CRIC: A VQA Dataset for Compositional Reasoning on Vision and Commonsense

Difei Gao, *Student Member, IEEE,* Ruiping Wang, *Senior Member, IEEE,* Shiguang Shan, *Fellow, IEEE,* and Xilin Chen, *Fellow, IEEE*

Abstract—Alternatively inferring on the visual facts and commonsense is fundamental for an advanced VQA system. This ability requires models to go beyond the literal understanding of commonsense. The system should not just treat objects as the entrance to query background knowledge, but fully ground commonsense to the visual world and imagine the possible relationships between objects, e.g., "fork, can lift, food". To comprehensively evaluate such abilities, we propose a VQA benchmark, CRIC, which introduces new types of questions about **C**ompositional **R**easoning on v**I**sion and **C**ommonsense, and an evaluation metric integrating the correctness of answering and commonsense grounding. To collect such questions and rich additional annotations to support the metric, we also propose an automatic algorithm to generate question samples from the scene graph associated with the images and the relevant knowledge graph. We further analyze several representative types of VQA models on the CRIC dataset. Experimental results show that grounding the commonsense to the image region and joint reasoning on vision and commonsense are still challenging for current approaches. The dataset is available at https://cricvqa.github.io.

Index Terms—visual question answering, compositional reasoning, commonsense reasoning, dataset construction.

1 Introduction

Visual intelligence has made great progress in many specific tasks, such as image classification [1], [2], [3], object detection [4], [5], and relationship detection [6], [7]. However, it is still a formidable challenge to answer a natural language question about an image (i.e., Visual Question Answering task, VQA), which requires a system to realize a wide range of abilities. In the past few years, [8], [9] first propose the VQA benchmarks, where the tasks are to answer relatively simple questions about the object name, attribute, like **Q1** in Fig. 1. Further works expand the scope of the VQA task along with two orthogonal directions: 1) [10], [11], [12] extend the questions about querying information of a *single* visual object to questions that require multi-hop reasoning on visual relations among *multiple* objects, like **Q2**. 2) [13], [14], [15] expand the questions from only querying relatively shallow visual information of an object to querying non-visual knowledge of an object in the image, like **Q3**. For visual-related abilities, this type of questions usually require relatively simple abilities, such as object recognition.

However,it is a pipe dream for AI to jointly infer on commonsense relations among entities and perform multi-hop reasoning on vision and knowledge. For example, to answer **Q4** in Fig. 1, an AI agent is required to not only infer the explicit semantic spatial relation, *eggs* on *plate* based on ***what it sees*** in the image, but more importantly infer the implicit commonsense relation between the objects based on ***what it knows*** about the world, *fork* can move *eggs*. It is a higher level of visual commonsense reasoning. The AI agents should not just treat objects as the entrance to query background knowledge, but fully ground commonsense to the visual world and imagine the possible relationships between objects, as shown in the top of Fig. 1. Therefore, this paper aims to extend the VQA task along with both directions and introduces a new VQA benchmark about

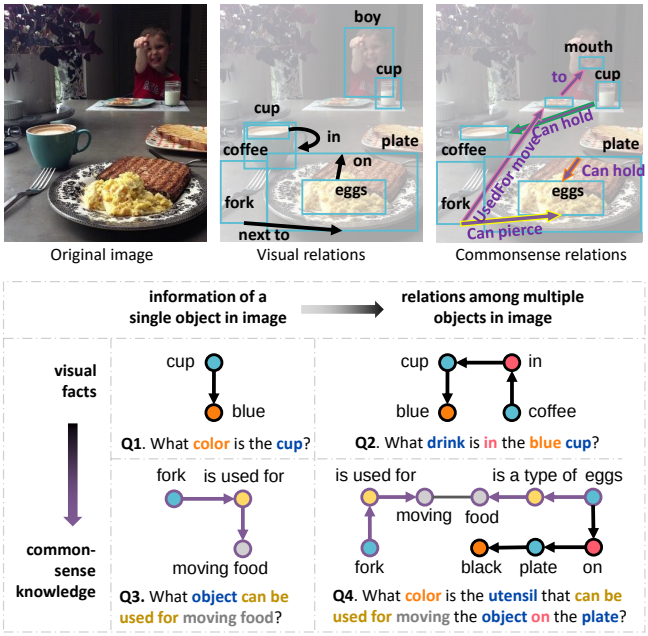


Fig. 1. Examples of four styles of questions. **Q1.** Querying visual information of an object. **Q2.** Multi-hop reasoning on visual relations. **Q3.** Querying non-visual knowledge about an object. Note that, though the Q3 involves two object names, "object" and "food", but only "object" refers to the object in the image. **Q4.** Multi-hop reasoning on both visual and commonsense relations. Black arrow indicates the visual relation, and purple arrow indicates the commonsense relation.

• *D. Gao, R. Wang, S. Shan, and X. Chen are with the Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China. E-mail: difei.gao@vipl.ict.ac.cn, {wangruiping, sgshan, xlchen}@ict.ac.cn*

Compositional Reasoning on vIsion and Commonsense.

The VQA task at the intersection of vision, language and commonsense makes fairly evaluating the models challenging. The commonsense-related questions derived from natural images are inevitable to mirror some priors inherent in the real world. These priors could be the hints for a model to achieve high scores by guessing the answers, e.g., the word *cut* in a question could be a hint for answering *knife*. Thus, to create a commonsense-related VQA dataset, it is crucial to reduce the impact of commonsense priors to fairly evaluate whether the model truly understands the vision and commonsense. To achieve this goal, we introduce two essential features to the CRIC. 1) We carefully design some new types of compositional questions to force models to look at the images, e.g., query the attribute of an object that meets a commonsense requirement, like **Q4**. 2) We not only use the correctness of the final answers to evaluate the model, but also the correctness of the intermediate grounding results, i.e., the model has to correctly find the object that meets the requirement of the question. Only when both two metrics are correct, one question is considered correctly answered.

To achieve these two features, we need strictly control the content of questions and collect rich annotations. The cost will be very high if the dataset is purely manually collected. Therefore, we propose a generator to automatically output question-answer pairs. Specifically, we dynamically assemble the question template from predefined template components for a given *scene graph* of the images and the relevant *knowledge graph*. Along with the question and the answer, we also automatically generate rich annotations for each sample to ease the difficulty of diagnosing a model, including the reasoning steps and their ground truth outputs of answering questions.

To support such a generator, we first need to collect scene graphs and knowledge graphs as the basis to provide object-level visual information and knowledge. Thanks to the Visual Genome dataset [16], the scene graphs of images are easy to get. However, the collection of knowledge graphs faces new challenges. To generate compositional questions, we need object-level knowledge which depicts the commonsense relations between objects. However, as shown in Fig. 2 (a), the current format of existing available knowledge items (e.g., items in ConceptNet [17]) are on the event-level, which makes commonsense relation between objects hard to be effectively represented. Rich information are simply stored in a triplet format $<head, relation, tail>$ to represent coarse-grained relations (e.g., *is used for*, *can*) between an object and an event, where the triplet has to mix up many detailed relations about multiple objects in phrase form (e.g., *moving food from plate to mouth*) as a *head* or *tail* entity. Obviously, the phrase type entity is difficult to be aligned to visual objects, let alone to represent our desired commonsense relations between objects. To tackle this issue, we collect the original items from existing knowledge graph (e.g., ConceptNet), then decompose phrase-formed entities into a finer granularity, i.e., object-level nodes, and re-organize the original *head* and *tail* entities as graphs to obtain graph-to-graph format, as shown in Fig. 2 (b). The graph-to-graph format item is much easier to be aligned to objects in images and can depict more informative commonsense relations

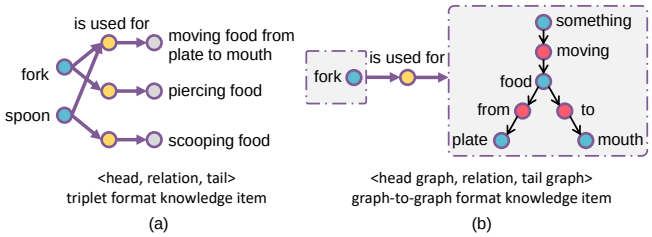


Fig. 2. Examples of triplet format versus graph-to-graph format knowledge items. **(a)**: The triplet format item (coming from ConceptNet [17]) usually depicts the relation between an object and an event. **(b)**: Our graph-to-graph format represents the item in a finer granularity (object-level nodes), which can be easily aligned to visual objects and depict more informative commonsense relations between objects. Here in this case, the head graph has only one single node.

(e.g., *can lift*, *can move*, etc.) between the objects (e.g., *fork*, *can lift*, *food*).

We further evaluate several representative types of VQA models on the CRIC dataset to analyze the advantages and disadvantages of them. In addition, we leverage the well studied modular network [18], [19] and our provided output of every function in programs to provide a detailed analysis of the main challenges of the CRIC. The experiments show that current joint representation of commonsense and vision limits the grounding the commonsense to the images and performing compositional reasoning on vision and commonsense, and cumulative error restricts the multi-step reasoning performance of modular networks.

To summarize, the contributions of this paper are as follows: 1) We propose a new benchmark CRIC which introduces new types of questions for fairly evaluating the ability of compositional reasoning on visual and commonsense. 2) To build a dataset to support such task at a proper cost, we propose an dynamic template assembly dataset construction method to collect question-answer pairs and rich additional annotations. 3) To collect satisfactory knowledge items to generate compositional questions, we introduce a new graph-to-graph format for representing the knowledge items. 4) Further experiments provide detailed analyses about the representation and reasoning abilities of the existing several representative types of VQA methods and the challenges of sub-tasks in CRIC.

2 Related Work

VQA Dataset. At the early stage, COCO-QA [8] and DAQUAR [9] focus on evaluating a range of visual abilities about querying the visual information of a visual entity, e.g., recognizing the category or attributes of an object. Further works [10], [25], [26], [27], [28], such as VQA [10], extend the scope of questions by requiring understanding visual relations between objects. [28] examines whether the model is sensitive to the changes of visual relations between objects by editing images. In addition, CLEVR [11] emphasizes the importance of a VQA system on compositional reasoning on spatial relations and provides compositional questions about synthetic images. More recently, GQA [12] introduces a real-image VQA dataset with compositional visual questions and a more balanced answer distribution.

TABLE 1
Comparison of various VQA datasets. The last three columns are about the additional annotations provided by the datasets. The CRIC contains compositional questions for commonsense reasoning and provides rich annotations.

Dataset	Year of Publication	Num. of Images	Num. of Questions	Task Focus	Scene Graph	Knowledge Graph	Functional Program
CRIC (Ours)	-	96K	494K	Commonsense (Compositional)	✓	✓	✓
OK-VQA [20]	2019	14K	14K	Unstructured Knowledge	✗	✗	✗
KVQA [14]	2019	24K	183K	Name Entities related Knowledge	✗	✓	✗
VCR [21]	2018	110K	290K	Commonsense	✓	✗	✗
FVQA [13]	2018	2.2K	5.8K	Commonsense	✗	✓	✗
KB-VQA [22]	2017	0.7K	2.4K	Commonsense	✗	✗	✗
GQA [12]	2019	113K	22M	Vision (Compositional)	✓	✗	✓
CLEVR [11]	2017	100K	999K	Vision (Compositional)	✓	✗	✓
PQA [23]	2021	32.9K	157K	Perceptual Reasoning	✗	✗	✗
VQA $v2$ [24]	2016	204K	1.1M	Vision	✗	✗	✗
VQA $v1$ [10]	2015	204K	614K	Vision	✗	✗	✗
VQA-abstract [10]	2015	50K	150K	Vision (Scene Graph)	✓	✗	✗
COCO-QA [8]	2015	69K	117K	Vision	✗	✗	✗
DAQUAR [9]	2014	1.4K	12K	Vision	✗	✗	✗

Another trend in recent studies [13], [14], [20], [21], [22], [29] is to expand the scope of the questions by requiring some commonsense-related abilities. [13] introduces the FVQA dataset, of which each question relates to one knowledge triplet in Knowledge Graph. The FVQA aims to evaluate the ability of VQA systems on understanding non-visual background knowledge of the objects. [14] introduces KVQA dataset containing questions about name entities knowledge in Wikipedia, rather than common objects, e.g., *Who is to the left of Barack Obama*. The OK-VQA [20] requires the model to mine the background knowledge of objects from outside natural language documents, rather than Knowledge Graph. The VCR dataset [21] focuses on challenging commonsense reasoning questions, e.g., inferring why something happened or the mental state of a person. Unlike previous datasets, which provide external knowledge sources, VCR requires understanding the knowledge about causal relations, social interactions and physics acquiring from training samples. [30], [31] introduce additional textual knowledge about cultural heritage for building VQA systems in more practical applications. [32], [33], [34], [35] focus on the joint reasoning on the scene text in the images, which can also be regarded as one special type of knowledge provided in images.

Compared to previous datasets, our proposed questions require the systems to not only use objects as the entrance to query background knowledge, but fully ground commonsense to the visual world and imagine the possible relationships of objects. We believe this is a crucial and fundamental ability for future AI agents. In Tab. 1, we display the basic statistics and main characteristics of major VQA datasets and our proposed CRIC dataset. And, a discussion about CRIC and the most relevant datasets is in Sec. 5.

Knowledge Graph. Structured Knowledge Graphs (KGs) are great sources to provide explicit and well-organized information for machines. There are two types of KGs widely used in AI researches, that is, world knowledge KGs, such as DBpedia [36], Freebase [37], and commonsense KGs, such as ConceptNet [17] and Webchild [38]. The world knowledge KGs are broadly used in NLP communities [39], [40], [41], [42] for supporting the AI agent in answering knowledge related question. In recent years, many inspiring works [43], [44] attempt to introduce commonsense KGs in scene understanding. [13], [45], [46], [47], [48], [49] also use external knowledge to expand the capability of VQA systems. [50] is a pioneering work introducing knowledge base into referring expression tasks.

In commonsense KGs, knowledge is typically represented by a large set of items in triplet format $<$*head*, *relation*, *tail*$>$, where *head* and *tail* are two entities and *relation* indicates the relationship between them. Compared with world knowledge KGs, the items in commonsense KGs have one unique characteristics: a large number of entities are informative phrases depicting an event rather than a real "entity", e.g., $<$*bus*, *is used for*, *transporting students to school*$>$. The rich information depicting the relationships between the objects is simply in a phrase, e.g., *bus* transport *students*, *bus* move to *school*. Therefore, compared to previous works that directly use these KGs into their tasks, we furthur decompose the triplet items into a new format to mine the information hidden in the entities and evaluate the VQA systems on understanding such more complicated knowledge.

Dataset Construction. Many existing VQA datasets [9], [10], [13], [20], [24] invite human annotators to collect free-form and open-ended visual questions. Another branch of works [8], [11], [12], [26] which focus on evaluating some specific abilities of VQA models, propose to generate questions by the template-based method automatically. [8] designs rules to convert image descriptions into some predefined types of questions. [11] proposes to generate compositional questions of synthetic images by filling predefined question templates with elements in scene graphs. [12], [26] design more diverse templates to generate rich questions querying about natural images.

Previous automatic question generation methods usually require predefining almost all possible templates and are thus less efficient and scalable for generating our desired questions, which involve large concept vocabulary and commonsense knowledge. To address this problem, we propose a new question generator to dynamically assemble the question template from predefined basic template components given the scene graph of an image and a collected knowledge graph.

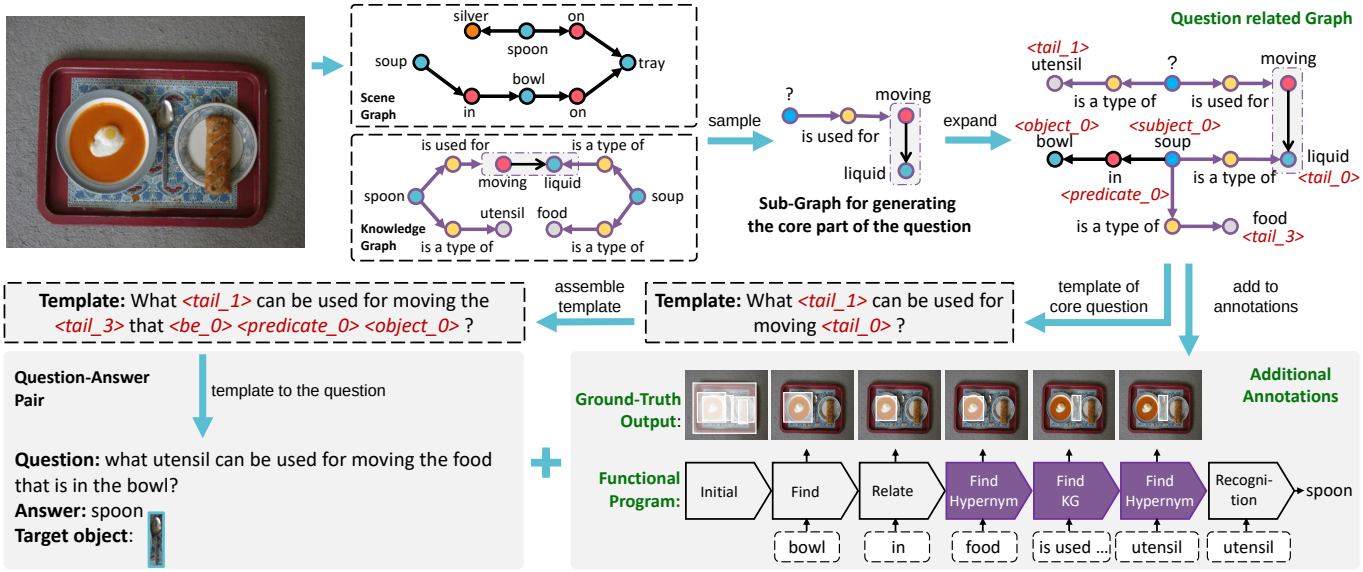


Fig. 3. Overview of the QA sample generation process. The question and corresponding annotations are automatically generated from the Scene Graph of a given image and the Knowledge Graph. Our method first selects the proper part of the Scene Graph and the Knowledge Graph that can generate a question, then assembles the question template from predefined template components, and finally generates the question-answer pair along with rich annotations.

3 Dataset Construction

Overview. We introduce CRIC, a new task that challenges AI systems to ground the commonsense into the visual world and perform multi-hop reasoning on the real image and the knowledge graph. To evaluate such abilities, the CRIC provides 494K balanced and compositional questions on 96K images along with 3.4K knowledge items. Besides, to ease the difficulty of designing robust and interpretable systems, the dataset collects rich annotations about task decomposition, scene graph and knowledge graph related to every question, reasoning steps, and corresponding results for answering every question. Our dataset is constructed in five main steps: 1) process scene graphs, 2) collect and parse knowledge triplets, 3) define basic functions that the question will involve, 4) automatically generate QA samples from the scene graphs and the knowledge graph, and 5) obtain additional annotations, as shown in Fig. 3.

Scene Graph Processing. The CRIC dataset utilizes the 108K images of Visual Genome and their corresponding Scene Graph annotations to generate QA samples. Scene graph is a structured representation of an image, where nodes are objects annotated with attributes and edges connect two related objects. In this stage, we first clean up the scene graphs by filtering rare concepts and merging synonyms. Our processed scene graphs contain 1291 distinct objects, 267 attributes, and 210 relationships. It is also observed that one object in the image might correspond to multiple object IDs and bounding boxes in the scene graph. For example, in Fig. 4, it may be because 1) the table is incomplete, and 2) the edges of the table are occluded, so the annotators have not reached a consensus on the table's bounding box. There are two bounding boxes that correspond to the same table in the annotation, i.e., the red box and the yellow one. This will introduce ambiguity in the later question generation procedure. Thus, we merge bounding boxes that correspond to the same object name and have a high IoU (> 0.7).

Fig. 4. An example that one object corresponds to multiple bounding boxes.

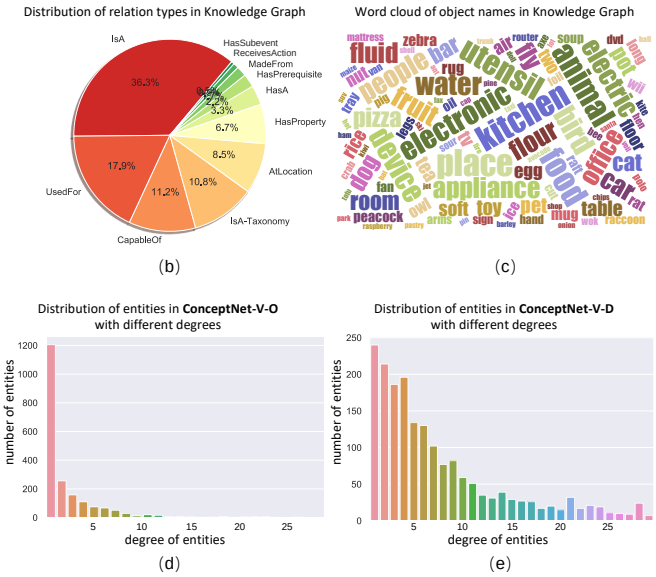
Knowledge Graph Processing. The purposes of this stage are 1) collecting knowledge that is useful in daily life and related to the images in Visual Genome and 2) parsing the triplet format knowledge items into the graph-to-graph format.

In this paper, our knowledge graph is extracted from a large-scale commonsense Knowledge Graph, ConceptNet [17]. The knowledge in ConceptNet is collected from a variety of resources, such as crowd-sourced resources (e.g., Open Mind Common Sense [51]) and expert-created resources (e.g., WordNet [52] and JMDict [53]), and is represented in triplet format $<head, relation, tail>$. To collect satisfactory knowledge triplets, we query the ConceptNet with all the concepts in the processed scene graphs and obtain about 225K triplets. Unfortunately, we find that many triplets of some specific relation types involve subjective opinion that are unnatural to appear in a vision-related question, e.g., $<person, Desires, own\ a\ house>$. Thus, we further invite annotators to check around 100 examples per relation type in ConceptNet to determine which relation types usually involve subjective information. CRIC finally uses 11 types of relations, as shown in Fig. 5 (a), which mainly state objective facts selected from all 34 relation types. In addition, we carefully refine the items to keep similar events expressed in the same style (e.g., same sentence structure and predicate) to avoid some special words being hints for guessing latter generated questions based on these items.

Moreover, we find that collected triplets from Concept-

Relation	Examples
IsA	<dog, is a type of, animal>, <car, is a type of, vehicle>
IsA-Taxonomy	<cat, is a type of, feline>, <antelope, is a type of, herbivore>
UsedFor	<menu, is used for, ordering food>
CapableOf	<jeep, can, climb hills>, <dog, can, guard your house>
AtLocation	<fork, at, kitchen>, <dresser, at, bedroom>
HasProperty	<lemon, is, sour>, <fast food, is, bad for your health>
HasA	<milk, has, calcium>, <wine, has, alcohol>
HasPrerequisite	<cooking, requires, food>, <cleaning the house, requires, broom>
ReceivesAction	<orange, can be, eaten>, <rice, can be, cooked>
HasSubevent	<cleaning clothing, has subevent, operating the washing machine>
MadeFrom	<cake, is made from, flour>, <fries, is made from, potato>

(a)



(b) (c) (d) (e)

Fig. 5. Statistics of collected knowledge items. **(a)**: Relation types and their examples in our selected knowledge items. **(b)**: The distribution of relation types in our selected knowledge graph. **(c)**: Word cloud for frequent object names in our knowledge graph. **(d)** and **(e)**: The distribution of entities in the original KG and processed KG with different degrees. Processing KG digs out the hidden relationships between the items and dramatically increases the density of the KG.

Net sometimes are incomplete, e.g., ConceptNet tries to record which objects contain calcium, while only a small portion of satisfied entities are recorded, like *milk* and *ice cream*. Since knowledge base reasoning usually needs to follow the closed-world assumption [54], i.e., what is not currently known to be true, is false, we need to make sure all the entities containing calcium are labeled in our KB. Otherwise, it could cause our generator to output ambiguous questions or incorrect answers. Thus, we further collect 372 knowledge items from Wikipedia to make the knowledge in ConceptNet more complete in such cases.

We also collect two types of categorization knowledge of the objects from Wikipedia and WordNet. One type is about trivial category knowledge, e.g., <*cat*, *IsA*, *animal*>, which is used for referring object in question, e.g., "which animal can ...?". Another type is about more professional taxonomy in specific disciplines, e.g., <*cat*, *IsA*, *feline*>, which is used for querying model whether know this knowledge, e.g., "which animal is a feline?". CRIC's professional taxonomy is about species classification in biology. We refer to NCBI's [55] biological classification; if one category appears in NCBI, we assume it belongs to the professional one.

Finally, we obtain 3,439 carefully selected knowledge triplets with 11 types of relations, e.g., *IsA*, *UsedFor*, *HasA*.

In Fig. 5 (a), we present the selected 11 types of relations and show some examples of each type. The distribution of relation types in the knowledge graph is shown in Fig. 5 (b). We can see that our collected knowledge items are roughly evenly distributed over the relation types of *UsedFor*, *CapableOf*, *IsA-Taxonomy*, *AtLocation*, *HasProperty*. In addition, the word cloud for frequent object names appearing in the knowledge items is shown in Fig. 5 (c). We can see that the knowledge items are mainly about common object used or seen in daily life, such as kitchen utensils, foods, appliances, and animals.

In the following, we parse the selected knowledge triplets into the graph-to-graph format, as shown in Fig. 2 (b). This goal is achieved by developing a simple rule-based phrase-to-graph parser. Our parser first utilizes the Stanford CoreNLP [56] to obtain POS tags of a phrase. Then, we build a set of POS-to-graph mapping templates, where the phrase-format entity can be automatically mapped to a graph according to the POS tags of the words, as shown in Fig. 6. In Fig. 5 (d) and (e), we compare the "degree" of entities (head and tail entities) in our processed KG (denoted as ConceptNet-V-D, where "-V" denotes "vision", and "-D" denotes "Decomposed") with the original triplet format KG (denoted as ConceptNet-V-O). For triplet format KG, since the basic elements in them are entities, we consider two entities as connected (contribute 1 degree) when they are exactly the same; while in graph-to-graph format KG, two entities (a.k.a graph-format entities) are considered as connected when they share at least one object node, since the basic element in the new format is the node in a graph-format entity. We can find that introducing the new format items digs out hidden relations between the items in the original KG and dramatically increases the density of the Knowledge Graph. The dense connected KG can facilitate us in later building challenging multi-hop reasoning questions.

Function Definition. At this stage, we define the basic functions in the CRIC. There are ten basic functions in the CRIC dataset, where three commonsense related functions are unique in CRIC, and the others are similar to those in CLEVR and GQA, as shown in Fig. 7.

Specifically, two functions relate to basic logical operations: "And", "Verify". Four functions are about basic abilities of reasoning on the image: "Find", "Relate", "Relate Reverse", "Recognition", where "Find" indicates find the object for an given object or attribute name, "Relate" indicates the task that given *subject* and *predicate* in a scene graph relationship <*subject*, *predicate*, *object*>, the model needs to locate the region of *object*, while "Relate Reverse" indicates that given *predicate* and *object*, the model locates the region of *subject*, and "Recognition" corresponds to a set of subtasks, such as, recognizing color, recognizing animal, etc. Moreover, we propose three new functions related to the commonsense reasoning: "Find KG", "Find KG Reverse", "Find Hypyernym". "Find KG" and "Find KG Reverse" require the model to find image regions that satisfy a commonsense query, e.g., find a proper object and fill it in the BLANK in query <*cleaning* BLANK, *HasSubevent*, *using the*

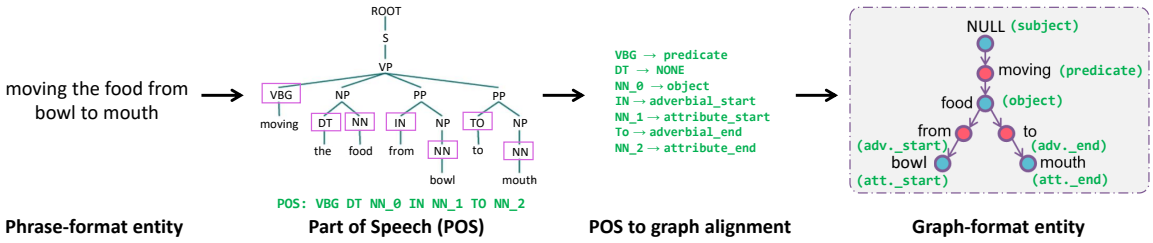


Fig. 6. Pipeline of parsing a phrase-format entity into a graph-format entity. The "NONE" in **POS to graph alignment** indicates that the word doesn't need to be assigned to the graph template. The "NULL" in **Graph-format entity** indicates that no word is assigned to this element in the template.

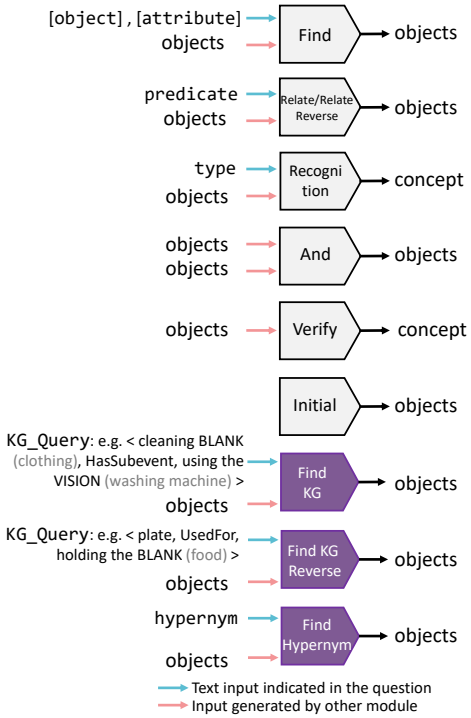


Fig. 7. Catalog of basic functions evaluated in questions of the CRIC dataset.

Question Type	Answer Type	Examples
QueryObjKG	Recognize	What vehicle in the image can carry cargo?
QueryObjSG	Recognize	What is next to the object that can hold liquid?
QueryAtt	Recognize	What material is the utensil that can cut food?
VerifyKG	Verify	Is there a vehicle that can carry cargo?
VerifyAtt	Verify	Is there an animal in black that can guard house?

(a)

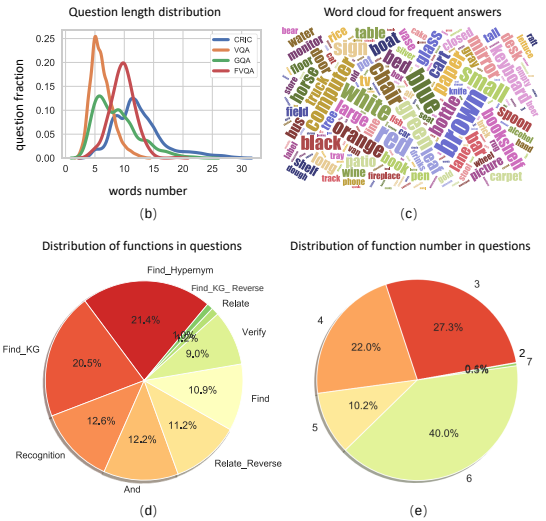


Fig. 8. Statistics of questions and answers in CRIC. **(a)**: 5 main categories of the CRIC questions and their examples. **(b)**: Comparison of question length distributions for different VQA datasets. The questions in the CRIC are generally much longer and have a wide range of lengths. **(c)**: Word cloud for frequent answers. **(d) & (e)**: Distributions of functions in overall questions and function number of programs in overall questions.

VISION *(washing machine)*> to make the statement in accordance with commonsense. Note that, the commonsense query could be multi-modal, e.g., the *washing machine* in the above query can be a text or a region containing *washing machine* which is outputted by another module. "Find Hypyernym" is required to find the object for a given category name, e.g., find all objects in an image which belong to *vehicle*. Finally, we design a simple function, "Initial" to attend on all objects, which is usually used at the functional program's start. In the supplementary material, we provide more details of these functions.

Template Collection & Question Generation. In this section, we introduce a scalable and low-cost question generator to automatically create numerous questions by imitating the procedure of humans creating a complex question. As shown in Fig. 3, one question is generated from a dynamically composed question template based on a subgraph composed of a sub-scene graph and a sub-knowledge graph. Specifically, we first build two types of template components. One type is to **query** one element in a visual triplet or a knowledge item, e.g., the template of querying color "what color/which color/... is the <*subject*>?", where <*subject*> will be filled in based on the graph annotation. The other one is about how to use a visual triplet or a knowledge item to **decorate** one object, e.g., the <*object_1*> (apple) that <*be_2*> (is) <*predicate_2*> (on) <*object_2*> (the plate). To increase the diversity of the question, one template component usually has multiple versions that will be randomly chosen to generate the question.

Then, the question is generated in the following steps: 1) One visual relation or knowledge item is selected to generate the core part of the question. 2) Proper relations and attributes are added to decorate the core question, when the core question contains limited information to precisely locate the image region, or we want to provide additional information to locate the image region. 3) The template of the question will be automatically composed of basic template components. 4) The blanks in the template will be filled in based on the scene graph and the knowledge

FVQA-style Commonsense:
Q1. What eating utensil can be used for moving food to the mouth? fork
Q2. What kitchenware can be used for turning food? spatula

Vision + Commonsense (Compositional):
Q3. Is the food on the plate a type of fast food? yes
Q4. What tableware can lift the object on the plate? fork

FVQA-style Commonsense:
Q5. What object can I use to hold drinks? bottle

Vision + Commonsense (Compositional):
Q6. What color is the object that can be used for sitting on? green
Q7. Who is sitting on the furniture that is likely to be found in living room? woman
Q8. Are the glasses that the woman is wearing used for correcting vision? no

FVQA-style Commonsense:
Q9. Which object can be used for protecting head? helmet

Vision + Commonsense (Compositional):
Q10. Is the object that is usually used for protecting the head in red? no
Q11. What color is the accessory that is used for protecting the head? black
Q12. Which object in the image can hit the flying object? bat

Fig. 9. Some example questions from the CRIC dataset. Our dataset contains not only the relatively simple commonsense question type similar to FVQA [13], but also the proposed unique compositional questions for reasoning on vision and commonsense knowledge. These compositional questions can avoid models answering by guessing through requiring recognizing visual attributes or spatial reasoning after commonsense reasoning to force models looking at the image to answer.

graph. In Fig. 9 and further in the supplementary material, we show some QA samples in the CRIC.

Note that, the questions derived from real images and common knowledge will naturally mirror some priors in the real world. These priors could be the hints for models to achieve high scores by guessing the answers without truly understanding the images and knowledge, e.g., the word *cut* in a question could be a hint for answering *knife*. Therefore, to propose questions that can fairly evaluate the models, we carefully design some **new types of compositional commonsense questions** (the question types in the CRIC are shown in Fig. 8 (a)). For example, we design *QueryAtt*, *QueryObjSG* and *VerifyAtt* types of questions to require recognizing visual attribute or spatial reasoning after commonsense reasoning to force models looking at the image, e.g., Q6, 7, 10, etc. in Fig. 9. Besides, we let the question generator sometimes replace an element in knowledge items with a referring expression, e.g., Q4 in Fig. 1 and Q4, 12 in Fig. 9, to avoid frequently involved knowledge expressions being used as hints.

Obtaining Additional Annotations. For every QA sample in the CRIC dataset, we provide the question and answer, along with additional annotations, including sub-graph needed for answering the question, the functional program of answering and the ground-truth output of every function in the program, as shown in Fig. 3. The sub-scene graph & sub-knowledge graph and the functional program can be automatically generated during the question generation. To collect the ground-truth of each function, at every step of the program, we search on the scene graph and knowledge graph to find candidate objects that satisfy all previous functions' requirements simultaneously.

Now, we obtain 3M automatically generated QA samples. However, these samples are highly unbalanced. To avoid the model overfitting on the bias of the dataset, we balance the dataset by filtering the QA samples based on the distribution of answers and the distribution of knowledge items involved in the questions. Finally, we obtain the CRIC dataset which contains **96,241 images** with **494,350 QA samples** and **3,439 knowledge triplets**.

Quality Control. To guarantee the quality of the dataset, it is crucial to validate the correctness of the annotations. We have manually validated the correctness during dataset construction to ensure quality at each key stage. *1) Module validation*: Our question and additional annotation generators are both modularized. We test on more than 100 samples in different images for each module to confirm they accurately work. *2) Consistency validation*: A natural language question and corresponding function layout are two expressions for the same question. Thus, we validate the correctness of both by checking if the two annotations are mutual-consistent, e.g., if their answers are the same. *3) Automatic Language Quality Assessment*: We also use a Transformer-based language assessment tool, grammaticality score (GS) in GRUEN [57], to automatically measure the quality of questions in CRIC. The GS of CRIC and VQA v2, a natural VQA dataset with human annotated questions, under each question length is shown in Fig. 10. It can be seen that there is a slight gap between our grammatical score and the VQA v2 dataset; however, we believe that the score of CRIC is within a reasonable range. but we believe that the score of CRIC is within a reasonable range. The paper of GRUEN showed the Grammaticality Scores of some example sentences. A correct sentence achieved 0.7 score, and a similar sentence with incorrect passive voice expression only got 0.2 score. In addition,

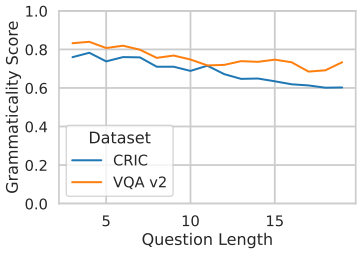


Fig. 10. The grammaticality score (GS) in GRUEN under each question length.

to more intuitively evaluate CRIC's grammar quality, we used GingerIt, a widely used grammar checker, to provide revision suggestions for CRIC questions. It reported that for 90.2% of CRIC questions, GingerIt suggests no modification is needed. For the remaining questions, the recommendations can be summarized as the following three categories: i) The use of prepositions. "What object is on the tub" is revised to "What object is in the tub". "writing at it" is revised to "writing on it". These prepositions are mainly derived from scene graph annotations and the knowledge graph. In many cases, GingerIt recommends to modify the uncommon phrase, but the original expression based on SG and KG may be more precise. ii) Spelling of words. For example, "upperbody" is revised to "upper body", "carnivora" is revised to "carnivore". ii) Adding comma. "Is the wooden object usually used for resting?" is revised to "Is the wooden object, usually used for resting?". In short, these suggestions indicate that some of the compositional questions in CRIC may be uncommon, but their grammar is basically qualified. *4) Human Study*: After generating, we then randomly select about 10,000 samples (i.e., about 2% out of the total 494,350 questions) from images covering various scenes to check their correctness.

The images and corresponding QA samples of the dataset are randomly split into the train (70%), validation (10%), and test (20%). The question contains on average 12 words and involves on average 6 functions. In Fig. 8 (b), we show the distribution of the question length of several VQA datasets. The CRIC has a relatively balanced distribution of question length and is relatively longer than other datasets. In Fig. 8 (c), from the presented frequent answers of the CRIC, we can see that although all questions are related to commonsense, a lot of answers do not come from the knowledge items, e.g., *brown*, *large*. These questions will force models to look at the images based on the results of commonsense reasoning. From Fig. 8 (d), we can see that our CRIC has a relatively balanced distribution of functions, which indicates that the CRIC provides plenty of QA samples for training each sub-task. Besides, from Fig. 8 (e), it shows that our auto-generated questions cover a wide range of complexities (the questions involve from 2 functions to 7 functions), and more than 70% questions, which involve more than 4 functions, are relatively complex and require multi-hop reasoning.

Evaluation Protocol. For evaluation, to further avoid models achieving high scores by guessing and fairly evaluate the performances of models, our evaluation metrics consider both the correctness of 1) the **answer** and 2) the **grounding results**. More specifically, for each QA sample, a VQA system is required to provide two results: the answer and one object selected from our provided candidate objects. Note that, for yes-no questions, if the answer is *yes*, the model should point out the eligible object; if the answer is *no*, the model should point out no object. A question is considered correctly answered when the two results are both correct. To better evaluate and diagnose the performance of the reasoning abilities, especially for grounding-related functions, we provide the bottom-up features [58] cropped by ground-truth bounding boxes as the image features. In addition, we classify the questions into two groups, Verify and Recognize, by checking if the answer is "yes" or "no".

4 Experiments

In this section, we evaluate the performances of four types of representative methods on the CRIC (Sec. 4.2.1), including classical monolithic VQA models, modular VQA models, KB-aware VQA models, and recent popular visual BERT to analyze the main challenges of CRIC.

4.1 Baselines

Q-Only: Q-Only model only takes the question features as input. We implement it with two different models, GRU and BERT, denoted as Q-Only-GRU and Q-Only-BERT, respectively.

I-Only: I-Only model only takes the image feature as input.

SF: SF [59] is a SOTA model on FVQA, which first uses visual concepts extracted by object, scene, action predictors, CNN image feature, and LSTM question feature to retrieve the Top-1 related knowledge item, then uses the question feature and retrieved knowledge item to predict the answer.

TRiG: TRiG [60] is a SOTA model on OK-VQA, which first transforms images into texts by image captioning, dense labeling, and OCR model, then uses these texts to retrieve the knowledge items, finally feeds all above texts to the T5 language model to predict the answer. In our implementation, since our questions don't involve OCR, we only use image captioning and dense labeling in knowledge retrieval and answer prediction.

SAN: Stacked Attention Network [61] is a classical monolithic VQA model on the VQA dataset which performs two-step soft attention on the image features.

Bottom-Up: Bottom-Up [58] is a classical monolithic VQA model which proposes object-level reasoning and implements soft-attention on object regions. The attended image features and question features are combined to generate the final answer.

Bottom-Up+l_{att}: Compared to Bottom-Up, this baseline adds a binary cross-entropy loss on attention score to supervise the model to attend on the correct region. Specifically, attention is supervised in the same way as that of VQA answers. The model predicts the score of each candidate object to determine whether it is the target object, denoted as $\hat{s} \in [0, 1]$. Then, it calculates the loss between the predicted score and the ground truth in annotations as follows:

$$L = -\sum_{i}^{M}\sum_{j}^{N} s_{ij} log(\hat{s}_{ij}) - (1 - s_{ij}) log(1 - \hat{s}_{ij}) \quad (1)$$

where the indices i and j run respectively over the M training questions and N candidate objects, and the ground-truth score s_{ij} is $1/n_i$, where n_i is the number of ground truth target objects for the i-th question.

NMN-CS: Neural Modular Network (NMN) [18], [19], [62] is another type of state-of-the-art modular VQA model on CLEVR dataset. However, its original versions cannot directly transfer to commonsense questions, so we add some visual commonsense reasoning modules, denoted as NMN-CS.

More specifically, the NMN-CS builds upon neural modular networks in [18] which contains a sequence-to-sequence

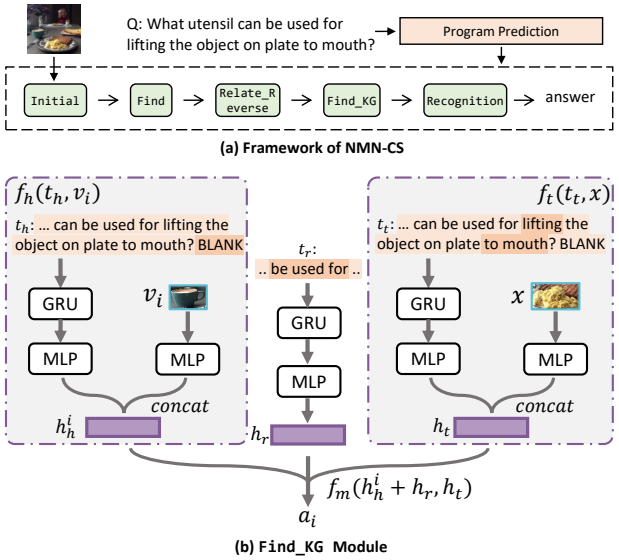


Fig. 11. Illustration of NMN-CS. (a) Framework of NMN-CS. A program prediction module generates the function layout, then predicted functions are performed to output the answer. (b) Find_KG module. Three networks extract the features of three parts of the knowledge item. Then, the Find_KG module calculates their matching score as the attention value. We add an element BLANK at the end of the question for the case that network f_h doesn't need to attend on any word in the question.

network (i.e., program prediction module). It generates the function layout (a sequence of function names) and the text inputs of each function (an attended question over words) for a given question, as shown in Figure 11 (a). In addition, NMN-CS contains a set of modules corresponding to each function in CRIC. The architecture of pure visual modules, such as `Find`, `Relate`, are similar to the corresponding modules in N2NMN [18]. For commonsense-related functions, we design some simple networks to achieve these functions.

First, for the `Find_KG` module, as shown in Fig. 11 (b), the goal is to generate the attention score a over all object candidates $\{v_1, ..., v_n\}$ for a given object feature x generated by another module and query sequence t. The query sequence contains three sub-sequences $t = [t_h, t_r, t_t]$ which indicate the head, relation, and tail in a knowledge item respectively. These sub-sequences are obtained by implementing self-attention on the given text inputs generated by program prediction network. `Find_KG` module first separately encodes the head, relation, and tail. Head feature $h_h = [h_h^1, ..., h_h^o]$ is the combination of each v_i and GRU features encoding the word sequence t_h. Relation feature h_r is encoded by text embedding layer. Tail feature h_t is the combination of the visual feature generated by the previous module and the GRU feature encoding t_t. Then, an MLP $f_m(h_h^i + h_r, h_t)$ outputs attention score on object v_i by calculating the matching score of current head, relation and tail. `Find_KG_Reverse` is similar to the `Find_KG` module, where the only difference is that the positions of x and v_i are exchanged. In addition, the architecture of `Find_Hypernym` is the same as the Find module, which uses the attended question feature to retrieve the objects in the image.

The NMN-CS is trained in two stages: training the program prediction module and training the neural modules.

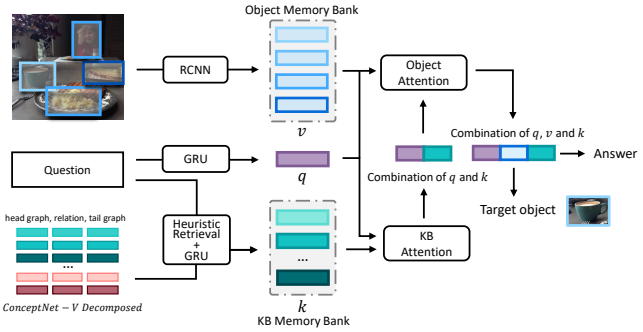


Fig. 12. Overall architecture of Memory-VQA model. The model encodes the image and the knowledge items into two memory banks, respectively, and then implements an iterative attention mechanism to locate relevant knowledge items and objects to answer the question.

For training the program prediction module, we minimize the cross-entropy losses of predicted function names. For training the neural modules, we use the predicted function layouts to assemble the neural modules and minimize the binary cross-entropy losses for the answers. More details about this baseline are shown in the supplementary material.

Memory-VQA: We also design a simple KB-aware baseline to explicitly utilize knowledge items to answer the questions, named as Memory-VQA. This baseline follows the spirit of memory network [40] which encodes the input materials (e.g., the knowledge items and the image in the CRIC) as memories, then uses the question to trigger an iterative attention process which allows the model to retrieve useful information to answer the question.

The overall architecture of this model is shown in Fig. 12. The whole model is composed of two parts. The first part realizes the feature extraction of three types of input: it implements a GRU to obtain question features $\boldsymbol{q}$, uses a Faster-RCNN to obtain image features, denoted as *object memory bank* $\boldsymbol{v}$, and applies a heuristic retrieval method and a GRU to roughly select candidate knowledge items and encode them, where the output is denoted as *KB memory bank* $\boldsymbol{k}$. More concretely for the heuristic KB retrieval, firstly every word in questions is used to retrieve the items in our collected ConceptNet-V-D by checking if the head or tail of a knowledge item contain such word; then a GRU selects specific relation type of items by predicting the relation type from the question. The second part implements a two-step attention to find proper knowledge items and object regions used for answering. The first step implements an attention mechanism f_k [58] over the knowledge items, which first predicts attentions $\boldsymbol{a}_k$ based on the question feature $\boldsymbol{q}$, object memory bank $\boldsymbol{v}$, and KB memory bank $\boldsymbol{k}$ and then gives a weighted average over the KB memory bank $\boldsymbol{k}$ as the output. We then combine the output with the question embedding to obtain $\boldsymbol{h}_{qk}$:

$$\boldsymbol{h}_{qk} = \mathrm{FC}(f_k(\boldsymbol{k}, \boldsymbol{q} + \mathrm{mean}(\boldsymbol{v}))) \odot \mathrm{FC}(\boldsymbol{q}), \qquad (2)$$

where FC denotes an fully connected C layer and $\odot$ denotes element-wise multiplication. The second step uses the same architecture to calculate the attention scores over object regions $\boldsymbol{a}_v$ and combines features $\boldsymbol{h}_{qkv}$ of $\boldsymbol{q}$, $\boldsymbol{k}$ and $\boldsymbol{v}$ as

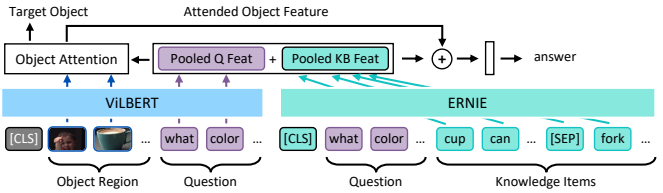


Fig. 13. Overall architecture of ViLBERT+ERNIE+l_{att} model. The ViLBERT encodes the image and the question, the ERNIE encodes the roughly retrieved knowledge items. Then, the combined question, knowledge and image features are used to predict the answer.

the output. Specifically,

$$\boldsymbol{h}_{qkv} = \mathrm{FC}(f_v(\boldsymbol{v}, \boldsymbol{h}_{qk})) \odot \mathrm{FC}(\boldsymbol{h}_{qk}), \tag{3}$$

where f_v is the attention module. Finally, an MLP predicts the answer probabilities and $\boldsymbol{a}_v$ is used to output the target object.

Memory-VQA+l_{att}: Compared to Memory-VQA, this baseline additionally applies a cross-entropy loss on attention score.

MAC: MAC [63] is a state-of-the-art modular VQA model for CLEVR and GQA which decomposes a question into a series of attention-based reasoning steps.

MAC-CS: We extend MAC to access of knowledge items, named as MAC - CommenSense reasoning (MAC-CS). Specifically, MAC-CS inherits the knowledge retrieval and knowledge representation modules in Memory-VQA to obtain a set of knowledge items related to the question. Then, we replace the original input image region feature of MAC with concatenation features of knowledge items and image region features. Thus, the reasoning module can perform attention on both vision and knowledge modality.

ViLBERT: Recently, many works [64], [65], [66], [67], [67], [68], [69], [70], [71], [72] propose powerful self-supervised learning approaches to learn the joint representations between image and language based on BERT model [73]. We select the ViLBERT [64] pre-training on 12 vision and language datasets [66] as the representative of this branch of works. In addition, to output the target object, we add an attention module over extracted visual features which is the same as in Memory-VQA, then use attended features to output the answer.

ViLBERT+l_{att}: Compared to ViLBERT, this baseline adds a cross-entropy loss on attention score.

ViLBERT w/o. PT+l_{att}: We also evaluate a version without pre-training for ablating the advantages of ViLBERT.

ViLBERT+ERNIE+l_{att}: Knowledge representation is one of the most important challenge of CRIC. However, since the knowledge base used in VQA is usually much smaller than the one in NLP tasks, it could be hard to obtain a promising knowledge representation with limited knowledge items. Thus, we propose to integrate a language Transformer pre-trained on large-scale knowledge bases, ERNIE [74], into the vision-language Transformer to improve the knowledge reasoning ability.

The architecture of the ViLBERT+ERNIE+l_{att} is shown in Fig. 13. It has three modules: (1) A vision-language Transformer, ViLBERT, aims to extract the image and question feature. (2) A knowledge Transformer, ERNIE, aims to extract the feature of candidate knowledge items. Specifically, the candidate knowledge items are obtained in the same way as Memory-QA. And the knowledge items are input in the form of word sequences, where each knowledge item is separated by a [SEP] token, as shown in Fig. 13. Finally, an attention module uses the pooled knowledge items feature generated by knowledge Transformer and the pooled question generated by vision-language Transformer to locate the target image region and then predict the answer. In implementation, for the pooling function, we follow the ViLBERT, simply taking the hidden state corresponding to the first token as the pooled feature.

TABLE 2
Results on the test set of the CRIC, where **Ans** indicates the answer accuracy (%), **Grd** indicates the grounding accuracy (%) and **Final** indicates the portion of questions on which the models both correctly generate answers and output grounding results.

Model	Verify		Recognize		Overall		
	Ans	Grd	Ans	Grd	Ans	Grd	Final
Q-Only-GRU	68.79	-	49.57	-	55.18	-	-
Q-Only-BERT	71.30	-	53.97	-	59.03	-	-
I-Only	48.47	-	00.12	-	14.24	-	-
SF	72.32	-	56.31	-	60.98	-	-
TRiG	75.44	-	60.72	-	65.01	-	-
SAN	75.19	46.45	59.36	8.38	63.98	19.50	17.07
Bottom-Up	75.81	48.50	60.18	8.18	64.71	19.88	18.27
Bottom-Up+l_{att}	73.83	52.90	57.72	32.06	62.39	38.10	29.25
MAC	78.71	52.19	64.91	23.00	68.91	31.46	26.19
MAC-CS	79.30	52.43	65.67	27.38	69.65	34.69	28.15
NMN-CS	79.09	48.69	64.82	22.60	68.96	30.17	25.03
Memory-VQA	76.93	51.36	62.36	17.99	66.59	27.67	23.17
Memory-VQA+l_{att}	77.44	61.39	62.64	44.65	66.93	49.51	38.87
ViLBERT	86.15	54.21	71.96	15.97	76.07	27.06	23.67
ViLBERT w/o. PT+l_{att}	83.56	72.36	70.70	53.94	74.46	59.34	50.39
ViLBERT+l_{att}	87.63	75.43	73.42	57.62	77.54	62.79	53.76
ViLBERT+ERNIE+l_{att}	89.21	76.45	75.98	58.89	79.85	64.02	55.24

4.2 Analysis and Diagnosis

In this section, we analyze model performances on different types of questions requiring different reasoning skills to compare existing reasoning mechanisms (Sec. 4.2.1, Sec. 4.2.2 and 4.2.3). Further qualitative and quantitative experiments are conducted to show the necessity of our collected additional annotations in model diagnose and training models (Sec. 4.2.4 and 4.2.5). Finally, we use modular network to investigate the main challenges of CRIC task (Sec. 4.2.6, Sec. 4.2.7, Sec. 4.2.8).

4.2.1 Overall comparison of different types of methods

The overall accuracy and the accuracy for each answer type are shown in Tab. 2. We can see that the current methods achieve at most 55.24% final score. And they struggle on grounding, where the grounding accuracy of the MAC method on another VQA dataset GQA can achieve 82.2% accuracy as reported in [12] while it achieves only 31.46% on the CRIC. These results suggest that the CRIC cannot be solved simply by transferring the standard VQA model to such task, but requires a more delicate model to build the connections between the commonsense and images. Comparing the Bottom-Up with MAC and NMN-CS, we observe that the compositional methods achieve better results on the CRIC dataset. This could be because these

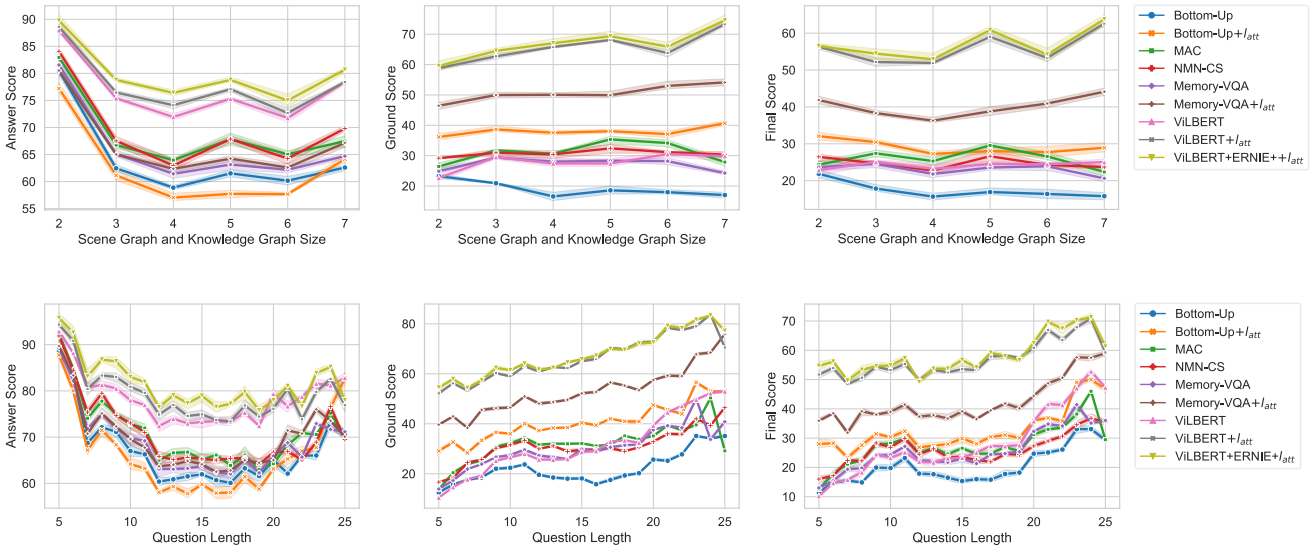


Fig. 14. **Top**: The performances of representative models on different sizes of question related scene and knowledge graph. **Bottom**: The performances of models on different question lengths.

models decompose the complex task into many simpler sub-tasks which is a more robust way to learn the answering skill. However, it is found that the performance gain of compositional models on CRIC is not expected as large as on CLEVR. The reason could be that the cumulative error impacts the final results. For real-image and commonsense-related QA, the sub-tasks are much more difficult than corresponding sub-tasks in synthetic-images, and the new proposed commonsense-related functions still need to be solved by some sophisticated designed modules. Comparing the results of Memory-VQA and Bottom-Up, we can see that the use of external knowledge also brings significant improvement. More analysis of Memory-VQA will be presented in Sec. 4.2.5.

Moreover, we can see that SF and TRIG both do not outperform Memory-VQA. The reason could be that these SOTAs on FVQA and OK-VQA focus more on knowledge retrieval, so they choose to transform the image into the text to better locate the related knowledge. However, when facing CRIC, which focuses more on the joint reasoning of the two modalities, transforming the image into texts will lose non-negligible key visual information in answer prediction.

In addition, ViLBERT with multi-task training (i.e., ViLBERT $+l_{att}$) shows substantial superiority compared to other models; still, the challenge is far from solved. Here we would like to further illustrate why the ViLBERT series models turn out to be the best-performing model. The advantages of ViLBERT could lie in three aspects: 1) BERT model pre-trained on large-scale text corpus may contain a certain level of commonsense, which is also helpful for commonsense VQA. 2) The ViLBERT architecture with lots of parameters has a larger model capacity and strong visual reasoning ability. 3) The multimodal pre-training of ViLBERT can better align vision and language and benefit visual reasoning. By comparing the results of Memory-VQA + l_{att}, ViLBERT w/o. PT + l_{att}, we can see that the ViLBERT architecture brings the greatest benefits (ViLBERT w/o. PT + l_{att} outperforms Memory-VQA + l_{att} by 11.52% on the Final Score (FS)). The language pre-training also helps (Q-Only-BERT outperforms Q-Only-GRU by 3.85% on FS). Moreover, vision-language pre-training contributes the smallest but still promising improvements (VilBERT + l_{att} outperforms VilBERT w/o. PT + l_{att} by 3.37% on FS).

It also can be seen that introducing pre-trained knowledge Transformer (i.e., ViLBERT+ERNIE+l_{att}) further benefits the performance at a certain level. However, we would expect that combining multiple types of Transformers have greater potential. Knowledge Transformer has a strong knowledge reasoning ability, as proven in KB field [75], however, due to the separate pre-training procedures of ERNIE and ViLBERT, the features of knowledge items may not be particularly well aligned with the visual features. As we can see, the improvement in QA (Ans Score) is larger than in grounding (Grd Score). We believe a future valuable direction is to design some novel objectives for joint pre-training multiple types of Transformers or prompt fine-tuning techniques to better align the features of different modalities.

4.2.2 *Effect of question size*

In this part, we compare the performances of models on questions in different difficulty degrees. Specifically, in Fig. 14, we present the performances of four types of representative models on different question sizes. The question size is measured in two metrics. (1) The question size is considered as the size of question-related scene graph and knowledge graph, that is, the total number of object-attribute tuples, visual relationship triples, and knowledge items. This measure of question size represents the number of reasoning steps. (2) The question size is equal to the question length. It indicates the amount of information used for object grounding.

From Fig. 14, we can see that for **answer score**, the shapes of curves for two metrics of question size both are U-like, rather than an intuitive result that longer questions

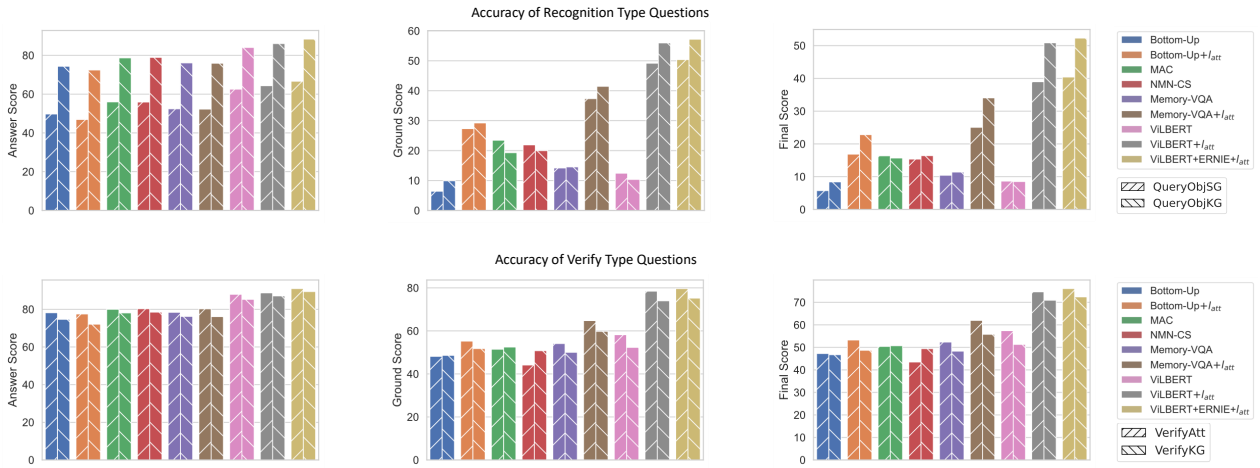


Fig. 15. The performances of models on different types of questions. The top row shows the scores on QueryObjSG and QueryObjKG questions. The bottom row shows the scores on VerifyAtt and VerifyKG questions.

should be harder. The curves of **ground score** are also somewhat counterintuitive; with the increase of question size, the performances of models don't drop, but are relatively stable and even slightly better. The primary causes of this interesting phenomenon could be some inherent commonsense priors in questions which mirror the bias of the world. The small size questions are easier to answer because these questions usually directly query a knowledge item, so the models are likely to overuse knowledge priors to guess the answer. However, overusing priors causes the damage of correctness and robustness on grounding. Besides, the short question also provides limited information to locate the target object, while longer questions usually depict the target object from more different perspectives. Thus, though shorter questions are easier to answer, locating the object region with a concise query is still challenging.

Such observations suggest that the answering long questions are difficult for current methods, for commonsense questions, and grounding to images with concise commonsense query is also challenging. The results further demonstrate the importance of comprehensive VQA model evaluation, especially for the commonsense-related questions. The answer score and ground score play complementary roles in evaluating VQA models to better reveal their superiorities and limitations.

4.2.3 *Effect of compositional commonsense questions*

In this section, we analyze the robustness of models by comparing the performances between the compositional and the simple problem. Specifically, the questions in CRIC are divided into two groups: one group of questions directly asks the content in a knowledge item which is less compositional, e.g., `QueryObjKG`, `VerifyKG`; another group requires an indirect use of knowledge items which are more compositional, e.g., `QueryObjSG`, `VerifyAtt`. In the Fig. 15, we compare the performances of models on these two groups of questions, `QueryObjSG` vs. `QueryObjKG`, and `VerifyKG` vs. `VerifySG`. From the results of **answer score** of query-type questions (top left figure), we can see a large gap between `QueryObjSG` and `QueryObjKG`. While for **ground score** (top middle figure), there is only a small gap between the two types of questions. In other words, under a similar grounding ability, it is more difficult to answer questions which indirectly query the commonsense. This demonstrates that the compositional questions effectively evaluate whether the model really understands the vision and commonsense. Besides, the performances of a model are very close on `VerifyAtt` and `VerifyKG` questions (bottom figures). This may be because the main function of knowledge prior is to reduce the number of candidate answers, while this function is invalid in answering verify-type questions.

These results also reflect that increasing the grounding performances is one of the most important direction for current methods. It limits the performance of compositional questions.

4.2.4 *Effect of attention supervision*

In this section, we show the importance of our collected additional attention annotations in model training and evaluation. In Tab. 2, we present three sets of models, Bottom-Up, Memory-VQA, ViLBERT, and their corresponding versions with attention supervision. Adding attention supervision brings significant improvements in ground scores and final scores for all three models and slight improvements in answer scores for Memory-VQA and ViLBERT. This shows that even if the model has an explicit attention module, it is still difficult for the model to learn a robust object localization spontaneously. Especially for questions whose answers come from knowledge items, in the top middle of Fig. 15, we can see that the models achieve larger improvement on `QueryObjKG` questions after adding attention supervision. Besides, it is found that the stronger the model, the more noticeable this phenomenon. ViLBERT itself is a very outstanding model in object localization proven in many other tasks [66], but it is difficult to give full play to this advantage without adding appropriate supervision information.

These results suggest that our attention annotations are also critical for evaluation and we may need to utilize the

attention annotations to increase the model's robustness and allow the model to output meaningful intermediate results.

4.2.5 *Effect of explicit use of knowledge items*

The main difference between Bottom-Up and Memory-VQA is that the latter has an additional branch that explicitly utilizes the knowledge items to answer the question. From the results of these two types of models in Table 2, we can see that the explicit use of commonsense knowledge not only brings an improvement in answering, but also a significant improvement in grounding ($> 9\%$ absolute improvement). This is an interesting phenomenon: what we provide to the model is actually more clear knowledge prior information, but this information does not exacerbate overfitting on priors (i.e., achieving higher answer score and lower ground score), instead, it helps the model gain better robustness and learn more meaningful intermediate result (i.e., achieving an obvious improvement in grounding).

In the Fig. 16, we show the visualized attention scores of Memory-VQA+l_{att} over knowledge items. First, we can see that the answering process of VQA is complicated, and errors may occur in every step, e.g., knowledge attention (Q3), and visual concept recognition (Q4). In addition, the existing model is not strictly modular design, but in an end-to-end manner. Therefore, the correct answer does not guarantee the correct intermediate result and vice versa. For example, the model may have located the correct region based on the knowledge information mentioned in the question, but made a mistake in recognizing the attribute (Q4). Or, although the model mistakenly selects the knowledge item, it can still have a chance to resort to the visual and language information in question to correctly guess the answer (Q3). This phenomenon further reveals that it is really very complex and important to conduct a comprehensive VQA evaluation. CRIC hopes to ease the difficulty in evaluation by providing various annotations to diagnose the complicated pipeline..

We can also see that the model usually attends on the relevant knowledge item with a higher score (the items in green or orange background). But, when many object categories meet the commonsense requirement of a question, it is still difficult to accurately locate the most relevant knowledge item, e.g., **Q3**. In other words, it is crucial for solving the CRIC task to design models to align knowledge and visual content locally. Besides, we also find that the distribution of the model's attention scores in some cases is relatively uniform. This is consistent with the intuition that sometimes the model needs to consider not only the knowledge item related to the target object, but also other knowledge items that help exclude the wrong objects.

The above experiments show that explicitly utilizing knowledge items is very effective. Only a proposed straightforward mechanism to explicitly use knowledge items shows obvious improvements. We believe that further exploring some techniques about jointly representing vision and knowledge will great benefit the performance, e.g., how to align the commonsense and vision, represent knowledge items, and use multiple knowledge items simultaneously.

Fig. 16. Visualization of Memory-VQA+l_{ans}'s attention scores on knowledge items. The highlighted knowledge items are the items used for generating their question. The red bounding boxes are predicted target objects.

4.2.6 *Challenging subtasks of the CRIC*

To better display the CRIC dataset's challenges, we conduct an additional experiment that tests the performance of each module in NMN-CS which can access ground-truth attention inputs and text inputs (denoted as NMN-CS-GT). More specifically, the outputted attention scores of grounding-related modules are passed through a sigmoid layer to determine if attending on the corresponding object candidate or not. In addition, the grounding modules are trained by ground-truth attention outputs with the weighted binary cross-entropy loss, where the weighted loss is to tackle the problem that attended objects and background objects are highly imbalanced. And the recognition module is trained by binary cross-entropy loss.

For program prediction, the accuracy of function name

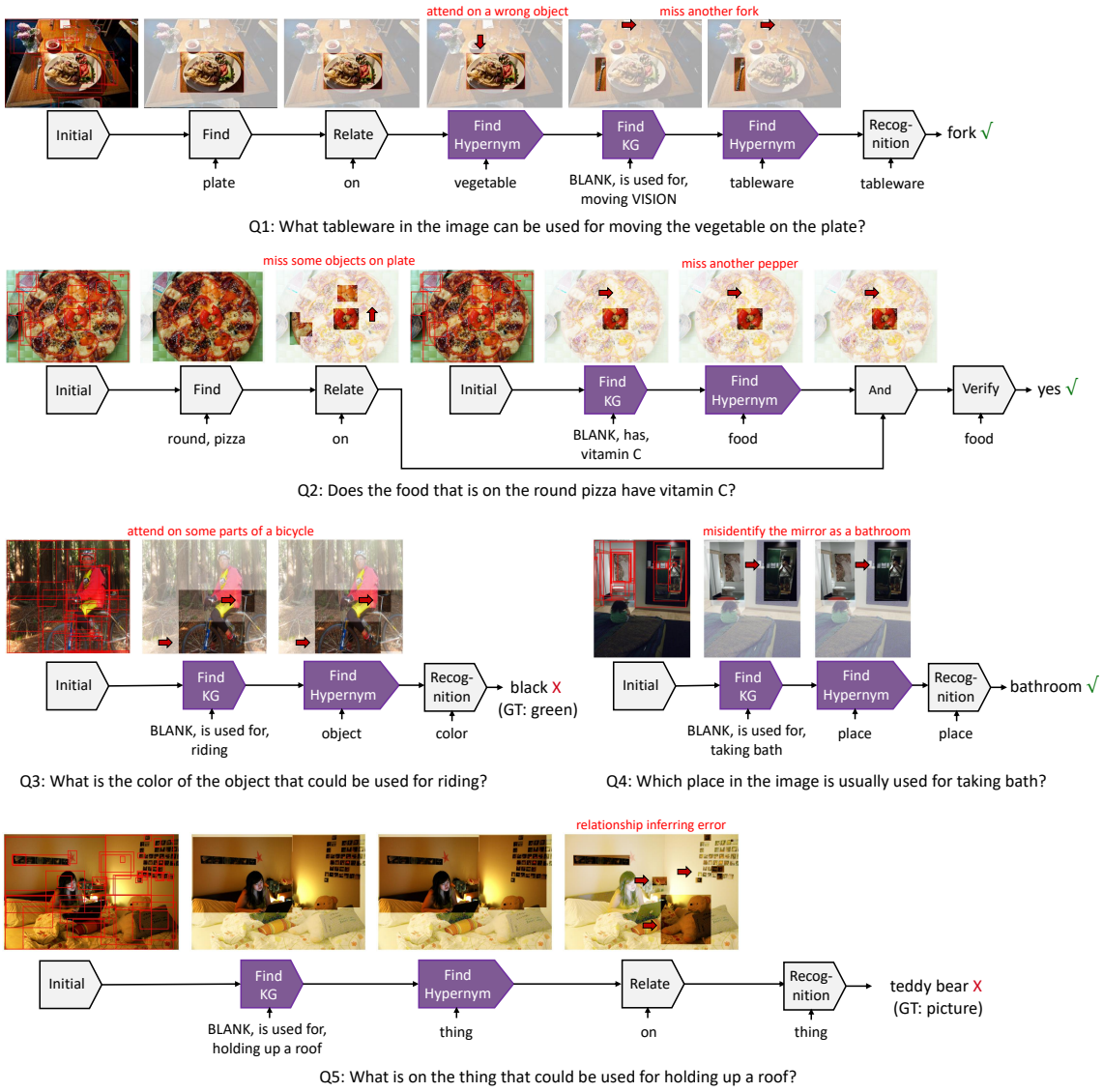


Fig. 17. Question answering examples of NMN-CS-GT which uses ground-truth text inputs. Though the model correctly answers the questions, the model still makes some mistakes in the intermediate modules. Precisely accomplishing each sub-task is still challenging.

prediction is 99.25%. This indicates that understanding the question is relatively easy. For grounding-related modules, we define the ground score as the IoU score of the output object set $O_{predict}$ and the ground-truth object set O_{gt},

$$score = \frac{|O_{predict} \cap O_{gt}|}{|O_{predict} \cup O_{gt}|}, \tag{4}$$

where $|.|$ indicates the number of elements in a set. The IoU score is a strict metric because a slight difference with O_{gt} will cause a relatively low score when $|O_{gt}|$ is small, so this metric can better expose the flaw of the grounding model.

From the results in Tab. 3, we can identify that `Find` and `Find_Hypernym` tasks are relatively easy, but it is still far from satisfactory. In addition, though the score of `Find_Hypernym` is higher than Find, it doesn't mean that `Find_Hypernym` is much easier. The `Find_Hypernym` sometimes requires the model to locate objects belonging to "object" or "thing", which is relatively easy, while correctly finding objects belonging to a category, such as "furniture" or "vegetable", is still challenging. Moreover, `Find_KG` related tasks are relatively more difficult than others. Along with the phenomenon in Tab. 2 that many models achieve high accuracy in answering, it shows that it is easy to understand the commonsense on semantic-level. Still, it isn't easy to learn to ground the commonsense knowledge into the images. For the `Recognition` task, the score is the accuracy of the output. We can see that the Recognition task is also relatively difficult for current models. This might be because, the visual genome dataset involves a large number

TABLE 3
The performance of each module in NMN-CS-GT. The scores of grounding-related module are IoU scores of predicted object set and ground-truth object set. The score of Recognition module is accuracy.

	Find	Relate	Find_Hypernym	Find_KG	Recognition
Score	40.56	18.03	47.61	16.72	55.34

of visual concepts, including thousands of object categories and attributes with diverse semantics.

Fig. 17 shows the qualitative results of the NMN-CS-GT, where modules are separately trained with ground-truth function output. It can be seen that even though the whole model predicts the correct answer, it is still challenging to provide precise intermediate attention results. For example, the model correctly predicts that *fork* can be used for moving the vegetable for **Q1**, while it doesn't correctly find the vegetable and all forks in the image.

In brief, many sub-tasks of CRIC is difficult and the cumulative error is the main bottle neck restricting the multi-step reasoning performance of modular networks. We may need to propose new representations to improve of each module, or present a new framework of modular network to avoid the cumulative error. Besides, it also shows that our provided rich ground-truth annotations can assist in diagnosing and improving the robustness of future models.

4.2.7 Will KB-related challenges be easily solved as long as the SG is correctly predicted?

To answer this question, we tested several new baselines with different types of inputs:

BERT - Predicted-SG. It takes the question and scene graph predicted by an existing SOTA scene graph generation (SGG) model [76] as input, then answer the question. Specifically, the scene graph is transformed into a sequence by listing the relationships separated by [SEP] token, e.g., "boy hold cup [SEP] boy wear hat [SEP] ...". Then, we concatenate the question with the transformed scene graph, and feed it into the BERT model to output the answer.

BERT - GT-SG. This model is similar to the BERT + Predicted-SG, but takes the question and Ground Truth Scene Graph (GT-SG) of the given image as input, then answers the question.

BERT - SG & KG for QAG. This model takes the question along with SG and KG used for generating the QA sample as input (i.e., the relationships unrelated to the question are filtered out), then predicts the answer.

TABLE 4
The answering score of BERT models with different types of inputs.

	Input	Verify	Recognize	Overall Score
BERT	Question (Q)	71.30	53.97	59.03
	Q + Image (i.e., ViLBERT + l_{att})	87.63	73.42	77.54
	Q + Predicted-SG	73.90	58.81	63.21
	Q + GT-SG	94.33	89.80	91.12
	Q + SG & KG for QAG	99.73	98.91	99.14

From the results in Tab. 4, we can see that given GT-SG, the model does have a significant performance improvement, but it is still far from 100% accurate. The accuracy is close to 100 only when the specific SG and KG for generating the question are given. This indicates that there are still many challenges remained. Specifically, 1) The model still needs to find the most relevant relationships to the questions from many relationships. 2) The models also need correctly retrieve knowledge items from the whole knowledge graph. Since we rephrase the knowledge entities in generating questions, e.g., using referring expressions to replace the object name in knowledge items, it challenges models' knowledge retrieval abilities. 3) The compositional nature of CRIC also introduces the challenge of multi-hop reasoning on multiple knowledge items and relationships.

In addition, the performance of BERT - Predicted-SG shows that scene graph predicted by current SGG methods is still hard to be an alternative of an image in QA reasoning. It may be difficult for models to output a scene graph with complete information about an image because a picture is worth a thousand words. Even ground truth SG is likely to miss some attributes of objects or relationships between objects in annotating. Thus, improving the quality of the predicted scene graph is undoubtedly one potential direction, but we want to emphasize that although CRIC is based on GT-SGs which are explicit representations of images, our dataset is also friendly to other methods which implicitly represent the image, like, NMN, ViLBERT. CRIC hopes to support the research of all possible types of methods.

4.2.8 Can a GQA method solve CRIC easily?

From the results of compositional models, MAC, MAC-CS, NMN-CS, in Tab. 2 with monolithic method like Bottom-Up, we can see that the compositional models do have obvious advantages over the monolithic model, but it is still far from satisfactory. We believe there are some additional challenges in knowledge retrieval, representation, and joint reasoning on knowledge and vision for GQA methods. We also train the ViLBERT model on CRIC and evaluate it on GQA, and vice-versa. From Tab. 5, we can see obvious performance drops under zero-shot transfer, both in GQA to CRIC and CRIC to GQA settings. The result reveals that the question domains of the two datasets are rather different. It indirectly shows that CRIC cannot be solved by simply transferring a GQA model.

TABLE 5
Zero-shot transfer Answer Score of ViLBERT between CRIC and GQA.

ViLBERT	Train Dataset: GQA	Train Dataset: CRIC
Test Dataset: GQA	60.51	19.42
Test Dataset: CRIC	13.80	76.07

5 Discussions

Comparison with FVQA, GQA, and OK-VQA. The CRIC extends the VQA task along two directions: towards multi-hop reasoning and understanding of non-visual knowledge of multiple objects, as illustrated in Fig. 1. Here, we introduce more details about the key features of the CRIC that differentiate it from existing ones, e.g., FVQA, GQA, and OK-VQA.

1) The CRIC introduces new types of compositional commonsense-related questions, e.g. QueryObjSG, Verify-Att, etc., to evaluate some unique capabilities. The questions in GQA mainly measure the understanding of visual contents (no external knowledge sources are used to generate questions). The pioneering commonsense VQA work, FVQA, primarily focuses on QueryObjKG type questions with limited visual reasoning. The questions of OK-VQA mainly evaluate the breadth of knowledge. It requires to

crawl information from the internet. In comparison, our new types of questions mainly aim to investigate whether the model can ground the commonsense into the visual world.

2) As far as we know, our dataset is the first large scale commonsense VQA datasets where attention results, the program for answering questions and question-related scene graph and knowledge graph are all available. Because of that, we can more comprehensively evaluate various types of methods. Besides, in Sec. 4.2.4 and Sec. 4.2.5, we also show that different types of annotations facilitate developing more robust commonsense VQA models. Multiple types of annotations also facilitate different types of models to be merged.

Open Issues of Auto-Generated Datasets. As another new automatically-generated dataset, our CRIC inherently has the following advantages: 1) Ease the risk of overusing priors. As stated above, the CRIC addresses this issue in each vital stage during construction. 2) Provide rich annotations for detailed evaluation and diagnosis. 3) Easy to measure the complexity of questions. The number and the types of sub-tasks involved in a question can assess its complexity for better diagnosis. 4) Easy to extend the dataset on new images or knowledge items. While embracing such favorite features at a low cost of human labor, like other auto-generated datasets (e.g., CLEVR, GQA), our CRIC faces the challenge of maintaining questions' naturalness. To tackle this issue, a certain level of human interference, like rephrasing the questions, would be necessary in the future.

6 Conclusion

This paper introduces the CRIC dataset that evaluates VQA systems on answering questions requiring compositional reasoning on the vision and commonsense. To build this dataset, we first propose a new Knowledge Graph format for easily aligning knowledge items to visual entities and depicting the commonsense relations between objects. Then, we propose an efficient method to generate numerous QA pairs and rich annotations automatically. Our generation method has better scalability and requires lower cost, easing the difficulty of building a complex VQA dataset.

Further experiments analyze the current four representative types of models. The results demonstrate our annotations' effectiveness on both comprehensive evaluation and enhancing the models' performances and robustness. The CRIC also brings new challenges for representation and reasoning of vision, question, and knowledge, e.g., how to design a model to capture the joint of graphs' global information in two modalities; how to conduct multi-hop reasoning on these two graphs explicitly; how to uniformly represent the commonsense and vision to better ground the commonsense. And various types of annotations will help researchers integrate the ideas of multiple types of models or propose a new unified framework to solve these challenges. For example, redesign the BERT as a modular network, equip modular networks with the ability of explicitly using the commonsense, or propose a pre-training model processing vision, language, and knowledge items simultaneously. In brief, we hope the CRIC can help drive the research of more transparent and robust models for designing more advanced AI agent reasoning on the vision and commonsense.

Acknowledgments

This work is partially supported by Natural Science Foundation of China under contracts Nos. U21B2025, U19B2036, 61922080, and National Key R&D Program of China No. 2021ZD0111901.

References

[1] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 248–255.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NIPS)*, 2012, pp. 1097–1105.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems (NIPS)*, 2015, pp. 91–99.

[5] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980–2988.

[6] H. Zhang, Z. Kyaw, S.-F. Chang, and T.-S. Chua, "Visual translation embedding network for visual relation detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5532–5540.

[7] Y. Li, W. Ouyang, X. Wang, and X. Tang, "Vip-cnn: Visual phrase guided convolutional neural network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 7244–7253.

[8] M. Ren, R. Kiros, and R. Zemel, "Exploring models and data for image question answering," in *Advances in Neural Information Processing Systems (NIPS)*, 2015, pp. 2953–2961.

[9] M. Malinowski and M. Fritz, "A multi-world approach to question answering about real-world scenes based on uncertain input," in *Advances in Neural Information Processing Systems (NIPS)*, 2014, pp. 1682–1690.

[10] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh, "Vqa: Visual question answering," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 2425–2433.

[11] J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. L. Zitnick, and R. Girshick, "Clevr: A diagnostic dataset for compositional language and elementary visual reasoning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1988–1997.

[12] D. A. Hudson and C. D. Manning, "Gqa: a new dataset for compositional question answering over real-world images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 6700–6709.

[13] P. Wang, Q. Wu, C. Shen, A. Dick, and A. van den Hengel, "Fvqa: Fact-based visual question answering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 10, pp. 2413–2427, 2018.

[14] S. Shah, A. Mishra, N. Yadati, and P. P. Talukdar, "Kvqa: Knowledge-aware visual question answering," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2019, pp. 8876–8884.

[15] Q. Cao, B. Li, X. Liang, K. Wang, and L. Lin, "Knowledge-routed visual question reasoning: Challenges for deep representation embedding," *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, 2021.

[16] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. S. Bernstein, and L. Fei-Fei, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *International Journal of Computer Vision*, vol. 123, no. 1, pp. 32–73, 2017.

[17] H. Liu and P. Singh, "Conceptnet—a practical commonsense reasoning tool-kit," *BT Technology Journal*, vol. 22, no. 4, pp. 211–226, 2004.

[18] R. Hu, J. Andreas, M. Rohrbach, T. Darrell, and K. Saenko, "Learning to reason: End-to-end module networks for visual question answering," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 804–813.

[19] J. Johnson, B. Hariharan, L. van der Maaten, J. Hoffman, L. Fei-Fei, C. L. Zitnick, and R. Girshick, "Inferring and executing programs for visual reasoning," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2989–2998.

[20] K. Marino, M. Rastegari, A. Farhadi, and R. Mottaghi, "Ok-vqa: A visual question answering benchmark requiring external knowledge," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 3195–3204.

[21] R. Zellers, Y. Bisk, A. Farhadi, and Y. Choi, "From recognition to cognition: Visual commonsense reasoning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 6720–6731.

[22] P. Wang, Q. Wu, C. Shen, A. Dick, and A. Van Den Henge, "Explicit knowledge-based reasoning for visual question answering," in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2017, pp. 1290–1296.

[23] Y. Qi, K. Zhang, A. Sain, and Y.-Z. Song, "Pqa: Perceptual question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 12 056–12 064.

[24] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh, "Making the v in vqa matter: Elevating the role of image understanding in visual question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6904–6913.

[25] Y. Zhu, O. Groth, M. Bernstein, and L. Fei-Fei, "Visual7w: Grounded question answering in images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4995–5004.

[26] K. Kafle and C. Kanan, "An analysis of visual question answering algorithms," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 1965–1973.

[27] L. Yu, E. Park, A. C. Berg, and T. L. Berg, "Visual madlibs: Fill in the blank description generation and question answering," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 2461–2469.

[28] V. Agarwal, R. Shetty, and M. Fritz, "Towards causal vqa: Revealing and reducing spurious correlations by invariant and covariant semantic editing," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 9690–9698.

[29] A. Kembhavi, M. Seo, D. Schwenk, J. Choi, A. Farhadi, and H. Hajishirzi, "Are you smarter than a sixth grader? textbook question answering for multimodal machine comprehension," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 4999–5007.

[30] L. Asprino, L. Bulla, L. Marinucci, M. Mongiovì, and V. Presutti, "A large visual question answering dataset for cultural heritage," in *International Conference on Machine Learning, Optimization, and Data Science*, 2021, pp. 193–197.

[31] P. Bongini, F. Becattini, A. D. Bagdanov, and A. Del Bimbo, "Visual question answering for cultural heritage," in *IOP Conference Series: Materials Science and Engineering*, vol. 949, no. 1, 2020, p. 012074.

[32] R. Hu, A. Singh, T. Darrell, and M. Rohrbach, "Iterative answer prediction with pointer-augmented multimodal transformers for textvqa," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 9992–10 002.

[33] A. F. Biten, R. Tito, A. Mafla, L. Gomez, M. Rusinol, E. Valveny, C. Jawahar, and D. Karatzas, "Scene text visual question answering," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019, pp. 4291–4301.

[34] A. Singh, V. Natarajan, M. Shah, Y. Jiang, X. Chen, D. Batra, D. Parikh, and M. Rohrbach, "Towards vqa models that can read," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 8317–8326.

[35] D. Gao, K. Li, R. Wang, S. Shan, and X. Chen, "Multi-modal graph neural network for joint reasoning on vision and scene text," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 12 746–12 756.

[36] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "Dbpedia: A nucleus for a web of open data," in *The Semantic Web*. Springer, 2007, pp. 722–735.

[37] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: a collaboratively created graph database for structuring human knowledge," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 2008, pp. 1247–1250.

[38] N. Tandon, G. De Melo, F. Suchanek, and G. Weikum, "Webchild: Harvesting and organizing commonsense knowledge from the web," in *Proceedings of the ACM international conference on Web search and data mining*, 2014, pp. 523–532.

[39] J. Berant, A. Chou, R. Frostig, and P. Liang, "Semantic parsing on freebase from question-answer pairs," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2013, pp. 1533–1544.

[40] J. Weston, S. Chopra, and A. Bordes, "Memory networks," *International Conference on Learning Representations (ICLR)*, 2015.

[41] S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus, "End-to-end memory networks," in *Advances in Neural Information Processing Systems (NIPS)*, 2015, pp. 2440–2448.

[42] Y. Yang, W.-t. Yih, and C. Meek, "Wikiqa: A challenge dataset for open-domain question answering," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2015, pp. 2013–2018.

[43] M. Qi, Y. Wang, J. Qin, and A. Li, "Ke-gan: Knowledge embedded generative adversarial networks for semi-supervised scene parsing," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 5237–5246.

[44] S. N. Aakur, F. D. M. de Souza, and S. Sarkar, "Generating open world descriptions of video using common sense knowledge in a pattern theory framework," in *Quarterly of Applied Mathematics*, 2019.

[45] J. Gu, H. Zhao, Z. Lin, S. Li, J. Cai, and M. Ling, "Scene graph generation with external knowledge and image reconstruction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 1969–1978.

[46] M. Narasimhan, S. Lazebnik, and A. Schwing, "Out of the box: Reasoning with graph convolution nets for factual visual question answering," in *Advances in Neural Information Processing Systems (NIPS)*, 2018, pp. 2654–2665.

[47] M. Narasimhan and A. G. Schwing, "Straight to the facts: Learning knowledge base retrieval for factual visual question answering," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 451–468.

[48] G. Li, H. Su, and W. Zhu, "Incorporating external knowledge to answer open-domain visual questions with dynamic memory networks," *arXiv preprint arXiv:1712.00733*, 2017.

[49] A. Zareian, S. Karaman, and S.-F. Chang, "Bridging knowledge graphs to generate scene graphs," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020, pp. 606–623.

[50] P. Wang, D. Liu, H. Li, and Q. Wu, "Give me something to eat: referring expression comprehension with commonsense knowledge," in *Proceedings of the ACM International Conference on Multimedia*, 2020, pp. 28–36.

[51] P. Singh, T. Lin, E. T. Mueller, G. Lim, T. Perkins, and W. L. Zhu, "Open mind common sense: Knowledge acquisition from the general public," in *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, 2002, pp. 1223–1237.

[52] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[53] J. Breen, "Jmdict: a japanese-multilingual dictionary," in *Proceedings of the Workshop on Multilingual Linguistic Ressources*, 2004, pp. 71–79.

[54] R. Reiter, "On closed world data bases," in *Readings in artificial intelligence*. Elsevier, 1981, pp. 119–140.

[55] S. T. Sherry, M.-H. Ward, M. Kholodov, J. Baker, L. Phan, E. M. Smigielski, and K. Sirotkin, "dbsnp: the ncbi database of genetic variation," *Nucleic acids research*, vol. 29, no. 1, pp. 308–311, 2001.

[56] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky, "The Stanford CoreNLP natural language processing toolkit," in *Association for Computational Linguistics (ACL) System Demonstrations*, 2014, pp. 55–60. [Online]. Available: http://www.aclweb.org/anthology/P/P14/P14-5010

[57] W. Zhu and S. Bhat, "Gruen for evaluating linguistic quality of generated text," *arXiv preprint arXiv:2010.02498*, 2020.

[58] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and vqa," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 6077–6086.

[59] M. Narasimhan and A. G. Schwing, "Straight to the facts: Learning knowledge base retrieval for factual visual question answering," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 451–468.
[60] F. Gao, Q. Ping, G. Thattai, A. Reganti, Y. N. Wu, and P. Natarajan, "A thousand words are worth more than a picture: Natural language-centric outside-knowledge visual question answering," *arXiv preprint arXiv:2201.05299*, 2022.
[61] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola, "Stacked attention networks for image question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 21–29.
[62] J. Andreas, M. Rohrbach, T. Darrell, and D. Klein, "Neural module networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 39–48.
[63] D. A. Hudson and C. D. Manning, "Compositional attention networks for machine reasoning," in *International Conference on Learning Representations*, 2018.
[64] J. Lu, D. Batra, D. Parikh, and S. Lee, "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," in *Advances in Neural Information Processing Systems (NIPS)*, 2019, pp. 13–23.
[65] H. Tan and M. Bansal, "Lxmert: Learning cross-modality encoder representations from transformers," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2019, pp. 5103–5114.
[66] J. Lu, V. Goswami, M. Rohrbach, D. Parikh, and S. Lee, "12-in-1: Multi-task vision and language representation learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10 437–10 446.
[67] Y.-C. Chen, L. Li, L. Yu, A. El Kholy, F. Ahmed, Z. Gan, Y. Cheng, and J. Liu, "Uniter: Universal image-text representation learning," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020, pp. 104–120.
[68] G. Li, N. Duan, Y. Fang, M. Gong, and D. Jiang, "Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2020, pp. 11 336–11 344.
[69] X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, F. Wei *et al.*, "Oscar: Object-semantics aligned pre-training for vision-language tasks," in *European Conference on Computer Vision*, 2020, pp. 121–137.
[70] L. Zhou, H. Palangi, L. Zhang, H. Hu, J. Corso, and J. Gao, "Unified vision-language pre-training for image captioning and vqa," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, pp. 13 041–13 049.
[71] A. Majumdar, A. Shrivastava, S. Lee, P. Anderson, D. Parikh, and D. Batra, "Improving vision-and-language navigation with image-text pairs from the web," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020, pp. 259–274.
[72] P. Zhang, X. Li, X. Hu, J. Yang, L. Zhang, L. Wang, Y. Choi, and J. Gao, "Vinvl: Revisiting visual representations in vision-language models," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 5579–5588.
[73] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2019, pp. 4171–4186.
[74] Y. Sun, S. Wang, Y. Li, S. Feng, H. Tian, H. Wu, and H. Wang, "Ernie 2.0: A continual pre-training framework for language understanding," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 34, no. 05, 2020, pp. 8968–8975.
[75] F. Petroni, T. Rocktäschel, P. Lewis, A. Bakhtin, Y. Wu, A. H. Miller, and S. Riedel, "Language models as knowledge bases?" *arXiv preprint arXiv:1909.01066*, 2019.
[76] K. Tang, Y. Niu, J. Huang, J. Shi, and H. Zhang, "Unbiased scene graph generation from biased training," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 3716–3725.

Difei Gao received the B.S. degree in electronic engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2015. He received Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2022. His research interests mainly include computer vision, pattern recognition, machine learning, and, in particular, visual question answering and commonsense reasoning.

Ruiping Wang (M'11−SM'22) received the B.S. degree in applied mathematics from Beijing Jiaotong University, Beijing, China, in 2003, and the Ph.D. degree in computer science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, in 2010. He worked as a Post-Doctoral Researcher with Tsinghua University from 2010 to 2012, and a Research Associate with University of Maryland at College Park from 2010 to 2011. In 2012, he joined the Faculty of ICT, CAS, where he has been a Professor since 2017. His research interests include computer vision, pattern recognition, and machine learning. He is currently an Associate Editor for Pattern Recognition, Neurocomputing. He has served as Area Chair for IEEE WACV18/19/20/22/23, ICME19/20, CVPR21/22, ICCV21, ECCV22 and ACCV22. He is a senior member of the IEEE.

Shiguang Shan (M'04−SM'15−F'21) received the M.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 1999, and the Ph.D. in computer science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, in 2004. In 2002, he joined ICT, CAS, where he has been a Professor since 2010. He is currently the Deputy Director of the Key Laboratory of Intelligent Information Processing of CAS. He has authored over 300 papers in refereed journals and proceedings in computer vision, pattern recognition, and machine learning. He was a recipient of the China's State Natural Science Award in 2015 and the China's State S&T Progress Award in 2005 for his research work. He is Associate Editor of several journals and has served as the Area Chair for a number of international conferences, including CVPR, ICCV, ICPR, ACCV, FG, AAAI, IJCAI, etc..

Xilin Chen (M'00−SM'09−F'16) is a professor with the Institute of Computing Technology, Chinese Academy of Sciences (CAS). He has authored one book and more than 400 papers in refereed journals and proceedings in the areas of computer vision, pattern recognition, image processing, and multimodal interfaces. He is currently an information sciences editorial board member of Fundamental Research, an editorial board member of Research, a senior editor of the Journal of Visual Communication and Image Representation, and an associate editor-in-chief of the Chinese Journal of Computers, and Chinese Journal of Pattern Recognition and Artificial Intelligence. He served as an organizing committee member for multiple conferences, including general co-chair of FG 2013 / FG 2018, VCIP 2022, and program co-chair of ICMI 2010. He is a fellow of the ACM, IEEE, IAPR, and CCF.