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

Dongsheng Zhu [1], Xunzhu Tang [2], Weidong Han [3], Jinghui Lu [4],
Yukun Zhao [1], Guoliang Xing [1], Junfeng Wang [1], Dawei Yin* [1]
[1] Baidu Inc., [2] University of Luxemburg
[3] Fudan University, [4] University College Dublin
{zhudongsheng, yindawei02}@baidu.com
xunzhu.tang@uni.lu, wdhan20@fudan.edu.cn

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 content. 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 13.1% and 9% increase in accuracy over the prior state-of-the-art on the TextVQA and HatefulMemes datasets. Our main code is available at `https://github.com/Zhudongsheng75/VisLingInstruct`.

1 Introduction

The integration of Large Language Models (LLMs) with vision or other multi-modalities, 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 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) are effective, while they may introduce significant challenges, particularly for users who lack expertise in crafting optimal instructions. This limitation leads to inconsistent or sub-optimal 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). The VisLingInstruct introduces an innovative method via In-Context Learning (ICL) (Min et al., 2022) based on the comparison between instruction cases. We incorporate it with our newly proposed Instruction Alignment Score (IAS) to exploit the inherent capacity of MMLMs to self-evaluate the quality of text instructions. Consequently, VisLingInstruct can guide the model towards the generation of 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. Our new approach enables MMLMs to more effectively process 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 distinc-

---

*Corresponding author

tive 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 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 for training and inference (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). To the best of our knowledge, we spearhead the manual-free optimization of textual instruction in zero-shot for multi-modal tasks.
- 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 13.1% and 9% on the TextVQA and HatefulMemes 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.

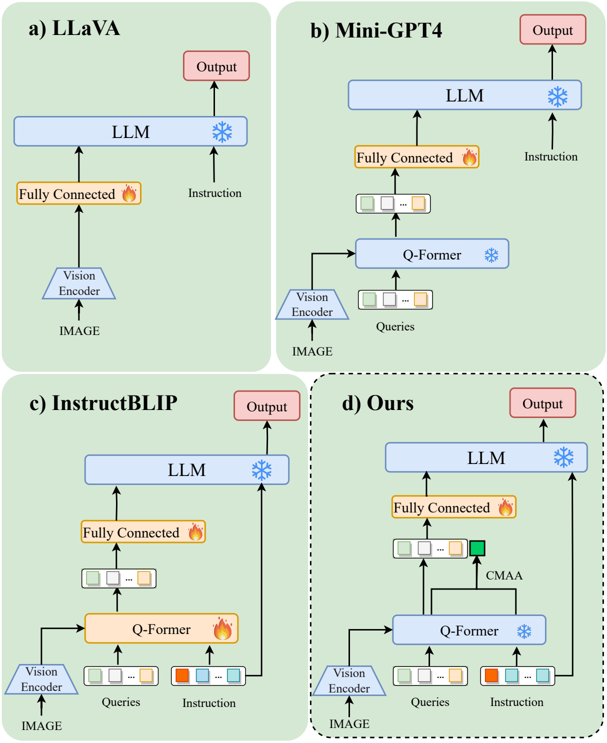
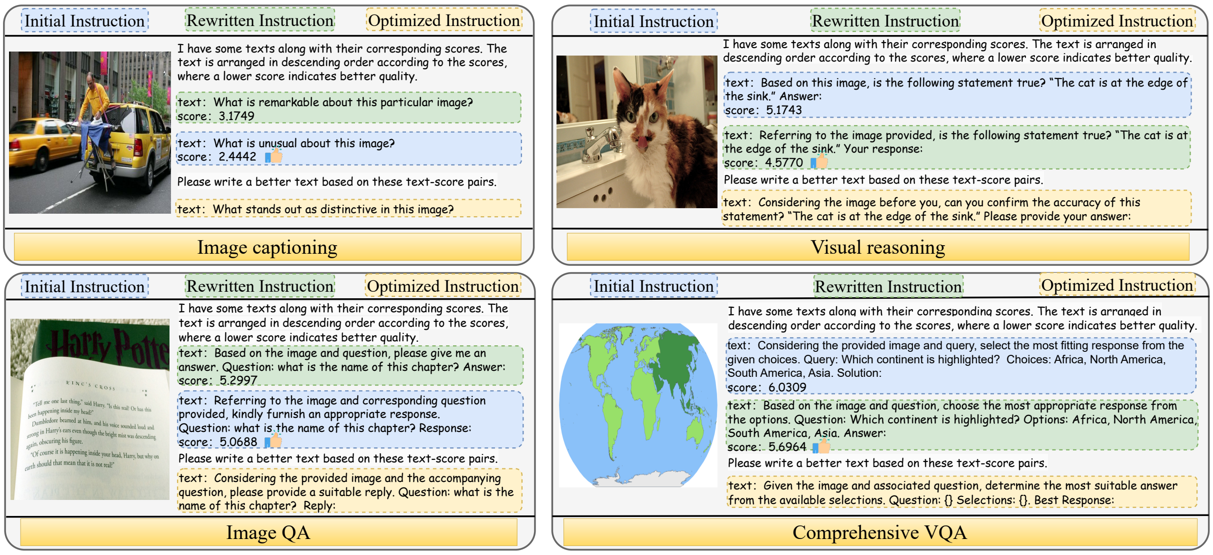


Figure 1: The structural comparison among the alignment modules of different MMLMs. The orange modules in the figure represent open weights, while the blue modules indicate frozen weights.

2.2 Optimizing Instructions for Large Models

Historically, Pre-trained Language Models (PLM) akin to BERT (Kenton and Toutanova, 2019) have utilized prompt crafting techniques (Brown et al., 2020; Jiang et al., 2022) to boost performance, with subsequent research exploring methods to discover higher-quality prompts (Gao et al., 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 textual instructions. (Zheng et al., 2023a) introduced STEP-BACK prompting, enabling LLMs to derive higher-level concepts from detailed instances.



I have some texts along with their corresponding scores. The text is arranged in descending order according to the scores, where a lower score indicates better quality.

text: What is remarkable about this particular image?
score: 3.1749

text: What is unusual about this image?
score: 2.4442 👍

Please write a better text based on these text-score pairs.

text: What stands out as distinctive in this image?

I have some texts along with their corresponding scores. The text is arranged in descending order according to the scores, where a lower score indicates better quality.

text: Based on this image, is the following statement true? "The cat is at the edge of the sink." Answer:
score: 5.1743

text: Referring to the image provided, is the following statement true? "The cat is at the edge of the sink." Your response:
score: 4.5770 👍

Please write a better text based on these text-score pairs.

text: Considering the image before you, can you confirm the accuracy of this statement? "The cat is at the edge of the sink." Please provide your answer:

Initial Instruction
Rewritten Instruction
Optimized Instruction
Image QA
Harry Potte
KING'S CROSS
"Tell me one last thing," said Harry. "Is this real? Or has this been happening inside my head?"
Dumbledore beamed at him, and his voice sounded loud and strong in Harry's ears even though the bright mist was descending again, obscuring his figure.
"Of course it is happening inside your head, Harry, but why on earth should that mean that it is not real?"

I have some texts along with their corresponding scores. The text is arranged in descending order according to the scores, where a lower score indicates better quality.

text: Based on the image and question, please give me an answer. Question: what is the name of this chapter? Answer:
score: 5.2997

text: Referring to the image and corresponding question provided, kindly furnish an appropriate response. Question: what is the name of this chapter? Response:
score: 5.0688 👍

Please write a better text based on these text-score pairs.

text: Considering the provided image and the accompanying question, please provide a suitable reply. Question: what is the name of this chapter? Reply:

I have some texts along with their corresponding scores. The text is arranged in descending order according to the scores, where a lower score indicates better quality.

text: Considering the provided image and query, select the most fitting response from the given choices. Query: Which continent is highlighted? Choices: Africa, North America, South America, Asia. Solution:
score: 6.0309

text: Based on the image and question, choose the most appropriate response from the options. Question: Which continent is highlighted? Options: Africa, North America, South America, Asia. Answer:
score: 5.6964 👍

Please write a better text based on these text-score pairs.

text: Given the image and associated question, determine the most suitable answer from the available selections. Question: {} Selections: {}. Best Response:

Figure 2: The examples of ranking with ICL in different domains. On the left side is the image input provided to the MMLM. On the right side, within the blue box, lies the initial instruction, while the rewritten instruction is contained within the green box. The 'score', referred to as IAS, indicates the quality of corresponding instructions with respect to the model, while the lower score (i.e., high quality) instruction ranks lower in the ICL demonstration. By utilizing the paradigm of ICL, MMLM learn the relationship between the scores of the two cases to generate higher-quality new instructions that lie in the yellow box.

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 Autonomous Instruction Optimization (AIO).

3.1 Enhancing Multi-modal Alignment

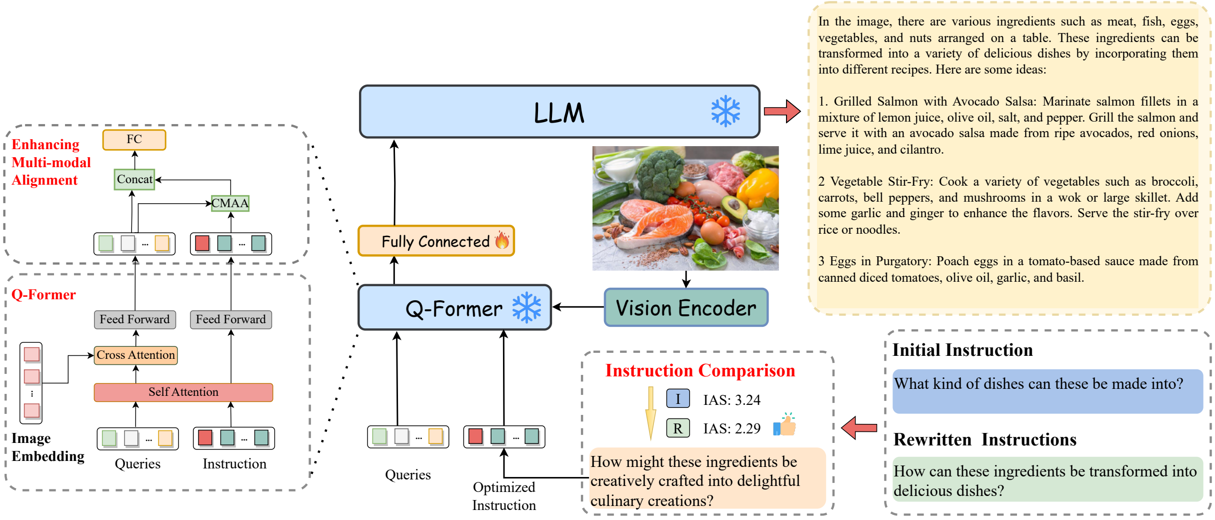
In the quest to refine MMLM, our focus shifts to bridging the gap between the realms of visual perception and linguistic expression. This section delves into our approach to enhancing the alignment between visual and textual modules within MMLM, introducing the architectural innovation and training optimization 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. The 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. CMAA leverages attention mechanisms (Bahdanau et al., 2014) and cross-modal feature fusion (Radford et al., 2021; Alayrac et al., 2022), to ensure that the resulting multi-modal representation encapsulates both the intricacies of language and the details of visual content:

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

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. Detailed information about CMAA can be referred to in Appendix A.1.

Optimized Model Training and Performance: In developing the new architecture, our approach extends beyond mere technical integration to encompass optimization of training and performance. We employ selective weight freezing strategy, where specific layers of the pre-trained model are



In the image, there are various ingredients such as meat, fish, eggs, vegetables, and nuts arranged on a table. These ingredients can be transformed into a variety of delicious dishes by incorporating them into different recipes. Here are some ideas:

1. Grilled Salmon with Avocado Salsa: Marinate salmon fillets in a mixture of lemon juice, olive oil, salt, and pepper. Grill the salmon and serve it with an avocado salsa made from ripe avocados, red onions, lime juice, and cilantro.

2 Vegetable Stir-Fry: Cook a variety of vegetables such as broccoli, carrots, bell peppers, and mushrooms in a wok or large skillet. Add some garlic and ginger to enhance the flavors. Serve the stir-fry over rice or noodles.

3 Eggs in Purgatory: Poach eggs in a tomato-based sauce made from canned diced tomatoes, olive oil, garlic, and basil.

What kind of dishes can these be made into?

How can these ingredients be transformed into delicious dishes?

How might these ingredients be creatively crafted into delightful culinary creations?

Figure 3: The figure depicts the complete pipeline of Instruction Comparison Optimization. The initial and rewritten instructions are processed through comparison optimization to generate optimized instruction. Subsequently, the optimized instruction is utilized for generation in MMLMs.

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 (Hu et al., 2021), thereby enhancing the learning efficiency and ensuring the robustness and scalability of the model (Toneva et al., 2018; Zhai et al., 2023). The objective function for training takes the following form:

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

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

3.2 Autonomous Instruction Optimization

During inference, the textual instruction has a significant impact on the generation results of MMLM. Therefore, we propose an approach that leverages the inherent text processing capabilities of MMLM to self-optimize textual instructions, thereby aligning the results more closely with user requirements. Our method comprises two stages: **Rewriting** Textual Instructions and Instruction **Comparison** Optimization.

Rewriting Textual Instruction: LLMs exhibit powerful text rewriting capabilities, preserving semantic information while modifying the content of the text. Therefore, our objective is to use the LLM in the MMLM to rewrite the initial textual instruction. The aim is obtain a pair of instructions that exhibit roughly equivalent semantics, thereby establishing a solid foundation for the next stage. It is important to note that the rewritten instruction that emerges from this process is not necessarily expected to surpass the initial instruction in quality. The mere occurrence of a difference between the pair is sufficient to satisfy the requirements of subsequent processes. This setting simplifies the text rewriting task, thereby lowering the barrier to its implementation.

Specifically, we designed a prompt tailored for LLM to rewrite the initial instruction. 'Initial instruction' refers to the original instruction sent by the user. The prompt directs LLM on how to rewrite the initial instruction while ensuring minimal semantic changes between the initial and the rewritten versions. The template of the prompt used in this stage can be referred to in the Appendix B.1. Notably, since this stage solely involves instruction rewriting, it does not necessitate the entire MMLM. Employing only the LLM part could marginally decrease the time consumed by the rewriting process.

Instruction Comparison Optimization: At this stage, we devise a method that allows the MMLM to identify the superior instruction via comparative analysis, with the aim to generate higher-quality instruction. As depicted in Figure 2, we innovatively apply ICL to rank cases, enabling the model to ascertain the quality of instructions solely through comparison between the pair of initial and rewritten instructions (Ren and Liu, 2023).

Considering that the ultimate purpose of the instructions is to aid inference by MMLM, we posit that the quality of these instructions should be evaluated by the MMLM themselves. Specifically, we enable MMLM to score the instruction independently, without the assistance of an external discriminator. As such, we proposed the Instruction Alignment Score (IAS), devised to measure the expected confidence of the evaluation instruction under the condition of a given image. We employ a prompt to guide MMLM in scoring the instruction. The template for this prompt can be found in Appendix B.2. Defined as the expectation of negative log-probability, 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, X_{img} is the input image, X_{prompt} denotes the prompt employed to guide the model in its computations, θ symbolizes our MMLM model and t_i represents the tokens from the textual instruction the MMLM are evaluating for quality. The negative log-probability, which originally served as the loss function for LLMs, is utilized in Equation 3 to assess the fluency of the given image and instruction under the current MMLM. A lower IAS indicates a higher alignment of the instruction with the model's understanding, enabling MMLM to perform better. After calculating IAS, as shown in Figure 2, we rank the **two** instruction-IAS pairs in descending order, and combine them into a prompt in the form of ICL. This is then input into MMLM to generate an optimized instruction. The optimized instruction will have better inference performance compared to the initial and rewritten instructions. The complete pipeline is presented in Figure 3 and Appendix A.2.

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. In Appendix C.1, this paper provides more details related to the training dataset.

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). In the Appendix C.2, we provide the details of the evaluation benchmarks as comprehensively as possible.

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 C.3.

4.3 Zero-shot Evaluation

We conducted zero-shot learning of our model against previous state-of-the-art (SOTA) works across 10 benchmarks in Table 1. It's evident that our model showcases a significant advantage in the majority of benchmarks, especially in Image QA and Comprehensive VQA domains. Specifically, our methods has improved the previous SOTA results by 13.1% and 9% in TextVQA and HatefulMemes. Furthermore, as our model weights are primarily inherited from InstructBLIP, a side-by-side comparison with InstructBLIP indicates that our method significantly enhances the overall ca-

	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$_{\mathrm{13B}}$)	74.9	107.5	50.9	41.0	40.6	19.6	42.5	45.1	61.0	53.7
MiniGPT-4 (Vicuna$_{\mathrm{13B}}$)	/	/	50.7	30.8	37.6	34.8	18.7	/	/	29.0
LLaVA (Vicuna$_{\mathrm{13B}}$)	/	/	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$_{\mathrm{7B}}$)	82.4	123.1	54.3	49.2	43.1	34.5	50.1	45.2	60.5	59.6
InstructBLIP (Vicuna$_{\mathrm{13B}}$)	82.8	121.9	52.1	49.5	44.8	33.4	50.7	45.4	63.1	57.5
BLIVA (Vicuna$_{\mathrm{13B}}$)	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$_{\mathrm{7B}}$)	87.9	**124.2**	60.1	52.0	44.2	42.7	60.6	45.7	74.6	**62.7**
Ours(Vicuna$_{\mathrm{13B}}$)	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 image-text 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).

pability of MMLMs. For example, based on the FlanT5-XXL model, our method improved upon InstructBLIP by 6% and 15.9% on Flickr30K and ScienceQA, respectively.

The results in Table 1 indicate that our proposed instruction optimization method exhibits very significant gains for image-text tasks. However, a small portion of the evaluation results shows some discrepancies with the overall trend. Specifically, our method exhibits very limited or even inferior performance compared to the baseline on NoCaps dataset. We attribute this phenomenon to potential biases introduced by the training set. The training set of InstructBLIP is much richer than ours, and fine-tuning solely on its subset may lead to a certain degree of catastrophic forgetting. Furthermore, another issue arises in the HatefulMemes where the smaller LLM backbone works better. The primary reason for this phenomenon might be the insufficient magnitude of parameter difference between LLMs, failing to establish a clear dominance. This observation is similarly reflected in the performance on InstructBLIP.

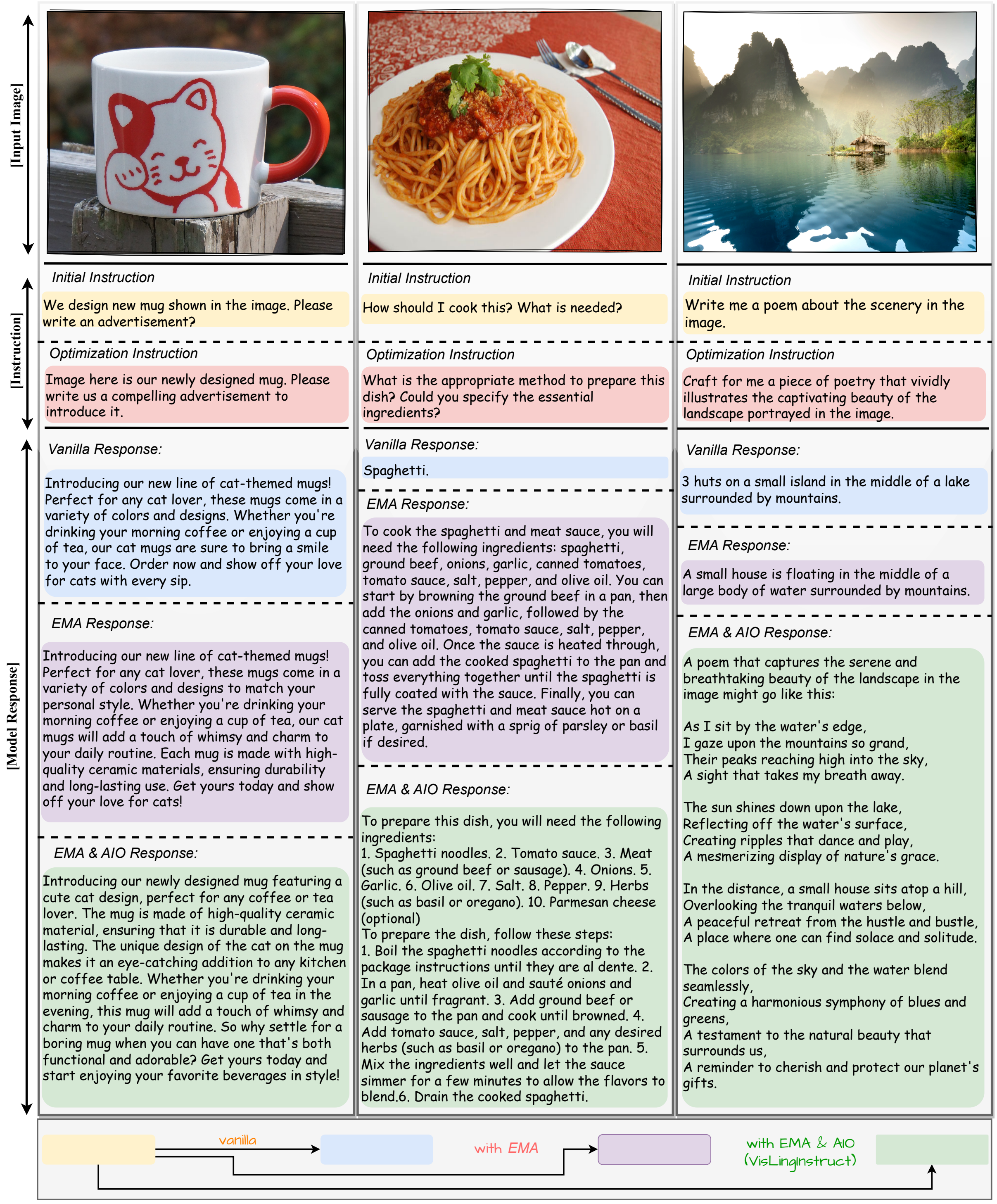
4.4 Ablation Study

To investigate the impact of EMA (Section 3.1) and 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.

Additionally, a particular phenomenon is observed in Table 2, where the encoder-decoder FlanT5 and the decoder-only Vicuna exhibit slight inconsistencies when our methods are applied. For instance, EMA is beneficial on the ScienceQA dataset for FlanT5, but performs poorly for Vicuna. The reasons for this phenomenon might be firstly because LLMs with different structures excel at different tasks. From our general understanding of model structures, the encoder part from FlanT5



Initial Instruction

We design new mug shown in the image. Please write an advertisement?

Optimization Instruction

Image here is our newly designed mug. Please write us a compelling advertisement to introduce it.

Vanilla Response:

Introducing our new line of cat-themed mugs! Perfect for any cat lover, these mugs come in a variety of colors and designs. Whether you're drinking your morning coffee or enjoying a cup of tea, our cat mugs are sure to bring a smile to your face. Order now and show off your love for cats with every sip.

EMA Response:

Introducing our new line of cat-themed mugs! Perfect for any cat lover, these mugs come in a variety of colors and designs to match your personal style. Whether you're drinking your morning coffee or enjoying a cup of tea, our cat mugs will add a touch of whimsy and charm to your daily routine. Each mug is made with high-quality ceramic materials, ensuring durability and long-lasting use. Get yours today and show off your love for cats!

EMA & AIO Response:

Introducing our newly designed mug featuring a cute cat design, perfect for any coffee or tea lover. The mug is made of high-quality ceramic material, ensuring that it is durable and long-lasting. The unique design of the cat on the mug makes it an eye-catching addition to any kitchen or coffee table. Whether you're drinking your morning coffee or enjoying a cup of tea in the evening, this mug will add a touch of whimsy and charm to your daily routine. So why settle for a boring mug when you can have one that's both functional and adorable? Get yours today and start enjoying your favorite beverages in style!

Initial Instruction

How should I cook this? What is needed?

Optimization Instruction

What is the appropriate method to prepare this dish? Could you specify the essential ingredients?

Vanilla Response:

Spaghetti.

EMA Response:

To cook the spaghetti and meat sauce, you will need the following ingredients: spaghetti, ground beef, onions, garlic, canned tomatoes, tomato sauce, salt, pepper, and olive oil. You can start by browning the ground beef in a pan, then add the onions and garlic, followed by the canned tomatoes, tomato sauce, salt, pepper, and olive oil. Once the sauce is heated through, you can add the cooked spaghetti to the pan and toss everything together until the spaghetti is fully coated with the sauce. Finally, you can serve the spaghetti and meat sauce hot on a plate, garnished with a sprig of parsley or basil if desired.

EMA & AIO Response:

To prepare this dish, you will need the following ingredients:
1. Spaghetti noodles. 2. Tomato sauce. 3. Meat (such as ground beef or sausage). 4. Onions. 5. Garlic. 6. Olive oil. 7. Salt. 8. Pepper. 9. Herbs (such as basil or oregano). 10. Parmesan cheese (optional)
To prepare the dish, follow these steps:
1. Boil the spaghetti noodles according to the package instructions until they are al dente. 2. In a pan, heat olive oil and sauté onions and garlic until fragrant. 3. Add ground beef or sausage to the pan and cook until browned. 4. Add tomato sauce, salt, pepper, and any desired herbs (such as basil or oregano) to the pan. 5. Mix the ingredients well and let the sauce simmer for a few minutes to allow the flavors to blend.6. Drain the cooked spaghetti.

Initial Instruction

Write me a poem about the scenery in the image.

Optimization Instruction

Craft for me a piece of poetry that vividly illustrates the captivating beauty of the landscape portrayed in the image.

Vanilla Response:

3 huts on a small island in the middle of a lake surrounded by mountains.

EMA Response:

A small house is floating in the middle of a large body of water surrounded by mountains.

EMA & AIO Response:

A poem that captures the serene and breathtaking beauty of the landscape in the image might go like this:

As I sit by the water's edge,
I gaze upon the mountains so grand,
Their peaks reaching high into the sky,
A sight that takes my breath away.

The sun shines down upon the lake,
Reflecting off the water's surface,
Creating ripples that dance and play,
A mesmerizing display of nature's grace.

In the distance, a small house sits atop a hill,
Overlooking the tranquil waters below,
A peaceful retreat from the hustle and bustle,
A place where one can find solace and solitude.

The colors of the sky and the water blend seamlessly,
Creating a harmonious symphony of blues and greens,
A testament to the natural beauty that surrounds us,
A reminder to cherish and protect our planet's gifts.

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
				FlanT5-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
				FlanT5-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 LLM backbones. 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.

is more suitable for tasks involving feature comprehension. Secondly, the corpus used during the pre-training of the model is also a crucial factor. FlanT5 might perform better in certain tasks simply because the model has encountered related content during pre-training.

We also conducted experiments and analyses on the number of instructions in ICL and the computational overhead of the proposed method. Detailed reports of these studies can be found in Appendices D.1 and D.2, respectively.

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 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 D.3.

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

Despite VisLingInstruct is an effective method for MMLMs, it still possesses certain limitations, which include:

Firstly, our autonomous instruction optimization framework has a relatively large computational overhead. We have carried out a comprehensive discussion on this subject in Appendix D.2. While we maintain that the added computations are quite justifiable and beneficial, it is undeniable that they augment the time required to yield MMLM results. We propose that future research should focus on optimizing the process of instruction optimization. We believe that such advancements will undoubtedly enhance the applicability and promotion of this technology.

Secondly, the experimental work presented within this paper is primarily concentrated on image and text modalities. There is also a real need to optimize instructions for other modalities. Consequently, we have earmarked this as future work, with the objective of verifying the efficacy of our framework on additional modalities, including 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.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

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.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

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.

Ting Jiang, Jian Jiao, Shaohan Huang, Zihan Zhang, Deqing Wang, Fuzhen Zhuang, Furu Wei, Haizhen Huang, Denvy Deng, and Qi Zhang. 2022. Promptbert: Improving bert sentence embeddings with prompts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8826–8837.

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.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763.

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*.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J Gordon. 2018. An empirical study of example forgetting during deep neural network learning. In

International Conference on Learning Representations.

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.

Yuexiang Zhai, Shengbang Tong, Xiao Li, Mu Cai, Qing Qu, Yong Jae Lee, and Yi Ma. 2023. Investigating the catastrophic forgetting in multimodal large language models. *arXiv preprint arXiv:2309.10313*.

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

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.1 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.2 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 Templates

B.1 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.
```

B.2 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: {}
```

C Data and Training Details

C.1 Training Dataset Format

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 input 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 input. Meanwhile, X_a^m serves as the ground truth.

C.2 Zero-shot Evaluation Datasets Details

As shown in Table 3, the evaluation parts chosen by different benchmarks are not the same. We have adopted the settings from InstructBLIP. It's important to note that for ScienceQA, we only evaluate the set with image context. 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.

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.

C.3 Training Details

We implement VisLingInstruct by LAVIS library (Li et al., 2022). We fine-tuned the fully connected

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.

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

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 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.

D More Experiments and Analyses

D.1 Number of Instructions in ICL

The process of instruction comparison is a crucial step, therefore we have explored the possibility of adding more instructions into ICL. In particular, we adopted two verification methods: firstly, we let the LLM generate multiple different **rewritten** instructions to increase the number of instructions involved in the ICL. Secondly, we continue to add optimized instructions generated by MMLM to the ICL for comparison and then generate new optimized instructions in a **loop**.

However, as shown in Table 5, neither of these operations could enhance the optimization effect of the instructions. On the contrary, the effect deteriorates as the number of rounds increases. We analyzed the possible reason is that the initial instruction is issued by the user, while the rewritten and optimized instructions are generated by MMLM. The statistical distribution of the user's instruction is significantly different from those generated by MMLM. The larger the difference, the greater the benefit derived from the comparison. However, the distribution of instructions generated by MMLM is similar, introducing them into the ICL for mutual comparison may introduce undesirable noise.

Meanwhile, the first method saw a more severe decline in effectiveness compared to the second. This is likely because the distributional difference between the additional instructions generated by rewriting is even smaller than that produced by looping, rendering the effect of ICL comparison almost negligible.

D.2 Computational Overhead

We conducted a comprehensive analysis of the computational overhead of our proposed method. VisLingInstruct, in the process of optimizing the initial instruction input by users, introduces some intermediate results, resulting in increased computation. For sample, the LLM from MMLM first generates the rewritten instruction based on the initial. Then, the MMLM needs to calculate the IAS of the two instructions. This step can rely on parallel computing, so it is equivalent to one computing time. Finally, the MMLM needs to further generate a refined instruction before finally producing a result. Therefore, the time cost is actually 3 times that of the vanilla baseline.

The aforementioned is purely a theoretical presumption. To quantify the specific computational overhead, we conducted several experiments. As depicted in Table 6, the time overhead varies across different benchmarks. This variability can be attributed to the fact that the intermediate results of our proposed method are all instruction-related,

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.

Backbone	1R	2R	3R	4R
		Rewriting		
FlanT5-XL	62.7	61.1	59.4	57.0
FlanT5-XXL	64.2	63.6	62.7	58.9
Vicuna-7B	65.5	64.8	63.2	60.4
Vicuna-13B	64.9	62.9	62.4	60.1
		Loop		
FlanT5-XL	62.7	61.5	60.2	58.4
FlanT5-XXL	64.2	64.4	62.6	59.5
Vicuna-7B	65.5	65.8	63.9	61.0
Vicuna-13B	64.9	63.7	62.1	60.3

Table 5: The results about the impact of the number in the ICL instruction comparison with different LLM backbones. 1R represents that we do not add new instructions, which is the standard setting in our method. 2R to 4R represent the corresponding rounds of instruction generation. All the results are the average values of 10 benchmarks.

Backbone	NoCaps	VSR	TextVQA	HM
FlanT5-XL	2.4	6.7	4.5	7.0
FlanT5-XXL	2.8	6.6	4.2	7.3
Vicuna-7B	2.7	7.1	4.3	7.2
Vicuna-13B	2.6	6.9	4.6	7.4

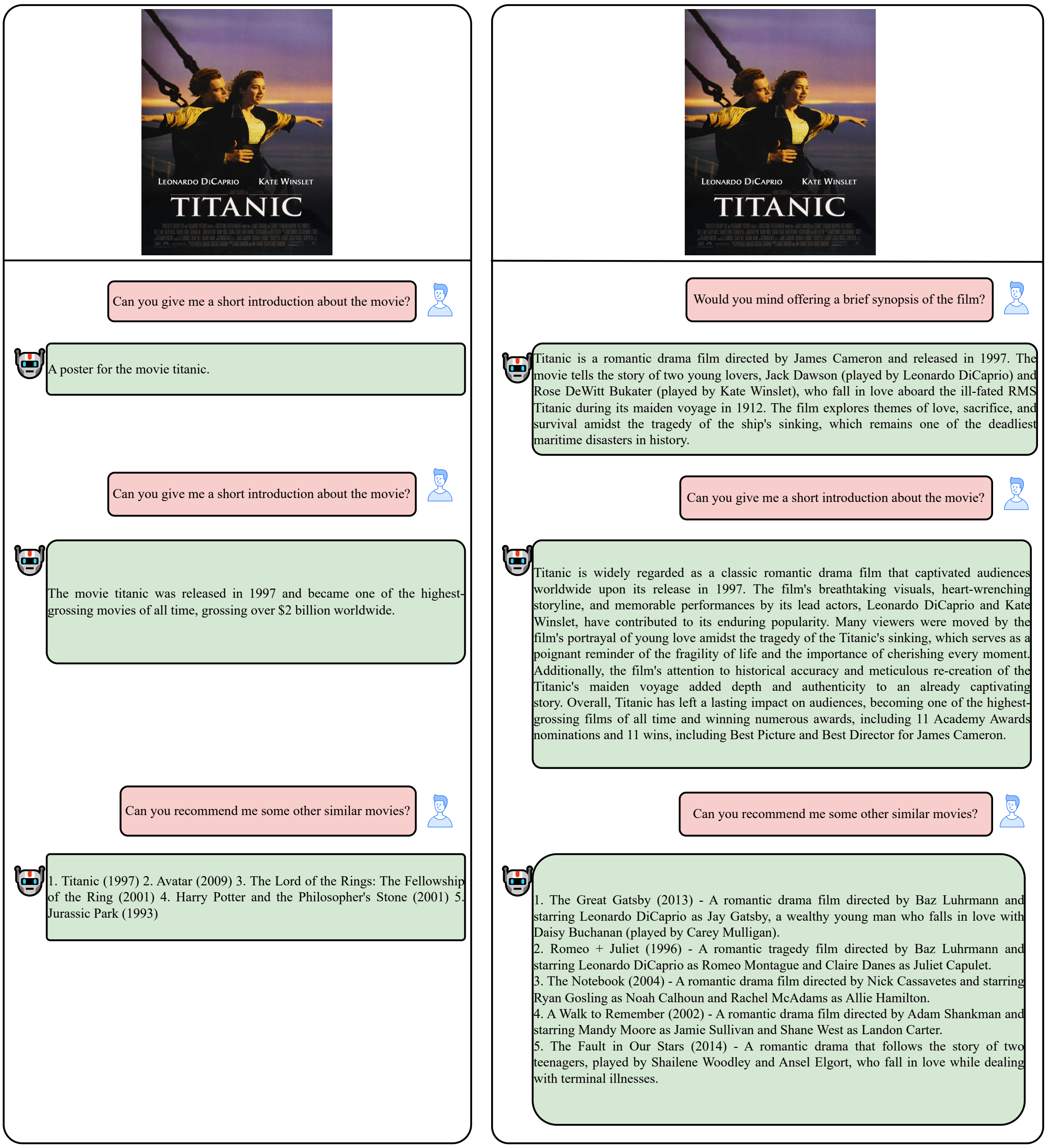
Table 6: The table records the computational overhead of VisLingInstruct on relevant benchmarks. We have selected four that are representative of their respective domains from ten benchmarks. The results denote that the multiple of time required by VisLingInstruct to complete the dataset compared to the time taken by the vanilla baseline.

and the length of the result returned by the model is the ultimate determinant of the computation time. For instance, in the VSR and HM tasks, the model only needs to respond with a simple ‘yes’ or ‘no’. As this is significantly shorter than the input instruction, the additional computation associated with the instruction becomes markedly impactful. Conversely, in tasks like NoCaps, the result returned by the model surpasses the length of the input instruction. As a result, the overall computational overhead of VisLingInstruct is diluted to less than 3 times that of the vanilla baseline.

The engineering of prompt and instruction always introduces additional computational overhead, which is inevitable. The key question is whether such overhead is worthwhile. In the real-world demands of MMLMs, instructions are often much shorter compared to the final generated results, such as advertisements or copywriting based on product images. Compared to the time spent producing results, the time cost centered around the instruction is not unacceptable. On the contrary, users maybe get a better experience because they get better results.

D.3 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.



Can you give me a short introduction about the movie?

A poster for the movie titanic.

Can you give me a short introduction about the movie?

The movie titanic was released in 1997 and became one of the highest-grossing movies of all time, grossing over $2 billion worldwide.

Can you recommend me some other similar movies?

1. Titanic (1997) 2. Avatar (2009) 3. The Lord of the Rings: The Fellowship of the Ring (2001) 4. Harry Potter and the Philosopher's Stone (2001) 5. Jurassic Park (1993)

Would you mind offering a brief synopsis of the film?

Titanic is a romantic drama film directed by James Cameron and released in 1997. The movie tells the story of two young lovers, Jack Dawson (played by Leonardo DiCaprio) and Rose DeWitt Bukater (played by Kate Winslet), who fall in love aboard the ill-fated RMS Titanic during its maiden voyage in 1912. The film explores themes of love, sacrifice, and survival amidst the tragedy of the ship's sinking, which remains one of the deadliest maritime disasters in history.

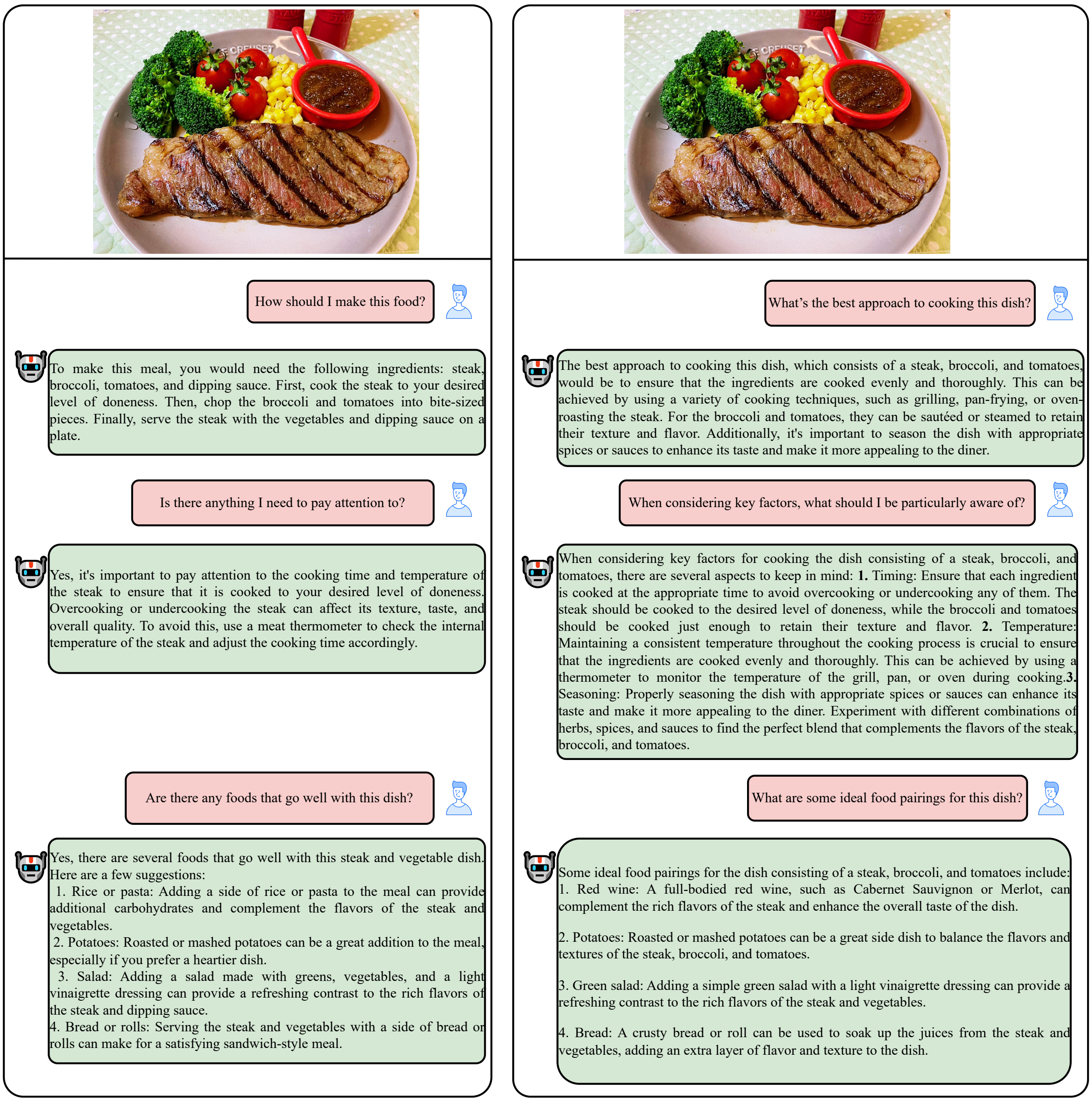
Can you give me a short introduction about the movie?

Titanic is widely regarded as a classic romantic drama film that captivated audiences worldwide upon its release in 1997. The film's breathtaking visuals, heart-wrenching storyline, and memorable performances by its lead actors, Leonardo DiCaprio and Kate Winslet, have contributed to its enduring popularity. Many viewers were moved by the film's portrayal of young love amidst the tragedy of the Titanic's sinking, which serves as a poignant reminder of the fragility of life and the importance of cherishing every moment. Additionally, the film's attention to historical accuracy and meticulous re-creation of the Titanic's maiden voyage added depth and authenticity to an already captivating story. Overall, Titanic has left a lasting impact on audiences, becoming one of the highest-grossing films of all time and winning numerous awards, including 11 Academy Awards nominations and 11 wins, including Best Picture and Best Director for James Cameron.

Can you recommend me some other similar movies?

1. The Great Gatsby (2013) - A romantic drama film directed by Baz Luhrmann and starring Leonardo DiCaprio as Jay Gatsby, a wealthy young man who falls in love with Daisy Buchanan (played by Carey Mulligan).
2. Romeo + Juliet (1996) - A romantic tragedy film directed by Baz Luhrmann and starring Leonardo DiCaprio as Romeo Montague and Claire Danes as Juliet Capulet.
3. The Notebook (2004) - A romantic drama film directed by Nick Cassavetes and starring Ryan Gosling as Noah Calhoun and Rachel McAdams as Allie Hamilton.
4. A Walk to Remember (2002) - A romantic drama film directed by Adam Shankman and starring Mandy Moore as Jamie Sullivan and Shane West as Landon Carter.
5. The Fault in Our Stars (2014) - A romantic drama that follows the story of two teenagers, played by Shailene Woodley and Ansel Elgort, who fall in love while dealing with terminal illnesses.

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).



How should I make this food?

To make this meal, you would need the following ingredients: steak, broccoli, tomatoes, and dipping sauce. First, cook the steak to your desired level of doneness. Then, chop the broccoli and tomatoes into bite-sized pieces. Finally, serve the steak with the vegetables and dipping sauce on a plate.

Is there anything I need to pay attention to?

Yes, it's important to pay attention to the cooking time and temperature of the steak to ensure that it is cooked to your desired level of doneness. Overcooking or undercooking the steak can affect its texture, taste, and overall quality. To avoid this, use a meat thermometer to check the internal temperature of the steak and adjust the cooking time accordingly.

Are there any foods that go well with this dish?

Yes, there are several foods that go well with this steak and vegetable dish. Here are a few suggestions:
1. Rice or pasta: Adding a side of rice or pasta to the meal can provide additional carbohydrates and complement the flavors of the steak and vegetables.
2. Potatoes: Roasted or mashed potatoes can be a great addition to the meal, especially if you prefer a heartier dish.
3. Salad: Adding a salad made with greens, vegetables, and a light vinaigrette dressing can provide a refreshing contrast to the rich flavors of the steak and dipping sauce.
4. Bread or rolls: Serving the steak and vegetables with a side of bread or rolls can make for a satisfying sandwich-style meal.

What's the best approach to cooking this dish?

The best approach to cooking this dish, which consists of a steak, broccoli, and tomatoes, would be to ensure that the ingredients are cooked evenly and thoroughly. This can be achieved by using a variety of cooking techniques, such as grilling, pan-frying, or oven-roasting the steak. For the broccoli and tomatoes, they can be sautéed or steamed to retain their texture and flavor. Additionally, it's important to season the dish with appropriate spices or sauces to enhance its taste and make it more appealing to the diner.

When considering key factors, what should I be particularly aware of?

When considering key factors for cooking the dish consisting of a steak, broccoli, and tomatoes, there are several aspects to keep in mind: **1.** Timing: Ensure that each ingredient is cooked at the appropriate time to avoid overcooking or undercooking any of them. The steak should be cooked to the desired level of doneness, while the broccoli and tomatoes should be cooked just enough to retain their texture and flavor. **2.** Temperature: Maintaining a consistent temperature throughout the cooking process is crucial to ensure that the ingredients are cooked evenly and thoroughly. This can be achieved by using a thermometer to monitor the temperature of the grill, pan, or oven during cooking.**3.** Seasoning: Properly seasoning the dish with appropriate spices or sauces can enhance its taste and make it more appealing to the diner. Experiment with different combinations of herbs, spices, and sauces to find the perfect blend that complements the flavors of the steak, broccoli, and tomatoes.

What are some ideal food pairings for this dish?

Some ideal food pairings for the dish consisting of a steak, broccoli, and tomatoes include:
1. Red wine: A full-bodied red wine, such as Cabernet Sauvignon or Merlot, can complement the rich flavors of the steak and enhance the overall taste of the dish.

2. Potatoes: Roasted or mashed potatoes can be a great side dish to balance the flavors and textures of the steak, broccoli, and tomatoes.

3. Green salad: Adding a simple green salad with a light vinaigrette dressing can provide a refreshing contrast to the rich flavors of the steak and vegetables.

4. Bread: A crusty bread or roll can be used to soak up the juices from the steak and vegetables, adding an extra layer of flavor and texture to the dish.

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.