Modality-Inconsistent Continual Learning of Multimodal Large Language Models

Anonymous authors
Paper under double-blind review

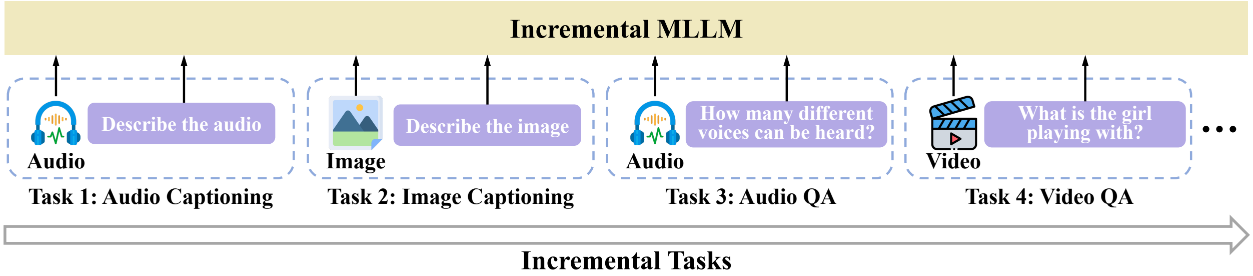


Figure 1: Illustration of our proposed Modality-Inconsistent Continual Learning (MICL), a novel and practical continual learning scenario of Multimodal Large Language Models (MLLMs), where tasks involve inconsistent modalities (image, video, or audio) and varying task types (captioning or question-answering).

Abstract

In this paper, we introduce Modality-Inconsistent Continual Learning (MICL), a new continual learning scenario for Multimodal Large Language Models (MLLMs) that involves tasks with inconsistent modalities (image, audio, or video) and varying task types (captioning or question-answering). Unlike existing vision-only or modality-incremental settings, MICL combines modality and task type shifts, both of which drive catastrophic forgetting. To address these challenges, we propose MoInCL, which employs a Pseudo Targets Generation Module to mitigate forgetting caused by task type shifts in previously seen modalities. It also incorporates Instruction-based Knowledge Distillation to preserve the model's ability to handle previously learned modalities when new ones are introduced. We benchmark MICL using a total of six tasks and conduct experiments to validate the effectiveness of our MoInCL. The experimental results highlight the superiority of MoInCL, showing significant improvements over representative and state-of-the-art continual learning baselines.

1 Introduction

Multimodal Large Language Models (MLLMs), leveraging the generative capabilities of LLMs, have demonstrated remarkable performance across diverse modality-specific tasks (Li et al., 2022b; 2023; Zhang et al., 2023b; Liu et al., 2023; Panagopoulou et al., 2023; Liu et al., 2024). MLLMs typically consist of a pre-trained modality encoder, like CLIP (Radford et al., 2021) for visual data, a pre-trained LLM, and a modality adapter that projects modality-specific features into the language token space. During training, the modality encoder is usually frozen to preserve its pre-trained knowledge, while the adapter and, optionally, the LLM are fine-tuned to align cross-modal representations and enhance task performance.

While fine-tuned MLLMs have demonstrated promising performance across various multimodal tasks, including impressive zero-shot capabilities on unseen instructions (He et al., 2023), adapting to novel tasks still requires task-specific fine-tuning. Nevertheless, existing studies (He et al., 2023; Zeng et al., 2024; Zheng et al., 2024) indicate that fine-tuning MLLMs on new tasks can lead to significant performance degradation on previously learned tasks, a phenomenon known as *catastrophic forgetting*, which remains the key challenge in continual learning. To address this issue, several works explore new approaches to enable continual training of MLLMs while mitigating the catastrophic forgetting issue. For instance, He et al. (2023) introduce the continual instruction tuning scenario for multimodal large language models, and propose an adapter-based method to handle it. Zheng et al. (2024) further explore the negative forward transfer problem in continual instruction tuning of MLLMs and propose a prompt-based method to mitigate these problems. Cao et al. (2024) propose a MLLM-based continual learning framework but mainly focusing on class-incremental image classification. While existing methods have demonstrated their abilities in alleviating the catastrophic problem in the continual learning scenario of MLLMs, they primarily focus on image modality, ignoring more general multimodal scenarios beyond image. Recently, Yu et al. (2024) introduced a modality-incremental setting for MLLMs, but treated each modality as a single, non-incremental task, ignoring the incremental nature of task types within modalities.

To address these issues, in this paper, we introduce Modality-Inconsistent Continual Learning (MICL), a novel continual learning scenario for MLLMs. In MICL, different task types, such as captioning and question-answering (QA), are introduced incrementally across learning steps incorporated with inconsistent modalities, as illustrated in Fig. 1. Unlike existing incremental learning settings of MLLMs, MICL not only highlights the modality-inconsistent (modality-incremental) scenario but also emphasizes the potential catastrophic forgetting problem arising from task type incrementality combined with modality inconsistency.

Moreover, we propose MoInCL (**Mo**dality-**In**consistent **C**ontinual **L**earning), a novel continual learning approach designed to address the MICL problem. By leveraging the generative capabilities of the LLM backbone, MoInCL introduces a *Pseudo Target Generation Module (PTGM)* to handle the task type shifts inherent in the task. Additionally, an *Instruction-based Knowledge Distillation (IKD)* constraint for LLM backbone is incorporated to preserve its ability to understand modality- and task-aware knowledge, preventing the degradation of its learned capabilities.

We evaluate our method across image, audio, and video modalities, combined with captioning and question-answering (QA) tasks, resulting in six multimodal incremental tasks (Image Captioning, Image QA, Audio Captioning, Audio QA, Video Captioning, and Video QA). Our experiments demonstrate that MoInCL significantly outperforms representative and state-of-the-art continual learning methods, effectively addressing both modality and task type shifts within MICL. In summary, this paper contributes the following:

- We propose the Modality-Inconsistent Continual Learning, a more general and practical continual learning scenario of MLLMs, where different modalities are introduced incrementally combined with different task types.
- We propose a novel continual learning approach named MoInCL to tackle the task. In MoInCL, a *Pseudo Target Generation Module (PTGM)* is introduced to address the task type shift problem of previously learned modalities through incremental steps. Moreover, we propose the *Instruction-based Knowledge Distillation (IKD)* constraint to prevent the LLM from the forgetting of learned both modality- and task-aware knowledge in old tasks.
- We benchmark the proposed MICL across three modalities—image, audio, and video—and two task types: captioning and question-answering, resulting in six incremental tasks. Experimental results demonstrate that our approach, MoInCL, significantly outperforms representative and state-of-the-art continual learning methods, showcasing its effectiveness in mitigating catastrophic forgetting from both modality and task type perspectives.

2 Related Work

2.1 Multimodal Large Language Models

Recent advances have extended Large Language Models (LLMs) to handle multimodal inputs such as images, audio, and video. Early work like CLIP (Radford et al., 2021) demonstrated the effectiveness of aligning textual and visual representations for zero-shot image classification. Flamingo (Alayrac et al., 2022) further integrated vision encoders with LLMs via cross-attention, significantly improving visual question answering

(VQA) and image captioning. Subsequent models like BLIP (Li et al., 2022b) and PaLM-E (Driess et al., 2023) scaled multimodal pre-training, with BLIP using a two-stage training strategy and PaLM-E incorporating embodied reasoning. More recently, LLaVA (Liu et al., 2023), InstructBLIP (Dai et al., 2023), X-InstructBLIP (Panagopoulou et al., 2023), Audio Flamingo (Kong et al., 2024; Ghosh et al., 2025; Goel et al., 2025), VideoLLaMA (Zhang et al., 2023a; Cheng et al., 2024; Zhang et al., 2025), Qwen-VL (Wang et al., 2024; Bai et al., 2025b), etc., have leveraged instruction tuning to refine the alignment between multimodal inputs and language, pushing the boundaries of multimodal reasoning and generation. Despite this progress, challenges persist as models scale to new modalities or tasks. Effectively integrating each modality without degrading performance on others remains a key issue. Moreover, robust continual learning strategies are crucial to prevent catastrophic forgetting and maintain knowledge across both previously learned and newly introduced modalities as new modalities or task types are integrated.

2.2 Continual Learning

Continual learning aims to enable models to learn incrementally while retaining previously acquired knowledge. Regularization-based methods, such as Elastic Weight Consolidation (EWC) (Kirkpatrick et al., 2017), assign importance to model parameters to prevent drastic updates (Kim et al., 2023). Knowledge distillation (KD) (Li & Hoiem, 2017; Rebuffi et al., 2017; Pian et al., 2023; Mo et al., 2023; Ahn et al., 2021; Douillard et al., 2020) and memory replay (Rebuffi et al., 2017; Pian et al., 2024; Chaudhry et al., 2019; Lopez-Paz & Ranzato, 2017) are other common strategies, where KD-based methods preserve past learned knowledge by aligning the predictions or internal features of a new model with those of an older one, and memory replay-based methods utilize a small memory set to store samples from old tasks, allowing the model to review a small number of old data while training on the current task (Rebuffi et al., 2017; Pian et al., 2024; Chaudhry et al., 2019; Lopez-Paz & Ranzato, 2017). Pseudo-rehearsal approaches (Odena et al., 2017; Ostapenko et al., 2019) take this a step further by generating synthetic examples via a generative model, reducing the need to store large amounts of data.

For MLLMs, where multiple modalities (e.g., images, audio, video) interact with language models, catastrophic forgetting is especially severe. Recent adapter-based continual instruction tuning (He et al., 2023) and prompt-based strategies (Zheng et al., 2024) help retain previously learned knowledge. HiDe-LLaVA (Guo et al., 2025) proposes a hierarchical decoupling strategy to separate instruction and perception components, allowing better task adaptation. SEFE (Chen et al., 2025) addresses forgetting by distinguishing between essential and superficial knowledge in continual instruction tuning. CL-MoE (Huai et al., 2025) introduces a dual momentum mixture-of-experts framework for continual visual question answering. However, these approaches mainly target image-text modalities. A modality-incremental scenario (Yu et al., 2024) has been explored, treating each modality as a separate task. However, it does not fully address evolving task types within each modality. To tackle this gap, we propose a new Modality-Inconsistent Continual Learning (MICL) scenario along with a novel approach to handle it effectively.

3 Method

3.1 Problem Formulation

In this subsection, we formalize the definition of our proposed Modality-Inconsistent Continual Learning (MICL). Given a sequence of T tasks $\{\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_T\}$, MICL aims to train the Multimodal Large Language Model (MLLM) $\mathcal{F}_{\boldsymbol{\Theta}}$ with parameters $\boldsymbol{\Theta}$ across these tasks incrementally. For the i-th task $\mathcal{T}_i$, we have $\mathcal{T}_i = \{(\boldsymbol{x}_{i,j}, \boldsymbol{t}_{i,j}, \boldsymbol{y}_{i,j})_{j=1}^{n_i}, M_i, P_i\}$, where M_i and P_i denote the modality and task type of task $\mathcal{T}_i$, respectively. $\boldsymbol{x}_{i,j}$, $\boldsymbol{t}_{i,j}$, and $\boldsymbol{y}_{i,j}$ present the modality's input data, the input text, and the target text of the j-th data sample of task $\mathcal{T}_i$. In our setting, the input text $\boldsymbol{t}_{i,j}$ varies depending on the task type. For captioning tasks, it may consist of a simple instruction, such as "`Describe the image/video/audio`." For question-answering (QA) tasks, the input text consists of sample-specific questions tailored to each instance. Moreover, the target text $\boldsymbol{y}_{i,j}$ typically consists of detailed description sentences for captioning tasks, while for QA tasks, it is usually limited to a few answer words. *Please note that, the output $\boldsymbol{y}_{i,j}$ is always a text sequence, consistent with the design of LLMs and MLLMs, which generate natural language outputs across diverse tasks. Tasks with*

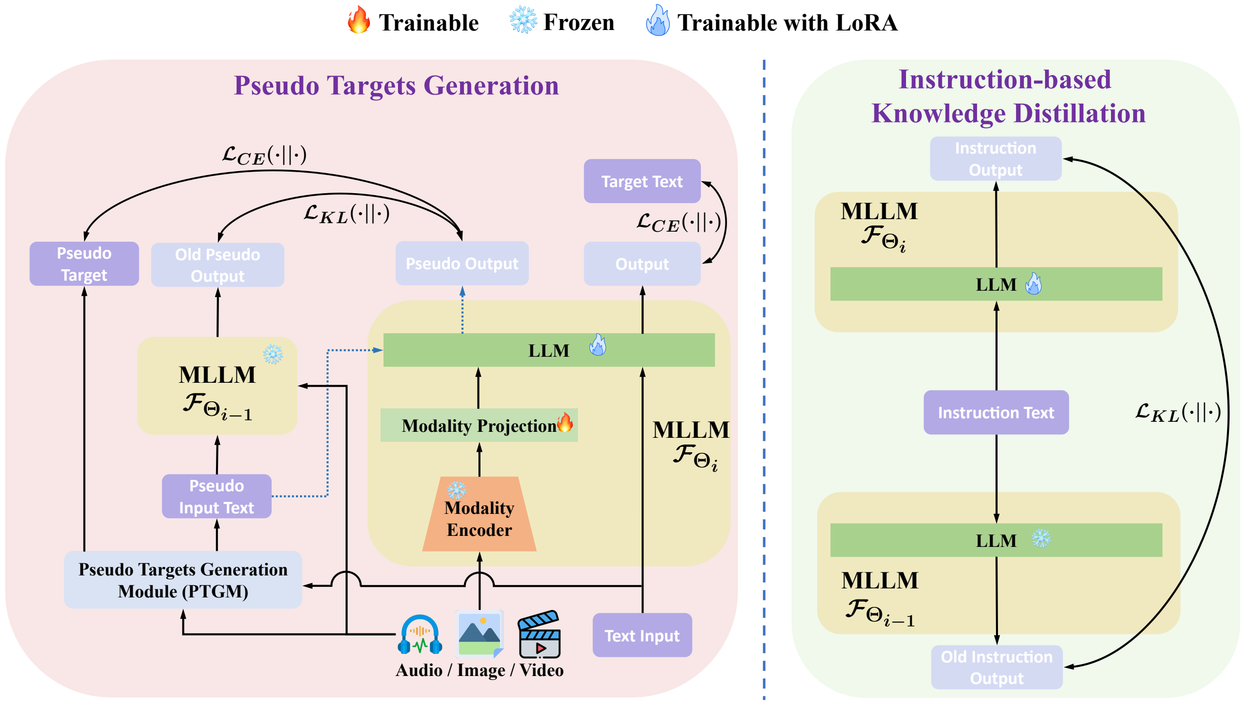

🔥 Trainable
❄ Frozen
Trainable with LoRA
Pseudo Targets Generation
$\mathcal{L}_{CE}(\cdot||\cdot)$
$\mathcal{L}_{KL}(\cdot||\cdot)$
Target Text
$\mathcal{L}_{CE}(\cdot||\cdot)$
Pseudo Target
Old Pseudo Output
Pseudo Output
Output
LLM
MLLM $\mathcal{F}_{\Theta_{i-1}}$
Modality Projection
Modality Encoder
MLLM $\mathcal{F}_{\Theta_i}$
Pseudo Input Text
Pseudo Targets Generation Module (PTGM)
Audio / Image / Video
Text Input
Instruction-based Knowledge Distillation
Instruction Output
MLLM $\mathcal{F}_{\Theta_i}$
LLM
Instruction Text
$\mathcal{L}_{KL}(\cdot||\cdot)$
LLM
MLLM $\mathcal{F}_{\Theta_{i-1}}$
Old Instruction Output

Figure 2: Overview of our proposed MoInCL, which mainly consists of a Multimodal Large Language Model (MLLM), a Pseudo Target Generation Module (PTGM), and a Instruction-based Knowledge Distillation (IKD). The red fire icon denotes the component is trainable in the current task, and the snowflake icon denotes the component is frozen during the training of the current task, while the blue fire icon means the associate component is trainable with LoRA (Hu et al., 2022) when training on the current task.

non-textual outputs (e.g., image or video generation) are beyond the scope of our current formulation, as they typically require fundamentally different architectures and objectives. We define $\mathcal{D}_i = \{(\boldsymbol{x}_{i,j}, \boldsymbol{t}_{i,j}, \boldsymbol{y}_{i,j})_{j=1}^{n_i}\}$ as the available training data when training the model $\mathcal{F}_{\boldsymbol{\Theta}}$ on task $\mathcal{T}_i$. Following the settings in modality-incremental learning (Yu et al., 2024), we do not include the memory set for replay in our MICL scenario, resulting in a memory-free continual learning setting. In summary, the training process on an incremental task $\mathcal{T}_i$ can be presented as:

$$\boldsymbol{\Theta}_i = \underset{\boldsymbol{\Theta}_{i-1}}{\arg\min}\, \mathbb{E}_{(\boldsymbol{x},\boldsymbol{t},\boldsymbol{y})\sim\mathcal{D}_i}[\mathcal{L}(\mathcal{F}_{\boldsymbol{\Theta}_{i-1}}(\boldsymbol{x},\boldsymbol{t}),\boldsymbol{y})], \tag{1}$$

where $\mathcal{L}$ denotes the cross-entropy loss function between the generated results and the target text for training the MLLM.

Please note that, in this work, we focus on two task types (captioning and question-answering) since they are among the most commonly studied in multimodal and continual learning scenarios (He et al., 2023; Yu et al., 2024). Following common practice, we adopted these tasks to establish benchmarks for comparison. Additionally, most of multimodal task types such as audio-visual event localization, vision-language navigation, etc, can often be reformulated into question-answering tasks, making these two task types a natural choice in our setting.

3.2 Framework Overview

To address our proposed Modality-Inconsistent Continual Learning (MICL), we introduce a novel continual learning method, **MoInCL**, as illustrated in Fig. 2. MoInCL primarily comprises a Pseudo Target Generation Module (PTGM) and an Instruction-based Knowledge Distillation (IKD) constraint. For the MLLM, we adopt the commonly used architecture, containing a modality encoder, a projection layer (MLP), and an

LLM, following the design paradigm used in models such as LLaVA (Liu et al., 2023), Qwen-VL family (Wang et al., 2024; Bai et al., 2025b;a), etc. However, we do not directly use the pre-trained parameters from these models, as most are designed to process only the visual modality, and their vision-language pre-training may introduce biases and unfairness in our continual learning setting. Please note that, for fair comparison, all the baseline methods use the same model architecture as our method, and during training, the modality encoders remain frozen, while the LLM is fine-tuned using LoRA (Hu et al., 2022).

3.3 Pseudo Target Generation Module

We now describe the Pseudo Target Generation Module (PTGM). Our key motivation is to leverage the text generation capability of the LLM component in the MLLM to address the task type shift challenge in continual learning. PTGM generates input and target text for different task types based on the modality input data of the current task. By utilizing the generated pseudo input text and pseudo targets, the model can effectively handle both the current task type and previously learned task types within the current modality.

In our PTGM, we maintain a set $LM = \{\}$ to represent all learned modalities. For example, $LM = \{$"*image*", "*audio*"$\}$ indicates that the model has been trained on tasks involving image or audio modalities. And for learned modalities, we maintain a modality-specific set $LT_M = \{\}$ to denote the learned task types of modality M. For instance, $LT_{image} = \{$"*captioning*"$\}$ if only image captioning task has been learned for image modality. Since different task types have distinct forms, the pseudo target generation process varies accordingly for each task type. Specifically, for a current task $\mathcal{T}_i$ with the modality of M_i, if M_i is a learned modality, *i.e.* $M_i \in LM$, the PTGM will be used to generate pseudo targets for task types within LT_{M_i}. If "*captioning*" $\in LT_{M_i}$, the pseudo input text should be a simple instruction guiding the model to generate a description of the input data. In this case, the pseudo input text generation process can be implemented by automatically filling the template to produce the result "`Describe the` M_i". On the other hand, if "*QA*" $\in LT_{M_i}$, directly applying a template is not suitable, as the pseudo QA pair should be specifically tailored to the modality's data rather than relying on generic templates. To overcome this issue, we utilize the generation ability of the LLM to generate the pseudo QA pair from the caption text of the current modality's data. Please note that in our MICL scenario, the task types considered are captioning and question-answering. Therefore, when generating pseudo QA pairs, the current task should correspond to the captioning task of the current modality. To generate QA pairs from captions, we employ a three-round generation process by prompting the pre-trained LLM component of the MLLM $\mathcal{F}$. Details of this process can be found in the Appendix. In summary, we use the following formulation to denote the pseudo target generation process:

$$\begin{gathered}\tilde{\boldsymbol{t}}, \tilde{\boldsymbol{y}} = PTGM(\boldsymbol{x}, \boldsymbol{y}, p), \\ s.t.\ M_i \in LM,\ P_i \notin LT_{M_i},\end{gathered} \tag{2}$$

where $p \in LT_{M_i}$ is a learned task type of modality M_i (please note that $p \neq P_i$), $\tilde{\boldsymbol{t}}$ and $\tilde{\boldsymbol{y}}$ denote the generated pseudo input text and pseudo target, respectively. $\boldsymbol{x}$ and $\boldsymbol{y}$ are the modality data and target text sampled from $\mathcal{D}_i$. Please note that only $\boldsymbol{x}$ is used for generating pseudo targets, while only $\boldsymbol{y}$ is utilized for generating pseudo QA pairs.

After obtaining the pseudo input text and pseudo target, a dual consistency constraint is applied between (1) the pseudo outputs of the current model $\mathcal{F}_{\boldsymbol{\Theta}_i}$ and the old model $\mathcal{F}_{\boldsymbol{\Theta}_{i-1}}$, and (2) the pseudo target and the pseudo output of the current model. This process is formulated as:

$$\begin{gathered}\mathcal{L}_{p.} = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{t})\sim\mathcal{D}_i}\Big[\lambda_i \mathcal{L}_{CE}(\hat{\boldsymbol{y}}' || \tilde{\boldsymbol{y}}) + \lambda_i' \mathcal{L}_{KL}(\hat{\boldsymbol{y}}' || \hat{\boldsymbol{y}}_{\boldsymbol{o}}')\Big], \\ s.t.\ \hat{\boldsymbol{y}}' = \mathcal{F}_{\boldsymbol{\Theta}_i}(\boldsymbol{x}, \tilde{\boldsymbol{t}}),\ \hat{\boldsymbol{y}}_{\boldsymbol{o}}' = \mathcal{F}_{\boldsymbol{\Theta}_{i-1}}(\boldsymbol{x}, \tilde{\boldsymbol{t}}),\end{gathered} \tag{3}$$

where $\hat{\boldsymbol{y}}_{\boldsymbol{o}}'$ and $\hat{\boldsymbol{y}}'$ denote the pseudo output from the old model and current model, respectively. λ_i and λ_i' present the weights to balance the two loss values for task $\mathcal{T}_i$.

3.4 Instruction-based Knowledge Distillation

In the previous subsection, we introduced the proposed PTGM to address the task type shift problem in the MICL scenario. However, when new modalities are introduced, the model faces a modality shift, leading to catastrophic forgetting of previously learned modalities. Additionally, as the PTGM generates pseudo targets only for seen modalities, the task type shift problem persists when training on tasks involving novel modalities. Furthermore, different modalities do *not* share the modality encoder or the modality projection, meaning that the shift problems primarily arise from updates to the LLM component in the MLLM. This results in the degradation of the LLM's ability to handle previously learned modalities. To address these issues, we propose Instruction-based Knowledge Distillation (IKD), a text instruction-based constraint designed to prevent the LLM from forgetting its learned capabilities in dealing with old modalities. Specifically, as illustrated in Fig. 2, IKD aligns the outputs of the LLM component from both the old and current models by applying a consistency loss, *i.e.* KL divergence, on their responses to the same text instruction input. In this way, instead of merely learning to handle tasks from new modalities, the current LLM's generative ability is also aligned with that of the previous LLM, thereby mitigating degradation in its ability to handle previously learned modalities. To achieve this, we introduce a pure text instruction set within IKD, which is maintained throughout the incremental steps. Since this pure text instruction set contains only text and no modality-specific data, it is not considered part of any multimodal tasks in our MICL scenario. As a result, maintaining this set does not violate the continual learning constraint that prohibits access to data from previous tasks during future tasks. This process can be formulated as:

$$\mathcal{L}_{ins.} = \mathbb{E}_{\boldsymbol{t}'\sim\mathcal{I}}\Big[\mathcal{L}_{KL}(f_{\boldsymbol{\theta}_i}(\boldsymbol{t}')||f_{\boldsymbol{\theta}_{i-1}}(\boldsymbol{t}'))\Big], \tag{4}$$

where $\mathcal{I}$ denotes the pure text instruction set, $f_{\boldsymbol{\theta}_i}$ and $f_{\boldsymbol{\theta}_{i-1}}$ denote the LLM component of the $\mathcal{F}_{\boldsymbol{\Theta}_i}$ and $\mathcal{F}_{\boldsymbol{\Theta}_{i-1}}$, respectively.

Algorithm 1 Training of MoInCL on task $\mathcal{T}_i$

```
Require: Old model $\mathcal{F}_{\boldsymbol{\Theta}_{i-1}}$, training set $\mathcal{D}_i$, pure text instruction set $\mathcal{I}$, current modality $M_i$, current task
    type $P_i$, learned modalities set $LM$, learned task type for the current modality $LT_{M_i}$ (only if $M_i \in LM$),
    learning rate $\eta$, scalars $\lambda_i, \lambda'_i, \alpha_i$
Initialize current model $\mathcal{F}_{\boldsymbol{\Theta}_i}$ from $\mathcal{F}_{\boldsymbol{\Theta}_{i-1}}$
if $M_i \notin LM$ then
    $\{\} \rightarrow LT_{M_i}$
end if
while not converged do
    Sample data $(\boldsymbol{x}, \boldsymbol{t}, \boldsymbol{y}) \sim \mathcal{D}_i$
    $\mathcal{L} = \mathcal{L}_{CE}(\mathcal{F}_{\boldsymbol{\Theta}_i}(\boldsymbol{x}, \boldsymbol{t})||\boldsymbol{y})$
    if $M_i \in LM$ and $LT_{M_i} \neq \varnothing$ then
        $\tilde{\boldsymbol{t}}, \tilde{\boldsymbol{y}} = PTGM(\boldsymbol{x}, \boldsymbol{y}, p)$, $s.t.\ p \in LT_{M_i}$
        $\hat{\boldsymbol{y}}' = \mathcal{F}_{\boldsymbol{\Theta}_i}(\boldsymbol{x}, \tilde{\boldsymbol{t}}), \hat{\boldsymbol{y}}'_{\boldsymbol{o}} = \mathcal{F}_{\boldsymbol{\Theta}_{i-1}}(\boldsymbol{x}, \tilde{\boldsymbol{t}})$
        $\mathcal{L}_{p.} = \lambda_i \mathcal{L}_{CE}(\hat{\boldsymbol{y}}'||\tilde{\boldsymbol{y}}) + \lambda'_i \mathcal{L}_{KL}(\hat{\boldsymbol{y}}'||\hat{\boldsymbol{y}}'_{\boldsymbol{o}})$
        $\mathcal{L} = \mathcal{L} + \mathcal{L}_{p.}$
    end if
    Sample instruction data $\boldsymbol{t}' \sim \mathcal{I}$
    $\mathcal{L}_{ins.} = \mathcal{L}_{KL}(f_{\boldsymbol{\theta}_i}(\boldsymbol{t}')||f_{\boldsymbol{\theta}_{i-1}}(\boldsymbol{t}'))$
    $\mathcal{L} = \mathcal{L} + \mathcal{L}_{ins.}$
    $\boldsymbol{\Theta}_i \leftarrow \boldsymbol{\Theta}_i - \eta \nabla \mathcal{L}$
    $\boldsymbol{\theta}_i \leftarrow \alpha_i \boldsymbol{\theta}_i + (1 - \alpha_i) \boldsymbol{\theta}_{i-1}$
end while
```

3.5 Overall Training Target

Above, we present our proposed Pseudo Target Generation Module (PTGM) and Instruction-based Knowledge Distillation (IKD) constraint. When training on a current task $\mathcal{T}_i$, we have the main loss function:

$$\begin{gathered}\mathcal{L}_{main} = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{t},\boldsymbol{y})\sim\mathcal{D}_i}\Big[\mathcal{L}_{CE}(\hat{\boldsymbol{y}}||\boldsymbol{y})\Big],\\ s.t.\ \hat{\boldsymbol{y}} = \mathcal{F}_{\boldsymbol{\Theta}_i}(\boldsymbol{x},\boldsymbol{t}),\end{gathered} \tag{5}$$

where $\hat{\boldsymbol{y}}$ is the output of the output of the current model $\mathcal{F}_{\boldsymbol{\Theta}_i}$ by taking data samples from current task's training data $\mathcal{D}_i$ as input.

Finally, in our overall training target, the dual consistency constraint for generated pseudo targets $\mathcal{L}_{pseudo}$ and the IKD constraint $\mathcal{L}_{ins.}$ are combined with the main training loss of task $\mathcal{T}_i$:

$$\mathcal{L} = \mathcal{L}_{main} + \mathcal{L}_{p.} + \mathcal{L}_{ins.} \tag{6}$$

Additionally, inspired by the parameters/weights fusion mechanism proposed in existing works (Xiao et al., 2023; Sun et al., 2024), which have demonstrated effectiveness in preserving learned knowledge from previous tasks by applying a weighted sum between the old and current models' parameters/weights, we also adopt the parameters fusion mechanism on the LLM component of the MLLM to further prevent it from forgetting the capabilities of handling previously learned modalities, which can be denoted as:

$$\boldsymbol{\theta}_i = \alpha_i\boldsymbol{\theta}_i + (1-\alpha_i)\boldsymbol{\theta}_{i-1}, \tag{7}$$

where $\boldsymbol{\theta}$ denotes the parameters of the LLM component of the MLLM, α_i is the weight for balancing the two groups of parameters. The overall algorithm of our proposed MoInCL is presented in Alg. 1.

3.6 Distinction from Existing Methods

Our MoInCL introduces two key innovations: 1) a Pseudo Target Generation Module (PTGM) to leverage the text generation capability of the LLM component in the MLLM to address the task type shift challenge in our proposed MICL scenario, and 2) an Instruction-based Knowledge Distillation (IKD) constraint to tackle the modality shift problem in the LLM component of the MLLM.

While existing works also utilize knowledge distillation techniques to preserve knowledge from old tasks, they primarily focus on distilling final outputs or internal features between old and current models by taking same training samples as input, as seen in methods like LwF (Li & Hoiem, 2017) and EWF (Xiao et al., 2023). These approaches do not perform well in our MICL scenario, as they significantly constrain the MLLM's ability to learn new tasks, particularly in settings with substantial gaps between tasks, such as in our proposed MICL. In contrast, our IKD leverages pure text instructions as the input to the LLM component for knowledge distillation, avoiding introducing negative impacts on the current training task. This approach allows us to directly distill knowledge of the LLM without imposing additional constraints on the MLLM's ability to learn new tasks, ensuring that both knowledge preservation and new task learning are achieved effectively in MICL.

As for the weight fusion strategy, we acknowledge that it is not one of our primary technical contributions. However, our experiments demonstrate that this strategy can be seamlessly integrated with PTGM and IKD to further enhance the anti-forgetting capability of our approach. For this reason, we also include the weight fusion strategy in our method.

4 Experiments

4.1 Experimental Setup

4.1.1 Dataset

In our proposed Modality-Inconsistent Continual Learning (MICL), we include six tasks: Image Captioning, Image QA, Audio Captioning, Audio QA, Video Captioning, and Video QA. Each task is represented by a commonly used dataset. Specifically, we use the Flickr30K (Young et al., 2014) dataset for the Image Captioning task, the OK-VQA (Marino et al., 2019) dataset for the Image QA task, the AudioCaps (Kim et al., 2019) dataset for the Audio Captioning task, the Clotho-AQA (Lipping et al., 2022) dataset for the Audio QA task, the MSR-VTT (Xu et al., 2016) dataset for the Video Captioning task, and the MSVD-QA (Xu et al., 2017) dataset for the Video QA task. More dataset details are provided in the Appendix B.

4.1.2 Baselines

In our experiments, we compare our proposed MoInCL with the following continual learning methods: Fine-tuning, LwF (Li & Hoiem, 2017), EWC (Kirkpatrick et al., 2017), EWF (Xiao et al., 2023), PathWeave (Yu et al., 2024), and BECAME (Li et al., 2025). Among these, LwF, EWC, EWF, and BECAME are representative general continual learning methods, while PathWeave is the most recent state-of-the-art continual learning method designed for MLLMs, which involves a modality-aware adapter-in-adapter mechanism to address the modality-shift problem in modality-incremental learning of MLLMs. Please note that, for a fair comparison, all baseline methods use the same model architecture as our approach, including the Large Language Model (LLM) component. We also conduct the experiment of joint training with all tasks as the Upper-Bound.

4.1.3 Evaluation Metrics

Following Panagopoulou et al. (2023), we use the CIDEr score (Vedantam et al., 2015) and prediction accuracy as evaluation metrics to evaluate captioning tasks and QA tasks, respectively. For all baselines and our method, we report the average final performance across all learned tasks, *i.e.*, the average performance of all tasks after completing the training of the final task. Since captioning and QA tasks use different evaluation metrics, we separately report the average final performance for each task type: the average final CIDEr score for captioning tasks and the average final accuracy for QA tasks. We formulate them as:

$$\begin{gathered} Avg.CIDEr = \frac{1}{N_{cap.}} \sum_{i=1}^{T} c_i^T, \\ s.t.\ P_i = \text{``}Captioning\text{''}, \end{gathered} \tag{8}$$

where $N_{cap.}$ denotes the number of captioning tasks, c_i^T denotes the CIDEr score of task $\mathcal{T}_i$ after completing the training of task $\mathcal{T}_T$ if task $\mathcal{T}_i$ is a captioning task. Similarly, the average final accuracy can be formulated as:

$$\begin{gathered} Avg.Acc. = \frac{1}{N_{QA}} \sum_{i=1}^{T} a_i^T, \\ s.t.\ P_i = \text{``}QA\text{''}, \end{gathered} \tag{9}$$

where N_{QA} denotes the number of QA tasks, a_i^T denotes the accuracy of task $\mathcal{T}_i$ after completing the training of task $\mathcal{T}_T$ if task $\mathcal{T}_i$ is a QA task. Furthermore, to evaluate the anti-forgetting capability of each method, we propose two metrics: the forgetting ratio and the average forgetting ratio. The forgetting ratio measures the proportion of performance drop for each task after completing the training of the final task, while the average forgetting ratio represents the mean forgetting ratio across all tasks, which can be formulated as:

$$\begin{aligned} Forget._i =& (s_i^i - s_i^T)/s_i^i, \\ Avg.Forget. =& \frac{1}{T} \sum_{i=1}^{T} Forget._i \end{aligned} \tag{10}$$

where s_i^i and s_i^T denotes the testing score of task $\mathcal{T}_i$ after the training of task $\mathcal{T}_i$ and $\mathcal{T}_T$, respectively.

4.1.4 Implementation Details

We implement our experiments using Pytorch (Paszke et al., 2019) and LaVIS (Li et al., 2022a) framework. For the LLM backbone of the Multimodal Large Language Model (MLLM), we adopt the Llama-3.2-1B-Instruct (Dubey et al., 2024), which is initialized with its official pre-trained parameters at the beginning

Table 1: Results on the two task orders for different methods. Bold values indicate the best results in each column, while underlined values represent the second-best results in each column.

Methods	Order 1			Order 2		
	Avg. CIDEr ↑	Avg. Acc. ↑	Avg. Forget. ↓	Avg. CIDEr ↑	Avg. Acc. ↑	Avg. Forget. ↓
Fine-tuning	30.64	40.58	41.17%	10.82	37.01	65.56%
LwF (Li & Hoiem, 2017)	34.80	40.21	39.26%	12.37	38.79	61.84%
EWC (Kirkpatrick et al., 2017)	39.06	37.04	38.79%	9.92	37.65	66.40%
EWF (Xiao et al., 2023)	24.59	36.34	48.55%	13.92	**45.85**	46.64%
PathWeave (Yu et al., 2024)	34.20	36.19	44.36%	11.11	41.13	61.47%
BECAME (Li et al., 2025)	24.36	38.50	46.96%	10.61	43.20	54.10%
MoInCL (Ours)	**55.31**	**42.29**	**14.21%**	**51.13**	45.22	**8.93%**
Upper Bound (Joint training)	66.69	48.97	-	66.69	48.97	-

of the first task. Following the implementation in (Panagopoulou et al., 2023), we apply the EVA-CLIP-ViT-G/14 (Fang et al., 2023) as the Image Encoder and Video Encoder, and the $\text{BEATs}_{\text{iter3+}}$ (Chen et al., 2023) as the Audio Encoder. Each video input consists of 4 frames, and the audio input also consists of 4 frames with the sampling rate of 11kHz. For the video and audio modalities, the Video Encoder and Audio Encoder process each frame individually and then concatenate the encoded patches from all frames, following the approach in (Panagopoulou et al., 2023). For the Image Projection, we use a two-layers MLP with the GELU (Hendrycks & Gimpel, 2016) activation function. For the Video and Audio Projection, both of them include a single convolutional layer as a pooling layer to reduce the total number of patches, followed by a two-layers MLP with the GELU activation function. For each task, we train the model using the AdamW (Loshchilov & Hutter, 2019) optimizer with an initial learning rate of 1e-5, adjusted using the cosine decay strategy, and a weight decay of 5e-2. We train our proposed MoInCL and all baseline methods on a NVIDIA RTX A6000 Ada GPU. During the training of our approach, the pure text instructions in the Instruction-based Knowledge Distillation (IKD) constraint are randomly sampled from the Natural Instructions (Mishra et al., 2022) dataset.

4.2 Experimental Comparison

We conduct experiments using two random task orders. For **Order 1**, the tasks are arranged as: *Audio Captioning* → *Image Captioning* → *Video QA* → *Audio QA* → *Image QA* → *Video Captioning*. For **Order 2**, the task sequence is: *Image Captioning* → *Video Captioning* → *Video QA* → *Image QA* → *Audio Captioning* → *Audio QA*. Additional experimental results on more task orders are provided in Appendix C, where we demonstrate that our framework can handle highly challenging task orders involving severe modality and task shifts.

The main results are shown in Tab. 1. We can see that our proposed MoInCL achieves state-of-the-art performance compared to all baseline methods. Except the average final accuracy of the Order 2, our method has the best performance on all three metrics across both orders. Specifically, in Order 1, our method surpasses the best baseline results by **16.25**, **1.71**, and **24.58** in terms of average final CIDEr score, average final accuracy, and average forgetting ratio, respectively. In Order 2, our method outperforms the best baseline results by **37.21** and **37.71** for average final CIDEr score and average forgetting ratio, respectively.

The testing results of the first three incremental tasks (Image Captioning → Video Captioning → Video QA) are shown in Tab. 2. From these results, we observe that when the modality shift occurs from the Image Captioning task to the Video Captioning task, the performance of the previous task (Image Captioning) drops significantly across all baseline methods, with CIDEr score reductions ranging from 8.34 to 23.63. Additionally, when the task type shift occurs from the Video Captioning task to the Video QA task, the performance of the previous task (Video Captioning) also decreases significantly, with CIDEr score reductions ranging from 35.61 to 41.66. These results further validate our insight that both modality shift and task type shift directly contribute to the catastrophic forgetting problem, underscoring the core challenges of our proposed MICL scenario. For our method, the performance drop for the Image Captioning task is only **3.91** when the modality shift occurs. Moreover, we observe that the performance of the Video Captioning task

Table 2: Detailed testing results of the first three tasks of Order 2. The evaluation metric used for the Flickr30K and MSR-VTT datasets is CIDEr score, while that for the MSVD-QA dataset is accuracy.

Methods		Flickr30k	MSR-VTT	MSVD-QA
Fine-tuning	Step 1	77.50	-	-
	Step 2	64.04	48.03	-
	Step 3	12.12	8.64	46.20
LwF (Li & Hoiem, 2017)	Step 1	77.50	-	-
	Step 2	53.87	48.70	-
	Step 3	10.20	7.80	47.64
EWC (Kirkpatrick et al., 2017)	Step 1	77.50	-	-
	Step 2	62.65	47.73	-
	Step 3	10.45	9.66	45.79
EWF (Xiao et al., 2023)	Step 1	77.50	-	-
	Step 2	69.16	45.30	-
	Step 3	56.10	9.69	45.33
PathWeave (Yu et al., 2024)	Step 1	77.22	-	-
	Step 2	53.60	50.01	-
	Step 3	7.36	8.35	47.87
BECAME (Li et al., 2025)	Step 1	77.50	-	-
	Step 2	77.22	47.64	-
	Step 3	52.16	9.82	47.35
MoInCL (Ours)	Step 1	77.50	-	-
	Step 2	73.59	48.03	-
	Step 3	70.88	48.34	43.11

Table 3: Forgetting ratio of each task in Order 1. Bold values denote the best results in each column, while underlined values indicate the second-best results in each column.

Methods	Forgetting Ratio ↓					
	AudioCaps	Flickr30k	MSVD-QA	Clotho-AQA	OK-VQA	MSR-VTT
Fine-tuning	57.51%	85.04%	51.33%	7.15%	4.81%	0.00%
LwF (Li & Hoiem, 2017)	54.79%	72.52%	59.32%	2.76%	6.92%	0.00%
EWC (Kirkpatrick et al., 2017)	62.47%	46.55%	61.55%	9.95%	13.42%	0.00%
EWF (Xiao et al., 2023)	69.65%	92.51%	79.07%	0.47%	1.03%	0.00%
PathWeave (Yu et al., 2024)	75.49%	58.16%	61.74%	16.25%	10.18%	0.00%
BECAME (Li et al., 2025)	72.82%	92.70%	66.04%	**-0.13%**	3.36%	0.00%
MoInCL (Ours)	**27.52%**	**9.18%**	**36.58%**	0.07%	**-2.28%**	0.00%

improves after training on the Video QA task which introduces the task type shift issue. These findings further highlight the effectiveness of our method in mitigating the catastrophic forgetting problem in MICL by addressing both modality shift and task type shift challenges. For detailed results of each task and qualitative analysis, please refer to E, D, and F in Appendix.

We present the forgetting ratio of each task in both orders in Tab. 3 and 4, from which we can see that, our method outperforms baseline methods significantly, further demonstrating the superiority of our proposed method in mitigating the catastrophic forgetting in our proposed MICL scenario.

Table 4: Forgetting ratio of each task in Order 2. Bold values denote the best results in each column, while underlined values indicate the second-best results in each column.

Methods	Forgetting Ratio ↓					
	Flickr30k	MSR-VTT	MSVD-QA	OK-VQA	AudioCaps	Clotho-AQA
Fine-tuning	93.02%	85.72%	31.23%	49.77%	68.06%	0.00%
LwF (Li & Hoiem, 2017)	91.20%	85.83%	31.51%	40.07%	60.60%	0.00%
EWC (Kirkpatrick et al., 2017)	91.08%	92.21%	40.49%	37.91%	70.31%	0.00%
EWF (Xiao et al., 2023)	89.86%	78.28%	6.04%	4.15%	54.89%	0.00%
PathWeave (Yu et al., 2024)	92.42%	87.54%	25.67%	35.51%	66.22%	0.00%
BECAME (Li et al., 2025)	90.50%	80.31%	15.48%	9.92%	74.29%	0.00%
MoInCL (Ours)	**22.04%**	**2.25%**	**2.60%**	**3.33%**	**14.43%**	0.00%

Table 5: Ablation results on the two task orders on each key component of our MoInCL. Bold values indicate the best results in each column, while underlined values represent the second-best results in each column.

Methods	Order 1			Order 2		
	Avg. CIDEr ↑	Avg. Acc. ↑	Avg. Forget. ↓	Avg. CIDEr ↑	Avg. Acc. ↑	Avg. Forget. ↓
MoInCL w/o PTGM	26.61	37.18	45.64%	9.95	**47.51**	49.62%
MoInCL w/o IKD	53.33	40.69	17.82%	49.32	43.40	13.03%
MoInCL	**55.31**	**42.29**	**14.21%**	**51.13**	45.22	**8.93%**

4.3 Ablation Studies

To further assess the effectiveness of each key component in our proposed MoInCL, we conduct ablation studies on the Pseudo Target Generation Module (PTGM) and Instruction-based Knowledge Distillation (IKD) across two random task orders. The experimental results, presented in Tab. 5, clearly demonstrate that removing either PTGM or IKD leads to a performance drop in both task orders. This highlights the significance of each component in our framework.

4.4 Analysis and Discussion

4.4.1 Results Analysis

We provide a more detailed analysis of the experimental results, specifically examining why the average accuracy of QA tasks in Order 2 does not achieve the best performance. In Order 2, the last four tasks follow the sequence: *Video QA* → *Image QA* → *Audio Captioning* → *Audio QA*, where QA tasks dominate. Consequently, the task type shift problem has a greater impact on captioning tasks than on QA tasks. For the baseline methods, as they focus less on addressing the task type shift problem, they prioritize QA tasks in the later stages of Order 2 rather than preserving knowledge from earlier tasks. This explains why most baseline methods perform better on QA tasks in Order 2 compared to Order 1. Nevertheless, our MoInCL still outperforms all other baselines in terms of average accuracy of QA tasks, except for EWF, where the difference is marginal. Additionally, MoInCL exhibits a lower average forgetting ratio compared to all baselines in both orders, and achieves lower forgetting ratio on each single task. Moreover, MoInCL maintains more stable performance across both task orders, further demonstrating its robustness.

4.4.2 Task Transfer Effectiveness

To investigate the mutual impact between different tasks, we evaluate the positive knowledge transfer across tasks that share the same modality or task type. Specifically, we conduct experiments to determine whether training on one task benefits a subsequent task within the same modality or task type. The experimental results are presented in Tab. 6. As shown, transferring the captioning ability from the image captioning task improves the CIDEr score of the video captioning task from 47.12 to 48.03. Similarly, transferring the question-answering capability from the video QA task enhances the accuracy of the audio QA task from

Table 6: Experimental results on task transfer effectiveness. We evaluate modality transfer effectiveness within the same task type and task type transfer effectiveness within the same modality.

Modality Transfer				Task Type Transfer	
Video Cap	Image Cap → Video Cap	Audio QA	Video QA → Audio QA	Audio QA	Audio Cap → Audio QA
47.12	48.03	58.28	59.94	58.28	61.75
Task Type Transfer					
Image QA	Image Cap → Image QA	Video Cap	Video QA → Video Cap	Image Cap	Image QA → Image Cap
35.00	36.50	47.12	51.25	77.50	81.93

58.28 to 59.94. These results further demonstrate that transferring knowledge from a previous task to a new task with the same task type enhances the performance of this new task. Additionally, the audio QA ability is enhanced by transferring knowledge from the learned audio captioning task, improving accuracy from 58.28 to 61.75. Similarly, positive knowledge transfer is observed within the image and video modalities, further demonstrating the benefits of transferring knowledge across tasks within the same modality.

4.4.3 Analysis on the Computational Cost

For each experiment, *i.e.*, training a single baseline method or our MoInCL, we use a single RTX A6000 Ada GPU with 48GB of memory. Compared to the pure fine-tuning baseline, the average training time for our MoInCL increases by approximately 40% per epoch, while the inference time remains the same. For example, during training on the audio captioning task with the AudioCaps dataset, pure fine-tuning takes around 45 minutes per epoch, and our method requires approximately 64 minutes per epoch.

5 Conclusion

In this paper, we explore the Modality-Inconsistent Continual Learning (MICL), a novel and practical continual learning scenario of Multimodal Large Language Models (MLLMs). To address the introduced MICL, we propose MoInCL, which incorporates a Pseudo Targets Generation Modul and an Instruction-based Knowledge Distillation constraint to mitigate the catastrophic forgetting caused by the inherent task type shift and modality shift problem in the context of MICL. Experiments on six multimodal incremental tasks demonstrate the effectiveness of our proposed MoInCL. This paper introduces a new direction for the continual learning of MLLMs.

Broader Impact. Our proposed continual modality-inconsistent continual learning allows the MLLMs to adapt to new modalities and task types without full retraining, which could enhance efficiency and privacy by reducing the need to transmit and store sensitive data.

Limitations

Our Modality-Inconsistent Continual Learning (MICL) introduces a novel and practical continual learning scenario by incorporating inconsistent modalities and varying task types across incremental tasks. However, the scope of our work is constrained by the limited number of modalities (audio, image, and video) and task types (captioning and question-answering) included in the experiments. This restricts the generalizability of MICL to scenarios involving a broader range of modalities and task types. Another limitation lies in the pseudo QA pairs generated by PTGM, which may not fully capture the complete answer space of prior QA tasks, leading to incomplete supervision when mitigating the task type shift from QA to captioning tasks. These imperfect pseudo targets may thus still hinder a full resolution of the task type shift problem.

In the future, we plan to enhance our MICL framework by incorporating additional modalities, such as depth, 3D, or even joint inputs like joint audio-visual modalities. We also aim to introduce a broader range of task types, such as reasoning, grounding, decision-making, etc. Furthermore, scaling up MICL to larger datasets within each task is also a key objective to better enable the model to address the complexity and diversity of real-world multimodal tasks in continual learning.

References

Hongjoon Ahn, Jihwan Kwak, Subin Lim, Hyeonsu Bang, Hyojun Kim, and Taesup Moon. Ss-il: Separated softmax for incremental learning. In *Proceedings of the IEEE/CVF International conference on computer vision*, pp. 844–853, 2021.

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

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

Xusheng Cao, Haori Lu, Linlan Huang, Xialei Liu, and Ming-Ming Cheng. Generative multi-modal models are good class incremental learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28706–28717, 2024.

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and Marc'Aurelio Ranzato. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486*, 2019.

Jinpeng Chen, Runmin Cong, Yuzhi Zhao, Hongzheng Yang, Guangneng Hu, Horace Ip, and Sam Kwong. SEFE: Superficial and essential forgetting eliminator for multimodal continual instruction tuning. In *Forty-second International Conference on Machine Learning*, 2025.

Sanyuan Chen, Yu Wu, Chengyi Wang, Shujie Liu, Daniel Tompkins, Zhuo Chen, Wanxiang Che, Xiangzhan Yu, and Furu Wei. Beats: Audio pre-training with acoustic tokenizers. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning*, volume 202, pp. 5178–5193. PMLR, 2023.

Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, et al. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. *arXiv preprint arXiv:2406.07476*, 2024.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *Computer vision–ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, proceedings, part XX 16*, pp. 86–102. Springer, 2020.

Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

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

Sreyan Ghosh, Zhifeng Kong, Sonal Kumar, S Sakshi, Jaehyeon Kim, Wei Ping, Rafael Valle, Dinesh Manocha, and Bryan Catanzaro. Audio flamingo 2: An audio-language model with long-audio understanding and expert reasoning abilities. In *Forty-second International Conference on Machine Learning*, 2025.

Arushi Goel, Sreyan Ghosh, Jaehyeon Kim, Sonal Kumar, Zhifeng Kong, Sang-gil Lee, Chao-Han Huck Yang, Ramani Duraiswami, Dinesh Manocha, Rafael Valle, et al. Audio flamingo 3: Advancing audio intelligence with fully open large audio language models. *arXiv preprint arXiv:2507.08128*, 2025.

Haiyang Guo, Fanhu Zeng, Ziwei Xiang, Fei Zhu, Da-Han Wang, Xu-Yao Zhang, and Cheng-Lin Liu. HiDe-LLaVA: Hierarchical decoupling for continual instruction tuning of multimodal large language model. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13572–13586. Association for Computational Linguistics, 2025.

Jinghan He, Haiyun Guo, Ming Tang, and Jinqiao Wang. Continual instruction tuning for large multimodal models. *arXiv preprint arXiv:2311.16206*, 2023.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Tianyu Huai, Jie Zhou, Xingjiao Wu, Qin Chen, Qingchun Bai, Ze Zhou, and Liang He. Cl-moe: Enhancing multimodal large language model with dual momentum mixture-of-experts for continual visual question answering. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 19608–19617, 2025.

Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. Audiocaps: Generating captions for audios in the wild. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 119–132, 2019.

Sanghwan Kim, Lorenzo Noci, Antonio Orvieto, and Thomas Hofmann. Achieving a better stability-plasticity trade-off via auxiliary networks in continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11930–11939, 2023.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

Zhifeng Kong, Arushi Goel, Rohan Badlani, Wei Ping, Rafael Valle, and Bryan Catanzaro. Audio flamingo: A novel audio language model with few-shot learning and dialogue abilities. In *Forty-first International Conference on Machine Learning*, 2024.

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

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pp. 12888–12900. PMLR, 2022b.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023.

Mei Li, Yuxiang Lu, Qinyan Dai, Suizhi Huang, Yue Ding, and Hongtao Lu. BECAME: Bayesian continual learning with adaptive model merging. In *Forty-second International Conference on Machine Learning (ICML)*, 2025.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

Samuel Lipping, Parthasaarathy Sudarsanam, Konstantinos Drossos, and Tuomas Virtanen. Clotho-aqa: A crowdsourced dataset for audio question answering. In *2022 30th European Signal Processing Conference (EUSIPCO)*, pp. 1140–1144. IEEE, 2022.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2023.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26296–26306, 2024.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3195–3204, 2019.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. Cross-task generalization via natural language crowdsourcing instructions. In *60th Annual Meeting of the Association for Computational Linguistics, ACL 2022*, pp. 3470–3487. Association for Computational Linguistics (ACL), 2022.

Shentong Mo, Weiguo Pian, and Yapeng Tian. Class-incremental grouping network for continual audio-visual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7788–7798, 2023.

Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In *International conference on machine learning*, pp. 2642–2651. PMLR, 2017.

Oleksiy Ostapenko, Mihai Puscas, Tassilo Klein, Patrick Jahnichen, and Moin Nabi. Learning to remember: A synaptic plasticity driven framework for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11321–11329, 2019.

Artemis Panagopoulou, Le Xue, Ning Yu, Junnan Li, Dongxu Li, Shafiq Joty, Ran Xu, Silvio Savarese, Caiming Xiong, and Juan Carlos Niebles. X-instructblip: A framework for aligning x-modal instruction-aware representations to llms and emergent cross-modal reasoning. *arXiv preprint arXiv:2311.18799*, 2023.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Weiguo Pian, Shentong Mo, Yunhui Guo, and Yapeng Tian. Audio-visual class-incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7799–7811, 2023.

Weiguo Pian, Yiyang Nan, Shijian Deng, Shentong Mo, Yunhui Guo, and Yapeng Tian. Continual audio-visual sound separation. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

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

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 2001–2010, 2017.

Zechao Sun, Haolin Jin, Weitong Chen, and Luping Zhou. Awf: Adaptive weight fusion for enhanced class incremental semantic segmentation. *arXiv preprint arXiv:2409.08516*, 2024.

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

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Jia-Wen Xiao, Chang-Bin Zhang, Jiekang Feng, Xialei Liu, Joost van de Weijer, and Ming-Ming Cheng. Endpoints weight fusion for class incremental semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7204–7213, 2023.

Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. Video question answering via gradually refined attention over appearance and motion. In *Proceedings of the 25th ACM international conference on Multimedia*, pp. 1645–1653, 2017.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5288–5296, 2016.

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

Jiazuo Yu, Haomiao Xiong, Lu Zhang, Haiwen Diao, Yunzhi Zhuge, Lanqing HONG, Dong Wang, Huchuan Lu, You He, and Long Chen. LLMs can evolve continually on modality for x-modal reasoning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Fanhu Zeng, Fei Zhu, Haiyang Guo, Xu-Yao Zhang, and Cheng-Lin Liu. Modalprompt: Dual-modality guided prompt for continual learning of large multimodal models. *arXiv preprint arXiv:2410.05849*, 2024.

Boqiang Zhang, Kehan Li, Zesen Cheng, Zhiqiang Hu, Yuqian Yuan, Guanzheng Chen, Sicong Leng, Yuming Jiang, Hang Zhang, Xin Li, et al. Videollama 3: Frontier multimodal foundation models for image and video understanding. *arXiv preprint arXiv:2501.13106*, 2025.

Hang Zhang, Xin Li, and Lidong Bing. Video-LLaMA: An instruction-tuned audio-visual language model for video understanding. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 543–553, 2023a.

Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023b.

Junhao Zheng, Qianli Ma, Zhen Liu, Binquan Wu, and Huawen Feng. Beyond anti-forgetting: Multimodal continual instruction tuning with positive forward transfer. *arXiv preprint arXiv:2401.09181*, 2024.

Appendix

In this appendix, we provide the process of generating three-round QA pairs from captions in Sec. A. We also include dataset details in Sec. B. Experimental results on additional task orders, upper bound results, detailed results of each task, and qualitative analysis are provided in Sec. C, D, E, and F, respectively.

A Three-Round QA Pairs Generation from Captions

Inspired by the question answering text generation process in (Panagopoulou et al., 2023), we adopt a similar three-round QA pair generation process from captions in our proposed Pseudo Targets Generation Module (PTGM). Given a caption from the dataset of the current captioning task $\mathcal{T}_i$, the objective is to generate a QA pair to address the task type shift problem when training on a captioning task within a seen modality. This process relies entirely on prompt engineering, where the caption is used as input to the pre-trained Large Language Model (LLM) component of our Multimodal Large Language Model (MLLM). Please note that, the LLM component employed in this process uses pre-trained weights, *i.e.*, the weights that are not fine-tuned on our incremental tasks.

In Round 1, the LLM takes an input with the format of:

> `Given the` M_i `context: "`$\boldsymbol{y}$`", generate a potential short answer from it. Provide just one or two words. The answer words should be strictly selected from the context. Provide only the answer, nothing else. Answer:`,

where M_i is the modality of the task $\mathcal{T}_i$, $\boldsymbol{y}$ denotes the sampled caption text. And the output of the LLM is used as the temporal short answer $\bar{\boldsymbol{y}}$.

In Round 2, the LLM takes the following prompt as input:

> `Given the` M_i `context: "`$\boldsymbol{y}$`" and the answer: "`$\bar{\boldsymbol{y}}$`", generate a question for the answer that can be inferred from the context. Provide only one question and nothing else. Question:`

and the output of the LLM in Round 2 is the question we aim to generate, which is denoted as $\tilde{\boldsymbol{t}}$.

Finally, in Round 3, the LLM processes the following prompt as input:

> `Answer the question using the given context. The answer should be only one or two words. Context: "`$\boldsymbol{y}$`". Question: "`$\tilde{\boldsymbol{t}}$`". Answer:`

and generates the final short answer $\tilde{\boldsymbol{y}}$.

Based the above three rounds, the pseudo QA pair is obtained, where $\tilde{\boldsymbol{t}}$ represents the pseudo question and $\tilde{\boldsymbol{y}}$ denotes the pseudo answer.

B Dataset Details

Table 7: Details of the datasets used in our experiments.

Task	Dataset	Sample number			
		Total	Training	Validation	Testing
Image Captioning	Flickr30K	31,784	29,783	1,000	1,000
Image QA	OK-VQA	14,055	8,007	1,002	5,046
Audio Captioning	AudioCaps	46,378	44,378	1,000	1,000
Audio QA	Clotho-AQA	10,480	6,181	1,823	2,476
Video Captioning	MSR-VTT	10,000	6,010	1,000	2,990
Video QA	MSVD-QA	50,476	30,904	6,415	13,157

In our experiments, we use the AudioCaps, Flickr30K, MSR-VTT, MSVD-QA, Clotho-AQA, and OK-VQA datasets for Audio Captioning, Image Captioning, Video Captioning, Video QA, Audio QA, and Image QA tasks, respectively. We summarize the details of these data in Tab. 7.

C Experimental Results on Additional Task Orders

Apart from the random task orders in Sec. 4.2, we also conduct additional experiments to further verify the effectiveness and robustness of our proposed MoInCL. Specifically, we construct a new random order: ***Video Captioning*** → ***Image QA*** → ***Image Captioning*** → ***Video QA*** → ***Audio Captioning*** → ***Audio QA***, which we refer to as **Order 3**.

Additionally, we also manually create another task order: ***Image QA*** → ***Video Captioning*** → ***Audio QA*** → ***Image Captioning*** → ***Video QA*** → ***Audio Captioning***, one of the most challenging task orders. This task order enforces frequent alternation between task types, following the pattern: *QA* → *Captioning* → *QA* → *Captioning* → *QA* → *Captioning*, which ensures no two tasks of the same task type appear consecutively. Moreover, this order also introduces more frequent modality shifts, avoiding repetition of the same modality in adjacent tasks. This setting helps mitigate task-recency bias and offers a more rigorous evaluation of each method's ability to generalize under highly dynamic conditions. We refer to this extreme task order as **Order 4**.

The experimental results on these two new task orders are reported in Tab. 8. As shown, our method consistently achieves significant improvements over the baseline methods. Furthermore, its performance remains in line with the results on the original task orders, further highlighting the stability and robustness of our approach.

D Upper Bound Results

We present the testing results of the Upper Bound (joint training) on each task in Tab. 9.

E Detailed Results of Each Task in Both Orders

We also present the detailed testing results for each task across the incremental steps in both orders in Tab. 10 and 11. These results show that our proposed MoInCL exhibits less performance drop compared to the baseline methods, demonstrating its superior ability to address catastrophic forgetting in the proposed Modality-Inconsistent Continual Learning (MICL) scenario.

Table 8: Experimental results on additional two task orders for different continual learning methods. Bold values indicate the best results in each column, while underlined values represent the second-best results in each column.

Methods	Order 3			Order 4		
	Avg. CIDEr ↑	Avg. Acc. ↑	Avg. Forget. ↓	Avg. CIDEr ↑	Avg. Acc. ↑	Avg. Forget. ↓
Fine-tuning	23.14	41.59	53.18%	46.16	19.94	56.11%
EWF (Xiao et al., 2023)	35.46	37.10	46.14%	46.92	36.72	29.66%
PathWeave (Yu et al., 2024)	28.46	40.50	51.73%	47.27	20.54	53.72%
MoInCL (Ours)	**57.18**	**45.39**	**13.07%**	**57.77**	**40.81**	**14.93%**
Upper Bound (Joint training)	66.69	48.97	-	66.69	48.97	-

Table 9: Experimental results of the Upper Bound (joint training) on each task.

Methods	Flickr30k	MSR-VTT	MSVD-QA	OK-VQA	AudioCaps	Clotho-AQA
Upper Bound (Joint training)	80.24	54.76	48.54	38.16	65.07	60.22

F Qualitative Analysis

We present the qualitative results of the Fine-tuning, LwF (Li & Hoiem, 2017), EWC (Kirkpatrick et al., 2017), EWF (Xiao et al., 2023), PathWeave (Yu et al., 2024), and our MoInCL in Fig. 3, 4, 5, 6, 7, and 8, respectively. From these results, we can see that our MoInCL can generate better results with the incremental step increases, demonstrating the better capability in mitigating the catastrophic forgetting problem in our proposed Modality-Inconsistent Continual Learning (MICL) scenario.

Table 10: Detailed testing results for each task across the incremental steps in Order 1. The evaluation metric used for the AudioCaps, Flickr30K, and MSR-VTT datasets is CIDEr score, while that for the MSVD-QA, Clotho-AQA, and OK-VQA datasets is accuracy.

		AudioCaps	Flickr30K	MSVD-QA	Clotho-AQA	OK-VQA	MSR-VTT
Fine-tuning	Step 1	57.66	-	-	-	-	-
	Step 2	26.42	85.83	-	-	-	-
	Step 3	8.34	30.83	47.67	-	-	-
	Step 4	4.28	21.89	44.52	62.64	-	-
	Step 5	4.06	6.49	39.36	57.51	42.41	-
	Step 6	24.50	12.84	23.20	58.16	40.37	54.59
LwF (Li & Hoiem, 2017)	Step 1	57.66	-	-	-	-	-
	Step 2	26.32	86.97	-	-	-	-
	Step 3	4.61	30.38	47.47	-	-	-
	Step 4	0.04	15.96	42.08	63.13	-	-
	Step 5	1.18	6.36	36.16	59.85	42.89	-
	Step 6	26.07	23.90	19.31	61.39	39.92	54.44
EWC (Kirkpatrick et al., 2017)	Step 1	57.66	-	-	-	-	-
	Step 2	38.59	85.27	-	-	-	-
	Step 3	5.67	25.23	46.03	-	-	-
	Step 4	2.04	14.21	43.78	63.29	-	-
	Step 5	3.85	6.31	38.85	56.70	42.09	-
	Step 6	21.64	45.58	17.70	56.99	36.44	49.95
EWF (Xiao et al., 2023)	Step 1	57.66	-	-	-	-	-
	Step 2	49.84	82.73	-	-	-	-
	Step 3	38.01	71.03	44.33	-	-	-
	Step 4	14.19	65.28	44.22	59.69	-	-
	Step 5	15.48	6.08	43.98	59.53	40.75	-
	Step 6	17.50	6.20	9.28	59.41	40.33	50.07
PathWeave (Yu et al., 2024)	Step 1	59.86	-	-	-	-	-
	Step 2	13.54	82.32	-	-	-	-
	Step 3	2.95	12.02	46.00	-	-	-
	Step 4	0.54	9.07	37.28	63.13	-	-
	Step 5	4.19	6.26	28.97	57.84	42.42	-
	Step 6	14.67	34.44	17.60	52.87	38.10	53.48
BECAME (Li et al., 2025)	Step 1	57.66	-	-	-	-	-
	Step 2	55.71	81.46	-	-	-	-
	Step 3	18.34	63.77	45.61	-	-	-
	Step 4	5.81	54.34	45.31	60.18	-	-
	Step 5	9.43	6.04	40.39	59.13	41.13	-
	Step 6	15.67	5.95	15.49	60.26	39.75	51.47
MoInCL (Ours)	Step 1	57.66	-	-	-	-	-
	Step 2	56.58	81.15	-	-	-	-
	Step 3	56.51	82.71	43.38	-	-	-
	Step 4	43.44	81.91	43.43	57.71	-	-
	Step 5	43.01	74.19	43.51	57.51	40.75	-
	Step 6	41.79	73.70	27.51	57.67	41.68	50.44
Upper Bound (Joint training)		65.07	80.24	48.54	60.22	38.16	54.76

Table 11: Detailed testing results for each task across the incremental steps in Order 2. The evaluation metric used for the AudioCaps, Flickr30K, and MSR-VTT datasets is CIDEr score, while that for the MSVD-QA, Clotho-AQA, and OK-VQA datasets is accuracy.

		Flickr30K	MSR-VTT	MSVD-QA	OK-VQA	AudioCaps	Clotho-AQA
Fine-tuning	Step 1	77.50	-	-	-	-	-
	Step 2	64.04	48.03	-	-	-	-
	Step 3	12.12	8.64	46.20	-	-	-
	Step 4	5.86	8.23	39.38	37.13	-	-
	Step 5	9.63	14.05	24.91	17.24	63.19	-
	Step 6	5.41	6.86	31.77	18.65	20.18	60.62
LwF (Li & Hoiem, 2017)	Step 1	77.50	-	-	-	-	-
	Step 2	53.87	48.70	-	-	-	-
	Step 3	10.20	7.80	47.64	-	-	-
	Step 4	7.41	8.44	37.14	36.51	-	-
	Step 5	12.51	18.08	31.44	19.47	59.37	-
	Step 6	6.82	6.90	32.63	21.88	23.39	61.87
EWC (Kirkpatrick et al., 2017)	Step 1	77.50	-	-	-	-	-
	Step 2	62.65	47.73	-	-	-	-
	Step 3	10.45	9.66	45.79	-	-	-
	Step 4	7.19	7.85	37.42	35.90	-	-
	Step 5	12.10	4.24	27.59	21.09	64.40	-
	Step 6	6.91	3.72	27.25	22.29	19.12	63.41
EWF (Xiao et al., 2023)	Step 1	77.50	-	-	-	-	-
	Step 2	69.16	45.30	-	-	-	-
	Step 3	56.10	9.69	45.33	-	-	-
	Step 4	8.26	9.85	44.74	34.95	-	-
	Step 5	8.04	10.24	43.31	33.10	53.36	-
	Step 6	7.86	9.84	42.59	33.50	24.07	61.47
PathWeave (Yu et al., 2024)	Step 1	77.22	-	-	-	-	-
	Step 2	53.60	50.01	-	-	-	-
	Step 3	7.36	8.35	47.87	-	-	-
	Step 4	6.99	7.14	41.17	36.38	-	-
	Step 5	8.01	7.86	33.89	22.27	62.90	-
	Step 6	5.85	6.23	35.58	23.46	21.25	64.34
BECAME (Li et al., 2025)	Step 1	77.50	-	-	-	-	-
	Step 2	77.22	47.64	-	-	-	-
	Step 3	52.16	9.82	47.35	-	-	-
	Step 4	7.24	9.59	46.36	34.48	-	-
	Step 5	8.04	8.81	43.11	31.62	58.74	-
	Step 6	7.36	9.38	40.02	31.06	15.10	58.52
MoInCL (Ours)	Step 1	77.50	-	-	-	-	-
	Step 2	73.59	48.03	-	-	-	-
	Step 3	70.88	48.34	43.11	-	-	-
	Step 4	63.32	47.56	42.27	33.35	-	-
	Step 5	61.91	47.78	42.24	33.46	53.79	-
	Step 6	60.42	46.95	41.99	32.24	46.03	61.43
Upper Bound (Joint training)		80.24	54.76	48.54	38.16	65.07	60.22

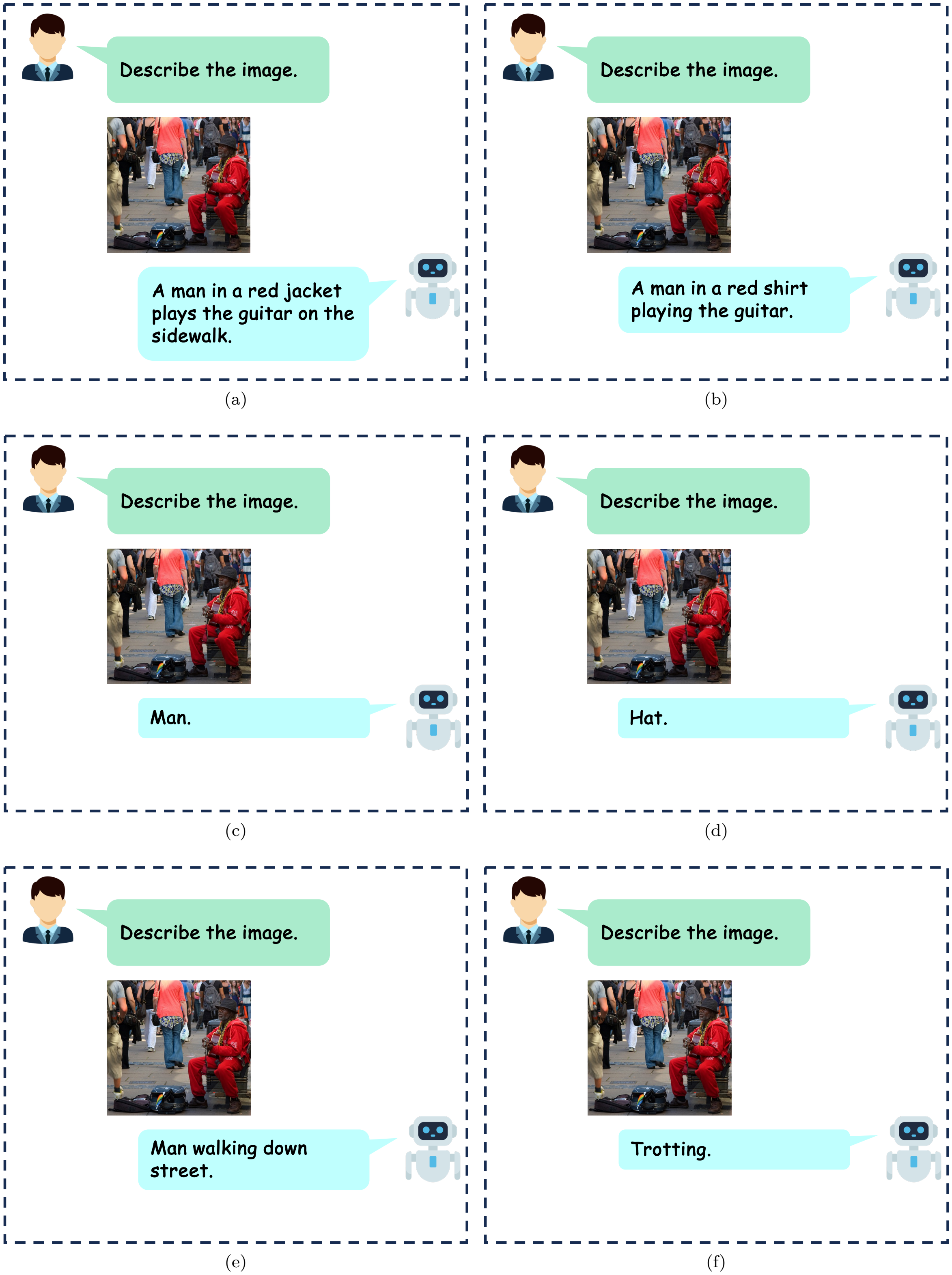
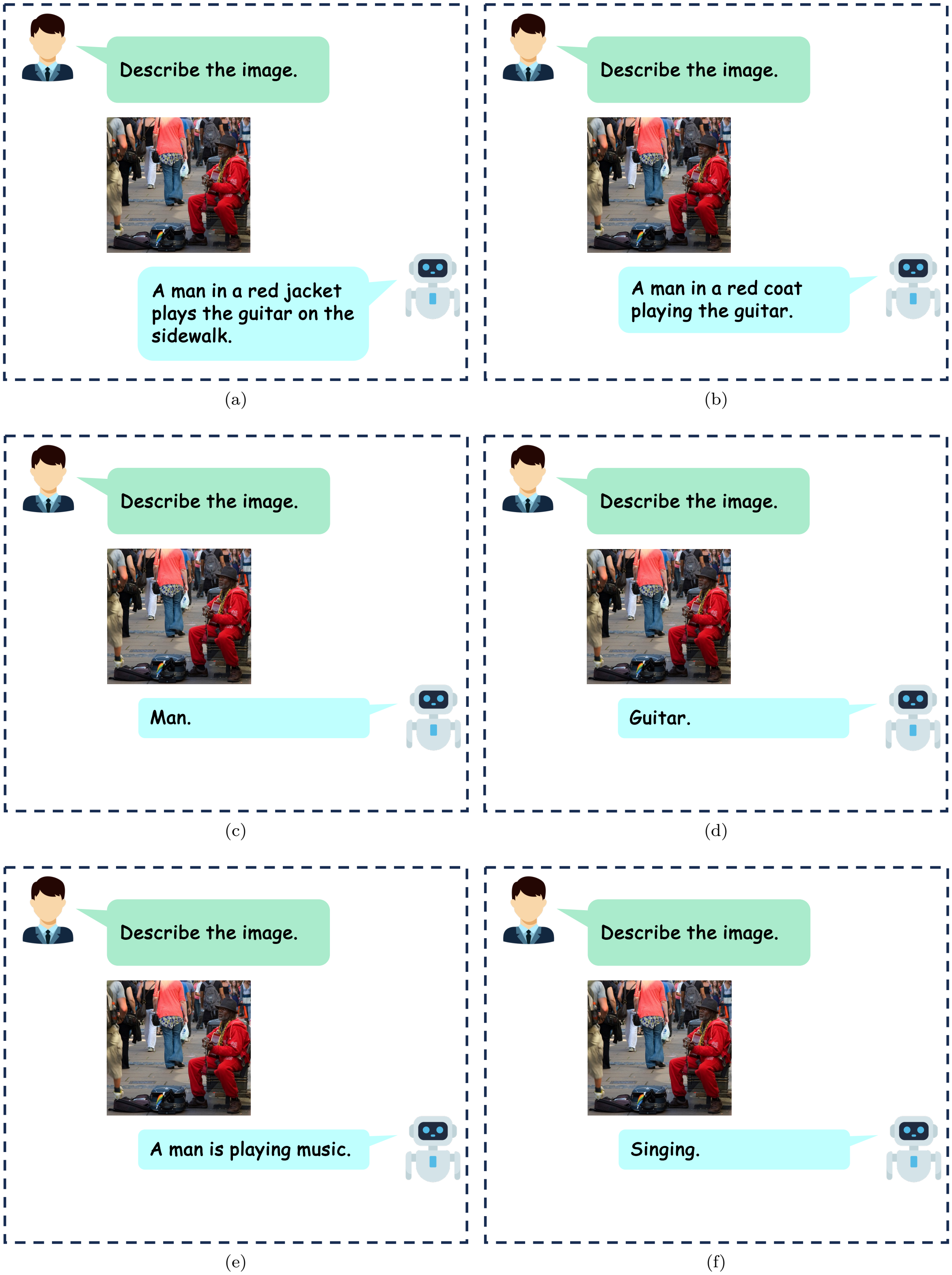


Figure 3: Qualitative results of the Fine-tuning method in Order 2. The sample is randomly selected from the test set of Task 1 (Image Captioning). The results are generated using models trained after after (a) Task 1, (b) Task 2, (c) Task 3, (d) Task 4, (e) Task 5, and (f) Task 6.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 4: Qualitative results of the LwF (Li & Hoiem, 2017) method in Order 2. The sample is randomly selected from the test set of Task 1 (Image Captioning). The results are generated using models trained after after (a) Task 1, (b) Task 2, (c) Task 3, (d) Task 4, (e) Task 5, and (f) Task 6.

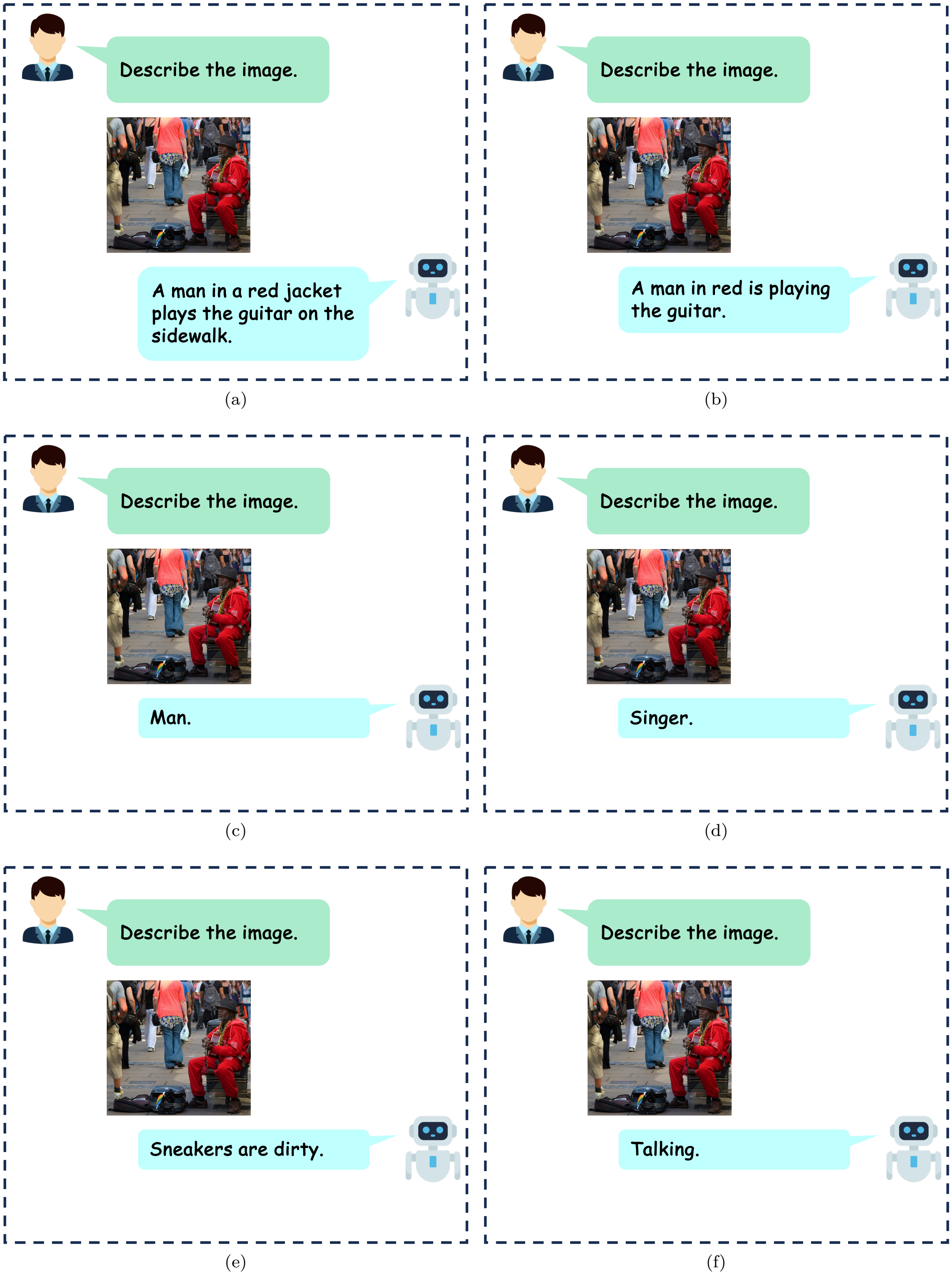


Figure 5: Qualitative results of the EWC (Kirkpatrick et al., 2017) method in Order 2. The sample is randomly selected from the test set of Task 1 (Image Captioning). The results are generated using models trained after after (a) Task 1, (b) Task 2, (c) Task 3, (d) Task 4, (e) Task 5, and (f) Task 6.

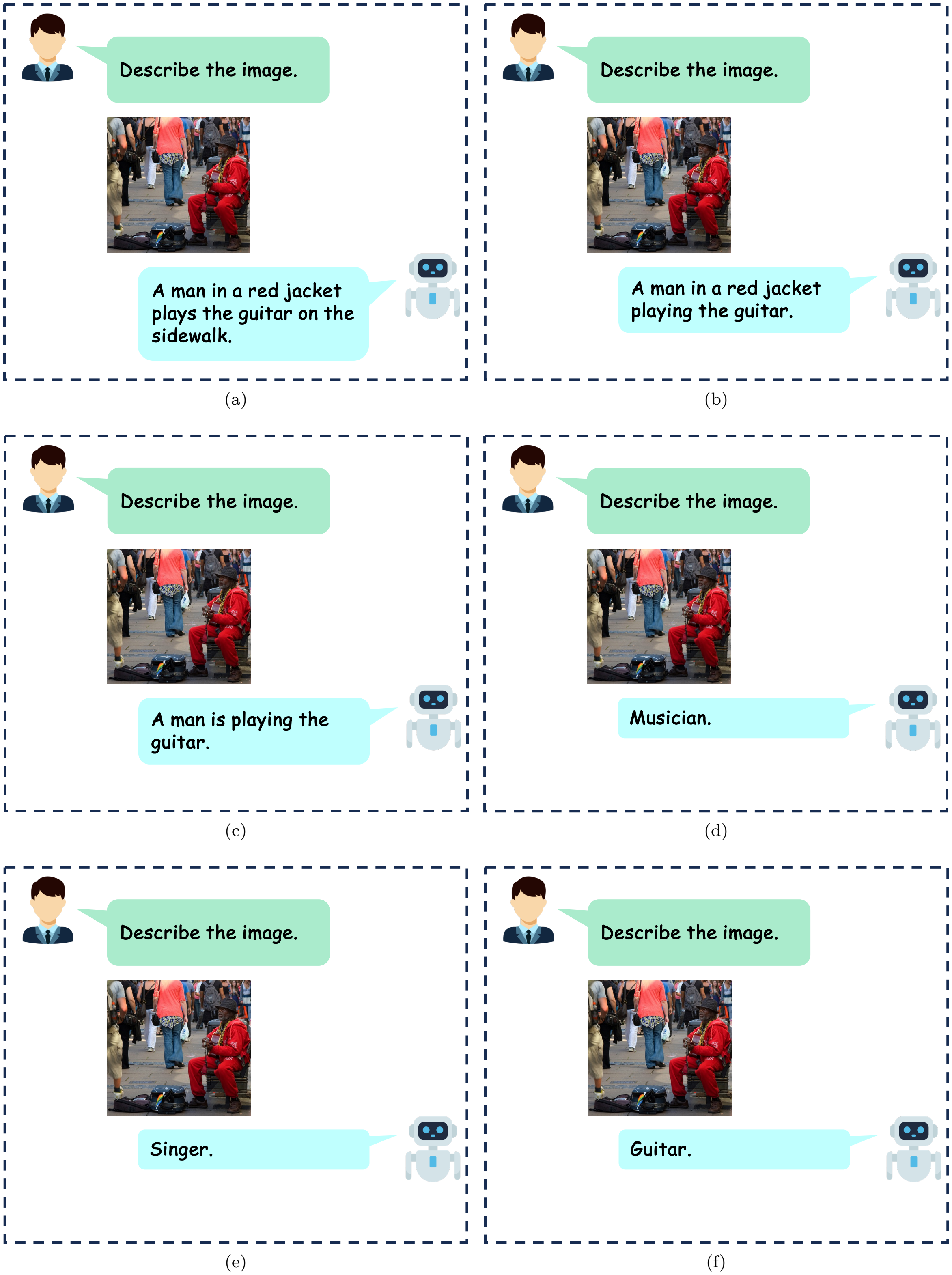
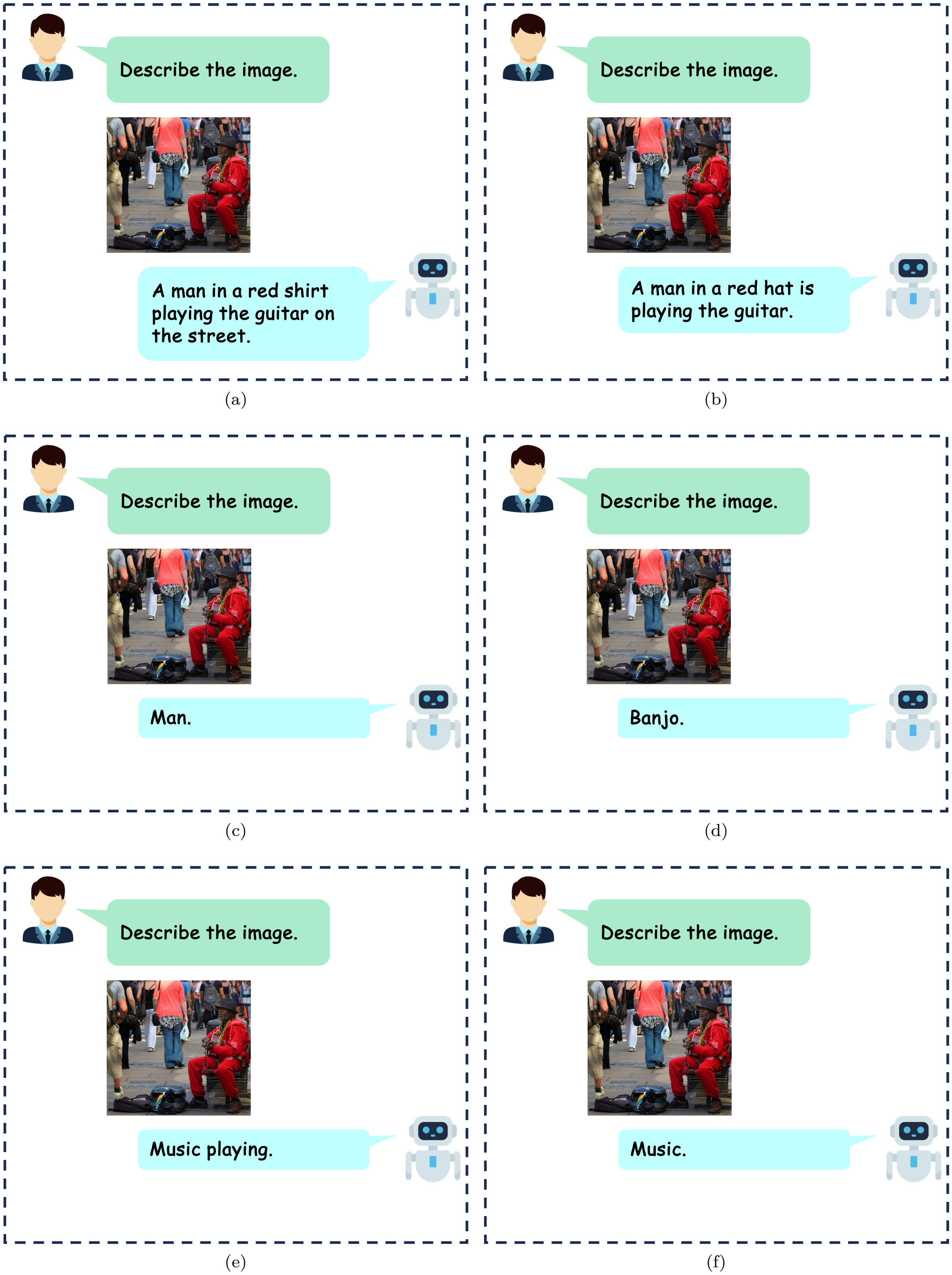
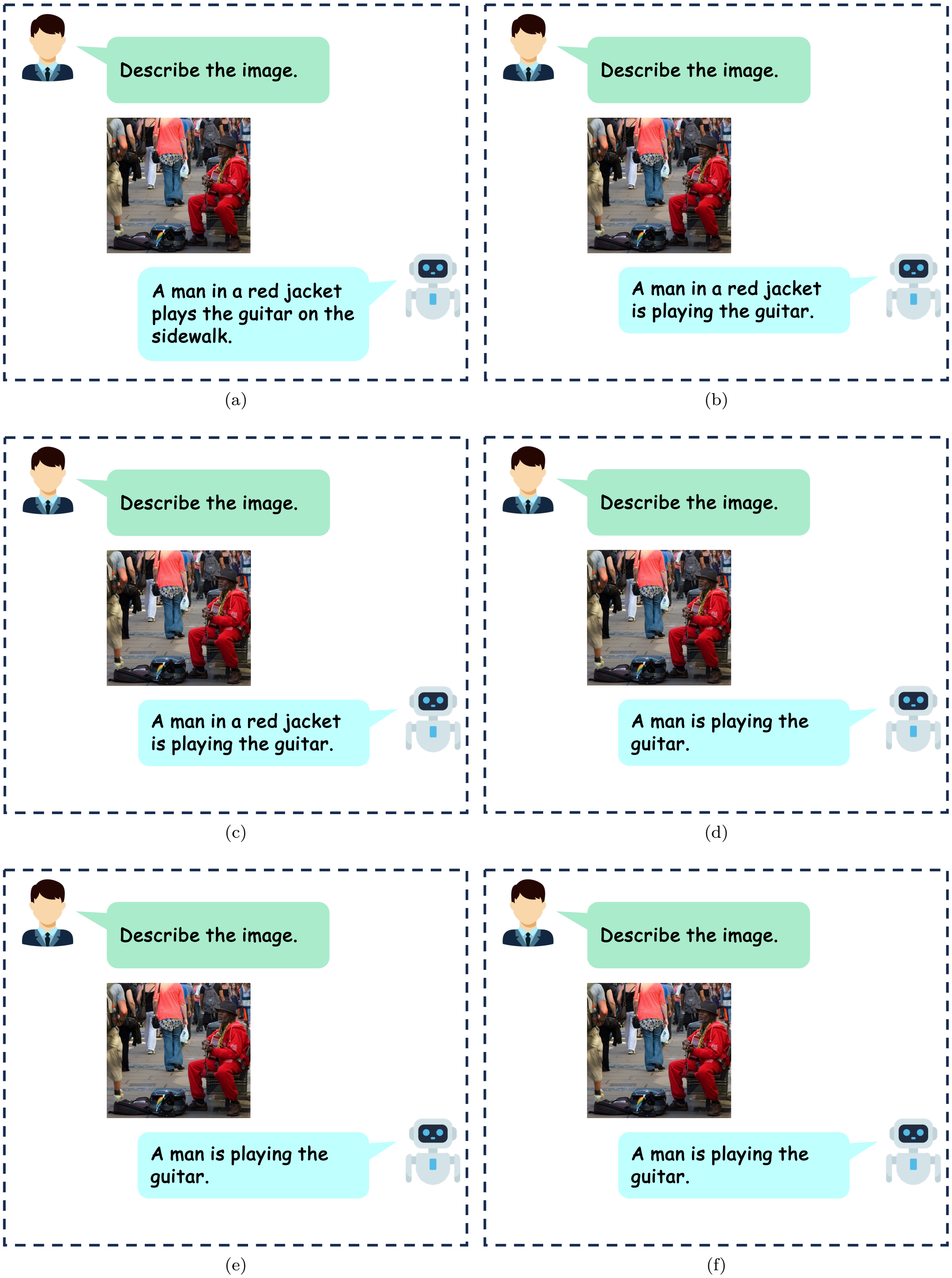


Figure 6: Qualitative results of the EWF (Xiao et al., 2023) method in Order 2. The sample is randomly selected from the test set of Task 1 (Image Captioning). The results are generated using models trained after after (a) Task 1, (b) Task 2, (c) Task 3, (d) Task 4, (e) Task 5, and (f) Task 6.



(a) (b) (c) (d) (e) (f)

Figure 7: Qualitative results of the PathWeave (Yu et al., 2024) method in Order 2. The sample is randomly selected from the test set of Task 1 (Image Captioning). The results are generated using models trained after after (a) Task 1, (b) Task 2, (c) Task 3, (d) Task 4, (e) Task 5, and (f) Task 6.



Describe the image.

A man in a red jacket plays the guitar on the sidewalk.

(a)

Describe the image.

A man in a red jacket is playing the guitar.

(b)

Describe the image.

A man in a red jacket is playing the guitar.

(c)

Describe the image.

A man is playing the guitar.

(d)

Describe the image.

A man is playing the guitar.

(e)

Describe the image.

A man is playing the guitar.

(f)

Figure 8: Qualitative results of our proposed MoInCL in Order 2. The sample is randomly selected from the test set of Task 1 (Image Captioning). The results are generated using models trained after after (a) Task 1, (b) Task 2, (c) Task 3, (d) Task 4, (e) Task 5, and (f) Task 6.