

Explicit Knowledge Incorporation for Visual Reasoning

Yifeng Zhang*, Ming Jiang*, Qi Zhao
University of Minnesota

{zhan6987, mjiang, qzhao}@umn.edu

Abstract

Existing explainable and explicit visual reasoning methods only perform reasoning based on visual evidence but do not take into account knowledge beyond what is in the visual scene. To address the knowledge gap between visual reasoning methods and the semantic complexity of real-world images, we present the first explicit visual reasoning method that incorporates external knowledge and models high-order relational attention for improved generalizability and explainability. Specifically, we propose a knowledge incorporation network that explicitly creates and includes new graph nodes for entities and predicates from external knowledge bases to enrich the semantics of the scene graph used in explicit reasoning. We then create a novel Graph-Relate module to perform high-order relational attention on the enriched scene graph. By explicitly introducing structured external knowledge and high-order relational attention, our method demonstrates significant generalizability and explainability over the state-of-the-art visual reasoning approaches on the GQA and VQAv2 datasets.

1. Introduction

Visual question answering (VQA) aims to answer natural language questions about a visual scene. It is a challenging task requiring a deep understanding of both vision and language inputs, as well as knowledge to answer open-ended questions. While deep neural networks (DNNs) are extraordinarily powerful, most DNN-based VQA methods are black boxes driven by superficial correlations between questions and answers [2]. These models are therefore limited in making inferences or generalizations. They also fall short in explaining their decision-making process, especially with complex questions requiring multiple reasoning steps to answer. The lack of generalizability or explainability in DNN models slows down their applications in many domains, such as healthcare, security, and finance.

Recent studies aim to address these problems by rep-

*These authors contributed equally.

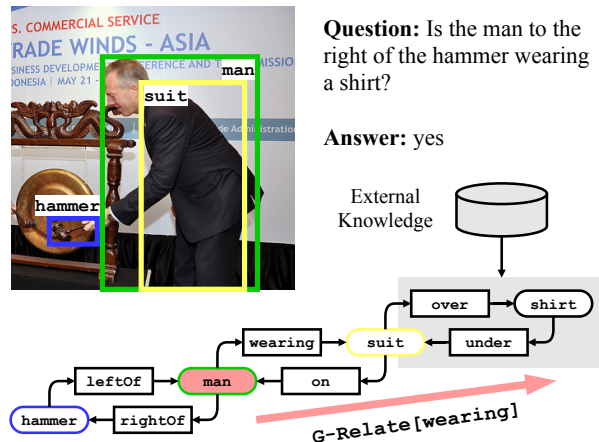


Figure 1. Explicit visual reasoning methods often fail when the observation does not provide sufficient knowledge. Our method addresses this problem by generating scene graphs with explicit knowledge incorporation (e.g., suit-over-shirt) and inferring high-order relations (e.g., man-wearing-suit-over-shirt) with a novel G-Relate neural module.

resenting the visual information as a structured scene graph [24] or converting the question into a program of executable neural modules [11, 12]. These explainable and explicit reasoning models have achieved remarkable performances on synthetic scenes and questions [14]. However, due to the complexity of real-world images and questions, they are still far from satisfactory when tested on more general VQA datasets [5, 13]. These data-driven methods depend on the accuracy and completeness of the detected objects and their relations, and are ignorant of commonsense or other useful knowledge beyond visual observations. For example, as shown in Fig. 1, to answer the question “Is the man to the right of the hammer wearing a shirt?” visual reasoning models need to detect the shirt and attend to it if it exists. The reasoning task in this example is challenging as the shirt is undetectable from the scene. On the other hand, humans can easily integrate the observation that “the man is wearing a suit” and the commonsense knowledge that “suits are commonly dressed over shirts”, to infer the high-order relation between man and shirt. In this work, to achieve generalizability and explainability in visual reason-

ing, we propose an explainable and explicit visual reasoning method based on *knowledge incorporation* and *high-order relational attention*. It depicts two major advantages over existing approaches:

First, existing visual reasoning studies either implicitly embed external knowledge as language features [12, 24] or propagate information from external knowledge graphs into a scene graph with static topology [32], which is not able to address undetected objects or missing concepts from the visual scene. Differently, in this work, we explicitly incorporate commonsense knowledge from an external knowledge graph into the scene graph by adding entities and predicates as new nodes. As shown in Fig. 1, with our proposed method, the external relations *shirt-under-suit* and *suit-over-shirt* can be added to the scene graph to enrich the scene graph. This enriched scene graph offers richer semantics enabling generalizable and explainable reasoning.

Second, existing methods depend on the detected binary relations but lack a mechanism to infer high-order relations between distant nodes in the scene graph. For example, as shown in Fig. 1, existing neural module networks cannot reason correctly with first-order *Relate* modules, because either no direct relations are detected between man and shirt or the question does not specify both (*e.g.*, *wearing* and *over*) relations. We address this challenge by designing a novel *Graph-Relate* module that enables high-order relational reasoning. Despite there is no direct relation between man and shirt, *G-Relate* can infer the probability of *man-wearing-shirt* based on the two direct relations *man-wearing-suit* and *suit-over-shirt*. This allows our model to efficiently transfer attention to non-adjacent graph nodes and answer the question correctly.

We summarize the contributions of this work as follows:

1. We propose the first explicit visual reasoning model that leverages external knowledge and neural modules to achieve generalizability and explainability.
2. We design a *Knowledge Incorporation Network (KI-Net)* that explicitly incorporates external knowledge as additional nodes and edges into a scene graph to provide rich semantics for reasoning.
3. We design a *Graph-Relate* module that achieves high-order relational attention based on the scene graph topology and semantics.
4. Our method outperforms state-of-the-art explicit reasoning methods on the *GQA* [13] and *VQAv2* [5] datasets, suggesting its superior generalizability and explainability.

2. Related Work

Scene graphs. Scene graphs have been pervasively adopted in various vision tasks, such as image captioning [8, 30, 31] and VQA [7, 24, 26]. A high-quality scene graph can accurately and reliably describe the visual contents of an image, and an incomplete or incorrect scene graph can de-

grade the performance of tasks of interest. To generate more accurate scene graphs, several studies have implicitly included external knowledge by representing knowledge as language features [28, 29, 33] or subject-predicate-object triplets [1]. Wu *et al.* [29] directly embeds external knowledge into language features and incorporates them with visual features. Gu *et al.* [9] queries class-wise relations by matching detected objects to classes in *Concept-Net* [19]. Zareian *et al.* [32] applies *Graph Convolutional Networks* [17] to propagate information across the scene graph and external knowledge graph. These methods only refine the features of graph nodes but not the graph topology, which cannot address issues about undetected objects or external concepts. Differently, by explicitly adding graph nodes for external entities and predicates, our method expands scene graphs to include richer semantics until the desired amount of external knowledge is incorporated. More importantly, it allows neural modules to be directly executed on these additional graph nodes, bridging the research gap of explainable visual reasoning with knowledge.

Explainable and explicit visual reasoning. Our method is related to a series of explainable and explicit reasoning methods [4, 10, 11, 12, 15, 20, 24]. Due to the remarkable learning ability of deep neural networks, end-to-end VQA models can easily learn the dataset bias without reasoning [14]. To address this problem, recent studies have developed composite reasoning models, by designing and executing neural modules based on image features [4, 10, 15, 20] or scene graphs [12, 24]. Recently, Shi *et al.* [24] proposes *eXplainable and eXplicit Neural Modules (XNMs)* that not only achieve 100% accuracy on the *CLEVR* [14] dataset but also allow to explicitly trace the attention shift in the scene graph following the reasoning progress. Similarly, *Neural State Machine (NSM)* [12] predicts a probabilistic graph and performs sequential reasoning over the graph with more generic modules. Our method differentiates itself from these related studies by introducing external knowledge and high-order relational attention. Our proposed *Graph-Relate* module propagates attention to non-adjacent nodes in the scene graph, enabling the efficient inference of high-order relations.

3. Approach

For the first time, we conduct explainable and explicit visual reasoning by leveraging scene graphs, external knowledge, and neural modules. Our method first creates an enriched scene graph by explicitly incorporating external knowledge and then executes a program of neural modules generated from the question. Fig. 2 highlights the two novelties of our method, including a *Knowledge Incorporation Network (KI-Net)* that explicitly incorporates entities and predicates from the external knowledge graph to the scene graph, and a *Graph-Relate (G-Relate)* module that infers

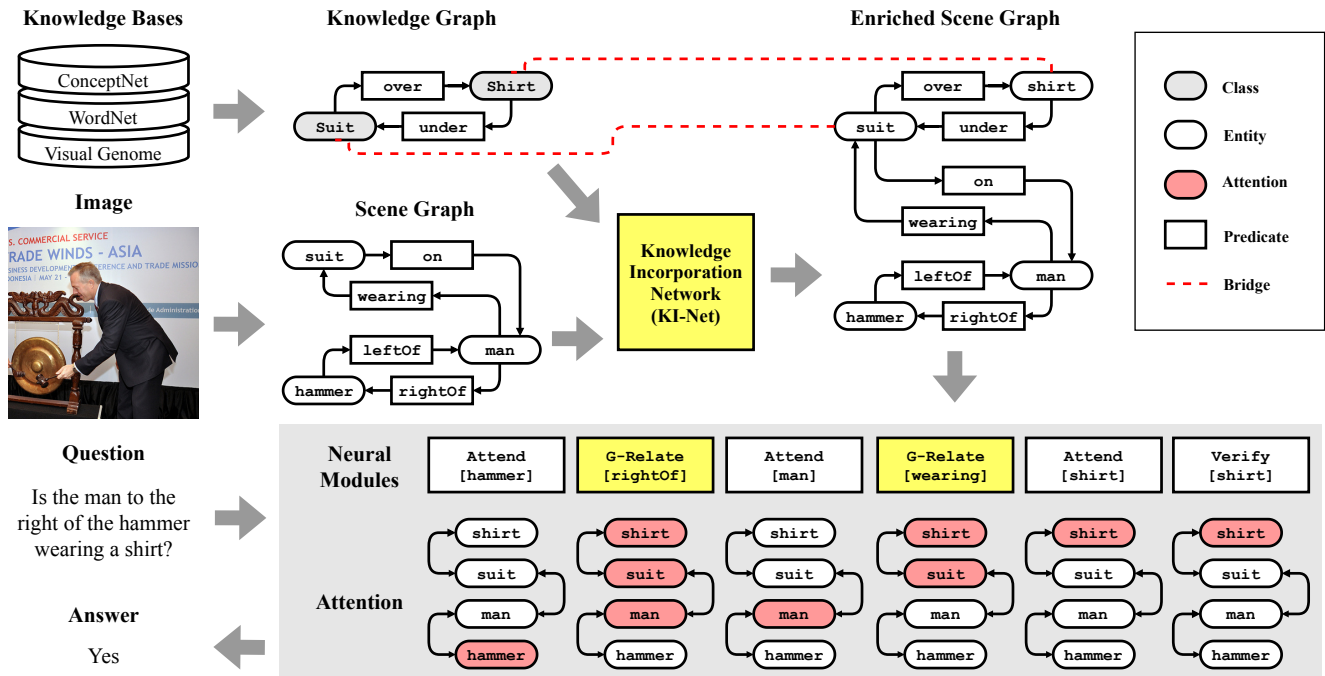


Figure 2. Overview of our proposed method. Our main contributions, the Knowledge Incorporation Network (KI-Net) and the Graph-Relate (G-Relate) module, are highlighted in yellow. Red nodes indicate the current attention.

high-order relations based on the enriched scene graph.

3.1. Knowledge Incorporation Network

Neural module networks are typically trained on datasets containing a specific set of semantics [13, 14], which makes them difficult to generalize and scale to a broader scope of knowledge. The proposed KI-Net aims to support explicit reasoning with richer semantics and allows the neural modules to trace the reasoning process beyond the visual observation. It is designed to explicitly incorporate external knowledge as scene graph nodes (see Fig. 3): Based on the topology of the external knowledge graph, it first performs *topological extension* to incorporate external relations into the scene graph (e.g., man-wearing-shirt and man-wearing-helmet in Fig. 3, by explicitly adding new candidate entities shirt and helmet to the scene graph). Then, taking the visual and semantic features into account, it performs *semantic refinement* to selectively discard the candidate entities with low relevance to the visual observation (e.g., the shirt in Fig. 3). The KI-Net results in an enriched scene graph that allows the neural modules to perform explicit reasoning on the incorporated semantics. It is supervised with ground-truth scene graph annotations using a cross-entropy loss.

Scene graph and knowledge graph. The KI-Net operates on an initial scene graph $\mathcal{G}_S = (\mathcal{V}_S, \mathcal{P}_S, \mathcal{E}_S)$ and an external knowledge graph $\mathcal{G}_K = (\mathcal{V}_K, \mathcal{P}_K, \mathcal{E}_K)$. The scene graph consists of entity nodes (i.e., object instances, denoted as \mathcal{V}_S) and predicate nodes (i.e., relations or interac-

tions between entities, denoted as \mathcal{P}_S) detected from the image. The knowledge graph consists of class nodes (i.e., general concepts, denoted as \mathcal{V}_K) and predicates (i.e., relations between concepts, denoted as \mathcal{P}_K) acquired from external knowledge bases. Both graphs can connect entities or classes with multiple predicates. They organize relations between entities or classes as a set of subject-predicate-object triplets, in which \mathcal{E}_S and \mathcal{E}_K contain directed edges linking from a subject to a predicate or from a predicate to an object. Each node is associated with a d_h -dimensional feature vector. Node features of the scene graph are initialized with regional features of the detected objects [3], while those of the knowledge graph are initialized with word embeddings [22]. The visual and external node features are fused with message passing following the GB-Net [32].

Topological extension. Based on the semantic matching between scene entities and external classes and the graph topology, we propose candidate entities to be added to the scene graph. First, each existing entity $e \in \mathcal{V}_S$ in the scene graph is bridged with a class node $g(e) \in \mathcal{V}_K$ with the same semantic meaning (i.e., the highest feature similarity above a threshold ϵ_{cls}). The bridging forms message passing paths between the scene graph and the knowledge graph. Next, we create candidate entity nodes to allow knowledge about unobserved but directly related concepts to be added to the scene graph. Let $d(\cdot, \cdot)$ denote the minimum number of predicates between a pair of input entities. We add a candidate entity e' along with its adjacent predicates p'

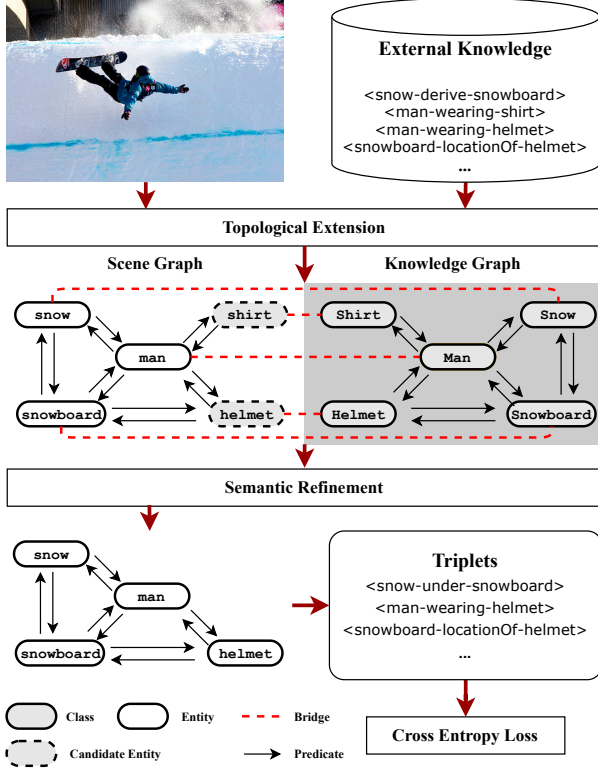


Figure 3. The knowledge incorporation process.

connecting to entity e if

$$\exists e \in \mathcal{V}_S, \quad d(g(e'), g(e)) = 1. \quad (1)$$

Finally, the features of e' and p' are directly copied from the corresponding nodes in the knowledge graph, and entity node e' is bridged with its class node $g(e')$ as they share the same features. This topological extension ensures that the candidate entities (e.g., shirt and helmet in the scene graph of Fig. 3) are directly related to the visual observation (e.g., man in Fig. 3) and semantically consistent with corresponding classes (e.g., Shirt and Helmet in the knowledge graph of Fig. 3). It builds abundant connections between the scene graph and the knowledge graph, so that their features can be jointly considered to compute the relevance between the new entities and the observed scene context.

Semantic refinement. The candidate entities have been added to the scene graph based on the knowledge graph topology, but their semantic relevance with the observed scene context is unknown. Therefore, we perform semantic refinement to maintain a compact scene graph while incorporating the most relevant external knowledge. To achieve this goal, with message passing, we compute a relevance matrix M measuring the feature relevance between different entities. The relevance weights in the matrix M are jointly decided by the visual features and the semantics from external knowledge.

Given two adjacent nodes $v_i, v_j \in \mathcal{V}_S \cup \mathcal{P}_S \cup \mathcal{V}_K \cup \mathcal{P}_K$ and their features h_i, h_j , the message passing is implemented as a Graph Attention Network [27]:

$$m_{ij} = \text{MLP}(h_i, h_j), \quad (2)$$

$$\phi_{ij} = \text{softmax}_{\mathcal{N}(v_i)}(m_{ij}), \quad (3)$$

$$h'_i = \sum_{\mathcal{N}(v_i)} \phi_{ij} h_j, \quad (4)$$

where $\mathcal{N}(v_i)$ denotes the set of adjacent nodes of v_i . The message passing results in the updated features h'_i for each node v_i , and a relevance matrix M contains all the pairwise relevance scores m_{ij} . We repeat this message passing K_{GAT} times to thoroughly propagate the features.

With the computed relevance matrix M , a candidate entity e'' is discarded when the sum of the top- K_p relevance scores between e'' and its adjacent nodes are smaller than a threshold ϵ_p . All its adjacent predicate nodes are also discarded. Finally, we remove all bridges and obtain an enriched scene graph with only the relevant nodes incorporated from the external knowledge graph. The topological extension and semantic refinement can be performed iteratively depending on the amount of knowledge required.

3.2. Reasoning with Neural Modules

Neural module networks are a class of reasoning methods that achieve explainable reasoning by composing and executing a set of handcrafted neural modules on top of image features [4, 10, 15, 20] or scene graphs [12, 24]. Recent neural module networks [24] have achieved perfect accuracy on synthetic visual reasoning datasets [14], but their generalization to semantically-rich real-world images is still an unsolved problem. Our KI-Net has generated an enriched scene graph with a broader scope of semantics, allowing explainable reasoning methods to generalize beyond the scope of training data. In this section, we focus on introducing the novel G-Relate module that can infer high-order relations by shifting attention to non-adjacent graph nodes.

To perform explicit reasoning on the enriched scene graph, we design three categories of neural modules: attention, logic, and output. These neural modules are grounded on four meta-types of atom modules that can represent all the question types in VQA datasets [5]. The attention modules compute the relative importance of different image contents (e.g., image features or scene graph nodes) during the reasoning process, which are essential to the answering of many questions. Attend computes the attention weights of entities based on their features, and G-Relate shifts attention to other related entities through a queried predicate. Besides the two attention modules, logic modules (i.e., And, Or, and Not) perform logical operations based on the attention weights, and output modules (i.e., Compare, Count, Exist, Choose, Describe, and Verify) compute output fea-

Modules	Category	Operation
Attend	Attention	$\mathbf{a} = \text{softmax}(\text{MLP}(\mathbf{h}, \mathbf{q}))$
G-Relate	Attention	$\mathbf{a}, \mathbf{h}, \mathbf{q} \rightarrow \mathbf{a}'$ (see Equ. (5))
Or	Logic	$\mathbf{a}' = \min(\mathbf{a}^1, \mathbf{a}^2)$
And	Logic	$\mathbf{a}' = \max(\mathbf{a}^1, \mathbf{a}^2)$
Not	Logic	$\mathbf{a}' = 1 - \mathbf{a}$
Compare	Output	$\mathbf{h}' = \text{MLP}(\mathbf{h}^1 - \mathbf{h}^2)$
Count	Output	$\mathbf{h}' = \text{MLP}(\text{sum}(\mathbf{a}))$
Exist	Output	$\mathbf{h}' = \text{MLP}(\text{sum}(\mathbf{a}))$
Choose	Output	$\mathbf{h}' = \text{softmax}(\text{MLP}(\mathbf{q}))\mathbf{W}(\mathbf{a} \circ \mathbf{h})$
Describe	Output	$\mathbf{h}' = \text{softmax}(\text{MLP}(\mathbf{q}))\mathbf{W}(\mathbf{a} \circ \mathbf{h})$
Verify	Output	$\mathbf{h}' = \text{softmax}(\text{MLP}(\mathbf{q}))\mathbf{W}(\mathbf{a} \circ \mathbf{h})$

Table 1. Our neural modules. $\text{MLP}(\cdot)$ indicates a multi-layer perceptron consisting of several fully-connected and ReLU layers, and \mathbf{W} is a matrix of learnable weights. The parameters \mathbf{a} , \mathbf{h} , and \mathbf{q} indicate attention, features, and query, respectively.

tures according to different question types. Tab. 1 summarizes the specific neural modules and their implementations. The three categories of neural modules are composed into a program to reason over the enriched scene graph. Taking both the graph topology and rich semantics into account, the neural program can explicitly trace the attention over the reasoning process to infer the answer.

Graph-Relate module. In neural module studies, relational inference is commonly implemented by reallocating attention considering the relevance to a predicate query [12]. Existing methods [24] either only shift attention between adjacent scene graph nodes, or learn a transfer matrix to propagate attention across all nodes regardless of the graph topology. In complex scene graphs, as the numbers of entities and predicates increase, high-order attention becomes a critical need that the existing neural modules cannot handle. For example, to answer the question ‘‘What is the phone on?’’, attention should be transferred from phone to both the adjacent entity table and the non-adjacent entity coffee (see Fig. 4). The features of coffee provide extra information about the table type. With first-order relate module, transferring attention to coffee is rather difficult, because no direct relation between phone and coffee can be extracted from the inputs. To address this challenge, we design a Graph-Relate module to infer high-order relations in the enriched scene graph, so that attention can be transferred along a path of relations to reach a distant entity.

Given the attention \mathbf{a} computed by the previous modules in the neural program, the G-Relate module computes a transfer matrix \mathbf{W}_h to propagate the attention over the scene graph. With this transfer matrix, the attention of the graph can be updated as:

$$\mathbf{a}' = \text{norm}(\mathbf{W}_h^T \mathbf{a}), \quad (5)$$

where $\text{norm}(\cdot)$ casts all attention weights of entity nodes

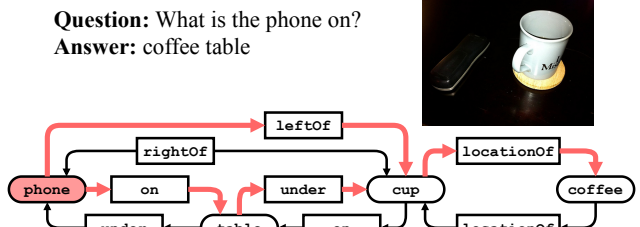


Figure 4. An example of attention transfer along different paths of high-order relations. Red nodes indicate the current attention, and red arrows indicate different paths to transfer attention from phone to coffee.

into $[0, 1]$ using a softmax function.

The transfer matrix \mathbf{W}_h can be computed in various ways. For example, in XNM [24], the encoded query \mathbf{q} and the edge features \mathbf{h}_{ij} are processed with a MLP to compute the transfer matrix. The edge features come from either the first-order ground-truth relations or the concatenation of two adjacent entity features. Differently, our G-Relate module considers high-order composite relations in the scene graph: we extract all possible relation paths $\mathcal{U}_{ij} = \{U_1, U_2, \dots, U_N\}$ connecting between e_i and e_j (within a maximum length L). For example (see Fig. 4), we extract two paths that describe the composite relation between coffee and phone: coffee-locationOf-cup-rightOf-phone and coffee-locationOf-cup-on-table-under-phone. Both paths consist of a set of first-order relations and contribute to the high-order relations between both entities. The transfer matrix is computed by considering different situations based on the topological distance $l_{ij} = d(e_i, e_j)$ between the entities e_i and e_j (i.e., the number of predicates along the path).

Formally, we compute the transfer weights w_{ij} between entities e_i and e_j based on predicate features and graph topology:

$$w_{ij} = \begin{cases} \text{softmax}_{\mathcal{N}(e_i)}(\max_{U_k \in \mathcal{U}_{ij}} (\text{MLP}(\mathbf{h}_k, \mathbf{q}))), & l_{ij} = 1 \\ \sum_{U_k \in \mathcal{U}_{ij}} \prod_{(e_a, e_b) \in U_k} w_{ab}, & 1 < l_{ij} \leq L \\ 0, & l_{ij} > L \end{cases} \quad (6)$$

where \mathbf{h}_k represents the features of the k -th predicate between entities e_i and e_j , and w_{ab} is the weight between adjacent entities e_a and e_b . The transfer weights of **first-order** relations (i.e., $l_{ij} = 1$) are computed directly based on the relevance between the predicate features and the query. A high transfer weight indicates that the predicate features are closely related to the query, and vice versa. Different from XNM, our graph structure allows multiple predicates to connect between two entities, and here we adopt their maximum weight. To measure the transfer weights of **high-order** relations ($1 < l_{ij} \leq L$), we compute the product of the first-order transfer weights along each path and linearly

combine them across multiple paths. We store the computed transfer weights w_{ij} into the relation matrix \mathbf{W}_h and update the attention at each entity node by propagating these weights across the whole graph. This process is integrated into the end-to-end training of neural modules.

4. Experiments and Results

We demonstrate our method with experiments on the GQA [13] and VQAv2 [5] datasets. Our method outperforms the state-of-the-art explicit reasoning methods, suggesting its superior ability to generate neural modules to explicitly reason over the enriched scene graph. Qualitative examples show that the complex reasoning process can be completely traced across multiple graph nodes. Our results also demonstrate the superior performance and generalizability of KI-Net on scene graph generation thanks to the incorporated external knowledge.

4.1. Implementation Details

Datasets. We conduct an extensive set of experiments to evaluate the proposed method on two VQA datasets: The GQA [13] dataset is a visual reasoning and compositional VQA dataset offering questions and answers about various real-world images. We conduct experiments on its balanced subset that includes 1.7M questions. The VQAv2 [5] dataset is particularly designed to test the generalizability of VQA models, which consists of 1.1M questions, each annotated with 10 ground-truth answers. These two datasets maintain a large size of versatile questions and rich annotations of the scene structure (*e.g.*, ground truth scene graph) and reasoning process (*e.g.*, semantic structure of question).

Scene graph and knowledge graph. We generate initial scene graphs with a VCTree [25] trained on Visual Genome [18]. To generate an external knowledge graph, we extract relations from three knowledge bases: ConceptNet [19], WordNet [21], and Visual Genome [18]. We initialize class nodes based on the nouns in the vocabulary of the training set. From ConceptNet and WordNet, we retrieve the first-order relations and add the corresponding classes and predicates to the knowledge graph. From the Visual Genome dataset, for each subject-object pair, we include the top-3 predicates according to their frequency of occurrence. For both the scene graph and the knowledge graph, the feature dimension is set to $d_h = 300$.

KI-Net training. Our KI-Net is trained on the GQA dataset using its ground-truth scene graphs. The KI-Net parameters are optimized with Adam optimizer [16] at a learning rate of 10^{-4} and a weight decay rate of 10^{-4} . We bridge entity and class nodes with top feature similarity and empirically set the feature similarity threshold $\epsilon_{cls} = 0.7$ following [32]. For the message passing, we set the number of iterations $K_{GAT} = 3$. We set the parameters $K_p = 3$ and $\epsilon_p = 0.8$ to limit the size of the enriched scene graph. Ablation studies

of the hyper-parameters are reported in the Supplementary Materials.

Neural program generation and training. We convert the input question into a program of neural modules following StackNMN [10]. The question is first converted into a sequence of $T = 4$ feature vectors (with dimensionality $d_s = 300$) using a bi-directional LSTM [23]. At each step t , we generate textual parameters q_t and weight parameters w_t using several layers of MLP in a time-dependent manner. The textual parameters are used as queries and the weight parameters are used for soft module selection. We feed the output features of the program into a softmax layer to predict the answer. The neural modules are trained by minimizing the cross-entropy loss of the predicted probabilities for the top 3000 answers. We use the Adam optimizer with a learning rate of 10^{-4} and a decay rate of 10^{-4} . The training process is approximately 20 epochs with early stopping based on validation accuracy. We set the max path length $L = 3$ to balance computational complexity and performance. Ablation studies of the hyper-parameters are reported in the Supplementary Materials.

Model evaluation and comparison. We compare our method with state-of-the-art neural module methods. While XNM [24] and NSM [12] are graph-based explicit reasoning models, StackNMN [10] and N2NMN [11] are based on image features. For a fair comparison, all compared models are trained and evaluated under the same settings, except that N2NMN requires ground truth layout policies to supervise the generation of neural programs.

4.2. Model Performance

Comparison with the state of the art. As shown in Tab. 2, our method achieves a 64.21% overall accuracy on the GQA test-dev dataset [13] and a 67.32% overall accuracy on the VQAv2 validation dataset [5], outperforming the state-of-the-art neural module models on both datasets. Our method also ranks the top regarding answer consistency, validity, and plausibility, while achieving the second-best distribution score. Among the compared models, NSM performs the second best thanks to its specifically designed state transition function that can be trained end-to-end to represent all possible neural modules. This end-to-end learning of neural modules improves the model performance at the expense of interpretability, as the semantic meaning of the learned neural modules is unclear.

It is noteworthy that our method shows a considerable improvement in the plausibility metric. The plausibility measures whether objects are described with a general level of world-knowledge (*e.g.*, the color of an apple can be red or green, but not blue). The higher plausibility score demonstrates that our method can effectively reason about commonsense knowledge based on the enriched scene graph.

Comparison with the baselines. Tab. 2 compares three

Method	GQA test-dev							VQAv2 val			
	Binary	Open	Consistency	Validity	Plausibility	Distribution	Overall	Yes/No	Number	Other	Overall
N2NMN [11]	74.68	41.33	87.78	96.03	84.15	6.07	56.97	77.54	40.38	56.39	63.28
StackNMN [10]	75.92	43.21	86.41	96.30	84.29	5.69	58.55	79.28	41.06	56.43	64.09
XNM [24]	76.88	43.24	88.24	96.21	84.92	5.81	59.01	79.92	41.16	57.12	64.70
NSM [12]	78.94	49.25	93.25	96.41	84.28	3.71	63.17	79.77	41.75	59.40	65.77
Baseline I	73.97	41.28	85.24	96.17	83.85	6.13	56.61	77.65	41.29	57.82	64.11
Baseline II (G-Relate)	76.22	43.31	88.25	96.12	84.71	5.48	58.74	79.80	40.97	58.73	65.38
Baseline III (KI-Net)	77.79	45.60	89.31	96.21	85.77	5.72	60.69	79.64	42.89	59.71	65.98
Ours	81.02	49.36	93.81	96.84	86.31	4.41	64.21	81.92	43.16	60.47	67.32

Table 2. Quantitative results on the GQA and VQAv2 datasets. The best results are highlighted in bold.

baseline models to evaluate the effectiveness of the proposed KI-Net and G-Relate. Baseline I replaces G-Relate with a basic Relate module following the XNM method [24], and performs reasoning without external knowledge incorporation. Baseline II (G-Relate) only uses G-Relate to infer high-order relations, and Baseline III (KI-Net) only uses KI-Net to incorporate knowledge. The results suggest that KI-Net and G-Relate can independently improve the VQA performance on both datasets. Altogether, they achieve total improvements of 7.6% on GQA and 3.2% on VQAv2, better than the sum of their independent improvements. This observation suggests that G-Relate is more effective on the enriched scene graph structure by resolving its semantic complexity. In particular, by improving the scene graph and attention transfer, KI-Net and G-Relate allow the attention to be more efficiently and accurately allocated to the correct nodes. As a result, our method significantly improves the accuracy of answers to attention-sensitive questions (*e.g.*, yes/no questions). For further analyses of the attention distribution, please refer to the Supplementary Materials.

Qualitative results. Fig. 5 presents qualitative examples and key relations incorporated from external knowledge that

help the reasoning model to predict the correct answer. As shown in the examples, our method predicts more accurate answers than the state-of-the-art methods. With the help of multi-source external knowledge, our method is more generalizable to questions with out-of-domain knowledge and answers more specifically and correctly to open questions (see Fig. 5a) and binary questions (see Fig. 5b-d). The performance improvement comes from the explicitly incorporated entities and predicates that allow the reasoning process to attend to these nodes and infer the correct answer. For example, in Fig. 5a-b, our method answers correctly because it can explicitly allocate attention to the incorporated entities (*i.e.*, bedroom and cheese). These examples also demonstrate the importance of reasoning with high-order relations. Since the entities (*i.e.*, pole, propeller, and aircraft, see Fig. 5c-d) are added from the external knowledge, it is not possible to directly generate multiple first-order Relate modules from the questions. Instead, our G-Relate can propagate attention directly along the high-order relation paths (*e.g.*, man-in-ski-requiring-pole in Fig. 5c and aircraft-has-airplane-has-propellers in Fig. 5d). These examples demonstrate the collaboration between KI-Net and G-Relate that improves the overall reasoning performance





	(a)	(b)	(c)	(d)
Image:				
Question:	Which room is it?	Are there both cheese and salad?	Is the man holding ski poles?	Are there any propellers on the aircraft?
Answers:	GT: Bedroom Ours: Bedroom XNM: Indoors StackNMN: Indoors NSM: Indoors N2NMN: Indoors	GT: Yes Ours: Yes XNM: No StackNMN: No NSM: No N2NMN: No	GT: Yes Ours: Yes XNM: No StackNMN: No NSM: No N2NMN: No	GT: Yes Ours: Yes XNM: No StackNMN: No NSM: No N2NMN: No
Knowledge:	bed-locationOf-bedroom	pizza-has-cheese	ski-requiring-pole	airplane-typeOf-aircraft propellers-partOf-airplane
Neural Modules:	Attend[room], Describe[name]	Attend[cheese], Attend[salad], And, Exist	Attend[man], G-Relate[hold], Attend[pole], Exist	Attend[aircraft], G-Relate[on], Attend[propeller], Exist

Figure 5. Qualitative examples of our method with the incorporated knowledge and the generated neural modules. Highlighted entities and predicates are incorporated from external knowledge.

Method	mR@50	mR@100	R@50	R@100
GB-Net [32]	6.1	6.9	25.5	29.8
KI-Net	6.2	7.3	25.7	30.6
Improvement	0.1	0.4	0.2	0.8

Table 3. Comparison between KI-Net and GB-Net on the VQAv2 validation set.

and model explainability.

4.3. Evaluation of Scene Graphs

To further demonstrate the effectiveness of KI-Net, we evaluate the enriched scene graphs on the VQAv2 dataset. We measure the quality of scene graphs with the Recall (R@50, R@100) and mean Recall (mR@50, mR@100) following the common practice [6]. The R@K measures how many ground-truth relations are hit in the top K predictions, and mR@K balances the uneven distribution of relations by measuring the average R@K across all relations.

Effectiveness of explicit knowledge incorporation. We compare KI-Net with GB-Net [32], a state-of-the-art scene graph generation model that implicitly distills semantic features from the external knowledge graph without adding graph nodes. In this experiment, both GB-Net and KI-Net are based on the same initial scene graph and external knowledge graph. Tab. 3 shows that the explicit incorporation of relevant entities and predicates allows KI-Net to generate better scene graphs on all metrics. Its performance gain over the GB-Net is more significant on the R@100 and mR100 metrics. This suggests that the less confident predictions of the original scene graph benefit the most from the KI-Net, due to the incorporated external relations.

Comparison of knowledge bases. To demonstrate the ability of KI-Net on the inclusion of multiple knowledge sources for the generation of enriched scene graphs, we compare the effectiveness of WordNet [21], ConceptNet [19], Visual Genome [18] or a combination of all three. Tab. 4 shows that the KI-Net can significantly improve the quality of the scene graph even with only one external knowledge base. With a combination of all three, KI-Net achieves the highest accuracy in scan-path generation, despite the semantic similarity of the knowledge bases and the training of KI-Net on the VQAv2.0 dataset.

4.4. Generalization across neural module networks

To validate the generalizability of the proposed KI-Net, we apply state-of-the-art graph-based neural module networks (*i.e.*, NSM [12] and XNM [24]) to the same scene graphs and compare their performances before and after knowledge incorporation. Specifically, since the NSM and XNM methods are based on different scene graph structures, for a fair comparison, we customize them to run on the same initial scene graph (*i.e.*, VCTree [25]) and the en-

Knowledge Base	mR@50	mR@100	R@50	R@100
None	5.5	6.7	25.3	28.9
WordNet [21]	5.9	7.0	25.4	30.2
ConceptNet [19]	6.1	7.2	25.5	30.1
Visual Genome [18]	6.0	7.2	25.6	30.4
All	6.2	7.3	25.7	30.6

Table 4. KI-Net performances with different knowledge bases on the VQAv2 validation set. The best results are highlighted in bold.

Method	Accuracy		
	w/o KI-Net	w/ KI-Net	Improvement
NSM [12]	63.41	63.68	0.27
XNM [24]	64.89	65.93	1.04
Ours	65.38	67.32	1.94

Table 5. Generalization results of KI-Net to other neural module networks on the VQAv2 validation set.

riched scene graph (*i.e.*, with KI-Net) as ours. As shown in Tab. 5, our KI-Net is generalizable to many neural module methods. It effectively enriches the scene graphs with relevant semantics and improves the accuracy of answers. Due to its more general neural module design, NSM is less sensitive to the quality improvement of scene graphs. Therefore, KI-Net is the least effective working with NSM. Similar to our method, XNM offers a set of explicitly defined neural modules, but it can only transfer attention along first-order relations. Therefore, with XNM, the enriched scene graph has a moderate level of effect on the VQA accuracy. Compared with NSM and XNM, our novel G-Relate module can better leverage the rich semantics in the scene graph and obtain a more significant performance improvement.

5. Conclusion

In this paper, we address the generalizability and explainability of visual reasoning by introducing an explainable and explicit visual reasoning method that emphasizes the explicit integration of external knowledge and high-order relational attention. It consists of a novel Knowledge Incorporation Network (KI-Net) that explicitly incorporates new entities and predicates to enrich the semantics of scene graphs, and a Graph-Relate (G-Relate) module to infer high-order relations. With these novel contributions, it can answer general questions about real-world images with both generalizability and explainability. Our method outperforms the state-of-the-art visual reasoning approaches on the GQA and VQAv2 datasets.

Acknowledgements

This work is supported by NSF Grants 1908711.

References

- [1] Somak Aditya, Yezhou Yang, and Chitta Baral. Explicit reasoning over end-to-end neural architectures for visual question answering. *AAAI*, 2018.
- [2] Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Anirudha Kembhavi. Don’t just assume; look and answer: Overcoming priors for visual question answering. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4971–4980, 2018.
- [3] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6077–6086, 2018.
- [4] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Neural module networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 39–48, 2016.
- [5] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual Question Answering. In *Int. Conf. Comput. Vis.*, 2015.
- [6] Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. Knowledge-embedded routing network for scene graph generation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6163–6171, 2019.
- [7] Shalini Ghosh, Giedrius Burachas, Arijit Ray, and Avi Ziskind. Generating natural language explanations for visual question answering using scene graphs and visual attention. *arXiv preprint arXiv:1902.05715*, 2019.
- [8] Jiuxiang Gu, Shafiq Joty, Jianfei Cai, Handong Zhao, Xu Yang, and Gang Wang. Unpaired image captioning via scene graph alignments. In *Int. Conf. Comput. Vis.*, pages 10323–10332, 2019.
- [9] Jiuxiang Gu, Handong Zhao, Zhe Lin, Sheng Li, Jianfei Cai, and Mingyang Ling. Scene graph generation with external knowledge and image reconstruction. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1969–1978, 2019.
- [10] Ronghang Hu, Jacob Andreas, Trevor Darrell, and Kate Saenko. Explainable neural computation via stack neural module networks. In *Eur. Conf. Comput. Vis.*, pages 53–69, 2018.
- [11] Ronghang Hu, Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Kate Saenko. Learning to reason: End-to-end module networks for visual question answering. In *Int. Conf. Comput. Vis.*, pages 804–813, 2017.
- [12] Drew Hudson and Christopher D Manning. Learning by abstraction: The neural state machine. In *Adv. Neural Inform. Process. Syst.*, pages 5903–5916, 2019.
- [13] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6700–6709, 2019.
- [14] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2901–2910, 2017.
- [15] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Judy Hoffman, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Inferring and executing programs for visual reasoning. In *Int. Conf. Comput. Vis.*, pages 2989–2998, 2017.
- [16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *Int. Conf. Learn. Represent.*, 2014.
- [17] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *Int. Conf. Learn. Represent.*, 2016.
- [18] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *Int. J. Comput. Vis.*, 123(1):32–73, 2017.
- [19] Hugo Liu and Push Singh. Conceptnet—a practical commonsense reasoning tool-kit. *BT technology journal*, 22(4):211–226, 2004.
- [20] David Mascharka, Philip Tran, Ryan Soklaski, and Arjun Majumdar. Transparency by design: Closing the gap between performance and interpretability in visual reasoning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4942–4950, 2018.
- [21] George A Miller. *WordNet: An electronic lexical database*. MIT press, 1998.
- [22] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. pages 1532–1543, 2014.
- [23] Mike Schuster and Kuldeep K Paliwal. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.*, 45(11):2673–2681, 1997.
- [24] Jiaxin Shi, Hanwang Zhang, and Juanzi Li. Explainable and explicit visual reasoning over scene graphs. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8376–8384, 2019.
- [25] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6619–6628, 2019.
- [26] Damien Teney, Lingqiao Liu, and Anton van Den Hengel. Graph-structured representations for visual question answering. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1–9, 2017.
- [27] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *Int. Conf. Learn. Represent.*, 2017.
- [28] Qi Wu, Chunhua Shen, Peng Wang, Anthony Dick, and Anton van den Hengel. Image captioning and visual question answering based on attributes and external knowledge. 40(6):1367–1381, 2017.
- [29] Qi Wu, Peng Wang, Chunhua Shen, Anthony Dick, and Anton Van Den Hengel. Ask me anything: Free-form visual question answering based on knowledge from external sources. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4622–4630, 2016.
- [30] Ning Xu, An-An Liu, Jing Liu, Weizhi Nie, and Yuting Su. Scene graph captioner: Image captioning based on structural visual representation. *J. Vis. Commun.*, 58:477–485, 2019.

- [31] Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. Auto-encoding scene graphs for image captioning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 10685–10694, 2019.
- [32] Alireza Zareian, Svebor Karaman, and Shih-Fu Chang. Bridging knowledge graphs to generate scene graphs. *Eur. Conf. Comput. Vis.*, 2020.
- [33] Yuke Zhu, Joseph J Lim, and Li Fei-Fei. Knowledge acquisition for visual question answering via iterative querying. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1154–1163, 2017.