# VisLingInstruct: Elevating Zero-Shot Learning in Multi-Modal Language Models with Autonomous Instruction Optimization

**Anonymous ACL submission**

## Abstract

This paper presents VisLingInstruct, a novel approach to advancing Multi-Modal Language Models (MMLMs) in zero-shot learning. Current MMLMs show impressive zero-shot abilities in multi-modal tasks, but their performance depends heavily on the quality of instructions. VisLingInstruct tackles this by autonomously evaluating and optimizing instructional texts through In-Context Learning, improving the synergy between visual perception and linguistic expression in MMLMs. Alongside this instructional advancement, we have also optimized the visual feature extraction modules in MMLMs, further augmenting their responsiveness to textual cues. Our comprehensive experiments on MMLMs, based on FlanT5 and Vicuna, show that VisLingInstruct significantly improves zero-shot performance in visual multi-modal tasks. Notably, it achieves a 15.9% increase in accuracy over the prior state-of-the-art on the ScienceQA dataset.

## 1 Introduction

The integration of Large Language Models (LLMs) with vision and multi-modality, epitomized by models like BLIP-2 (Chen et al., 2022; Alayrac et al., 2022; Li et al., 2023), has marked a significant evolution in the Natural Language Processing (NLP) field. This advancement led to the emergence of Multi-Modal Language Models (MMLMs), blending visual and linguistic data processing to enhance complex multimodal information understanding and generation. InstructBLIP (Dai et al., 2023), a notable example, utilizes advanced instruction tuning for image-text pairs, significantly improving the Q-Former module's zero-shot learning capabilities in a variety of vision-language multi-modal tasks. This progression underscores the potential of MMLMs in navigating the intricacies of multi-modal data, setting a new benchmark in the intersection of language, vision, and machine learning.

However, the effectiveness of MMLMs is highly constrained by the quality of textual instructions. Current instruction-tuned models (Ouyang et al., 2022; Zheng et al., 2023b), effective as they may be, while potentially effective, introduces significant challenges, particularly for the users lacking expertise in crafting optimal instructions. The limitation leads to inconsistent and sometimes suboptimal outputs, thus impeding the practical utility of MMLMs in the real world scenarios. To mitigate this issue, we propose a novel autonomous optimization method for textual instruction, named **Vis**ual, **Ling**uistic, **Instruct**ion optimization (VisLingInstruct). VisLingInstruct introduces an innovative method through In-Context Learning (ICL) (Min et al., 2022) based on the comparison between instruction cases, using MMLMs' linguistic capabilities to autonomously enhance and evaluate textual instruction. This method can guide the model towards generating more effective and contextually appropriate instructions.

Complementing our instructional optimization strategy, we present an architectural innovation aimed at enhancing the alignment between visual and textual modules within MMLMs. Inspired by recent advancements in models such as MiniGPT4 (Zhu et al., 2023), LLaVA (Liu et al., 2023b), mPLUG-Owl (Ye et al., 2023), and BLIVA (Hu et al., 2023), our architecture enhances the integration of textual and visual data. This integrative approach enables MMLMs to more effectively process and interpret complex tasks that require an understanding of both textual and visual elements, thereby improving accuracy and contextual understanding. Figure 1 offers a visual comparison of the alignment modules in different MMLMs, highlighting the distinctive features and benefits of our proposed method. Through this architectural enhancement, we aim to bridge the existing gaps in multi-modal data processing, creating a more cohesive and efficient model capable of tackling the



*Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2122–2135

nuanced demands of multi-modal interactions.

In summary, our contributions are as follows:

- We introduce substantial architectural improvements for better integration of multi-modal data within MMLMs during training (Section 3.1).
- We present an autonomous method for optimizing instruction quality, tailored to improve the effectiveness of textual instruction during inference (Section 3.2).
- We conduct comprehensive experiments and ablation studies to demonstrate the effectiveness of VisLingInstruct and the success of each component. Notably, VisLingInstruct has improved the performance by a significant margin of 15.9% on the ScienceQA dataset.

## 2 Related Work

### 2.1 Instruction Tuning in MMLMs

Instruction tuning has emerged as a cost-effective alternative to the expensive pre-training of large models, focusing on fine-tuning a few foundational models for downstream tasks. In this context, models like InstructGPT (Ouyang et al., 2022), Flan-T5 (Chung et al., 2022), and Vicuna (Zheng et al., 2023b) represent significant strides in conversational models obtained through instruction tuning based on LLMs. These models have showcased exceptional question-answering capabilities, underscoring the importance of instruction-based approaches in language generation. In the multi-modal domain, advancements such as Mini-GPT4 (Zhu et al., 2023), LLaVA (Liu et al., 2023b), mPLUG-Owl (Ye et al., 2023), InstructBLIP (Dai et al., 2023), and BLIVA (Hu et al., 2023) have focused on instruction fine-tuning. These methods typically involve aligning images and text by introducing transitional layers, like Q-Former and fully connected layers, between visual encoders and LLMs. Our work builds upon these foundations, aiming to further optimize the instruction tuning process for enhanced performance in MMLMs.

### 2.2 Optimizing Instructions for Large Models

Historically, models akin to BERT (Kenton and Toutanova, 2019) have utilized prompt crafting techniques (Brown et al., 2020) to boost performance, with subsequent research exploring methods to discover higher-quality prompts (Gao et al.,

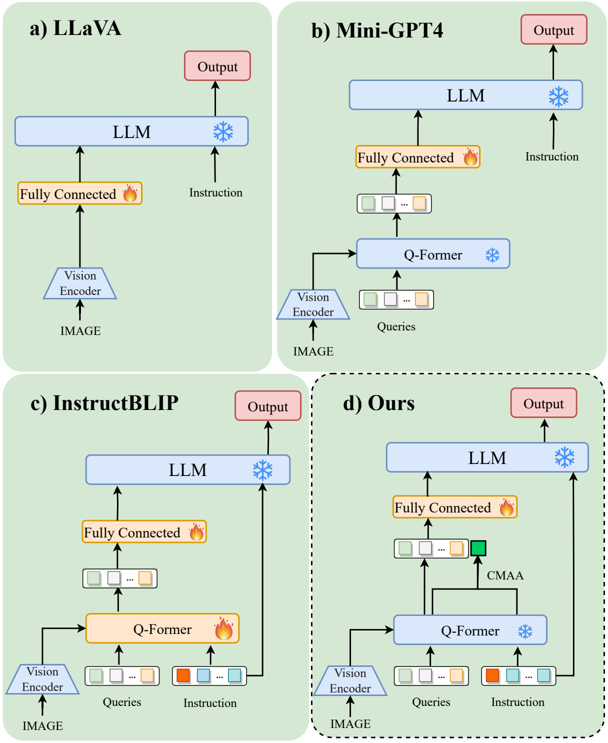


Figure 1: The structural comparison among the alignment modules of different MMLMs. The orange modules in the figure represent open weights, while the blue modules indicate frozen weights.

2021; Lu et al., 2023). In generative models, this concept has evolved into optimizing 'instructions', leading to a series of works focused on prompt and instruction optimization (Wei et al., 2022; Min et al., 2022). Notably, UPRISE (Cheng et al., 2023) trained a prompt retriever for acquiring superior instructions, while OPRO (Yang et al., 2023) conceptualized LLMs as optimizers, formulating optimization tasks in natural language. Zheng et al. introduced STEP-BACK prompting, enabling LLMs to derive higher-level concepts from detailed instances. To the best of our knowledge, we spearhead the manual-free optimization of textual instruction in zero-shot manner for a wide range of multi-modal tasks.

## 3 Methods

Our approach comprises two components: First, we refine the architecture of existing multi-modal models and their fine-tuning mechanisms to augment their perceptivity of instruction, that is, the Enhanced Multi-modal Alignment (EMA). Second, subsequent to the model's fine-tuning, we concentrate on the autonomous optimization of instructions during the inference, referred to as the Au-

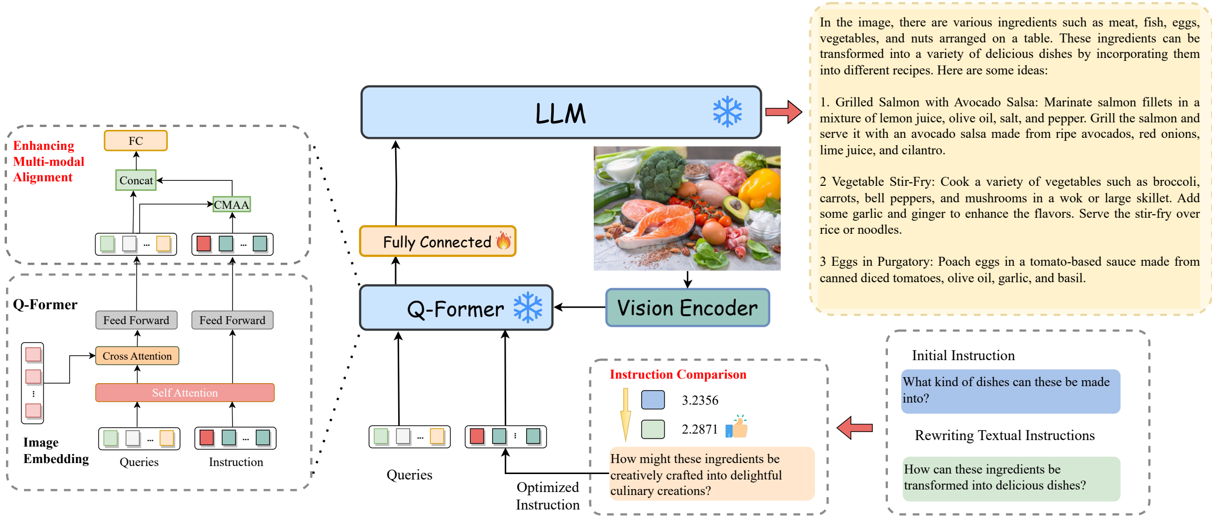


Figure 2: The figure depicts the complete workflow of Instruction Comparison Optimization. Initial and rewritten instructions are processed through comparison optimization to generate optimized instructions. Subsequently, the optimized instructions are utilized for generation in MMLMs.

tonomous Instruction Optimization (AIO).

### 3.1 Enhancing Multi-modal Alignment

In the quest to refine MMLMs, our focus shifts to bridging the gap between the realms of visual perception and linguistic expression. This section delves into our pioneering approach to enhancing the alignment between visual and textual modules within MMLMs, introducing a series of architectural innovations and training optimizations designed to synergize these two distinct modalities seamlessly.

**Integrative Processing of Text and Image:** At the core of our architectural enhancements is the integrative processing of textual and visual data. This process involves constructing a unified representation by merging detailed textual embeddings with rich visual information. We introduce the Cross-Modal Alignment Attention (CMAA) algorithm to achieve this integration, specifically designed to harmonize these disparate data modalities. This algorithm leverages attention mechanisms and cross-modal feature fusion, to ensure that the resulting multi-modal representation encapsulates both the intricacies of language and the finer details of visual content:

$$U_{mm} = \sum_{i=1}^{N} \text{softmax}(\text{emb}_{\text{que}} \cdot \text{emb}_{\text{text}}^{T}) \cdot \text{emb}_{\text{text}}(i) \quad (1)$$

where $\text{emb}_{\text{text}}(i)$ and $\text{emb}_{\text{vis}}(i)$ represent the embedding of the textual instruction and Queries for the $i$-th element respectively. Simultaneously, $\text{emb}_{\text{text}}(i)$ serves as both the key (K) and value (V) in traditional attention mechanism, while $\text{emb}_{\text{vis}}(i)$ functions as the query (Q). The textual instruction, after undergoing CMAA, transforms into $U_{mm}$. Subsequently, $U_{mm}$ concatenate onto the output of Queries in the form of Figure 1, culminating in the final integration of visual and textual elements.

**Optimized Model Training and Performance:** In developing this new architecture, our approach extends beyond mere technical integration to encompass strategic training and performance optimization. We employ selective weight freezing methods, where specific layers of the pre-trained model are kept static to preserve learned features, and targeted fine-tuning, where newly introduced components or layers are specifically trained to adapt to the task at hand. This targeted approach allows us to fine-tune the model's performance without the need for extensive retraining, thereby enhancing the learning efficiency and ensuring the robustness and scalability of the model. The loss function used for training takes the following form:

$$p(Y_{text}|X_{img}) = \prod_{i=1}^{L} p_\theta(y_i|X_{img}, Y_{text}^{[1:i-1]}) \quad (2)$$

where $\theta$ is the trainable parameters, $X_{img}$ and $Y_{text}$ respectively denote the input image and the output text, $Y_{text}^{[1:i-1]}$ represents the input instruction and the text already generated up to the $i-1$ step.

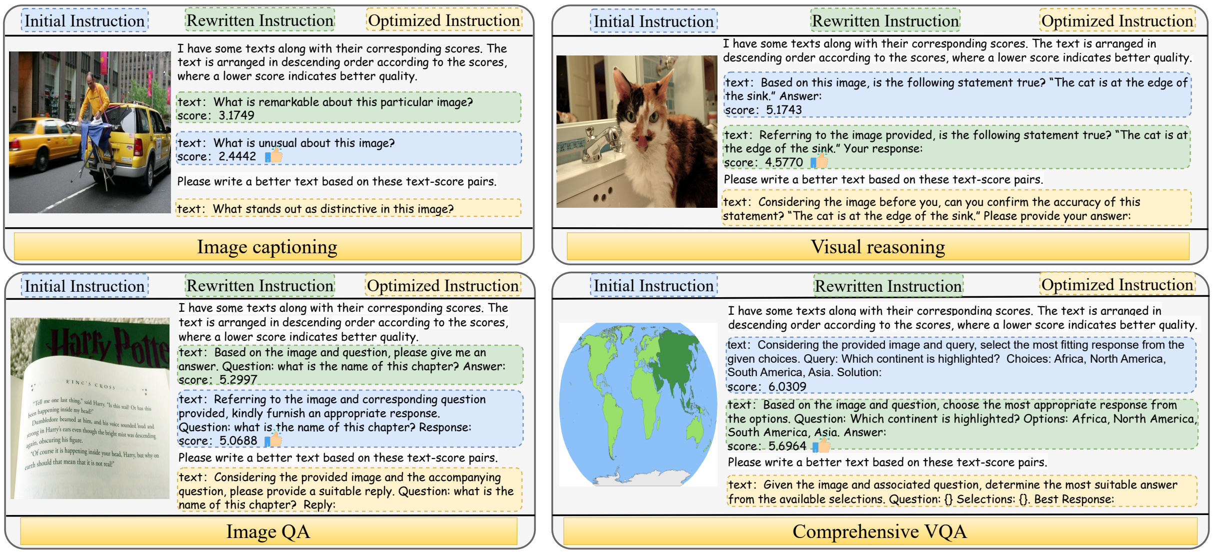


Figure 3: The examples of IAS ranking in different domains. On the left side is the image input provided to the MMLMs. On the right side, within the blue box, lies the initial instruction, while the rewritten instruction is contained within the green box. The 'score' indicates the quality of corresponding instructions with respect to the model, while the lower score (i.e., high quality) instruction always ranks later in the ICL demonstration. By utilizing the paradigm of ICL, MMLMs learn the relationship between the scores of the two cases to generate higher-quality new instructions that lie in the yellow box.

### 3.2 Autonomous Instruction Optimization

During inference, the textual instruction has a significant impact on the generation results of MMLMs. Therefore, we propose an approach that leverages the inherent text processing capabilities of LLMs to optimize textual instructions, thereby aligning the results generated by MMLMs more closely with user requirements. This method comprises two stages: Rewriting Textual Instructions and Instruction Alignment Optimization.

**Rewriting Textual Instructions:** Rewriting is the first stage in our methodology. LLMs exhibit strong text rewriting capabilities, maintaining semantic information and changing the content of the text. Therefore, our goal is to use the ability of LLM to rewrite the initial textual instructions, aiming to obtain a pair of instruction with roughly similar semantics to lay the foundation for the second stage. Furthermore, the rewritten instruction resulting from this process is not necessarily required to be of higher quality than the initial instruction. As long as there is a difference between the two, it is enough to meet the requirements of subsequent processes. This reduces the complexity of the rewrite task, making the barrier to implementation relatively low.

Specifically, we have intricately designed a prompt tailored for LLMs to rewrite textual instruction. The prompt should explicitly instruct LLMs on how to rewrite the textual instruction while ensuring minimal semantic changes between the initial and rewritten version. The template of the prompt used in this stage can be referred to in the Appendix B. An important consideration to note is that since this stage solely involves rewriting instructions, the entire MMLMs aren't required. Involving only LLMs could slightly reduce the time consumption caused by the rewriting process.

**Instruction Comparison Optimization:** At this stage, we design a method that enables MMLMs to identify which instruction is better through comparative analysis and strive to generate instruction with better quality. As illustrated in Figure 2, we innovatively rank the cases in ICL so that the model can learn the quality of instructions only through the comparison between initial instruction and rewritten instruction (Ren and Liu, 2023).

Given that the ultimate goal of the instruction is to facilitate inference by MMLMs, we believe that the quality of these instructions should be evaluated by the MMLMs themselves. Specifically, we allow MMLMs to score the instruction solely by the itself without the help of external discriminator. Therefore, we proposed the Instruction Alignment Score (IAS), which is formulated to quantify the divergence between the model's predicted output and the expected output for a given instruction. We score the instruction by using a prompt to guide MMLMs. Please see the Appendix C for the prompt template. Defined as the expectation of negative log-likelihood, IAS is calculated as follows:

$$\mathrm{IAS} = \mathbb{E}[-\log P(t_i|\mathrm{X}_{img}, \mathrm{X}_{prompt}, t_{[1:i-1]}; \theta)] \quad (3)$$

Here, $\mathrm{X}_{img}$ is the input image, $\mathrm{X}_{prompt}$ denotes textual prompt employed to guide the model in its computational processes, $\theta$ represents our MMLMs model and $t_i$ represents the textual tokens that need to be generated, which is the instruction whose quality MMLMs is tasked to evaluate. A lower IAS indicates a higher alignment of the instruction with the model's understanding, enabling MMLMs to perform better. After calculating IAS, as shown in Figure 3, we rank the instruction-IAS pairs in descending order, and combine them into a prompt in the form of ICL to input to MMLMs to generate a optimized instruction. The optimized instruction will have better inference performance compared to the initial and rewritten instructions.

# 4 Experiments

## 4.1 Datasets

The datasets in this paper primarily consists of a training dataset and the zero-shot evaluation benchmarks. The training data is sourced from LLaVA, which is also a subset of the InstructBLIP training datasets. The data was collected by the authors of LLaVA using ChatGPT/GPT-4 (OpenAI, 2023a,b), following a multi-modal instruction format. We believe that using the same dataset as previous work enables a fairer comparison in the experiments. For an image $X_v$, there is an associated question-answer pair <$X_q$, $X_a$> related to $X_v$. In some cases, there are multi-turn dialogues represented as (<$X_q^1$, $X_a^1$>,...,<$X_q^m$, $X_a^m$>). During training, for single-turn dialogue data, $X_q$ serves as the initial instruction, while $X_a$ corresponds to the ground truth. Likewise, for multi-turn dialogue data, it is essential to concatenate the historical dialogues (excluding the last turn) and append them along with $X_q^m$ as the initial instruction. Meanwhile, $X_a^m$ serves as the ground truth.

For zero-shot evaluation benchmarks, to ensure alignment for comparison, we also follow InstructBLIP. The evaluation domains include: Image captioning: Flickr30K (Young et al., 2014), Nocaps (Agrawal et al., 2019). Visual Reasoning: VSR (Liu et al., 2023a), GQA (Hudson and Manning, 2019), IconQA (Lu et al., 2021). Image QA: VizWiz (Gurari et al., 2018), TextVQA (Mishra et al., 2019). Comprehensive VQA: Visual Dialog (Das et al., 2017), ScienceQA (Lu et al., 2022), HatefulMemes (Kiela et al., 2020). It's important to note that for ScienceQA, we only evaluate the set with image context. The utilization of the overall evaluation benchmarks can be referenced in Appendix D. The evaluation metrics vary across benchmarks: NoCaps and Flickr30K employ CIDEr scores (Vedantam et al., 2015), HatefulMemes utilizes AUC scores, and Visual Dialog employs Mean Reciprocal Rank (MRR). For all remaining datasets, top-1 accuracy is used as the metric. All evaluation benchmarks have no data overlap with the training set, ensuring the authenticity of zero-shot. In the Appendix F, we provide the initial instructions used for all benchmarks in zero-shot learning.

## 4.2 Implementation details

In terms of the model architecture, we opted for the ViT-G/14 from EVA-CLIP (Fang et al., 2023) as the visual encoder, removing the final layer of the ViT and utilizing the output features from the penultimate layer. In line with InstructBLIP, we employed two distinct LLMs: FlanT5 and Vicuna. FlanT5, derived from the instruction-tuning of the encoder-decoder Transformer T5 (Raffel et al., 2020), encompasses two sizes: FlanT5-XL and FlanT5-XXL. Vicuna, on the other hand, is refined from the instruction-tuning of the decoder-only Transformer LLaMA (Touvron et al., 2023), and also includes two sizes: Vicuna-7B and Vicuna-13B. The weights of both Q-Former and the fully connected layers are sourced from InstructBLIP and need to correspond to different LLMs. Our entire model framework requires freezing the weights of the visual encoder, Q-Former, and LLMs, allowing only the fully connected layers to be unfrozen. Further details regarding training hyperparameters can be found in Appendix E.

## 4.3 Zero-shot Evaluation

During the evaluation process, we employed two different generation methods tailored to different benchmarks. For the domain of benchmarks such as Image Captioning, results were directly generated from instructions. These results were then compared against ground truth to calculate metrics. On the other hand, for classification-based VQA

| | Image Captioning | | Visual Reasoning | | | Image QA | | Comprehensive VQA | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Flickr30K | Nocaps | VSR | GQA | IconQA | VizWiz | TextVQA | Visdial | SciQA | HM |
| BLIP-2 (FlanT5$_{\mathrm{XXL}}$) | 73.7 | 104.5 | 68.2 | 44.6 | 45.4 | 29.4 | 44.1 | 46.9 | 64.5 | 52.0 |
| BLIP-2 (Vicuna$_{13\mathrm{B}}$) | 74.9 | 107.5 | 50.9 | 41.0 | 40.6 | 19.6 | 42.5 | 45.1 | 61.0 | 53.7 |
| MiniGPT-4 (Vicuna$_{13\mathrm{B}}$) | / | / | 50.7 | 30.8 | 37.6 | 34.8 | 18.7 | / | / | 29.0 |
| LLaVA (Vicuna$_{13\mathrm{B}}$) | / | / | 56.3 | 41.3 | 43.0 | 37.7 | 28.3 | / | / | 9.2 |
| InstructBLIP (FlanT5$_{\mathrm{XL}}$) | 84.5 | 119.9 | 64.8 | 48.4 | 50.0 | 32.7 | 46.6 | 46.6 | 70.4 | 56.6 |
| InstructBLIP (FlanT5$_{\mathrm{XXL}}$) | 83.5 | 120.0 | 65.6 | 47.9 | 51.2 | 30.9 | 46.6 | 48.5 | 70.6 | 54.1 |
| InstructBLIP (Vicuna$_{7\mathrm{B}}$) | 82.4 | 123.1 | 54.3 | 49.2 | 43.1 | 34.5 | 50.1 | 45.2 | 60.5 | 59.6 |
| InstructBLIP (Vicuna$_{13\mathrm{B}}$) | 82.8 | 121.9 | 52.1 | 49.5 | 44.8 | 33.4 | 50.7 | 45.4 | 63.1 | 57.5 |
| BLIVA (Vicuna$_{13\mathrm{B}}$) | 87.1 | / | 62.2 | / | 44.9 | 42.9 | 58.0 | 45.6 | / | 55.6 |
| BLIVA (FlanT5$_{\mathrm{XXL}}$) | 87.7 | / | **68.8** | / | **52.4** | 44.0 | 57.2 | 36.2 | / | 50.0 |
| Ours(FlanT5$_{\mathrm{XL}}$) | 85.3 | 119.5 | 64.1 | 47.9 | 50.4 | 33.0 | 48.7 | 47.0 | 71.0 | 60.0 |
| Ours(FlanT5$_{\mathrm{XXL}}$) | **88.5** | 120.4 | 66.9 | 48.1 | 51.2 | 31.3 | 48.8 | **49.2** | **81.8** | 55.7 |
| Ours(Vicuna$_{7\mathrm{B}}$) | 87.9 | **124.2** | 60.1 | 52.0 | 44.2 | 42.7 | 60.6 | 45.7 | 74.6 | **62.7** |
| Ours(Vicuna$_{13\mathrm{B}}$) | 84.0 | 119.8 | 56.2 | **52.9** | 50.3 | **45.0** | **65.6** | 45.7 | 71.0 | 58.9 |

Table 1: Zero-shot results on general VQA benchmarks. Here, Visdial, SciQA, and HM respectively refer to Visual Dialog, ScienceQA, and HatefulMemes. The results for MiniGPT-4 and LLaVA are sourced from BLIVA (Hu et al., 2023), while the remaining results originate from their respective papers (Li et al., 2023; Dai et al., 2023).

tasks, we followed previous work (Alayrac et al., 2022; Dai et al., 2023) by computing the language model loss for each candidate option and selecting the one with the lowest loss as the final prediction. This method was applied to ScienceQA, IconQA, HatefulMemes, and Visual Dialog.

We conducted zero-shot learning of our model against previous state-of-the-art works across 10 benchmarks in Table 1. It's evident that our model showcases a significant advantage across the majority of benchmarks, especially in Image QA and Comprehensive VQA domains. At the same time, due to the primary inheritance of our model weights from InstructBLIP, a horizontal comparison with InstructBLIP reveals that our method significantly strengthens the generative capability of MMLMs. For instance, with model based on FlanT5-XXL, our approach exhibits a comparative increase of 6.0% and 15.9% in performance over InstructBLIP concerning Flickr30K and ScienceQA, respectively. These results demonstrate that the instruction optimization approach we proposed exhibits highly favorable gains for multi-modal tasks in the domain of image-text.

### 4.4 Ablation study

To investigate the impact of Enhancing Multi-modal Alignment (EMA, Section 3.1) and Autonomous Instruction Optimization (AIO, Section 3.2) on the final results, we conducted ablation studies by individually removing them during evaluation.

As depicted in Table 2, after integrating the EMA mechanism on the vanilla baseline, the overall performance of all models is significantly enhanced. This indicates that our EMA method indeed enhances the alignment between images and text. Moreover, if AIO continues to be integrated on the basis of EMA, the evaluation results can be further improved. This adequately shows that the two mechanisms can strengthen each other. EMA, by enhancing its perception of instructions, can serve as a booster to further enhance AIO.

As for the AIO part, we also further split it to conduct ablation experiments. We discuss Rewriting Textual Instructions and Instruction Comparison Optimization separately. It can be clearly seen from the results in Table 2 that instruction rewriting cannot continue to improve the effect on the basis of EMA. On the contrary, it is even inferior to the vanilla baseline in many results. This phenomenon fully demonstrates that just rewriting cannot stably optimize the instruction, and requires correction by our Instruction Comparison Optimization mechanism.

### 4.5 Qualitative evaluation

Beyond the benchmarks-driven experimental analyses, we diversified our qualitative evaluation by incorporating real-world images and instructions. As shown in Figure 4, we have enumerated three cases for comprehensive analysis. The process


Figure 4: The one on the left is a case written for a product advertisement, the one in the middle is a recipe description, and the one on the right is a poetry creation. Qualitative comparison of three responses from different ablations: initial instruction with vanilla model (blue), initial instruction with EMA model (purple), and optimized instruction with EMA model (green).

| Vanilla | EMA | AIO | | Image Captioning | | Visual Reasoning | | | Image QA | | Comprehensive VQA | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Rewriting | Comparison | Flickr30K | Nocaps | VSR | GQA | IconQA | VizWiz | TextVQA | Visdial | SciQA | HM |
| | | | | ***FlaxT5-XL*** | | | | | | | | | |
| ✓ | | | | 84.5 | **119.9** | 64.8 | 48.4 | 50.0 | 32.7 | 46.6 | 46.6 | 70.4 | 56.6 |
| ✓ | ✓ | | | 85.1 | 119.7 | 63.5 | **48.6** | 50.0 | 32.8 | 48.5 | 46.9 | 70.6 | **60.8** |
| ✓ | ✓ | ✓ | | 84.7 | 118.1 | **66.8** | 48.5 | 49.0 | 31.8 | 47.5 | 44.8 | 70.4 | 57.3 |
| ✓ | ✓ | ✓ | ✓ | **85.3** | 119.5 | 64.1 | 47.9 | **50.4** | **33.**0 | **48.7** | **47.0** | **71.0** | 60.0 |
| | | | | ***FlaxT5-XXL*** | | | | | | | | | |
| ✓ | | | | 83.5 | 120.0 | 65.6 | 47.9 | 51.2 | 30.9 | 46.6 | 48.5 | 70.6 | 54.1 |
| ✓ | ✓ | | | 86.3 | 120.3 | 55.7 | 48.0 | **51.6** | **31.5** | 48.3 | 49.0 | **82.0** | 55.2 |
| ✓ | ✓ | ✓ | | 85.3 | 120.1 | 66.5 | 48.1 | 50.9 | 31.1 | 46.7 | 48.5 | 73.5 | 54.1 |
| ✓ | ✓ | ✓ | ✓ | **88.5** | **120.4** | **66.9** | **48.3** | 51.2 | 31.3 | **48.8** | **49.2** | 81.8 | **55.7** |
| | | | | ***Vicuna-7B*** | | | | | | | | | |
| ✓ | | | | 82.4 | 123.1 | 54.3 | 49.2 | 43.1 | 34.5 | 50.1 | 45.2 | 60.5 | 59.6 |
| ✓ | ✓ | | | 81.6 | 124.5 | **60.6** | 51.9 | 43.2 | 40.5 | 49.9 | 45.3 | 55.4 | 60.8 |
| ✓ | ✓ | ✓ | | 82.3 | **124.5** | 55.4 | 47.6 | 44.0 | 40.3 | 58.3 | 43.4 | 63.0 | 62.2 |
| ✓ | ✓ | ✓ | ✓ | **87.9** | 124.2 | 60.1 | **52.0** | **44.2** | **42.7** | **60.6** | **45.7** | **74.6** | **62.7** |
| | | | | ***Vicuna-13B*** | | | | | | | | | |
| ✓ | | | | 82.8 | **121.9** | 52.1 | 49.5 | 44.8 | 33.4 | 50.7 | 45.4 | 63.1 | 57.5 |
| ✓ | ✓ | | | 84.4 | 120.2 | **58.9** | 51.6 | 48.4 | 43.0 | 56.9 | 43.0 | 48.4 | **61.0** |
| ✓ | ✓ | ✓ | | 80.4 | 120.6 | 52.5 | 51.1 | 49.3 | 41.5 | 62.4 | 44.4 | 68.0 | 58.7 |
| ✓ | ✓ | ✓ | ✓ | **84.0** | 120.8 | 56.2 | **52.9** | **50.3** | **45.0** | **65.6** | **45.7** | **71.0** | 58.9 |

Table 2: Results of ablation studies for Enhancing Multi-modal Alignment (EMA) and Autonomous Instruction Optimization (AIO) in different LLMs models. Among them, EMA is split into Rewriting Textual Instructions (Rewriting) and Instruction Comparison Optimization (Comparison) for discussion respectively. Vanilla represents the baseline model without any of our proposed modules and ✓indicates that the module has been integrated.

commences with the input of an image, subsequent questions and answers revolve around this visual context. This is followed by the presentation of instructions, encompassing both the initial instructions and the optimized by the AIO module. Conclusively, model response is delineated. The output section for evaluation includes: the results obtained by inputting the initial instructions into the vanilla model (Vanilla Response); the results obtained by inputting the initial instructions into the integrated EMA module model (EMA Response); and the results from inputting the optimized instructions into the integrated EMA module model (EMA & AIO Response), which is VisLingInstruct.

The outcome as observed in the figure suggests that the EMA Response demonstrates an improvement over the Vanilla Response, both in terms of content accuracy and richness of detail. For instance, within the case of poetry creation, the erroneously presented '3 huts' is accurately identified as 'a small house'. In the case of recipe description, the narrative about spaghetti is much more detailed in the EMA Response. Furthermore, the EMA & AIO response also surpasses the EMA response alone, evident in the former's answers possessing a superior logical organization and better fulfillment of user intent. This is well illustrated in all three cases presented in the figure. And for more on the performance in multi-turn dialogues, we have provided a demonstration and discussion in the Appendix G.

## 5 Conclusion

This paper proposes VisLingInstruct, a novel autonomous instruction optimization framework for visual-linguistic multi-modal models. We conducted a comprehensive study on multi-modal models and demonstrated the powerful autonomous instruction optimization capabilities of the VisLingInstruct model, demonstrating strong zero-shot learning capabilities in a series of benchmarks. At the end of the experiment, qualitative examples were used to demonstrate the specific situation of VisLingInstruct in autonomous instruction optimization, such as knowledge-based image description, image-based text creation and multi-turn dialogue. We hope that VisLingInstruct can inspire more new research on autonomous optimization of multi-modal instruction.

## Limitations

Limitations of the current work are discussed below. Firstly, while the proposed multimodal autonomous instruction optimization framework performs well in terms of effectiveness, its structure is relatively intricate. Streamlining the process while maintaining effectiveness would greatly facilitate the practical application of this technology. Secondly, the experiments in this paper are focused on image and text modalities, and further validation is needed to determine the effectiveness of our framework in other modalities, such as video and audio.

## References


Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. 2019. Nocaps: Novel object captioning at scale. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8948–8957.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. 2022. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. 2022. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*.

Daixuan Cheng, Shaohan Huang, Junyu Bi, Yuefeng Zhan, Jianfeng Liu, Yujing Wang, Hao Sun, Furu Wei, Denvy Deng, and Qi Zhang. 2023. Uprise: Universal prompt retrieval for improving zero-shot evaluation. *arXiv preprint arXiv:2303.08518*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. *arXiv preprint arXiv:2305.06500*.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. 2017. Visual dialog. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 326–335.

Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. 2023. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19358–19369.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL-IJCNLP 2021*, pages 3816–3830.

Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. 2018. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3608–3617.

Wenbo Hu, Yifan Xu, Y Li, W Li, Z Chen, and Z Tu. 2023. Bliva: A simple multimodal llm for better handling of text-rich visual questions. *arXiv preprint arXiv:2308.09936*.

Drew A Hudson and Christopher D Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. 2020. The hateful memes challenge: Detecting hate speech in multimodal memes. *Advances in neural information processing systems*, 33:2611–2624.

Dongxu Li, Junnan Li, Hung Le, Guangsen Wang, Silvio Savarese, and Steven CH Hoi. 2022. Lavis: A library for language-vision intelligence. *arXiv preprint arXiv:2209.09019*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*.

Fangyu Liu, Guy Emerson, and Nigel Collier. 2023a. Visual spatial reasoning. *Transactions of the Association for Computational Linguistics*, 11:635–651.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023b. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Jinghui Lu, Dongsheng Zhu, Weidong Han, Rui Zhao, Brian Mac Namee, and Fei Tan. 2023. What makes pre-trained language models better zero-shot learners? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pages 2288–2303.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.

Pan Lu, Liang Qiu, Jiaqi Chen, Tony Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu. 2021. Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning. *arXiv preprint arXiv:2110.13214*.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11048–11064.

Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. 2019. Ocr-vqa: Visual question answering by reading text in images. In *2019 international conference on document analysis and recognition (ICDAR)*, pages 947–952.

OpenAI. 2023a. Chatgpt. Technical report.

OpenAI. 2023b. Gpt-4. Technical report.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Ruifeng Ren and Yong Liu. 2023. In-context learning with transformer is really equivalent to a contrastive learning pattern. *arXiv preprint arXiv:2310.13220*.

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

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. 2023. Large language models as optimizers. *arXiv preprint arXiv:2309.03409*.

Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. 2023. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

Huaixiu Steven Zheng, Swaroop Mishra, Xinyun Chen, Heng-Tze Cheng, Ed H Chi, Quoc V Le, and Denny Zhou. 2023a. Take a step back: Evoking reasoning via abstraction in large language models. *arXiv preprint arXiv:2310.06117*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023b. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

# A Algorithm

## A.1 Algorithm Overview

The algorithmic core of our approach in VisLingInstruct is structured around two main processes: Cross-Modal Alignment Attention and Autonomous Instruction Optimization. The former process harmonizes the integration of text and image, while the latter refines the textual instructions for MMLMs.

## A.2 Cross-Modal Alignment Attention

The Cross-Modal Alignment Attention (CMAA) algorithm focuses on the integration of textual and visual embeddings, creating a unified text representation.

**Algorithm 1** Cross-Modal Alignment Attention

**Require:** Textual embeddings $E_{\text{text}}$, Queries embeddings $E_{\text{que}}$
**Ensure:** Unified multi-modal representation $U_{\text{mm}}$

Initialize cross-modal alignment mechanism
**for** each element $i$ in $E_{\text{text}}$ **do**
    Compute attention between $E_{\text{text}}(i)$ and $E_{\text{que}}$
    Assign attention weight on $E_{\text{text}}(i)$
**end for**
$U_{\text{mm}} \leftarrow$ Aggregate of aligned and weighted $E_{\text{text}}$ **return** $U_{\text{mm}}$

## A.3 Autonomous Instruction Optimization

The Autonomous Instruction Optimization (AIO) is designed to transform initial instruction into an optimized format.

**Algorithm 2** Autonomous Instruction Optimization

**Require:** Initial instructions $I_i$
**Ensure:** optimized instruction $I_{opt}$

Initialize autonomous instruction optimization
Rewriting the initial instruction $I_i$ to obtain $I_j$
Calculating the IAS for $I_i$ and $I_j$
Ranking the instruction-IAS pairs
$I_{refined} \leftarrow$ Constructing the prompt input for Instruction Comparison in MMLMs **return** $I_{refined}$

# B Instruction rewriting templates

Here is the template used for Instruction rewriting in this paper, where '{}' signifies the instruction that requires modification:

```
There is the text {}. Please modify the text to make it better while retaining the sentence structure and keywords.
```

# C IAS templates

In the following prompt template, {} is used to place instructions requiring MPG calculation.

```
<Image>Based on the image given, the most appropriate instruction should be: {}
```

# D Zero-shot evaluation datasets details

| Dataset Name | Part | count |
|---|---|---|
| Flickr30K | test | 1000 |
| NoCaps | val | 4500 |
| VSR | test | 1222 |
| GQA | test-dev | 12578 |
| IconQA | test | 6316 |
| VizWiz | test-dev | 4319 |
| TextVQA | val | 5000 |
| Visual Dialog | val | 2064 |
| ScienceQA | test | 2017 |
| HatefulMemes | val | 1040 |

Table 3: The selected part in all zero-shot evaluation benchmarks, and accompanied by specific data count.

# E Training details

We implement VisLingInstruct by LAVIS library (Li et al., 2022). We fine-tuned the fully connected layers for 3 epochs, employing different hyperparameters across distinct LLMs. We employ a batch size of 32, 128 and 256 for the Vicuna-7B/13B, FlanT5-XL and FlanT5-XXL, respectively. For each model, we conduct validation every 1K steps. A consistent aspect across our training procedures was the utilization of the AdamW (Loshchilov and Hutter, 2018) optimizer with $\beta_1$ = 0.9, $\beta_2$ = 0.999, and a weight decay of 0.05. Furthermore, we implemented a linear warm-up of the learning rate over the initial 1K steps, escalating from $10^{-}8$ to $10^{-}5$, followed by cosine decay towards a minimum learning rate of 0.

All our model's trainable parameter counts are maintained within the range of a few million. Under the conditions of 8 A100 40G, the training durations for FlanT5, Vicuna 7B, and Vicuna 13B

| Dataset | Initial instruction |
|---|---|
| Flickr30K/ NoCaps | <Image>A short image description: |
| VSR | <Image>Based on the image, is this statement true or false? {} |
| GQA/ Visdial | <Image>Question: {} \n Short answer: |
| IconQA | <Image>Question: {} Options: {} \n Answer: |
| VizWiz | <Image>Answer the question based on the image. Reply in one phrase/word or say 'unanswerable'. Question: {} \n Short answer: |
| TextVQA | <Image>OCR tokens: {} Question: \n Short answer: |
| SciQA | <Image>Given the image, choose the correct option for the following question. Question: {} \n Options: {} |
| HM | <Image>This is an image with: {} written on it. Is it hateful? |

Table 4: Presentation of initial instructions for each benchmark.

are 105 minutes, 135 minutes, and 210 minutes, respectively.

## F Instructions in benchmarks

Table 4 illustrates the initial instructions for all benchmarks. The initial instructions were predominantly sourced from InstructBLIP. '{}' contains entities such as questions from each individual case. For instructions with options, we separate the choices alphabetically, for instance: (a) apple (b) banana (c) pineapple.

## G Multi-turn dialogues case studies

As illustrated in Figures 5 and 6, we present additional cases regarding multi-turn dialogue to showcase the strengths and distinctive features of our approach. The multi-turn generation results on the right originate from the models incorporating the EMA and AIO proposed in this paper (VisLingInstruct), while those on the left exclude this aspect, representing the results from Vanilla Response.

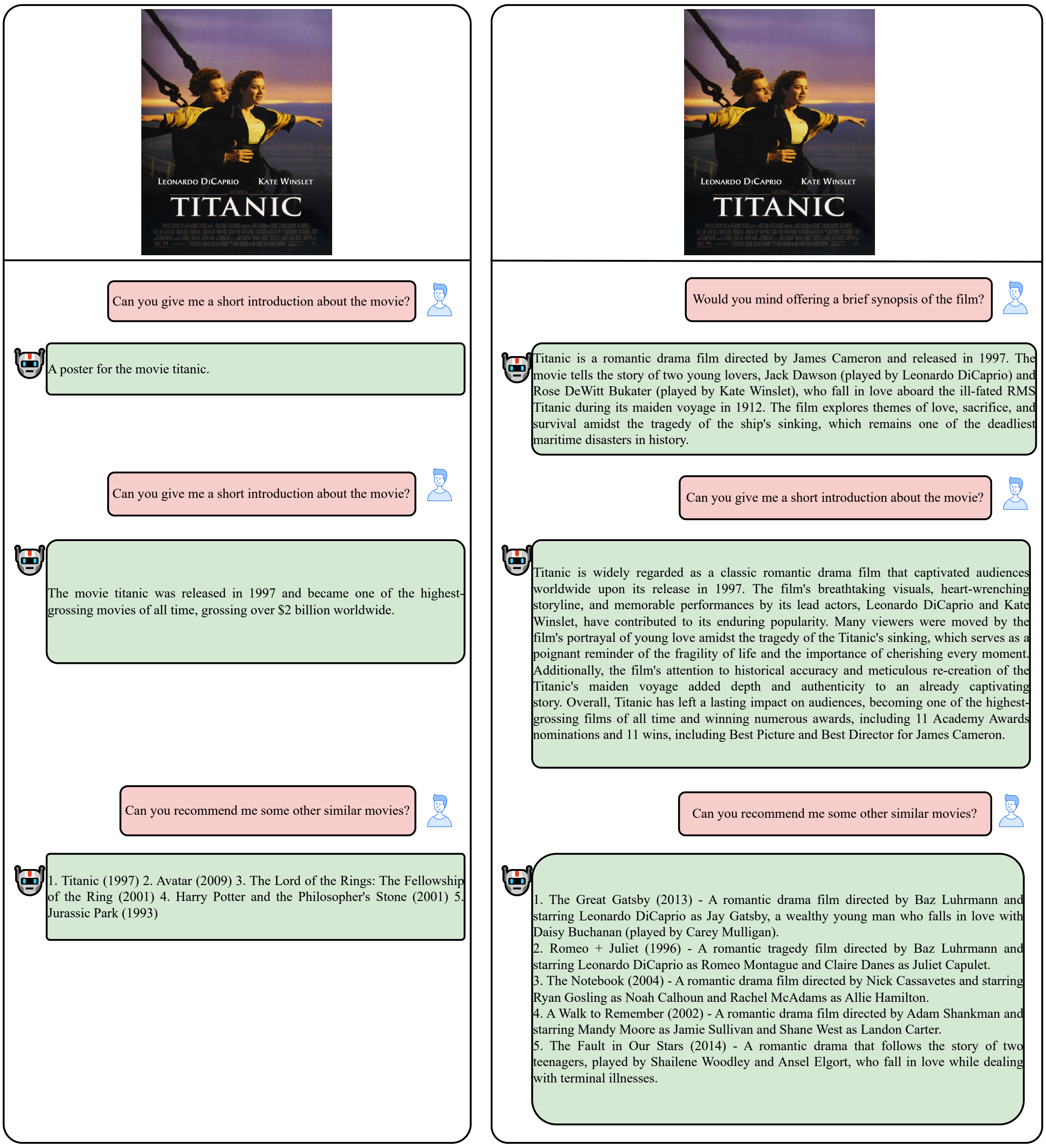


Figure 5: This multi-turn dialogue example demonstrates the effectiveness of VisLingInstruct in movie introduction and recommendation. Our generated results (right) exhibit significantly improved accuracy in identifying instructional content and richness in answers compared to the vanilla model's results (left).

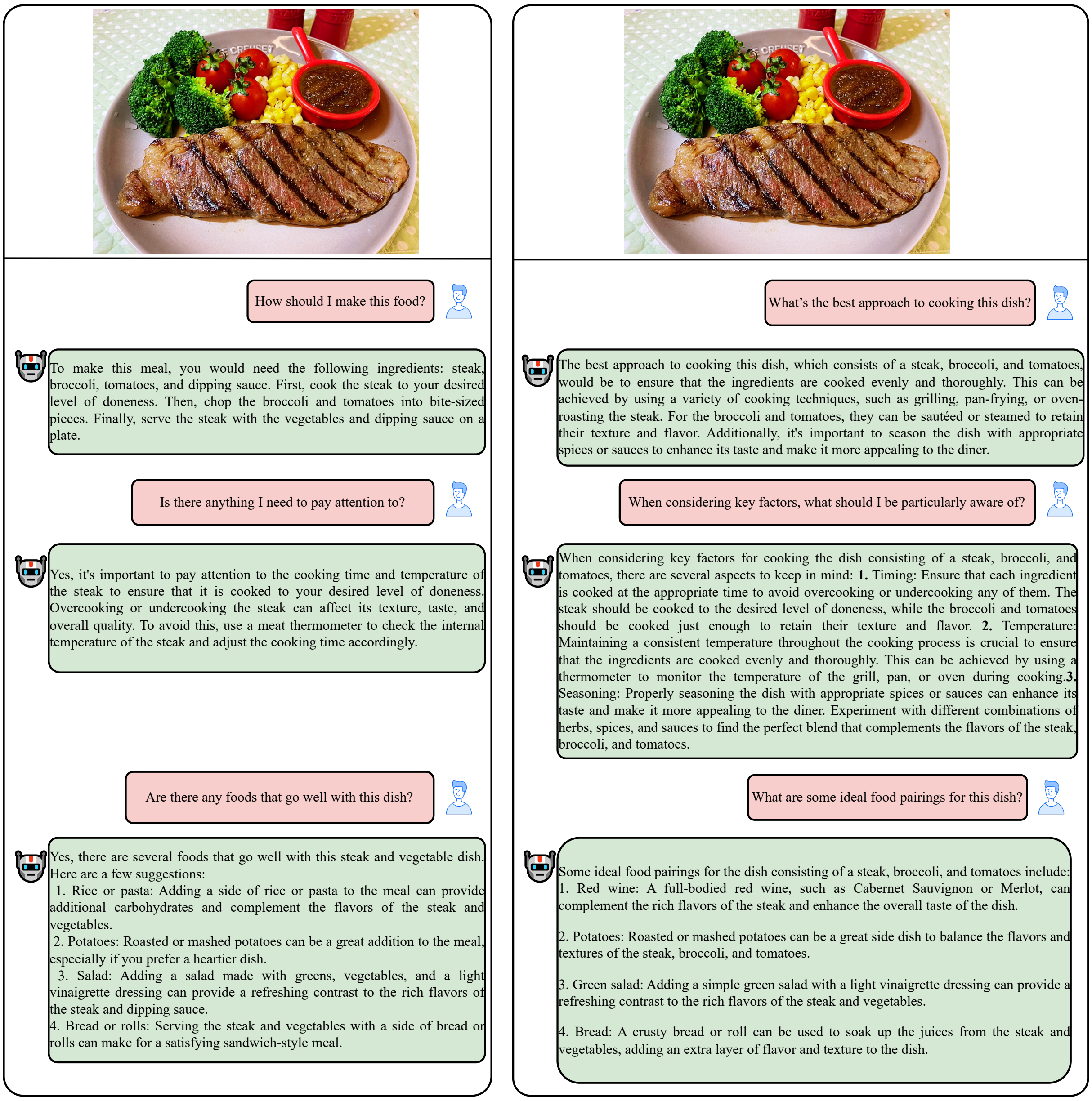


Figure 6: This is a multi-turn dialogue about cooking topics. The two generated answers exhibit similar accuracy in content recognition. However, in terms of richness and the final food pairing, our response (right) surpasses the vanilla model's response (left) significantly