Consistency of Compositional Generalization Across Multiple Levels

Chuanhao Li[1,2*], **Zhen Li**[1*], **Chenchen Jing**[3†], **Xiaomeng Fan**[1], **Wenbo Ye**[2,1], **Yuwei Wu**[1,2†], **Yunde Jia**[2,1]

[1]Beijing Key Laboratory of Intelligent Information Technology,
School of Computer Science & Technology, Beijing Institute of Technology
[2]Guangdong Laboratory of Machine Perception and Intelligent Computing, Shenzhen MSU-BIT University
[3]School of Computer Science, Zhejiang University
{lichuanhao, li.zhen, fanxiaomeng, yewenbo, wuyuwei, jiayunde}@bit.edu.cn
jingchenchen@zju.edu.cn

Abstract

Compositional generalization is the capability of a model to understand novel compositions composed of seen concepts. There are multiple levels of novel compositions including phrase-phrase level, phrase-word level, and word-word level. Existing methods achieve promising compositional generalization, but the consistency of compositional generalization across multiple levels of novel compositions remains unexplored. The consistency refers to that a model should generalize to a phrase-phrase level novel composition, and phrase-word/word-word level novel compositions that can be derived from it simultaneously. In this paper, we propose a meta-learning based framework, for achieving consistent compositional generalization across multiple levels. The basic idea is to progressively learn compositions from simple to complex for consistency. Specifically, we divide the original training set into multiple validation sets based on compositional complexity, and introduce multiple meta-weight-nets to generate sample weights for samples in different validation sets. To fit the validation sets in order of increasing compositional complexity, we optimize the parameters of each meta-weight-net independently and sequentially in a multilevel optimization manner. We build a GQA-CCG dataset to quantitatively evaluate the consistency. Experimental results on visual question answering and temporal video grounding, demonstrate the effectiveness of the proposed framework.

Repository — https://github.com/NeverMoreLCH/CCG

Introduction

Compositionality is an important property of human cognition (Fodor and Pylyshyn 1988). Compositional generalization, the capability of a model to understand novel compositions composed of seen concepts, is critical for artificial intelligence systems to mimic the compositionality. Previous work (Pierrot et al. 2019; Liu et al. 2020; Yang et al. 2023; Xu et al. 2023) has shown that novel compositions exist at multiple levels, including phrase-phrase level, phrase-word level and word-word level, as shown in Figure 1, but the consistency of compositional generalization across multiple levels of novel compositions remains unexplored. The consistency refers to the model's ability to generalize to both phrase-phrase level novel compositions and phrase-word/word-word level novel compositions, which are derived from the words within the phrase-phrase structures. For example, when a model generalize to a phrase-phrase level composition like *"golden dog"+"white cat"*, it should also be able to generalize to phrase-word and word-word level compositions, such as *"golden"+"white cat"* and *"golden"+"cat"*. Understanding *"golden"+"white cat"* and *"golden"+"cat"* are the premise for understanding *"golden dog"+"white cat"*. We investigate if existing vision-and-language models exhibit the consistency. Our observations show that the models even with 37B parameters only achieve ∼40% in the consistency, which indicates that existing models misunderstands the concepts in novel compositions.

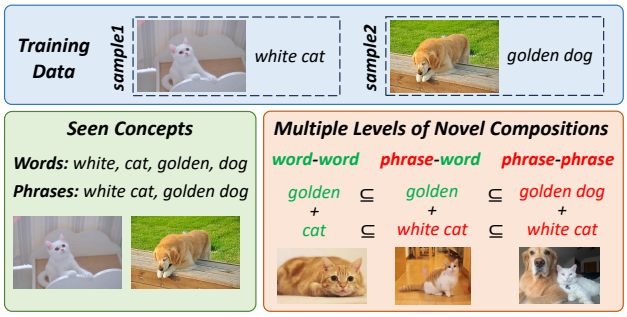

Figure 1: Illustration of multiple levels of novel compositions, including word-word level, phrase-word level and phrase-phrase level.

In this paper, we propose a meta-learning based framework applicable to different types of models for consistent compositional generalization (CCG) across multiple levels of novel compositions. The basic idea behind our framework is to progressively learn compositions from simple to complex by making models learn difficult compositions only after learning simple compositions. To this end, we explicitly distinguish samples with different compositional complexities by generating different sample weights for them,

*Equal contribution
†Corresponding author: Chenchen Jing and Yuwei Wu

and adaptively update the sample weights to ensure learning compositions from simple to complex. Specifically, we divide the original training set into multiple validation sets based on their compositional complexities, and introduce a set of meta-weight-nets to generate sample weights for samples in different validation sets. To learn compositions from simple to complex, we make the model to fit the validation sets progressively in order of increasing compositional complexity, by training the model and the meta-weight-nets independently and sequentially in a multilevel optimization manner. In doing so, multiple levels of compositions are learned in a progressive consistent manner, thus achieving consistent compositional generalization across multiple levels of novel compositions.

To enable the quantitative evaluation for the consistency of compositional generalization across multiple levels, we build a new dataset in the context of visual question answering (VQA), *i.e.*, GQA-CCG, based on the GQA dataset (Hudson and Manning 2019), a large-scale dataset organized for compositional VQA. We filter out samples that include novel compositions at the phrase-phrase level from the val_all split of the GQA dataset to construct a sub-split. We select some samples in the sub-split, and manually annotate them with new questions containing novel compositions at simple levels including phrase-word level and word-word level for them. Based on the annotated questions, we employ GPT-3.5 in an in-context learning manner to generate new questions for other samples in the sub-split, and conduct automatic postprocessing and manual review on the generated questions. Furthermore, we introduce a consistency metric to measure whether a model achieves consistent compositional generalization across multiple levels.

We incorporate various types of methods of two tasks including VQA and temporal video grounding (TVG) into our framework, and conduct experiments on our GQA-CCG dataset, the GQA dataset and the Charades-CG dataset (Li et al. 2022), for validating the effectiveness and generalizability of our framework. Experimental results show that our framework effectively enhances the consistency of compositional generalization across multiple levels and improves the accuracy of compositional generalization at different levels, while maintaining comparable independent and identically distributed (IID) generalization capability.

To sum up, our contributions are as follows: (1) To our knowledge, we are the first to explore the consistency of compositional generalization across multiple levels of novel compositions, which is critical for understanding the concepts in the novel compositions. (2) We propose a meta-learning based framework for consistent compositional generalization across multiple levels of novel compositions. (3) We present a GQA-CCG dataset to evaluate the consistency of compositional generalization across multiple levels of novel compositions for VQA models.

Related work

Compositional Generalization

Compositional generalization has received increasing attention as its importance in mimicking the fundamental compositionality of human cognition (Fodor and Pylyshyn 1988). Numerous benchmarks (Li et al. 2022, 2024) have been proposed to evaluate the compositional generalization capacity, and a substantial amount of research (Wang et al. 2023a; Li et al. 2023b) has been proposed to boost the compositional generalization capacity.

An important property regarding composition is that the process of composition is recursive (Bienenstock 1996), which revealed that compositions exist at multiple levels and compositional generalization capacity can be evaluated at multiple levels. Recently, there have been several attempts that improve compositional generalization capacity at multiple levels. For example, works (Pierrot et al. 2019; Liu et al. 2020) perform recursive reasoning over a decomposed tree layout to achieve compositional generalization at multiple levels. Yang *et al.* (2023) proposed a coarse-to-fine contrastive ranking loss for learning a composite representation that is sensitive to different levels of granularity of both queries and actions. Xu *et al.* (2023) optimized models on multiple virtual sets in a bi-level optimization scheme to handle various levels of novel compositions. These works focus on the accuracy on samples with multiple levels of novel compositions. Differently, we explore the consistency of compositional generalization across multiple levels, requiring a model to generalize not only to complex phrase-phrase level novel compositions but also to their associated simple phrase-word/word-word level novel compositions.

Consistency

Checking for consistency can be likened to conducting a Turing Test (Radziwill and Benton 2017), and the research community has demonstrated significant interest in assessing consistency. Xu *et al.* (2018) explored the consistency across image variations by performing adversarial attack on vision systems, while (Shah et al. 2019; Ribeiro, Guestrin, and Singh 2019) measure the consistency across linguistic variations by generating new questions with the same visual facts in the original question. (Ray et al. 2019; Tascon-Morales, Márquez-Neila, and Sznitman 2023) focus on logical consistency about logically consistent entailed questions or sufficient/necessary conditions. (Selvaraju et al. 2020; Yuan et al. 2021) test perception consistency on low-level perception questions generated for reasoning questions. Jing *et al.* (2022) improved reasoning consistency, which requires a VQA model make correct answers for a series of sub-questions about a compositional question. Other works have also looked into consistency, such as spatial-temporal consistency (Wang et al. 2024) in video-related tasks, multi-view consistency (Yang et al. 2024) and 2D-3D relational consistency (Zhang, Luo, and Lei 2024) in 3D-related tasks. By contrast, we focus on the consistency of compositional generalization across multiple levels, which is critical for understanding novel compositions but remains unexplored.

Framework

Overview

The overview of the proposed framework is shown in Figure 2. In our framework, we make models learn composi-

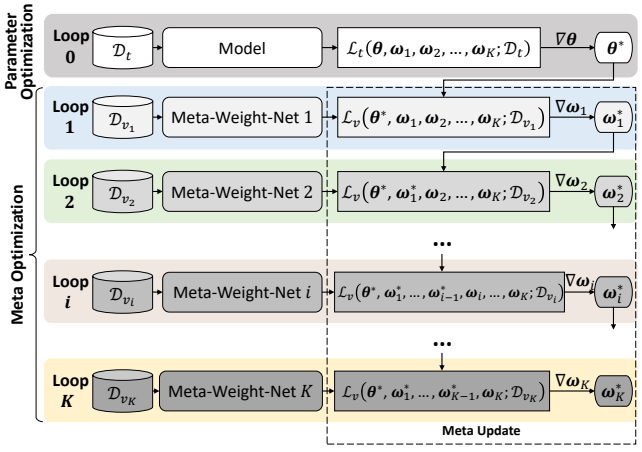

Figure 2: Overview of the proposed framework.

tions from simple to complex by progressively fitting samples in order of increasing compositional complexity. For a training set $\mathcal{D}_t$, we first divide $\mathcal{D}_t$ into multiple validation sets $\{\mathcal{D}_{v_i}\}_{i=1}^{K}$ based on the compositional complexity of samples. A larger i indicates more complex samples in $\mathcal{D}_{v_i}$. Then we introduce K meta-weight-nets into the model, and use i-th meta-weight-net to generate sample weights for the samples in $\mathcal{D}_{v_i}$. Finally, we train the model and the meta-weight-nets via a multilevel optimization process (Migdalas, Pardalos, and Värbrand 2013; Choe et al. 2023), where parameters of the model and different meta-weight-nets are optimized by their own unique objectives in a certain order.

To sum up, we learn compositions in a progressive and consistent manner by: (1) constructing validation sets to validate different training objectives of fitting samples with different complexities; (2) using meta-weight-nets to generate sample weights that control which samples should be learned; (3) updating the meta-weight-nets to fit the validation sets from simple to complex to control when to learn which samples.

Validation Set Construction

To learn samples with different levels of compositional complexity, we divide the original training set to construct multiple validation sets. Each validation set is expected to contain samples with a certain level of compositional complexity. For clarity, we introduce how to construct validation sets in the context of VQA, as shown in Figure 3. VQA requires models to learn to provide a correct answer A for a natural language question Q about an image V. For a training set $\mathcal{D}_t$ of VQA, compositional generalization is the capacity of a VQA model trained on $\mathcal{D}_t$ to correctly answer questions with novel compositions composed of primitives (*i.e.*, words and phrases) seen in $\mathcal{D}_t$.

Specifically, given a sample $(Q, V, A) \in \mathcal{D}_t$, we first use the benepar toolkit (Kitaev, Cao, and Klein 2019) to extract phrases in Q. The phrases are denoted as $\{P_i\}_{i=1}^{N_P}$, where N_p represent the number of phrases. Generally, the compositional complexity is proportional to the length of the longest phrase in a question, with longer longest phrases indicating more complex questions. As a result, we count the length of the longest phrase in the question as an approximated compositional complexity, and denote it as $\mathrm{L}(Q)$. Next, we count the number of samples whose longest phrase length is $L(Q)$ using a function $S(\cdot)$, and denote the number as $S(L(Q))$. Finally, we construct validation sets according to two principles: (1) Samples with similar compositional complexity should be placed in the same validation set as much as possible. (2) The number of samples in each validation set should be as close as possible. Based on these two principles, we construct the i-th validation set by using

$$\begin{aligned}\mathcal{D}_{v_i} = \Big\{ & (Q,V,A) | (Q,V,A) \in \mathcal{D}_t, \\ & \sum\nolimits_{1\le j\le L(Q)} S(j) \ge \lfloor \max(|\mathcal{D}_t| \times (i-1)/K, 0) \rfloor, \\ & \sum\nolimits_{1\le j\le L(Q)} S(j) < \lfloor \max(|\mathcal{D}_t| \times i/K, |\mathcal{D}_t|) \rfloor \Big\},\end{aligned} \tag{1}$$

where $|\cdot|$ is the sample number of the input dataset, and K is a hyperparameter denoting the number of expected validation sets. In doing so, each sample in the original training set is assigned to a unique validation set and satisfies $\sum_{1\le i\le K} \mathcal{D}_{v_i} = \mathcal{D}_t$, while samples in different validation sets have different complexities, the larger i is, the more complex the samples in $\mathcal{D}_{v_i}$ are.

Sample Weight Generation

As different samples have different importance in learning a certain level of compositions, we introduce a set of meta-weight-nets (Shu et al. 2019) to automatically generate sample weights for samples in different validation sets, to explicitly control which samples should be learned. Each meta-weight-net is only responsible for generating sample weights for a specific validation set. These meta-weight-nets use the

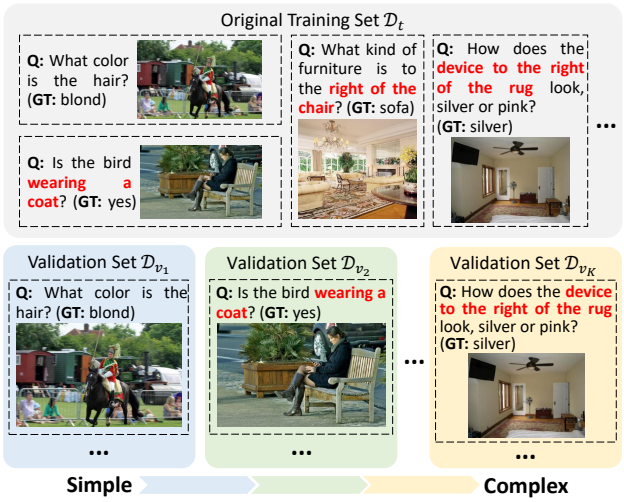

Figure 3: Validation set construction in the context of VQA, where the words in red font denote the longest phrase in the question.

same arc hitectures—three stacked fully connected layers and a sigmoid layer, but have different parameters.

For a sample $d \in \mathcal{D}_{v_i}$, i-th meta-weight-net is used to generate its sample weight. The meta-weight-net accepts question features as input to output sample weight by

$$w_d = f_i(\boldsymbol{\omega}_i; g(d)), \tag{2}$$

where $f_i(\cdot)$ and $\boldsymbol{\omega}_i$ denote the feedforward process and the parameters of the i-th meta-weight-net, respectively. $g(\cdot)$ is the question feature of the input sample extracted by the language encoder of the VQA model.

Multilevel Optimization

To make the VQA model fit the validation sets progressively from simple to complex, we optimize the model and the meta-weight-nets in a multilevel optimization process. The process consists of two continuously alternating steps: parameter optimization and meta optimization. During parameter optimization, we freeze the meta-weight-nets, and optimize the model to fit current sample weights. During meta optimization, we update sample weights by optimizing meta-weight-nets to fit the validation sets. The two steps are performed in sequence until the model training converges. For model testing, we conventionally test the model and exclude the meta-weight-nets. Below we first introduce the formulation of the multilevel optimization process, and then discuss the details of the parameter optimization and the meta optimization in the process.

Formulation. Let $\boldsymbol{\theta}$ and $\boldsymbol{\omega}_i$ denote the parameters of the model and the i-th meta-weight-net, respectively, the multilevel optimization process can be formulated as sequentially performing the following nested loops from $LOOP_0$ to $LOOP_K$:

$$\begin{aligned}
LOOP_K &: \boldsymbol{\omega}_K^* = \arg\min_{\boldsymbol{\omega}_K} \mathcal{L}_v(\boldsymbol{\theta}^*, \{\boldsymbol{\omega}_j^*\}_{j=1}^{K-1}, \boldsymbol{\omega}_K; \mathcal{D}_{v_K}) \\
&\ddots \\
LOOP_2 &: \text{ s.t. } \boldsymbol{\omega}_2^* = \arg\min_{\boldsymbol{\omega}_2} \mathcal{L}_v(\boldsymbol{\theta}^*, \boldsymbol{\omega}_1^*, \boldsymbol{\omega}_2, ..., \boldsymbol{\omega}_K; \mathcal{D}_{v_2}) \\
LOOP_1 &: \quad \text{s.t. } \boldsymbol{\omega}_1^* = \arg\min_{\boldsymbol{\omega}_1} \mathcal{L}_v(\boldsymbol{\theta}^*, \boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_K; \mathcal{D}_{v_1}) \\
LOOP_0 &: \quad \text{s.t. } \boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_t(\boldsymbol{\theta}, \boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_K; \mathcal{D}_t),
\end{aligned} \tag{3}$$

where $\mathcal{L}_t$ and $\mathcal{L}_v$ denote the training loss and validation loss, respectively. The $\mathcal{L}_t$ is determined by the selected model, as different models are trained by different losses. Given a model trained by loss $\mathcal{L}$, by applying the proposed framework, $\mathcal{L}_t$ can be given by

$$\mathcal{L}_t(\boldsymbol{\theta}, \boldsymbol{\omega}_1, \boldsymbol{\omega}_2, ..., \boldsymbol{\omega}_K; \mathcal{D}_t) = \sum_{1 \le i \le K} \sum_{d \in \mathcal{D}_{v_i}} w_d \mathcal{L}(\boldsymbol{\theta}; d), \tag{4}$$

where w_d is the sample weight of d calculated by Eq. (2). Furthermore, $\mathcal{L}_v$ can be written as

$$\mathcal{L}_v(\boldsymbol{\theta}^*, \boldsymbol{\omega}_1^*, ..., \boldsymbol{\omega}_{i-1}^*, \boldsymbol{\omega}_i, ..., \boldsymbol{\omega}_K; \mathcal{D}_{v_i}) = \sum_{d \in \mathcal{D}_{v_i}} \mathcal{L}(\boldsymbol{\theta}^*(\omega_i); d). \tag{5}$$

Parameter Optimization. Parameter optimization aims to find the optimal parameters $\boldsymbol{\theta}^*$ such that minimizing the training loss $\mathcal{L}_t$. Specifically, for the initial parameter $\boldsymbol{\theta}^{(0)}$, we train for T_p iterations to update $\boldsymbol{\theta}$ by performing gradient descent. At each iteration ($i = 1, ..., T_p$), we update the parameter as follows:

$$\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}^{(i)} - \beta_{\boldsymbol{\theta}}^{(i)} \nabla \boldsymbol{\theta}^{(i)}, \tag{6}$$

where $\nabla \boldsymbol{\theta}^{(i)} = \frac{d\mathcal{L}_t}{d\boldsymbol{\theta}^{(i)}}$, and $\beta_{\boldsymbol{\theta}}^{(i)}$ is the learning rate of $\boldsymbol{\theta}$ at iteration i. We take the final parameter $\boldsymbol{\theta}^{(T_p)}$ as the optimal parameter $\boldsymbol{\theta}^*$.

Meta Optimization. At this step, we find the optimal parameter $\boldsymbol{\omega}_i^*$ from $i = 1$ to K sequentially, for progressively fitting the validation sets from simple to complex. For i-th meta-weight-net, we train it for T_m iterations to update its parameter $\boldsymbol{\omega}_i$ to the optimal $\boldsymbol{\omega}_i^*$. At each iteration ($j = 1, ..., T_m$), we perform a meta update operation to $\boldsymbol{\omega}_i$ as follows:

$$\boldsymbol{\omega}_i^{(j+1)} = \boldsymbol{\omega}_i^{(j)} - \beta_{\boldsymbol{\omega}_i}^{(j)} \nabla \boldsymbol{\omega}_i^{(j)}, \tag{7}$$

where $\nabla \boldsymbol{\omega}_i^{(j)} = \frac{d\mathcal{L}_v}{d\boldsymbol{\omega}_i^{(j)}}$, and $\beta_{\boldsymbol{\omega}_i}^{(j)}$ is the learning rate of $\boldsymbol{\omega}_i$ at iteration j. However, $\nabla \boldsymbol{\omega}_i^{(j)}$ is difficult to directly calculate due to two aspects: (1) There are multiple inevitable computationally onerous matrix-matrix multiplications during calculating $\nabla \boldsymbol{\omega}_i^{(j)}$. (2) The calculation needs to save the calculation graph during multiple iterations, which increases the demand for GPU memory. To solve this issue, we follow (Lorraine, Vicol, and Duvenaud 2020) with implicit function theorem to approximate the best-response Jacobian, and the details are provided in our **repository**.

GQA-CCG Dataset

In this section, we illustrate how we construct GQA-CCG based on the GQA dataset, the overview of which is shown in Figure 4. We use the the train_balanced split and the val_all split of GQA in the process of constructing GQA-CCG, and here we denote them as $\mathcal{D}_t$ and $\mathcal{D}_v$, respectively.

Preparations

Candidate Filter. First, we extract words and phrases by benepar (Kitaev, Cao, and Klein 2019) for all questions in $\mathcal{D}_t$ and $\mathcal{D}_v$, respectively. Then we count seen compositions including phrase-phrase (pp), phrase-word (pw) and word-word (ww) in $\mathcal{D}_t$. Next, we filter out the samples in $\mathcal{D}_v$, whose question has at least a novel phrase-phrase composition. We collected all question-answer pairs (denoted as $[Q_{pp}, A_{pp}]$) of the filtered samples as a candidate set $\mathcal{C}$.

In-context Filter. Based on $\mathcal{C}$, we select M questions for each type of question prefix by a diversity maximization method. For a type of question prefix, we first randomly select a QA pair $[Q_{pp}, A_{pp}]$ of the type to construct an initial set $\mathcal{P} = \{[Q_{pp}, A_{pp}]\}$. Then we find a QA pair, in which the question has the lowest average similarity to all questions in $\mathcal{P}$ of this type. We add the found QA pair to $\mathcal{P}$, and repeat the above step until $|\mathcal{P}| = M$. The similarities between the two questions are computed by the cosine similarity of their

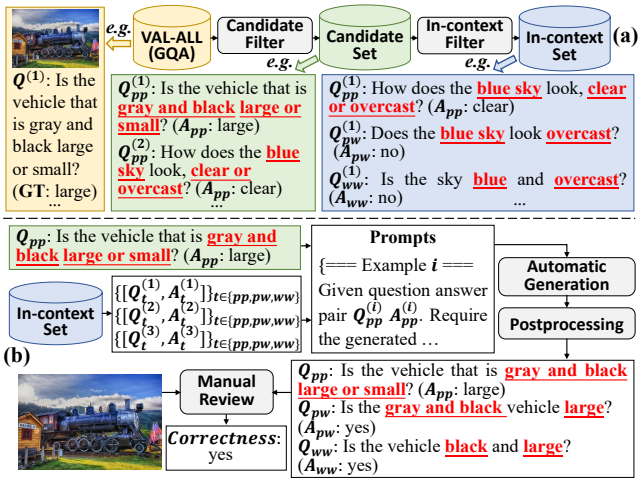

Figure 4: Overview of the pipeline for constructing our GQA-CCG dataset. (a) Preparations for constructing GQA-CCG. (b) Illustration of the sample generation process. The underlined words/phrases in red font denote the components of novel compositions. Detailed prompts are provided in our **repository.**

BERT embeddings (Devlin et al. 2019). As a result, we obtain a set of in-context sets $\{\mathcal{I}_t\}_{t\in\mathcal{T}}$, where $\mathcal{T}$ denotes the set containing all types of question prefix, and $\mathcal{I}_t$ represents the set of selected M QA-pairs with type t. Finally, we update $\mathcal{C} = \mathcal{C} - \mathcal{I}$, where $\mathcal{I} = \sum_{t\in\mathcal{T}} \mathcal{I}_t$, and $\mathcal{I}_t$ can be represented as $\{[Q^{(i)}_{pp}, A^{(i)}_{pp}]\}^M_{i=1}$.

For a QA pair $[Q_{pp}, A_{pp}] \in \mathcal{I}$, where Q_{pp} has a novel phrase-phrase composition p_1-p_2 (notes that p_1 and p_2 can be exchanged), we manually annotate it with: (1) A QA pair $[Q_{pw}, A_{pw}]$ with a novel phrase-word composition p_1-w_2. (2) A QA pair $[Q_{ww}, A_{ww}]$ with a novel word-word composition w_1-w_2. The relationship between p_1, p_2, w_1 and w_2 satisfies

$$w_1 \subsetneq p_1,\ w_2 \subsetneq p_2. \tag{8}$$

After the manual annotation, we rewrite $\mathcal{I}_t$ as $\{T_i\}^M_{i=1}$, where T_i is a triplet that is denoted as $\{[Q^{(i)}_t, A^{(i)}_t]\}_{t\in\{pp,pw,ww\}}$.

Sample Generation Pipeline

Automatic Generation. For a QA pair $[Q_{pp}, A_{pp}] \in \mathcal{C}$ with type t, we first select 3 triplets, in which the questions have the maximum cosine similarity of BERT embeddings with Q from $\mathcal{I}_t$. We denote the novel phrase-phrase composition in $[Q_{pp}, A_{pp}]$ as p_1-p_2, i-th selected triplet as $P_i = \{[Q^{(i)}_t, A^{(i)}_t]\}_{t\in\{pp,pw,ww\}}$, the components of novel compositions in P_i as $p^{(i)}_1$, $p^{(i)}_2$, $w^{(i)}_1$ and $w^{(i)}_2$. Then we iterate over the words in p_2 as w_2. For each w_2, we iterate over the words in p_1 as w_1. For each pair of w_1-w_2, we fill associated infos into the prompt template in Figure 4 (b), and then use GPT-3.5 to generate $\{[Q_t, A_t]\}_{t\in\{pw,ww\}}$ to form a triplet $\{[Q_t, A_t]\}_{t\in\{pp,pw,ww\}}$. Eventually, the generated triplets are collected as $\mathcal{G}$.

Postprocessing. For a triplet $\{[Q_t, A_t]\}_{t\in\{pp,pw,ww\}}$ in $\mathcal{G}$, we denote the novel phrase-phrase composition in Q_{pp} as p_1-p_2, and retain the triplet that if it satisfies: (1) There is a novel word-word composition w_1-w_2 and a novel phrase-word composition p_1-w_2 in Q_{ww} and Q_{pw}, respectively. (2) The relationship between p_1, p_2, w_1 and w_2 satisfies Eq. (8). (3) There are no novel compositions that are not w_1-w_2 and p_1-w_2 in Q_{ww} and Q_{pw}, respectively.

Manual Review. We recruited volunteers to verify the correctness of the generated QA pairs according to the image for the original QA pair. For a triplet $\{[Q_t, A_t]\}_{t\in\{pp,pw,ww\}}$ and the image I for $[Q_{pw}, A_{pw}]$, we retain the triplet if it satisfies both $[Q_{pw}, A_{pw}]$ and $[Q_{ww}, A_{ww}]$ are correct based on I. We add the associated images to the retained triplets, forming our dataset $\mathcal{D}_{CCG}$. We get 18983 samples with novel compositions at different levels that consist of 8702 triplets of $\{[Q_t, A_t, I]\}_{t\in\{pp,pw,ww\}}$. The number of samples with novel compositions at phrase-phrase, phrase-word and word-word level are 5125, 8102 and 5756, respectively.

Consistency Score

To quantitatively evaluate the consistency on our $\mathcal{D}_{CCG}$, we devise a metric $Cons$, which is computed by

$$Cons = \frac{\sum_{T\in\mathcal{D}_{CCG}} \prod_{t\in T} \mathrm{Correct}(t)}{\mathrm{triplet\_num}(\mathcal{D}_{CCG})}, \tag{9}$$

where $\mathrm{Correct}(\cdot)$ is an indicator function, $\mathrm{triplet\_num}(\cdot)$ is a function that counts the triplet number of the input dataset. The value range of $Cons$ is $[0, 1]$, and the larger $Cons$, the better the consistency.

Experiment

Experimental Settings

Datasets. We apply the proposed framework to two tasks, VQA and TVG, to evaluate its effectiveness. For VQA, we evaluate the framework on our GQA-CCG dataset and the GQA dataset (Hudson and Manning 2019). The GQA-CCG dataset is used to test the consistency of compositional generalization across multiple levels and the accuracy of compositional generalization at multiple levels. We use the GQA dataset to test whether our framework is harmful to the IID generalization capability. For TVG, we use the recently released Charades-CG dataset (Li et al. 2022) that contains compositional referring expressions about real-world videos to further test the compositional generalization capability of our framework on different tasks.

Baseline Methods. For VQA, we incorporate our framework into five methods including MAC (Hudson and Manning 2018), LCGN (Hu et al. 2019), MMN (Chen et al. 2021), VL-T5 (Cho et al. 2021) and CFR (Nguyen et al. 2022). and dub the incorporated methods X+MLO, where X is a method name. These methods belong to different types, thus the experiments on these methods allow for a comprehensive assessment of the effectiveness of our framework. For TVG, we apply the proposed framework to MS-2D-TAN (Zhang et al. 2021), which uses a 2D temporal map to model the temporal adjacent relations of video moments, and demonstrates good compositional capability.

Type	Method	Accuracy				Consistency
		overall	*phrase-phrase*	*phrase-word*	*word-word*	
Attention-based	MAC (Hudson and Manning 2018)	62.07	70.97	59.84	57.28	30.82
	+ MLO (Ours)	63.98	72.06	61.78	59.90	34.10
Graph-based	LCGN (Hu et al. 2019)	66.38	76.00	62.92	62.68	38.53
	+ MLO (Ours)	67.53	76.92	64.14	63.86	40.46
NMN-based	MMN(Chen et al. 2021)	70.14	84.87	66.57	62.07	42.41
	+ MLO (Ours)	71.07	**84.95**	67.83	63.26	43.81
Pretrain-based ($\geq$ 7B)	OpenFlamingo (9B) (Awadalla et al. 2023)	53.47	54.15	49.67	58.20	19.58
	BLIP-2 (FlanT5$_{\text{XXL}}$) (Li et al. 2023c)	54.12	58.20	51.42	54.29	28.02
	Otter (7B) (Li et al. 2023a)	55.86	59.14	53.74	55.94	23.85
	LLaVA-1.5-Xtuner (7B) (Contributors 2023b)	65.28	57.66	66.14	70.85	35.89
	XComposer2 (7B) (Dong et al. 2024)	55.08	50.48	56.73	56.85	28.01
	mPLUG-Owl2 (7B) (Ye et al. 2024)	65.08	58.22	65.96	69.96	36.32
	LLaVA-1.6 (7B) (Liu et al. 2024)	61.50	59.06	61.91	63.10	34.52
	CogVLM (17B) (Wang et al. 2023b)	67.47	61.37	68.56	71.37	38.30
	emu2 (37B) (Sun et al. 2024)	68.46	63.22	68.65	**72.86**	40.58
Pretrain-based ($\leq$ 0.2B)	LXMERT (Tan and Bansal 2019)	71.26	80.61	68.58	66.71	45.43
	VL-T5 (Cho et al. 2021)	70.19	77.22	67.66	67.51	42.78
	+ MLO (Ours)	71.08	77.89	68.78	68.25	44.69
	CFR (Nguyen et al. 2022)	72.95	83.95	70.31	66.87	46.46
	+ MLO (Ours)	**74.23**	84.50	**71.81**	68.75	**49.27**

Table 1: Accuracy (%) and Consistency (%) of the state-of-the-art methods on GQA-CCG.

Compositional Generalization Performance

We compare with different types of VQA methods including large vision-language models (LVLMs) varies in parameters (7B to 37B) on GQA-CCG, including MAC (Hudson and Manning 2018), LCGN (Hu et al. 2019), MMN (Chen et al. 2021), OpenFlamingo (Awadalla et al. 2023), BLIP-2 (Li et al. 2023c), Otter (Li et al. 2023a), LLaVa-v1.5-Xtuner (Contributors 2023b), XComposer2 (Dong et al. 2024), mPLUG-Owl2 (Ye et al. 2024), LLaVA-1.6 (Liu et al. 2024), CogVLM (Wang et al. 2023b), emu2 (Sun et al. 2024), LXMERT (Tan and Bansal 2019), VL-T5 (Cho et al. 2021), and CFR (Nguyen et al. 2022). We evaluate pretrain-based models with parameters $\geq$ 7B via VLMEvalKit (Contributors 2023a), which provides model-specific prompts and answer matching rules. We design additional matching patterns for each model with respect to its answer format. For example, we use the matching pattern "The answer is XXX." for XComposer2 as it often answers in this format.

The experimental results on GQA-CCG are listed in Table 1, where "*overall*" represents the accuracy on all test samples of GQA-CCG, and "Consistency" is the consistency score computed by Eq. (9). The "*phrase-phrase*", "*phrase-word*" and "*word-word*" denote the accuracy on samples with corresponding levels of novel compositions. We observe that: (1) CFR+MLO achieves the best performance on both the accuracy and consistency. (*e.g.*, 74.23% and 49.27% in overall accuracy and consistency, respectively). (2) For all five baseline methods of different types, our framework consistently improves their accuracy and consistency (*e.g.*, 3.28% and 2.81% absolute performance gains in consistency for MAC and CFR, respectively). (3) LVLMs are better at simple word-word level compositions than at complex phrase-phrase level and phrase-word level compositions. Although several LVLMs outperform CFR+MLO in word-word accuracy, they have more than thirty times the scale of parameters of CFR+MLO and have been trained on much more VQA samples. These observations show that the proposed framework is efficient in improving not only the consistency but also the accuracy of compositional generalization at multiple levels for different types of baseline methods. Furthermore, LVLMs still struggle to the consistency of compositional generalization, although they've been trained on a large amount of VQA samples.

IID Generalization Performance

The experimental results on GQA are listed in Table 2. We can observe that: (1) Overall, CFR+MLO achieves the best performance among state-of-the-art methods. (2) Compared to baseline methods, our framework improves their performance. (*e.g.*, 0.4% absolute performance gains in accuracy for MAC). The reason for the limited performance improvement of the proposed framework on GQA is that we mainly focus on the compositional generalization capability, which can be viewed as a capability of out-of-distribution (OOD) generalization, while GQA is used more to evaluate the independent and identically distributed (IID) generalization. The experimental results show that our framework is beneficial for IID setting (*e.g.*, GQA) apart from the OOD setting (*e.g.*, GQA-CCG), compared to most existing methods that provide performance gains in the OOD testing at the expense of IID performance (Cho et al. 2023).

Type	Method	test-dev
Attention-based	MAC (Hudson and Manning 2018)	52.43
	+ MLO (Ours)	52.83
Graph-based	LCGN (Hu et al. 2019)	55.63
	+ MLO (Ours)	55.95
NMN-based	MMN (Chen et al. 2021)	59.14
	+ MLO (Ours)	59.32
Pretrain-based (zero-shot)	BLIP-2 (FlanT5$_{XXL}$) (Li et al. 2023c)	44.70
	MiniGPT-4 (Zhu et al. 2023)	43.50
Pretrain-based (fine-tuned)	VL-T5 (Cho et al. 2021)	58.40
	+ MLO (Ours)	58.59
	CFR (Nguyen et al. 2022)	70.27
	+ MLO (Ours)	**70.51**

Table 2: Accuracy (%) of the state-of-the-art methods on GQA (Hudson and Manning 2019).

Meta-Weight-Nets	Multilevel Optimization	GQA-CCG		Charades-CG	
		Acc	Cons	R1@0.5	R5@0.5
-	-	62.07	30.82	42.04	77.16
✓	-	63.12	32.54	43.73	78.39
✓	✓	**63.98**	**34.10**	**44.10**	**78.65**

Table 3: Ablation studies about components of our framework on GQA-CCG and the Novel-Composition split of Charades-CG (Li et al. 2022), on which we use MAC (Hudson and Manning 2018) and MS-2D-TAN (Zhang et al. 2021) as baseline methods, respectively. The performance of baseline methods is shown in the first line.

Ablation Studies

Firstly, we investigate the effectiveness of different components of our framework on the consistency and accuracy of compositional generalization, and the results are shown in Table 3. We evaluate the effectiveness of meta-weight-nets by training them simultaneously rather than in a multilevel optimization manner. We observe that the performance is better than the baseline methods but worse than the methods trained with our full framework, which suggests that introducing meta-weight-nets is effective in improving the capability of compositional generalization and using multi-level optimization can obtain further improvements. More ablation studies are provided in our **repository**, including different variations of validation set construction, different optimization manners.

Analysis of Validation Set Learning Sequence

We analyze the importance of learning validation sets from simple to complex, to find whether learning from complex to simple is also effective to improve the compositional consistency. The experimental results of using different learning procedures are list in Table 4. which shows that learning validation sets from complex to simple (C $\rightarrow$ S) has little improvement over the baseline model, and even suffers a slight decline in some metrics (*e.g.*, 0.14% performance drop in phrase-phrase accuracy). The main reason is that without the accumulation of simple knowledge, it is difficult to directly learn complex knowledge, which is proved in the human cognitive theory (Plass, Moreno, and Brünken 2010).

Learning Sequence	Accuracy				Cons
	overall	*phrase-phrase*	*phrase-word*	*word-word*	
-	62.07	70.97	59.84	57.28	30.82
C $\rightarrow$ S	62.05	70.83	59.89	57.28	30.95
S $\rightarrow$ C	**63.98**	**72.06**	**61.78**	**59.90**	**34.10**

Table 4: Performance of using different learning procedures of validation sets on GQA-CCG, where we use MAC (Hudson and Manning 2018) as baseline methods (the first line), "C $\rightarrow$ S" and "S $\rightarrow$ C" denote learning from complex to simple and simple to complex, respectively.

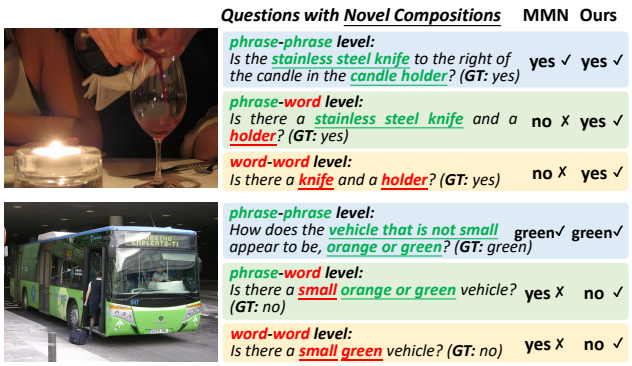

Figure 5: Qualitative comparisons between MMN+MLO (Ours) and MMN (Chen et al. 2021).

Qualitative Analysis

We provide several qualitative examples in the context of VQA in Figure 5. For a triplet that consists of questions with novel compositions at different levels, we provide the predictions of MMN+MLO and MMN. We can obverse that: (1) MMN makes correct predictions for the questions with complex novel phrase-phrase compositions, but fails for the questions with associated simple phrase-word/word-word compositions. (2) MMN+MLO (Ours) makes predictions accurately on all questions. These observations show that our framework is effective to help the baseline method MMN maintain consistency of compositional generalization across different levels of novel compositions. More qualitative examples are provided in our **repository**.

Conclusion

In this paper, we have explored the consistency of compositional generalization across multiple levels of novel compositions, and have presented that existing vision-and-language models even with 37B parameters struggle to the consistency. We've proposed a meta-learning based framework that can improve the consistency of different models, by making the models progressively learn compositions from simple to complex in a multilevel optimization process. Moreover, a GQA-CCG dataset has been presented to enable the qualitative evaluation of the consistency for VQA models. Experimental results show that our framework can improve not only the consistency of compositional generalization across multiple levels, but also the capacity of compositional generalization at different levels.

Acknowledgments

This work was supported by the Natural Science Foundation of China (NSFC) under Grants No. 62176021 and No. 62172041, Natural Science Foundation of Shenzhen under Grant No. JCYJ20230807142703006, Key Research Platforms and Projects of the Guangdong Provincial Department of Education under Grant No.2023ZDZX1034.

References

Awadalla, A.; Gao, I.; Gardner, J.; Hessel, J.; Hanafy, Y.; Zhu, W.; Marathe, K.; Bitton, Y.; Gadre, S.; Sagawa, S.; Jitsev, J.; Kornblith, S.; Koh, P. W.; Ilharco, G.; Wortsman, M.; and Schmidt, L. 2023. OpenFlamingo: An Open-Source Framework for Training Large Autoregressive Vision-Language Models. *arXiv preprint arXiv:2308.01390*.

Bienenstock, E. 1996. Composition. In *Brain theory*, 269–300. Elsevier.

Chen, W.; Gan, Z.; Li, L.; Cheng, Y.; Wang, W.; and Liu, J. 2021. Meta module network for compositional visual reasoning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 655–664.

Cho, J.; Lei, J.; Tan, H.; and Bansal, M. 2021. Unifying vision-and-language tasks via text generation. In *Proceedings of the International Conference on Machine Learning*, 1931–1942. PMLR.

Cho, J. W.; Kim, D.-J.; Ryu, H.; and Kweon, I. S. 2023. Generative bias for robust visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11681–11690.

Choe, S. K.; Neiswanger, W.; Xie, P.; and Xing, E. 2023. Betty: An Automatic Differentiation Library for Multilevel Optimization. In *Proceedings of the International Conference on Learning Representations*.

Contributors, O. 2023a. OpenCompass: A Universal Evaluation Platform for Foundation Models. https://github.com/open-compass/opencompass.

Contributors, X. 2023b. XTuner: A Toolkit for Efficiently Fine-tuning LLM. https://github.com/InternLM/xtuner.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 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)*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics.

Dong, X.; Zhang, P.; Zang, Y.; Cao, Y.; Wang, B.; Ouyang, L.; Wei, X.; Zhang, S.; Duan, H.; Cao, M.; Zhang, W.; Li, Y.; Yan, H.; Gao, Y.; Zhang, X.; Li, W.; Li, J.; Chen, K.; He, C.; Zhang, X.; Qiao, Y.; Lin, D.; and Wang, J. 2024. InternLM-XComposer2: Mastering Free-form Text-Image Composition and Comprehension in Vision-Language Large Model. *arXiv preprint arXiv:2401.16420*.

Fodor, J. A.; and Pylyshyn, Z. W. 1988. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1-2): 3–71.

Hu, R.; Rohrbach, A.; Darrell, T.; and Saenko, K. 2019. Language-conditioned graph networks for relational reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10294–10303.

Hudson, D. A.; and Manning, C. D. 2018. Compositional Attention Networks for Machine Reasoning. In *Proceedings of the International Conference on Learning Representations*.

Hudson, D. A.; and Manning, C. D. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6700–6709.

Jing, C.; Jia, Y.; Wu, Y.; Liu, X.; and Wu, Q. 2022. Maintaining reasoning consistency in compositional visual question answering. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5099–5108.

Kitaev, N.; Cao, S.; and Klein, D. 2019. Multilingual Constituency Parsing with Self-Attention and Pre-Training. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3499–3505. Florence, Italy: Association for Computational Linguistics.

Li, B.; Zhang, Y.; Chen, L.; Wang, J.; Yang, J.; and Liu, Z. 2023a. Otter: A Multi-Modal Model with In-Context Instruction Tuning. *arXiv preprint arXiv:2305.03726*.

Li, C.; Li, Z.; Jing, C.; Jia, Y.; and Wu, Y. 2023b. Exploring the Effect of Primitives for Compositional Generalization in Vision-and-Language. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19092–19101.

Li, C.; Li, Z.; Jing, C.; Wu, Y.; Zhai, M.; and Jia, Y. 2024. Compositional Substitutivity of Visual Reasoning for Visual Question Answering. In *Proceedings of the European Conference on Computer Vision*, 143–160. Springer.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023c. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*.

Li, J.; Xie, J.; Qian, L.; Zhu, L.; Tang, S.; Wu, F.; Yang, Y.; Zhuang, Y.; and Wang, X. E. 2022. Compositional temporal grounding with structured variational cross-graph correspondence learning. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3032–3041.

Liu, H.; Li, C.; Li, Y.; Li, B.; Zhang, Y.; Shen, S.; and Lee, Y. J. 2024. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge.

Liu, Q.; An, S.; Lou, J.-G.; Chen, B.; Lin, Z.; Gao, Y.; Zhou, B.; Zheng, N.; and Zhang, D. 2020. Compositional generalization by learning analytical expressions. *Advances in Neural Information Processing Systems*, 33: 11416–11427.

Lorraine, J.; Vicol, P.; and Duvenaud, D. 2020. Optimizing millions of hyperparameters by implicit differentiation. In *International conference on artificial intelligence and statistics*, 1540–1552. PMLR.

Migdalas, A.; Pardalos, P. M.; and Värbrand, P. 2013. *Multilevel optimization: algorithms and applications*, volume 20. Springer Science & Business Media.

Nguyen, B. X.; Do, T.; Tran, H.; Tjiputra, E.; Tran, Q. D.; and Nguyen, A. 2022. Coarse-to-Fine Reasoning for Visual Question Answering. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 4557–4565.

Pierrot, T.; Ligner, G.; Reed, S. E.; Sigaud, O.; Perrin, N.; Laterre, A.; Kas, D.; Beguir, K.; and de Freitas, N. 2019. Learning compositional neural programs with recursive tree search and planning. *Advances in Neural Information Processing Systems*, 32.

Plass, J. L.; Moreno, R.; and Brünken, R. 2010. Cognitive load theory.

Radziwill, N. M.; and Benton, M. C. 2017. Evaluating quality of chatbots and intelligent conversational agents. *arXiv preprint arXiv:1704.04579*.

Ray, A.; Sikka, K.; Divakaran, A.; Lee, S.; and Burachas, G. 2019. Sunny and Dark Outside?! Improving Answer Consistency in VQA through Entailed Question Generation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing*, 5860–5865.

Ribeiro, M. T.; Guestrin, C.; and Singh, S. 2019. Are red roses red? evaluating consistency of question-answering models. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 6174–6184.

Selvaraju, R. R.; Tendulkar, P.; Parikh, D.; Horvitz, E.; Ribeiro, M. T.; Nushi, B.; and Kamar, E. 2020. Squinting at vqa models: Introspecting vqa models with sub-questions. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10003–10011.

Shah, M.; Chen, X.; Rohrbach, M.; and Parikh, D. 2019. Cycle-consistency for robust visual question answering. In 2019 IEEE. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6642–6651.

Shu, J.; Xie, Q.; Yi, L.; Zhao, Q.; Zhou, S.; Xu, Z.; and Meng, D. 2019. Meta-weight-net: Learning an explicit mapping for sample weighting. *Advances in neural information processing systems*, 32.

Sun, Q.; Cui, Y.; Zhang, X.; Zhang, F.; Yu, Q.; Wang, Y.; Rao, Y.; Liu, J.; Huang, T.; and Wang, X. 2024. Generative multimodal models are in-context learners. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14398–14409.

Tan, H.; and Bansal, M. 2019. LXMERT: Learning Cross-Modality Encoder Representations from Transformers. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing*, 5100–5111.

Tascon-Morales, S.; Márquez-Neila, P.; and Sznitman, R. 2023. Logical Implications for Visual Question Answering Consistency. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6725–6735.

Wang, K.; Wu, Y.; Cen, J.; Pan, Z.; Li, X.; Wang, Z.; Cao, Z.; and Lin, G. 2024. Self-Supervised Class-Agnostic Motion Prediction with Spatial and Temporal Consistency Regularizations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14638–14647.

Wang, Q.; Liu, L.; Jing, C.; Chen, H.; Liang, G.; Wang, P.; and Shen, C. 2023a. Learning Conditional Attributes for Compositional Zero-Shot Learning. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11197–11206.

Wang, W.; Lv, Q.; Yu, W.; Hong, W.; Qi, J.; Wang, Y.; Ji, J.; Yang, Z.; Zhao, L.; Song, X.; et al. 2023b. Cogvlm: Visual expert for pretrained language models. *arXiv preprint arXiv:2311.03079*.

Xu, L.; Huang, M. H.; Shang, X.; Yuan, Z.; Sun, Y.; and Liu, J. 2023. Meta compositional referring expression segmentation. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19478–19487.

Xu, X.; Chen, X.; Liu, C.; Rohrbach, A.; Darrell, T.; and Song, D. 2018. Fooling vision and language models despite localization and attention mechanism. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4951–4961.

Yang, L.; Kong, Q.; Yang, H.-K.; Kehl, W.; Sato, Y.; and Kobori, N. 2023. Deco: Decomposition and reconstruction for compositional temporal grounding via coarse-to-fine contrastive ranking. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 23130–23140.

Yang, X.; Zuo, Y.; Ramasinghe, S.; Bazzani, L.; Avraham, G.; and van den Hengel, A. 2024. ViewFusion: Towards Multi-View Consistency via Interpolated Denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9870–9880.

Ye, Q.; Xu, H.; Ye, J.; Yan, M.; Hu, A.; Liu, H.; Qian, Q.; Zhang, J.; and Huang, F. 2024. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13040–13051.

Yuan, Y.; Wang, S.; Jiang, M.; and Chen, T. Y. 2021. Perception matters: Detecting perception failures of vqa models using metamorphic testing. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16908–16917.

Zhang, S.; Peng, H.; Fu, J.; Lu, Y.; and Luo, J. 2021. Multi-scale 2d temporal adjacency networks for moment localization with natural language. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12): 9073–9087.

Zhang, Y.; Luo, H.; and Lei, Y. 2024. Towards CLIP-driven Language-free 3D Visual Grounding via 2D-3D Relational Enhancement and Consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13063–13072.

Zhu, D.; Chen, J.; Shen, X.; Li, X.; and Elhoseiny, M. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.