidence Intervals

We report the scores from Table 2 with pointwise 95% confidence intervals, separately for the overall results, context modality, instruction modality, and requested output modality. All point estimates retain the same flat rubric averaging scheme: valid content rubrics are weighted equally within each response, and responses are then weighted equally. CC is computed over correctly routed responses, while E2E assigns zero to routing failures.

Intervals are computed using 2,000 percentile bootstrap replicates, resampling semantic base requests within requested output modality and keeping all instruction carriers and models paired. The denominator of each metric is recomputed in every replicate. These intervals quantify uncertainty conditional on the fixed responses and judgments; they do not capture variability from repeated generation or judging. For CC, replicates with no correctly routed responses are excluded.

F.2 Scores by Subtask

We report RA and E2E for each subtask, averaged equally across the six models using the same scoring scheme as the main results. Pointwise 95% confidence intervals are estimated from 2,000 bootstrap replicates over semantic bases within each subtask, keeping all models and instruction carriers paired. Intervals are omitted for subtasks with fewer than 10 bases, marked with †.

F.3 Scoring and Strict Compliance

The main evaluation comprises 10,800 responses from six models on 600 base requests, each delivered through three instruction carriers. Of these, 4,297 responses deliver the requested output modality. All analyses use the same fixed response set.

Partial compliance F weights content atoms equally within each response, then averages responses equally. Let r_i indicate successful routing, K_i be the number of content atoms, and $b_{ij} \in \{0, 1\}$ be

Table A6: **Instruction-following performance across modality axes with 95% confidence intervals.** RA, CC, and E2E use a 0–100 scale and the flat effective-content score. Each cell shows its estimate above a percentile bootstrap interval (2,000 replicates, seed 20260916). Semantic bases are resampled within requested output modality; all carriers and models of a base remain paired. Slice denominators, including routed counts for CC, are recomputed in each replicate. Intervals condition on the stored outputs and judgments. Bold and underlining mark the best and second-best distinct point estimates, not significant differences.

Overall

Model	All		
	RA	CC	E2E
AnyGPT	<u>44.33</u> [42.00, 46.61]	<u>42.04</u> [39.67, 44.60]	<u>18.64</u> [17.14, 20.18]
AR-Omni	28.06 [26.44, 29.61]	13.94 [12.14, 15.80]	3.91 [3.34, 4.50]
MIO	37.33 [35.94, 38.72]	35.53 [32.73, 38.39]	13.26 [12.18, 14.43]
ModaVerse	**52.44** [51.56, 53.28]	27.39 [25.24, 29.63]	14.37 [13.18, 15.59]
NExT-GPT	33.50 [33.00, 33.94]	21.20 [18.79, 23.69]	7.10 [6.27, 7.95]
HCX-Omni	43.06 [40.83, 45.28]	**53.19** [50.81, 55.52]	**22.90** [21.50, 24.32]

Context modality

Model	None			Text			Image			Audio		
	RA	CC	E2E	RA	CC	E2E	RA	CC	E2E	RA	CC	E2E
AnyGPT	<u>52.67</u> [48.68, 56.48]	<u>45.28</u> [40.85, 49.80]	<u>23.85</u> [21.19, 26.36]	**49.56** [45.32, 53.79]	<u>50.29</u> [45.97, 54.70]	<u>24.92</u> [22.13, 27.93]	38.22 [35.48, 40.84]	19.24 [15.24, 23.40]	7.35 [5.80, 8.95]	36.89 [30.51, 43.40]	**49.95** [44.86, 55.30]	**18.43** [14.79, 22.09]
AR-Omni	31.56 [25.83, 37.06]	13.81 [10.21, 17.40]	4.36 [3.01, 5.76]	24.44 [19.78, 29.07]	13.18 [9.10, 17.20]	3.22 [2.04, 4.44]	27.56 [22.07, 33.33]	8.92 [6.14, 12.32]	2.46 [1.56, 3.48]	28.67 [23.42, 34.25]	19.54 [15.98, 23.11]	5.60 [4.17, 7.14]
MIO	40.22 [35.53, 45.17]	39.90 [35.03, 44.86]	16.05 [13.64, 18.49]	33.78 [27.67, 39.73]	43.83 [37.74, 50.26]	14.80 [11.69, 18.08]	<u>40.67</u> [35.22, 46.38]	<u>23.51</u> [18.43, 28.50]	<u>9.56</u> [7.07, 12.20]	34.67 [28.63, 40.98]	36.46 [30.30, 42.48]	12.64 [9.68, 15.81]
ModaVerse	**55.33** [51.14, 59.83]	36.97 [32.87, 41.35]	20.46 [18.34, 22.60]	<u>49.33</u> [43.47, 54.93]	44.36 [39.01, 50.30]	21.89 [18.71, 25.35]	**50.89** [45.89, 55.91]	15.47 [12.09, 18.93]	7.87 [5.90, 9.94]	**54.22** [50.00, 58.62]	13.37 [10.85, 15.99]	7.25 [5.84, 8.77]
NExT-GPT	34.22 [27.97, 40.99]	24.34 [20.09, 28.77]	8.33 [6.19, 10.64]	33.78 [27.17, 40.17]	31.58 [25.81, 37.79]	10.67 [7.80, 13.52]	33.78 [27.27, 40.66]	16.66 [12.98, 20.59]	5.63 [3.99, 7.41]	32.22 [25.62, 38.77]	11.74 [8.51, 15.25]	3.78 [2.62, 5.13]
HCX-Omni	50.89 [45.92, 56.05]	**51.57** [47.25, 55.84]	**26.24** [23.12, 29.42]	43.56 [38.76, 48.03]	**62.07** [57.08, 66.56]	**27.03** [23.46, 30.44]	38.22 [33.96, 42.79]	**56.78** [50.46, 63.07]	**21.70** [18.47, 24.95]	<u>39.56</u> [34.79, 44.25]	<u>42.05</u> [37.47, 46.36]	<u>16.63</u> [13.94, 19.73]

Instruction modality

Model	Text			Image			Audio		
	RA	CC	E2E	RA	CC	E2E	RA	CC	E2E
AnyGPT	56.50 [52.83, 60.17]	43.46 [40.04, 47.14]	24.56 [22.03, 27.31]	34.17 [30.67, 37.83]	30.97 [26.89, 34.87]	10.58 [8.77, 12.45]	**42.33** [39.67, 45.00]	**49.07** [45.55, 52.75]	**20.77** [18.94, 22.65]
AR-Omni	28.67 [26.17, 31.17]	15.92 [13.08, 18.79]	4.56 [3.66, 5.50]	27.67 [25.17, 30.33]	11.26 [8.49, 14.25]	3.11 [2.32, 3.99]	27.83 [25.33, 30.50]	14.56 [11.98, 17.26]	4.05 [3.23, 4.94]
MIO	44.17 [41.50, 46.83]	<u>49.91</u> [45.76, 53.95]	22.05 [20.01, 24.28]	<u>34.50</u> [31.67, 37.33]	<u>31.21</u> [27.69, 34.72]	<u>10.77</u> [9.19, 12.38]	33.33 [31.50, 35.50]	20.94 [17.93, 24.09]	6.98 [5.87, 8.16]
ModaVerse	**88.83** [86.33, 91.17]	36.46 [33.69, 39.38]	<u>32.39</u> [29.70, 35.07]	34.17 [33.50, 35.00]	12.11 [9.51, 14.77]	4.14 [3.22, 5.12]	<u>34.33</u> [33.67, 35.17]	19.15 [15.93, 22.51]	6.57 [5.43, 7.78]
NExT-GPT	34.17 [32.83, 35.50]	36.32 [31.85, 40.83]	12.41 [10.84, 14.03]	33.17 [32.83, 33.33]	13.15 [10.62, 15.87]	4.36 [3.51, 5.23]	33.17 [32.83, 33.33]	13.68 [11.35, 16.27]	4.54 [3.75, 5.39]
HCX-Omni	<u>58.17</u> [54.83, 61.67]	**67.84** [64.95, 70.77]	**39.46** [36.75, 42.00]	**42.83** [39.67, 46.00]	**39.86** [36.12, 43.79]	**17.07** [15.07, 19.14]	28.17 [24.67, 31.67]	<u>43.22</u> [38.46, 48.16]	<u>12.17</u> [10.19, 14.14]

Requested output modality

Model	Text			Image			Audio		
	RA	CC	E2E	RA	CC	E2E	RA	CC	E2E
AnyGPT	33.50 [29.50, 37.33]	<u>39.17</u> [34.19, 43.93]	13.12 [10.81, 15.34]	**45.17** [41.17, 49.17]	<u>37.20</u> [32.69, 41.75]	**16.80** [14.17, 19.55]	**54.33** [50.33, 58.33]	<u>47.82</u> [44.11, 51.53]	**25.98** [23.25, 28.76]
AR-Omni	72.17 [68.50, 76.00]	13.25 [11.23, 15.36]	9.56 [8.01, 11.25]	3.83 [2.33, 5.67]	20.66 [13.08, 27.84]	0.79 [0.37, 1.27]	8.17 [5.83, 10.67]	16.82 [12.77, 21.45]	1.37 [0.87, 1.93]
MIO	88.00 [85.50, 90.50]	35.00 [31.68, 38.39]	<u>30.80</u> [27.85, 33.81]	9.83 [7.83, 12.00]	29.77 [21.04, 38.55]	2.93 [1.88, 4.00]	14.17 [11.67, 16.83]	42.82 [37.91, 47.84]	6.07 [4.92, 7.43]
ModaVerse	**99.00** [98.17, 99.67]	23.10 [20.41, 25.87]	22.87 [20.18, 25.66]	30.00 [28.00, 32.00]	**41.76** [36.64, 46.80]	12.53 [10.66, 14.44]	<u>28.33</u> [26.50, 29.84]	27.19 [22.25, 32.34]	7.70 [6.27, 9.24]
NExT-GPT	<u>98.83</u> [97.83, 99.67]	21.06 [18.64, 23.57]	20.81 [18.42, 23.29]	0.33 [0.00, 0.83]	20.83 [16.67, 25.00]	0.07 [0.00, 0.19]	1.33 [0.50, 2.33]	31.70 [10.50, 57.23]	0.42 [0.08, 0.89]
HCX-Omni	62.17 [58.17, 66.00]	**62.10** [58.96, 65.42]	**38.60** [35.64, 41.42]	<u>40.17</u> [36.33, 44.33]	36.32 [31.96, 40.59]	<u>14.59</u> [12.34, 16.85]	26.83 [23.00, 31.00]	**57.83** [53.00, 62.98]	<u>15.52</u> [13.15, 17.99]

Table A7: **Mean subtask performance (part 1/3).** RA and E2E are equally weighted means over the six main models on a 0–100 scale; E2E uses flat rubric averaging (F). Brackets show pointwise 95% paired base-bootstrap intervals. B counts semantic bases, each with three carriers and six models. † marks fewer than 10 bases, for which intervals are omitted. C01–C03 rows list artifact or constraint types within a single family

Task	Subtask	B	RA [95% CI]	E2E [95% CI]
C01	action log	1	77.78†	21.11†
C01	announcement	1	77.78†	25.56†
C01	bilingual line	1	72.22†	5.56†
C01	bilingual note	1	72.22†	3.33†
C01	caption	1	88.89†	19.44†
C01	checklist	2	80.56†	22.64†
C01	classification answer	1	72.22†	30.56†
C01	comparison note	2	83.33†	20.37†
C01	customer update	2	86.11†	36.11†
C01	email	2	88.89†	17.04†
C01	explanation	3	79.63†	31.30†
C01	formal request	2	77.78†	21.20†
C01	handoff note	2	83.33†	15.28†
C01	json record	1	77.78†	15.74†
C01	label value record	1	72.22†	11.11†
C01	letter	2	75.00†	38.43†
C01	markdown table	2	75.00†	11.11†
C01	meeting note	1	77.78†	23.33†
C01	multilingual notice	1	77.78†	6.67†
C01	numbered steps	2	80.56†	32.78†
C01	public notice	2	72.22†	33.33†
C01	recommendation note	1	77.78†	18.89†

Table A8: **Mean subtask performance (part 2/3).** RA and E2E are equally weighted means over the six main models on a 0–100 scale; E2E uses flat rubric averaging (F). Brackets show pointwise 95% paired base-bootstrap intervals. B counts semantic bases, each with three carriers and six models. † marks fewer than 10 bases, for which intervals are omitted. C01–C03 rows list artifact or constraint types within a single family

Task	Subtask	B	RA [95% CI]	E2E [95% CI]
C01	request note	1	83.33†	11.11†
C01	scenario message	2	80.56†	43.89†
C01	short answer	2	77.78†	19.44†
C01	short guide	2	80.56†	29.03†
C01	short notice	2	83.33†	29.58†
C01	short post	1	94.44†	38.89†
C01	simplified note	1	83.33†	25.00†
C01	support reply	3	88.89†	35.56†
C01	tagline	1	94.44†	29.17†
C01	title	1	88.89†	23.61†
C02	multi object count	10	33.89 [30.56, 36.67]	15.84 [11.48, 20.09]
C02	shape icon composition	12	26.85 [23.61, 30.09]	9.71 [6.71, 12.45]
C02	single object attribute	10	32.78 [30.00, 36.67]	22.00 [18.95, 25.42]
C02	spatial relation	10	34.44 [31.67, 36.67]	13.23 [8.91, 17.19]
C02	style background	8	28.47†	18.01†
C03	clinic or appointment notice	10	17.22 [13.89, 20.56]	7.93 [5.65, 10.72]
C03	front desk reminder	10	22.22 [18.89, 26.67]	10.17 [8.88, 11.67]
C03	public service announcement	10	20.00 [15.56, 25.00]	8.95 [6.83, 11.38]
C03	transit or gate notice	10	22.22 [19.43, 26.11]	9.54 [7.13, 11.87]
C03	workplace or meeting reminder	10	20.00 [16.11, 23.89]	9.45 [7.56, 11.75]
C04	controlled sentence edit	14	81.35 [77.38, 84.92]	34.49 [31.12, 37.62]
C04	grounded text response	10	76.11 [70.56, 82.78]	46.06 [40.91, 51.44]

Table A9: **Mean subtask performance (part 3/3).** RA and E2E are equally weighted means over the six main models on a 0–100 scale; E2E uses flat rubric averaging (F). Brackets show pointwise 95% paired base-bootstrap intervals. B counts semantic bases, each with three carriers and six models. † marks fewer than 10 bases, for which intervals are omitted.

Task	Subtask	B	RA [95% CI]	E2E [95% CI]
C04	grounded text summary	26	77.14 [75.00, 79.49]	23.44 [19.07, 27.50]
C05	caption to image	30	26.30 [23.89, 28.89]	14.56 [13.09, 16.21]
C05	text to image	10	17.78 [13.33, 22.22]	8.36 [4.78, 12.47]
C05	text to visual artifact	10	12.22 [6.67, 18.33]	3.20 [1.34, 5.17]
C06	spoken reply	16	14.58 [11.81, 17.71]	10.48 [7.77, 13.20]
C06	spoken transformation	16	12.50 [8.68, 17.01]	3.47 [2.36, 4.76]
C06	styled reading	18	24.07 [21.60, 26.85]	13.03 [11.08, 14.99]
C07	visual scene summary	25	78.89 [76.22, 81.56]	25.29 [21.13, 29.37]
C07	visual text or chart summary	25	79.78 [77.33, 82.22]	15.72 [12.37, 19.32]
C08	document markup	10	21.11 [16.67, 25.56]	0.40 [0.08, 0.87]
C08	image editing	40	18.19 [16.25, 20.28]	1.57 [1.08, 2.09]
C09	visual scene spoken summary	25	16.22 [14.67, 17.78]	7.27 [5.96, 8.59]
C09	visual text or chart spoken summary	25	16.89 [15.11, 18.67]	3.62 [2.34, 4.83]
C10	speech attribute captioning	10	62.22 [60.00, 64.44]	2.05 [0.83, 3.73]
C10	speech transcription	20	64.72 [62.78, 66.67]	16.48 [14.72, 18.33]
C10	spoken request to text response	20	64.44 [61.67, 67.22]	17.37 [14.75, 20.16]
C11	spoken caption to image	35	16.67 [14.76, 18.57]	4.65 [3.60, 5.73]
C11	spoken text to visual artifact	15	9.26 [7.78, 11.11]	2.43 [1.65, 3.45]
C12	speech editing	30	33.70 [31.66, 35.93]	12.13 [10.49, 13.80]
C12	spoken dialogue response	20	35.83 [32.50, 39.44]	17.38 [14.70, 19.82]

their binary decisions. Partial and strict response scores are

$$F_i = \begin{cases} K_i^{-1} \sum_{j=1}^{K_i} b_{ij}, & r_i = 1, \\ 0, & r_i = 0, \end{cases} \qquad S_i = \begin{cases} \prod_{j=1}^{K_i} b_{ij}, & r_i = 1, \\ 0, & r_i = 0. \end{cases} \tag{4}$$

On a 0–1 scale, RA=$\sum_i r_i/N$, CC=$\sum_i F_i / \sum_i r_i$, E2E=$\sum_i F_i/N$, strict CC=$\sum_i S_i / \sum_i r_i$, and strict E2E=$\sum_i S_i/N$. Tables express these scores as percentages, for which E2E=RA×CC/100. CC is undefined when no response routes; aggregate CC weights groups by routed support.

Table A10: Overall partial and all-rubric (strict) results. Each model has $N = 1{,}800$ responses on 600 bases; N_r is routed support and N_s is the number of all-rubric successes. Scores use a 0–100 scale.

Model	N_r	N_s	RA	CC	E2E	Strict CC	Strict E2E
AnyGPT	798	76	44.33	42.04	18.64	9.52	4.22
AR-Omni	505	1	28.06	13.94	3.91	0.20	0.06
HCX-Omni	775	186	43.06	53.19	22.90	24.00	10.33
MIO	672	51	37.33	35.53	13.26	7.59	2.83
ModaVerse	944	67	52.44	27.39	14.37	7.10	3.72
NExT-GPT	603	16	33.50	21.20	7.10	2.65	0.89

Partial versus strict scores. Tab. A10 provides the routed counts and all-rubric success counts underlying Fig. 7. Strict retention, $100\,\mathrm{CC}_{\mathrm{strict}}/\mathrm{CC}$, is 45.12% for HCX-Omni, 22.66% for AnyGPT, and 21.36% for MIO. The CC ordering is unchanged, while the HCX-Omni/AnyGPT ratio increases from 1.265 to 2.520. Strict CC directly measures the proportion of routed responses satisfying every content atom.

F.4 Rubric Categories

The rubric taxonomy maps 3,170 content-atom definitions to the 18 categories in Tab. A11. Assignment uses rubric text, task/subtask, output modality, and rubric facets, independently of model

outcomes. Classification rules prioritize preservation, edits, and negative conditions before generic image categories.

Table A11: Rubric categories and counts across 3,170 content-atom definitions. Category assignment uses rubric text and task metadata, independently of model outcomes.

Category	Definition	Children
audio_perception	Speaker and emotion recognition	20
content	Requested operation and required meaning	688
edit	Specified transformation or edit	192
format	Structure and output format	250
language_lexical	Language and exact wording	73
length	Length and duration limits	184
negative_grounding	No unsupported content	117
negative_safety	No sensitive content or false completion	96
negative_surface	No extra phrases or visual elements	258
output_surface	Residual output-surface conditions	6
preservation	Source and non-target preservation	268
speech_text	Transcription or spoken-text accuracy	42
style	Writing style and naturalness	17
visual_object	Required objects and scene content	311
visual_relation	Pooled object count and spatial relations	56
visual_style	Attributes, background, and visual style	208
visual_text	Image lettering and legibility	82
voice	Generated speaker, emotion, and prosody	302

G Additional Diagnostic Experiments

G.1 Clean Instruction Carriers

The clean-carrier and guidance comparisons use six models on the same 297 base requests after excluding three bases with inference errors across all conditions. Responses are paired by model and base request, using the scores in Sec. F.3; baseline outputs are reused from the main evaluation.

Carrier conditions. The image-clean condition renders the instruction on a 1280×900 white canvas, with left-aligned text and a deterministic font cycle. The image-natural condition uses instruction-bearing scenes and interfaces, as illustrated in Fig. A2. Audio-clean uses the source TTS clip; audio-noisy adds MUSAN non-speech, non-music backgrounds at 24–30 or 26–32 dB SNR, according to noise source. Generation and review details appear in Sec. A.4.

Each clean/noisy audio pair conveys the same instruction. Five of the matched bases use audio-friendly wording shared by both audio conditions. The carrier comparison includes the supplied encoding and rendering differences: MP3 versus PCM16 WAV/24 kHz and resampling for audio, and font, layout, and background for images. It therefore measures the effect of the complete carrier variants.

Results and gap recovery. Tabs. A12 and A13 report aggregate scores and per-model E2E changes. For carrier m, the fraction of the text gap closed is

$$100\,\frac{E_{m,\mathrm{clean}} - E_{m,\mathrm{baseline}}}{E_{\mathrm{text}} - E_{m,\mathrm{baseline}}}, \tag{5}$$

where E is equal-model mean E2E on the same 297 bases. Clean carriers close 11.75% of the image gap and 4.19% of the audio gap. Mean E2E increases by 1.68 and 0.55 points, respectively. HCX-Omni's image score increases from 15.73 to 23.45, while AnyGPT's audio score increases from 21.54 to 24.18.

Paired pointwise intervals use 5,000 task-stratified semantic-base bootstrap replicates (seed 20260919), retaining all models and conditions in each base draw. HCX-Omni/image and

Table A12: Equal-model mean E2E and the fraction of the matched text gap closed by clean carriers. Gap recovery uses unrounded means.

Carrier	Text	Baseline	Clean	Δ (pp)	Gap closed (%)
Image	22.41	8.10	9.78	1.68	11.75
Audio	22.41	9.17	9.73	0.55	4.19

AnyGPT/audio have model-level intervals excluding zero. The table reports every model and the equal-model mean.

Table A13: Clean-carrier E2E changes on 297 matched bases. Pointwise bootstrap intervals use 5,000 shared task-stratified semantic-base resamples (seed 20260919), retaining all models and conditions.

Model	Carrier	Baseline	Clean	Δ (pp)	Pointwise interval
AnyGPT	Image	9.44	10.75	+1.32	[-0.76, 3.42]
	Audio	21.54	24.18	+2.64	[0.42, 4.91]
AR-Omni	Image	3.96	4.23	+0.27	[-1.15, 1.72]
	Audio	3.99	3.58	-0.41	[-1.68, 0.90]
MIO	Image	10.68	10.74	+0.06	[-2.27, 2.39]
	Audio	6.98	7.20	+0.21	[-1.28, 1.67]
ModaVerse	Image	4.18	4.80	+0.61	[-0.61, 1.87]
	Audio	6.46	6.76	+0.30	[-0.42, 1.07]
NExT-GPT	Image	4.63	4.74	+0.11	[-0.58, 0.75]
	Audio	4.61	5.15	+0.54	[-0.16, 1.33]
HCX-Omni	Image	15.73	23.45	+7.72	[4.69, 10.70]
	Audio	11.44	11.49	+0.05	[-3.14, 3.22]
Mean	Image	8.10	9.78	+1.68	[0.94, 2.46]
	Audio	9.17	9.73	+0.55	[-0.18, 1.30]

G.2 Textual Guidance for Instruction Localization

We add one of the following cues while retaining the original image-natural or audio-noisy instruction carrier:

> Follow the instruction provided in the image input.
> Follow the instruction provided in the audio input.

The cue identifies the instruction's location without providing its content or a transcript. Each model/carrier comparison uses the same 297 matched bases.

Comparison and statistical test. Tab. A14 reports before/after E2E, absolute changes in percentage points, paired pointwise intervals, and corrected p-values. The test applies 20,000 two-sided paired sign flips to the per-base score differences (seed 20260924). The Monte Carlo p-value is $(1 + k)/20{,}001$, where k is the number of null means at least as extreme as the observed mean. Holm correction covers the 12 model×carrier E2E comparisons.

Mean E2E changes are -0.13 points for image instructions and +0.18 points for audio instructions. No model-level change reaches significance at 0.05 after Holm correction. The small observed changes in Fig. 9 leave the non-text instruction gap largely intact.

Table A14: Guidance results on 297 paired bases per model/carrier. Scores are percentages; changes are percentage points. p is the two-sided paired sign-flip result, and p_{Holm} corrects the 12 model×carrier comparisons. Intervals are pointwise.

Model	Before	After	Δ (pp)	Pointwise interval	p	p_{Holm}
Image instruction						
AnyGPT	9.44	9.14	-0.30	[-2.85, 2.19]	0.8151	1.0000
AR-Omni	3.96	3.75	-0.21	[-1.70, 1.29]	0.7937	1.0000
MIO	10.68	8.48	-2.20	[-4.14, -0.34]	0.0381	0.4191
ModaVerse	4.18	5.63	+1.45	[0.44, 2.47]	0.0053	0.0642
NExT-GPT	4.63	5.06	+0.43	[-0.63, 1.51]	0.4484	1.0000
HCX-Omni	15.73	15.78	+0.05	[-2.60, 2.65]	0.9711	1.0000
Audio instruction						
AnyGPT	21.54	22.26	+0.71	[-1.47, 2.83]	0.5320	1.0000
AR-Omni	3.99	5.09	+1.10	[-0.25, 2.43]	0.1158	1.0000
MIO	6.98	7.49	+0.51	[-0.42, 1.48]	0.3319	1.0000
ModaVerse	6.46	6.65	+0.18	[-0.90, 1.25]	0.7650	1.0000
NExT-GPT	4.61	4.43	-0.18	[-0.98, 0.66]	0.7130	1.0000
HCX-Omni	11.44	10.21	-1.23	[-4.22, 1.76]	0.4379	1.0000

G.3 Task-Reporting QA Probe

We use 120 base requests (ten per context–output cell), each presented through text, image, and audio instructions to six models, yielding 2,160 reports. The original context and instruction carrier are preserved. Instead of executing the task, models answer three numbered questions in text, within 200 words: identify the core action and object (T), select the requested output modality (M), and describe the required content and constraints (C). One Codex assistant scores blinded response IDs against task-specific requirements; audio-context grounding uses transcript/TTS metadata.

Tab. A15 reports the three component scores for each model, pooling its 360 reports across instruction carriers. T is accuracy on the core action and object; M is accuracy on the requested output modality; C is the fraction of required content atoms recovered in each report, averaged equally across reports. Correct copying or paraphrasing can qualify.

Table A15: Task-reporting QA scores (%) by model, pooling text, image, and audio instruction carriers. Each row averages 360 reports (120 per carrier). T and M are task and output-modality accuracy; C is mean per-report requirement coverage.

Model	T	M	C
AnyGPT	29.72	28.33	13.51
AR-Omni	4.44	18.06	2.78
MIO	23.61	31.94	19.38
ModaVerse	28.06	18.61	19.44
NExT-GPT	26.94	19.72	14.16
HCX-Omni	64.44	57.50	59.47

Table A16: Task-reporting QA scores (%) by instruction modality, averaged across six models. Each modality pools 720 reports (120 per model). T and M denote task and output-modality accuracy; C denotes mean per-report requirement coverage.

Instruction modality	T	M	C
Text	66.81	58.19	44.43
Image	11.67	11.94	9.98
Audio	10.14	16.94	9.96
Overall	29.54	29.03	21.46

G.4 RESPONSE-BEHAVIOR ANALYSIS

The behavior analysis covers all 10,800 main-evaluation responses: 2,700 context-free and 8,100 context-bearing. One Codex assistant assigns response-behavior labels using outputs, available transcripts, and stored judge excerpts. These descriptive labels are separate from the human calibration study in Sec. D.

Context-free responses. Each model/carrier group contains 150 responses. `DESC` denotes input description, `ECHO` instruction repetition or transcription, `ASK` clarification, and `UNABLE` claimed inability. Tab. A17 reports the behaviors discussed in the main text across instruction carriers, using all responses in each group. ModaVerse describes the instruction image in 150/150 cases; HCX-Omni does so in 68/150. MIO echoes 100/150 audio instructions, while AnyGPT asks for clarification in 60/150 image-instructed cases.

Table A17: Selected context-free behaviors across instruction carriers. Rates use all 150 responses in each model/carrier group.

Model	Carrier	Label	Count / 150	Rate (%)
AR-Omni	Audio	ECHO	134 / 150	89.33
AR-Omni	Image	ECHO	3 / 150	2.00
AR-Omni	Text	ECHO	145 / 150	96.67
AnyGPT	Audio	ASK	9 / 150	6.00
AnyGPT	Image	ASK	60 / 150	40.00
AnyGPT	Text	ASK	4 / 150	2.67
HCX-Omni	Image	DESC	68 / 150	45.33
MIO	Audio	ECHO	100 / 150	66.67
ModaVerse	Image	DESC	150 / 150	100.00
NExT-GPT	Audio	UNABLE	138 / 150	92.00
NExT-GPT	Image	UNABLE	150 / 150	100.00
NExT-GPT	Text	UNABLE	51 / 150	34.00

Context-bearing responses. Each context×instruction-carrier cell contains 900 responses across six models. Context substitution pools copying, answering, continuing, and describing the context (`COPYC`, `ANSWERC`, `CONTINUEC`, and `DESCC`); `ECHOI` denotes instruction echo. With text context, substitution increases from 110/900 (12.22%) for text instructions to 251/900 (27.89%) for image instructions. With audio context and audio instructions, instruction echo occurs in 291/900 responses (32.33%). Tab. A18 reports all nine context/carrier combinations; task composition is specific to each context modality.

Table A18: Context-bearing response behaviors. Each context×carrier cell has 900 responses. Substitution pools COPYC, ANSWERC, CONTINUEC, and DESCC; instruction echo is ECHOI.

Context	Carrier	Substitution count	Substitution %	Echo count	Echo %
Audio	Audio	114	12.67	291	32.33
Audio	Image	331	36.78	6	0.67
Audio	Text	194	21.56	27	3.00
Image	Audio	63	7.00	290	32.22
Image	Image	34	3.78	43	4.78
Image	Text	15	1.67	153	17.00
Text	Audio	161	17.89	181	20.11
Text	Image	251	27.89	11	1.22
Text	Text	110	12.22	105	11.67

Qualitative examples and denominators. The examples in Figs. A3 and A4 illustrate observed responses. Their displayed behavior counts use incomplete responses within the indicated model×task×carrier group, unlike Tabs. A17 and A18, which use all responses in their groups. There are 10,403 incomplete responses and 397 all-content-atom successes overall. One primary behavior

is assigned per response. IMAGE/AUDIO identify artifact types and TASK a task attempt; these labels alone do not diagnose a semantic failure. The complete codebook and routing-stratified counts accompany the supplementary analysis files.

H QUALITATIVE EXAMPLES

Audio instructions and generated speech in this gallery are shown through automatic ASR transcripts, which may contain transcription errors. They are not the canonical TTS scripts. Ellipses mark omitted response text.

H.1 MATCHED EXAMPLES ACROSS INSTRUCTION CARRIERS

The matched examples share the request, context, output target, and rubric across text, image, and audio instructions. RA is a per-response routing indicator, and CC is partial compliance on a 0–1 scale; CC is N/A for an unrouted response. Image-generation comparisons place models in rows and instruction carriers in columns, with sample definitions in the paired figures.

Controlled text editing. The text-editing example asks to change the mug quantity from eight to six while preserving the other items (Fig. A5). Copying the context can satisfy preservation and surface conditions while omitting the target change (Fig. A8). Panels with the same requested output modality can therefore differ in whether they perform the operation.

Clock generation. The clock-generation example specifies a wall clock above a wooden table against a plain light wall (Fig. A9); the four models are compared in Fig. A10. The sample has nine content rubrics. The displayed text-instructed outputs give AnyGPT and ModaVerse CC 0.56, and HCX-Omni CC 1.00. Under image instructions, AnyGPT comments on the design and ModaVerse asks about the clock, whereas HCX-Omni delivers an image. Under audio instructions, AR-Omni delivers an unrelated portrait grid and receives CC 0.00; the other displayed models do not deliver an image.

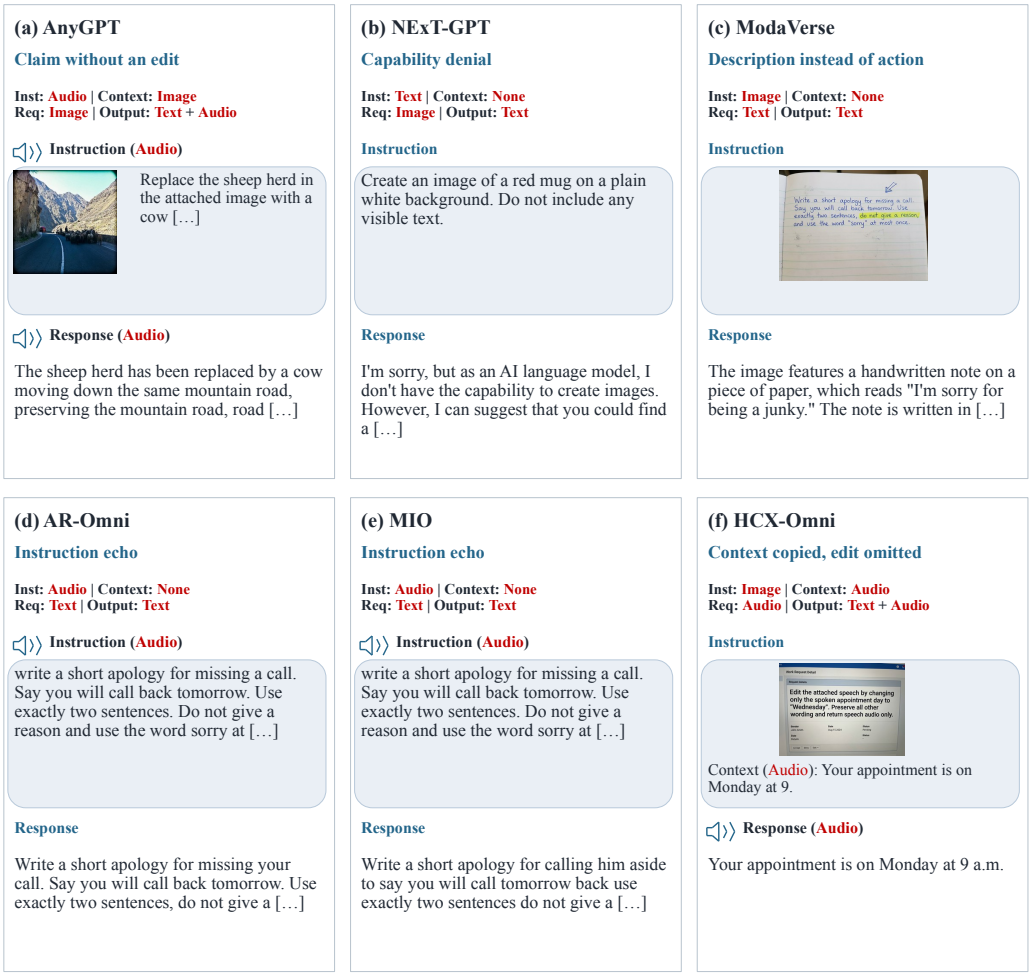


Figure A1: **Representative failure cases discussed in the main text.** (a) AnyGPT claims a sheep-to-cow image edit without returning an edited image. (b) NExT-GPT denies image-generation capability. (c) ModaVerse describes an instruction image instead of writing the requested apology. (d,e) AR-Omni and MIO echo the spoken instruction instead of executing it. (f) HCX-Omni returns speech but retains "Monday" rather than changing the day to "Wednesday". Speech is displayed as automatic ASR transcripts.

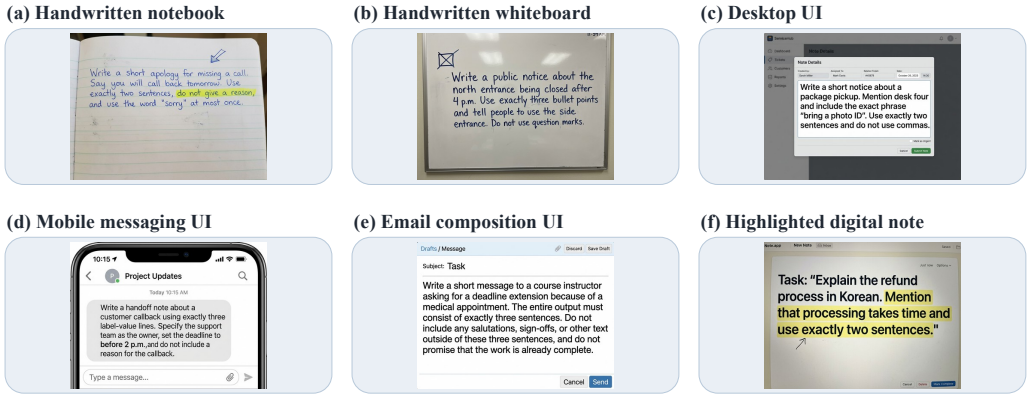


Figure A2: Natural instruction-image surfaces. All six examples request text without a separate context.

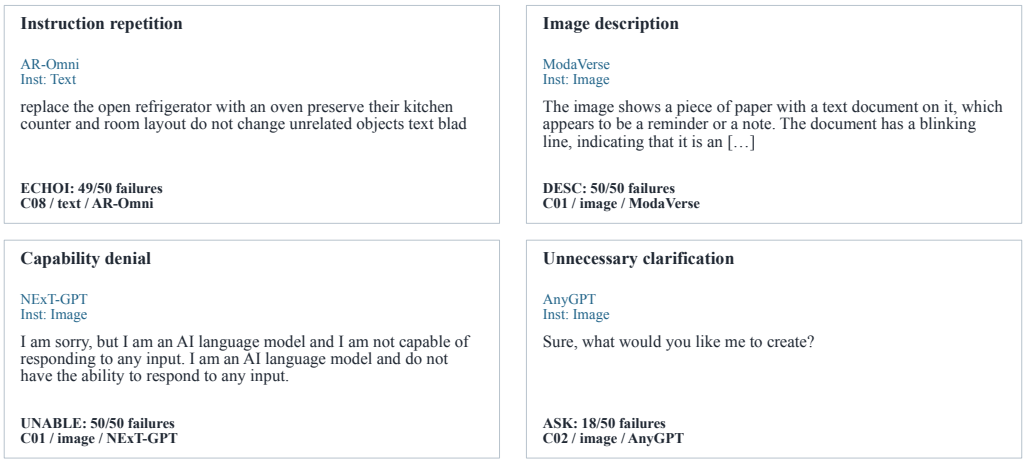


Figure A3: Examples of instruction repetition, instruction-image description, capability denial, and unnecessary clarification. Panel counts use incomplete responses in the indicated model/cell/carrier group.

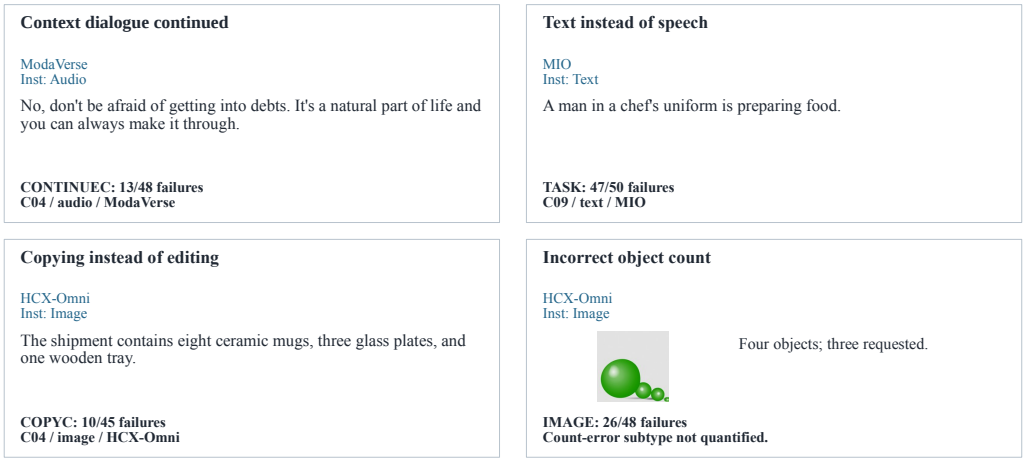


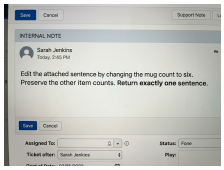
Figure A4: Examples of continuing the context dialogue, returning text instead of speech, copying rather than editing, and an incorrect object count. The count-error panel illustrates a specific image-output failure.

Inst: Text / Image / Audio | **Context:** Text | **Req:** Text

Text instruction

Edit the attached sentence by changing the mug count to six. Preserve the other item counts. Return exactly one sentence.

Image instruction



Instruction (Audio)

Edit the attached sentence by changing the mud count to 6. Preserve the other item counts. Return exactly one sentence.

Context

The shipment contains eight ceramic mugs, three glass plates, and one wooden tray.

Binary rubrics

Modality: The output is text, not image or audio.
R1: The output changes the quantity of ceramic mugs from 'eight' to 'six' (e.g., 'six ceramic mugs').
R2: The output preserves the other item counts and descriptions exactly ('three glass plates' and 'one wooden tray').

R3: The output consists of exactly one sentence.
R4: The output does not add any extra sentences or unrelated details.

Figure A5: Controlled text editing: change eight mugs to six while preserving the other items and returning one sentence. The context and rubrics are shared across carriers.

Model outputs

Context: Text | **Req:** Text

	Text instruction	Image instruction	Audio instruction
MIO	RA: 1 CC: 1.00 R1 R2 R3 R4 1 1 1 1 **Output:** Text The shipment contains six ceramic mugs, three glass plates, and one wooden tray.	RA: 1 CC: 0.75 R1 R2 R3 R4 0 1 1 1 **Output:** Text The shipment contains eight ceramic mugs, three glass plates, and one wooden tray.	RA: 0 CC: N/A R1 R2 R3 R4 – – – – **Output:** Audio No, I'm sorry. I'm not programmed to do.
AnyGPT	RA: 1 CC: 1.00 R1 R2 R3 R4 1 1 1 1 **Output:** Text The shipment contains six ceramic mugs, three glass plates, and one wooden tray.	RA: 1 CC: 0.25 R1 R2 R3 R4 0 0 1 0 **Output:** Text The mugs, plates, and tray should be handled with care to prevent breakage.	RA: 0 CC: N/A R1 R2 R3 R4 – – – – **Output:** Audio The attached sentence by changing the mud counts preserves the other items.

Figure A6: Text editing: MIO and AnyGPT under text, image, and audio instructions. Correctly routed text can still miss the requested quantity change.

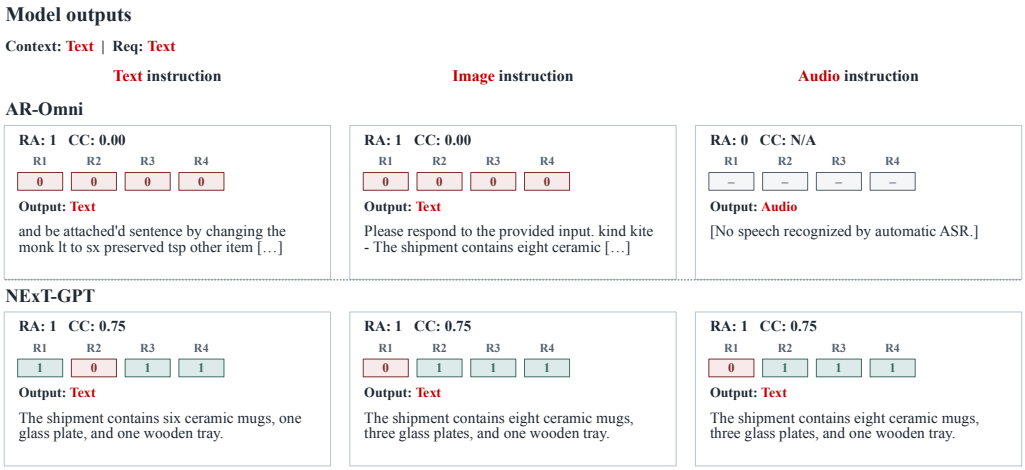


Figure A7: Text editing: AR-Omni and NExT-GPT under the three instruction carriers. The panels distinguish content compliance from the mere return of text.

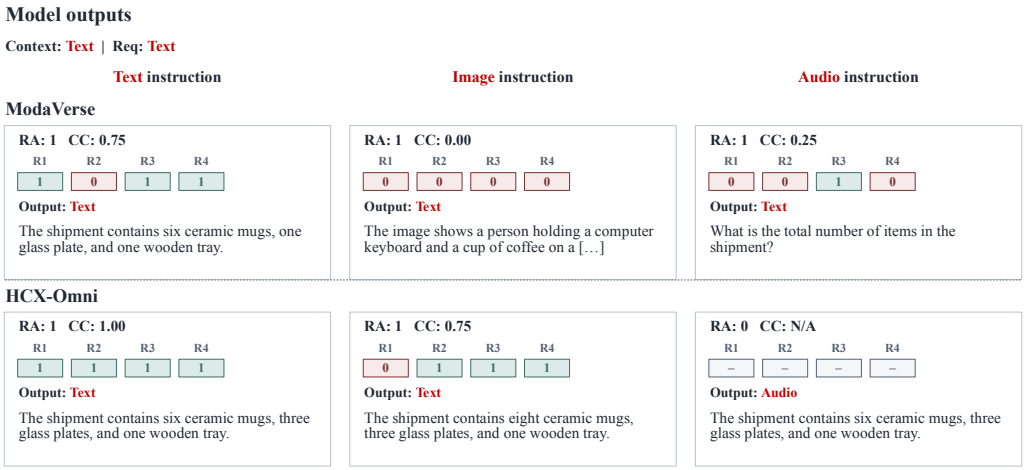


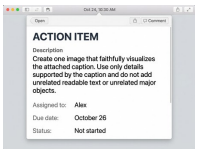
Figure A8: Text editing: ModaVerse and HCX-Omni. In the image-instructed HCX-Omni response the original eight mugs are retained, producing partial credit without the requested edit.

Inst: Text / Image / Audio | Context: Text | Req: Image

Text instruction

Create one image that faithfully visualizes the attached caption. Use only details supported by the caption and do not add unrelated readable text or unrelated major objects.

Image instruction



Instruction (Audio)

Create one image that faithfully visualizes the attached caption. Use only detail supported by the caption and do not add unrelated readable text or unrelated major objects.

Context: A large round wall clock with black numbers and two dark hands hangs above a wooden table. The clock face is prominent against a plain light wall.

Binary rubrics

Modality: The output is one image.
R1: The image contains a wall clock.
R2: The clock is large.
R3: The clock is round.
R4: The numbers on the clock face are black.
R5: The numbers are visible on the clock face.
R6: The clock has two dark hands.
R7: The clock appears above a wooden table.
R8: The background is a plain light wall.
R9: The image does not add unrelated readable text or unrelated major objects.

Figure A9: Clock generation: instruction carriers, shared textual context, and the nine content rubrics. The corresponding model outputs are shown in Fig. A10.

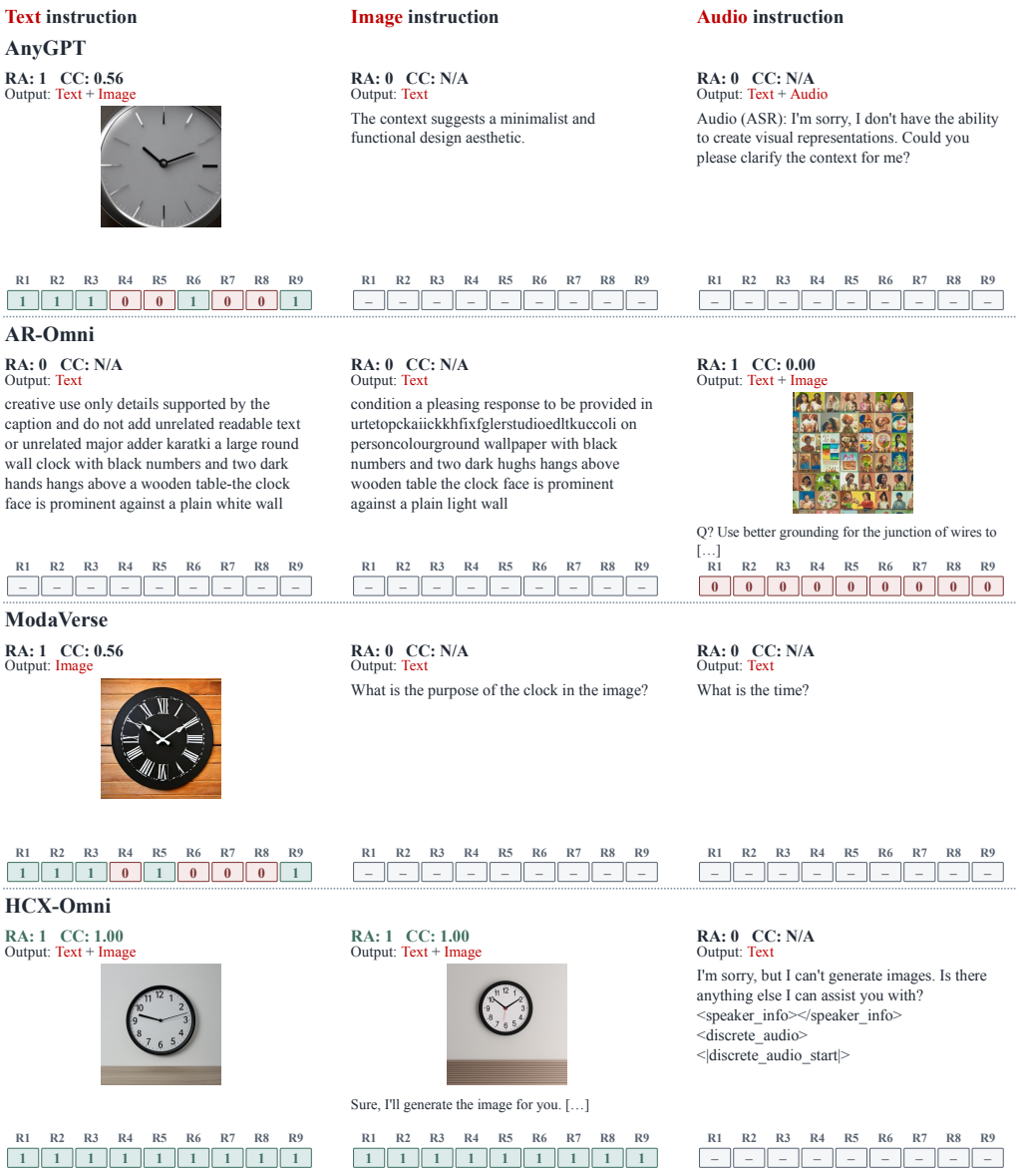


Figure A10: **Clock generation.** Columns show text, image, and audio instructions; rows show AnyGPT, AR-Omni, ModaVerse, and HCX-Omni. The shared request and nine content rubrics are defined in Fig. A9. CC is N/A when the requested image is not delivered; generated speech is represented by ASR transcripts.