REASONING-AWARE GRAPH CONVOLUTIONAL NETWORK FOR VISUAL QUESTION ANSWERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Relational reasoning methods based on graph networks are currently state-of-the-art models for Visual Question Answering (VQA) tasks involving real images. Although graph networks are used in these models to enrich visual representations by encoding question-adaptive inter-object relations, these simple graph networks is arguably insufficient to perform visual reasoning for VQA tasks. In this paper, we propose a Reasoning-Aware Graph Convolutional Networks (RA-GCN) that goes one step further towards visual reasoning for GCNs. Our first contribution is the introduction of visual reasoning ability into conventional GCNs. Secondly, we strengthen the expressive power of GCNs via introducing node-sensitive kernel parameters based on edge features to address the limitation of shared transformation matrix for each node in GCNs. Finally, we provide a novel iterative reasoning network architecture for solving VQA task via embedding the RA-GCN module into an iterative process. We evaluate our model on the VQA-CP v2, GQA and Clevr dataset. Our final RA-GCN network successfully achieves state-of-the-art accuracy which is 42.3% on the VQA-CP v2, and highly competitive 62.4% accuracy on the GQA, as well as 90.0% on val split of Clevr dataset.

1 INTRODUCTION

Since the Convolutional Neural Networks (CNNs) have successuflly tackled many classic computer vision problems such as image classification (Krizhevsky et al., 2012) (Simonyan & Zisserman, 2014), object detection (He et al., 2017) (Ren et al., 2015a) and generation (Radford et al., 2015), generalizing CNNs to inputs with graph-like structures is an important topic in the field of deep learning. Graph Neural Networks (GNNs) were introduced in Kipf & Welling (2016) Scarselli et al. (2008) as a common solution to handle arbitrary graph data. Beyond GNNs' outstanding performances when applied to 3D mesh deforamtion (Ranjan et al., 2018), image captioning (Yao et al., 2018), scene understanding (Yang et al., 2018), GNNs have also been successfully used for Visual Question Answering (VQA) task (Li et al., 2019) (Norcliffe-Brown et al., 2018b) (Hu et al., 2019). VQA requires a high level understanding of images and questions and is often considered to be a good proxy for visual reasoning. However, like the CNNs, it is not straightforward to use GNNs in a situation where a high level of reasoning is required. In this paper, we investigate the usage of GNNs for visual reasoning requried by VQA task, which is a core problem of comuter vision tasks in a lot of real-world applications.

How should we build a model based on GNNs to perform reasoning in VQA task? Recently, most state-of-the-art approaches based on graph networks to VQA (Cadene et al., 2019) (Li et al., 2019) (Norcliffe-Brown et al., 2018b) (Hu et al., 2019), are focusing on encoding question-adaptive inter-object relations to enrich visual representations via different graph networks. Figure 1 shows an overview of the framwork of methods mentioned above. Hu et al. (2019) enhances each object features in the scene with a relational contextualized feature collected by multiple iterations of meassage passing between objects. Cadene et al. (2019) provides an iterative reasoning process of a MuRel cell which is designed to encode interactions between question and image objects. Li et al. (2019) adopts a graph attention network to learn multi-type question-adaptive relation representations, in order to enrich visual representations. Norcliffe-Brown et al. (2018b) constructs question-adaptive graph structure and uses a spatial GCN to learn the visual representaions. We stress that the enriched visual features from above methods will then fuse with question embeddings via a multimodal fusion module to produce a joint representation, which is used in the answer prediction. Without the

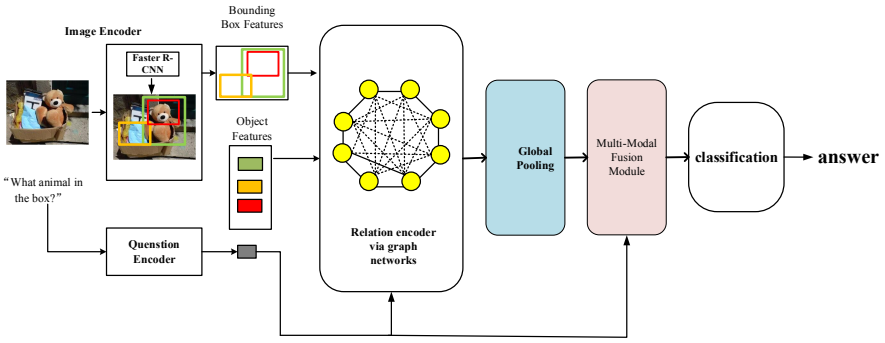

Figure 1: Overview of the framwork of the current Relational reasoning networks based on graph networks.

last multimodal fusion module, those methods will lost huge performance. To some extent, the graph networks adopted by above approaches are mainly responsible for learning question-adaptive inter-object relationships instead of visual reasoning.

In this work, we propose a novel Reasoning-Aware Graph Convolutional Network (RA-GCN) that goes one step further towards visual reasoning for GCNs. Our first contribution is to introduce the RA-GCN module which achieves visual reasoning capability to modulate visual representations based on question information. Our second contribution is to introduce node-sensitive kernel parameters based on edge features to address the limitation of shared transformation matrix for each node in GCNs. By building edge features to capture relationships between nodes and providing node-specific kernel parameters, the expressive power of model visual feature patterns is enhanced. Our last contribution is to provide a novel iterative reasoning network architecture for solving VQA task via embedding the RA-GCN module into an iterative process.

In the experments, we provide various ablative studies to validate our RA-GCN module and the iterative reasoning architecture. By evaluating our model on the VQA-CP v2 (Agrawal et al., 2018), GQA (Hudson & Manning, 2019) and Clevr (Johnson et al., 2017) dataset, our model remarkably achieves state-of-the-art accuracy which is 42.3% on the VQA-CP v2, and highly competitive 62.4% accuracy on the GQA and 90.0% on val split of Clevr dataset.

2 Related Work

2.1 Visual Question Answering

The current dominant framework for VQA systems contains an image encoder, a question encoder, multimodal fusion, and an answer predictor. In,Ren2015FasterRTstead of using CNN-based feature extractors to obtain features, Yang et al. (2015); Fan & Zhou (2018); Ren et al. (2015b); Malinowski et al. (2018) apply various image attention machanism to locate regions that are relevant to the question. Besides, Lu et al. (2016); Nam et al. (2016); Fan & Zhou (2018) carry out question-guided image attention and image-guided question attention to merge knowledge from both visual and textual modalities in the encoding stage, learning a better representation of the question. In addition, some works Li et al. (2018; 2017); Wu et al. (2016) explored high-level semantic information in the image, attributes, for example captions and visual relation facts. These methods applied VQA-independent models to obtain semantic knowledge, while Lu et al. (2018) built a Relation-VQA dataset and directly mined VQA-specific relation facts to feed additional semantic information to the model.

2.2 Graph Convolutional Networks

In the field of deep learning, generalizing CNNs to inputs with graph-like structures is an important topic. The principle of constructing graph CNNs (GCNs) on graph generally follows two streams: the spectral perspective Kipf & Welling (2016)Defferrard et al. (2016) and the spatial perspective

Monti et al. (2017)Velickovic et al. (2018)Boscaini et al. (2016). Our work belongs to the second stream, where the convolution filters are applied diretly on the graph nodes and their neighbors.

Spatial GCNs tend to be more engineered as they require the definition of a node ordering and a convolution filter. Several approaches have been proposed to learn graph structures via learning edge weights based on nodes. Monti et al. Monti et al. (2017) provided a spatial GCNs which learns edge weights based on a mixture of Gaussians of nodes' spatial distance. Graph Attention Networks were proposed in Velickovic et al. (2018), which performs an attention operation on node neighbours to calculate attention weights as edge weights. The question-adaptive edge weights introduced in Norcliffe-Brown et al. (2018a) is the most related work to ours but with following differences. The work Norcliffe-Brown et al. (2018a) computes the similarities of two joint embedding vectors, obtained by performing a non-linear function on the concatenation of question and node feature, as their edge weights, while we use a simple neural network working on edge features to calculate edge weights.

3 Reasoning-Aware Graph Convolutional Networks

Our VQA approach is depicted in Figure 2. Given an image $v \in \mathcal{I}$ and a question $q \in \mathcal{Q}$, we want to predict an answer $\hat{a} \in \mathcal{A}$ that matches the ground truth answer a^*. As common practice in the VQA literature, this can be defined as a classification problem:

$$\hat{a} = \arg\max_{a \in \mathcal{A}} p_\theta(a|v, q) \tag{1}$$

where p_θ is our trained model. In our system, the image is represented by a set of objects $\mathcal{V} = \{v_i\}_{i=1}^K$, where each object v_i is associated with visual feature vector $\mathbf{v}_i \in \mathbb{R}^{d_v}$ and a bounding-box spatial coordinates $\mathbf{b}_i$. Each $\mathbf{b}_i = [x, y, w, h]$ corresponds to 4-dimensional spatial coordinate, where (x, y) denotes the coordinate of the top-left point and (h, w) is the height, width of the bounding box. Note that x and w (respectively y and h) are normalized by the width (resp. height) of the image. For the question, we adopt dynamic RNN with a GRU cell to provide a sentence embedding $\mathbf{q} \in \mathbb{R}^{d_q}$ ($d_q = 1024$ in our experiments) as suggested in Norcliffe-Brown et al. (2018b).

In Section 3.1, we provide the background of GCNs and a feature-wise linear modulation (FiLM) transformation approach (Perez et al., 2017). Next, in Section 3.2, we present the RA-GCN module, a novel graph convolution network that learns to perform visual reasoning operations by blending conditional question information into the set of spatially grounded visual representations. Finally, in Section 3.3, by leveraging the reasoning power of RA-GCN module, a novel architecture is established by iterating through the RA-GCN module to reason about the visual representations with respect to a question.

3.1 Preliminaries

GCN. We will start by a brief recap of the 'vanilla' as proposed in Kipf & Welling (2016). A graph is represented as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ where $\mathcal{V}$ is the set of K nodes and $\mathcal{E}$ are edges, while $\mathbf{x}_i^l \in \mathbb{R}^{D_l}$ and $\mathbf{x}_i^{l+1} \in \mathbb{R}^{D_{l+1}}$ are the feature of node x_i before and after the l-th convolution respectively. Generally, there has two steps to perform a graph convolutional filter on node x_i. First, Node representations are transformed by a learnable parameter matrix $\mathbf{W} \in \mathbb{R}^{D_{l+1} \times D_l}$. Second, node x_i gathers these transformed node representations from its neighbour nodes $j \in \mathcal{N}(i)$, followed by a non-linear function (ReLU). If node representations are collected into a matrix $\mathbf{X}^l \in \mathbb{R}^{D_l \times K}$, the convolutional operation can be written as:

$$\mathbf{X}^{l+1} = \sigma\big(\mathbf{W}\mathbf{X}^l\tilde{\mathbf{A}}\big), \tag{2}$$

where $\tilde{\mathbf{A}}$ is symmetrically normalized from $\mathbf{A}$ in conventional GCNs. $\mathbf{A} \in [0, 1]^{K \times K}$ is the adjacency maxtrix of $\mathcal{G}$, and we have $a_{ij} = 1$ for node $j \in \mathcal{N}(i)$ and $a_{ii} = 1$.

FiLM. Feature-wise linear modulation (FiLM) transformation approach (Perez et al., 2017) is a general-purpose conditioning method for granting neural networks the reasoning ability based on conditioning information. Specifically, it modulate one feature via a simple, feature-wise transformation based on another domain feature. More formally, FiLM learns functions f and h which output $\gamma_{i,c}$ and $\beta_{i,c}$ as a function of condition encoding $\mathbf{x}^i$:

$$\gamma^{i,c} = f_c(\mathbf{x}_i) \qquad \beta_{i,c} = h_c(\mathbf{x}_i), \tag{3}$$

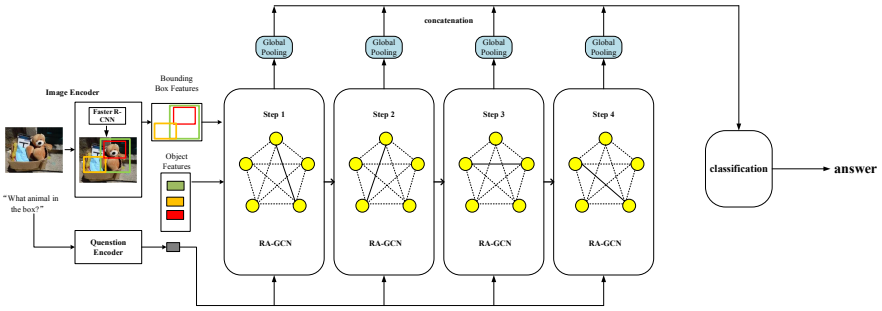

Figure 2: RA-GCN network. The RA-GCN network performs visual reasoning by iterating through several RA-GCN modules and concatenates all the global pooling information from each module as the last vector feature for classification.

where $\gamma_{i,c}$ and $\beta_{i,c}$ modulate features $\mathbf{F}_{i,c}$, whose subscripts refer to the i^{th} input's c^{th} feature, via a feature-wise affine transformation:.

$$FiLM(\mathbf{F}_{i,c}|\gamma_{i,c}, \beta_{i,c}) = \gamma_{i,c}\mathbf{F}_{i,c} + \beta_{k,c}. \tag{4}$$

3.2 RA-GCN MODULE

The RA-GCN module takes as input a set of K visual features $\mathbf{v}_i \in \mathbb{R}^{d_v}$, along with their bounding box coordinates $\mathbf{b}_i$. Before using a GCN filter to learn representation for each object, we need construct a graph (adjacency matrix) of image objects based on those relationships among objects. It is natural that graph convolutions should focus not only on the objects, but also on the object ralationships that are the most relevant to the question. Hence, we first provide a graph learner module to introduce a sparser graph, in which relationships are learned based on question-guided edge features. With the learned question-specific sparse graph, we provide a novel GCN filter which not only learns to capture semantic information between objects but is also able to perform conditional visual reasoning.

Graph learner

The goal of this sub-module is to produce a sparse graphical representations of objects based on a question. Specifically, we learn to define the most k-relevant nodes for each node as its neighbours based on question. This sparsity constraint make sense as most VQA questions requires attending only to a small subset of the graph nodes and it can also reduce the amount of computation. As the interactive dynamics between different objects is very crucial for answering semantically-complicated questions. Here, we try to exploit both geometric and semantic relationships between objects via explictly constructing edge features concated by question-adaptive semantic edge features and geometric edge features. Such edge features allows the storing of more specific information about what precise characteristic of a particular relationship is important in a given question context.

We seek to construct a adjacency matrix $\mathbf{A} \in \mathbb{R}^{K\times K}$ and each a_{ij} resprents the importance of edge e_{ij}. For building semantic feature of each edge $e_{ij} \in \mathcal{E}$, we first fuse the neighbour node features $\mathbf{v}_i$ and $\mathbf{v}_j$ as $Fuse(\mathbf{v}_i, \mathbf{v}_j)$. Then we choose the FiLM method to modulate these fused features based on question feature $\mathbf{q}$ as, $\mathbf{e^s}_{ij} = FiLM(Fuse(\mathbf{v}_i, \mathbf{v}_j)|\mathbf{q})$. The reasoning ability of FiLM layer can ensure that the semantic edge features is question-specific.

Zhuang et al. (2017) has shown that the relative spatial feature between two objects is a strong representation to encode their relationship. Similarly, we model the geometric edge features between node v_i and v_j based on their relative spatial information. Suppose the top-left coordinate, bottom-right coordinate, width and height of node v_i are represented as $[x_{tl_i}, y_{tl_i}, x_{br_i}, y_{br_i}, w_i, h_i]$, then the geomtric edge features are represented as, $\mathbf{e}^g_{ij} = [\frac{x_{tl_i}-x_{tl_j}}{w_i}, \frac{y_{tl_i}-y_{tl_j}}{h_i}, \frac{x_{br_i}-x_{br_j}}{w_i}, \frac{y_{br_i}-y_{br_j}}{h_i}, \frac{w_j}{w_i}, \frac{h_j}{h_i}, \frac{w_j * h_j}{w_i * h_i}, \frac{w_j+h_j}{w_i+h_i}]$. Finally, the edge features are represented as $\mathbf{e}_{ij} = [\mathbf{e^s}_{ij}, \mathbf{e}^g_{ij}]$.

After constructing edge features, we provide a simple discriminative network F^e consisting of two fully-connected layers and a softmax layer to measure its importance $a_{ij} = F^e(\mathbf{e}_{ij})$. Such definition does not impose any constraints on the graph sparsity, and therefore yield a fully connected adjacency matrix. Like the works (Norcliffe-Brown et al., 2018b) (Gao & Ji, 2019), we adapt ranking strategy for each node to learn a sparse heighbourhood system:

$$\mathcal{N}(i) = topm(\mathbf{a}_i), \tag{5}$$

where $topm$ returns the indices of the m largest values of an input vector, and $\mathbf{a}_i$ denotes the i^{th} row of the adjacency matrix. For each row $\mathbf{a}_i$, we choose a non-linear function to normalizes the subset $\{a_{ij}, j \in \mathcal{N}(i)\}$, have $a_{ii} = 1$ and $a_{ij} = 0$ for $j \in \mathcal{N}(i)$. In other words, the neighbourhood system of a given node will correspond to the nodes which it has the strongest connections to.

Conditioned Graph Convolution

Given a question specific graph structure, we then exploit a novel graph convolution filter to learn new object representations that are informed by performing visual reasoning among a neighbouhood system tailored to answer the given question. Except for adding the reasoning ability, we enhance the conventional GCN's representation power by addressing the limitaion of shared transformation matrix for each node.

First, motivated by Perez et al. (2017), we use the FiLM layer to update each node features in the context of question as $FiLM(\mathbf{X}^l|\mathbf{q}) \in \mathbb{R}^{D_{l+1}}$. Although such FiLM transformation is still shared for each node, the question specific way enables the learning of each node's representation to explore the most relevant information with question and fuse question information into node features. Then the Eq. 2 is transformed to:

$$\mathbf{X}^{l+1} = \sigma\big(FiLM(\mathbf{X}^l|\mathbf{q})\tilde{\mathbf{A}}\big). \tag{6}$$

Second, in order to make the graph convolution work on nodes with arbitrary topologies, the learned kernel matrix $\mathbf{W}$ is shared for all nodes. In contrast, CNN learns a different transformation matrix for each position inside the kernel and owns expressive power to model feature patterns. By building on the concept of CNN, we try to adapt node-specific kernels in our graph convolution filter. Su et al. (2019) solves the limitation of spatially shared weights in CNN via multipling filter weights with a spatially varying kernel. The spatially varying kernel is generated by learnable, locally pixel features. Motivated by this, we use the edge features to genearate node-specific kernel parameters, which are spatial sensitive.

Specifically, we use a simple neural network F^w to convert edge features into kernel parameters as $\mathbf{m}_{ij} \in \mathbb{R}^{D_{l+1}}$ and then perform channel-wise multiplication with updated node feature $FiLM(\mathbf{x}_j^l|\mathbf{q})$. The generated kernel parameters not only make the transformation of node features node-adaptive, but also can fully explore semantic interaction as well as graph structures information into node representation. The network F^w consists of two fully-connected layers and a tanh layer which limits the element of weight vector to range $[-1, 1]$ to avoid increasing the scale of output features. All the kernel parameters can be collected into a tensor $\mathbf{P} \in \mathbb{R}^{K\times K\times D_{l+1}}$. Then Eq. 6 is transformed to:

$$\mathbf{X}^{l+1} = \|_{d=1}^{D_{l+1}} \sigma\big(FiLM_d(\mathbf{X}^l|\mathbf{q}) * (\mathbf{P}_d \odot \tilde{\mathbf{A}})\big), \tag{7}$$

where $\|$ represents channel-wise concatenation, $FiLM_d$ outputs the channel d of output features and $\mathbf{M}_d \in \mathbb{R}^{K\times K}$ is the d-th slice across channel in $\mathbf{M}$.

The learnable weighting matrix based on edge features introduced in Zhao et al. (2019) is the most related work to ours but with following sharp difference. Our design is based on the concept of CNN that has espressive power to model feature patterns because the kernal parameters can be both positive and negative. So we use a tanh layer to generate the kernel paramters. However, Zhao et al. (2019) use edge features to genearate a wighting mask which is always positive as they use a sofamax layer. As a result, our model is more similar to CNN and owns better capability to fit the data mapping, showed in Section.

3.3 Network Architecture

Our network architecture mimics a simple form of iterative reasoning by leverage the reasoning power of our RA-GCN module to iteratively adjust visual features based on question information

to answer questions. As we can see in Figure 2, the object features $\{\mathbf{v}_i\}$ are updated by RA-GCN module through multiple steps. More specifically, for each step $t = 1..T$ where T is the total number of steps, a RA-GCN module processes and updates the object features as follows:

$$\{\mathbf{v}_i^t\} = RAGCN(\{\mathbf{v}_i^{t-1}\}; \{\mathbf{b}_i\}, \mathbf{q}) \tag{8}$$

At step $t = T$, the object representations $\{\mathbf{v}_i^t\}$ are aggregated with a global pooling operation to provide a single vector $\mathbf{v} \in \mathbb{R}^{d_v}$. This vector contains information about all the objects, their spatial and semantic relations conditioned on questions for current step T.

Like the CNN architecures, the information about the inputs or gradients can vanish when our architecture becomes increasingly deep. We opt to use dense-wise connections proposed in Huang et al. (2017) to concatenate all the global pooled features from each RA-GCN module as the last vector feature. Without fusing this vector feature with questin information, we directly use a classifier layer consisting of 2-layers MLP with ReLU actiations to predict the answer.

We stress that our model relies solely on RA-GCN module to use question information to modulate the object features. This method distinguishes itself from previous methods (Cadene et al., 2019) (Hu et al., 2019) (Li et al., 2019) Norcliffe-Brown et al. (2018b) which fuse object features and question information into a single embedding via element-wise produce, concatenation, attention, and/or more advanced methods.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets: We evaluate our proposed model on VQA-CP v2 (Agrawal et al., 2018), GQA (Hudson & Manning, 2019) and Clevr (Johnson et al., 2017) datasets for question-conditioned visual reasoning. Although VQA 2.0 (Goyal et al., 2017) dataset is the most used dataset, it still contains some kinds of language bias. Here we adapt VQA-CP v2 dataset, a derivation of the VQA 2.0 (Goyal et al., 2017) dataset, which was introduced to evaluate and reduce the question-oriented biases in the VQA models. In particular, the distribution of answers differs between training and test splits. This dataset can denonstrate the generalization ability of our model. We provide a fine grained analysis on the test set. Then, we use the GQA dataset to further demonstrate our model's visual reasoning capacity as the VQA dataset pays less attetion to reasoning, since 19.5% of its questions have relations, 8% have spatial reasoning questions and only 3% have compoistional questions. while, the GQA dataset makes much effort on generating questions that need multi-step reasoning and balancing the answer distributions to overcome the question-condition biases. Finally, we also use the Clevr dataset to construct a more detailed analysis of our model's performance on very complicated relational questions, such as *what number of other objects are there of the same size as the brown shiny object*. The (Shrestha et al., 2019) appoints that a good VQA model should be capable of datasets across from natural images to synthetic datasets that test reasoning and most methods do not generalize across the two domains.

Hyper-parameters: No matter which dataset we use, we both adapt the same GRU based question encoder provided in Norcliffe-Brown et al. (2018b). For the VQA-CP v2 dataset, we use the Bottom-up features provided by Norcliffe-Brown et al. (2018b) to represent our image as a set of 36 localized regions. We also use the same encoder from Norcliffe-Brown et al. (2018b) to embed the question tokens. For the GQA dataset, we adopt the same preprocess operation to process dataset as the Hu et al. (2019). We use the object detection features of size $N_{det} \times 2048$ (where N_{det} is the number of detected objects in each image with a maximum of 100 per image) and object bounding box coordinates provided by GQA dataset itself. The embedding process of question tokens is same as Hu et al. (2019). For the Clevr dataset, we use the $14 \times 14 \times 1024$ convolutional grid features extracted from the C4 block of an ImageNet-pretrained ResNet-101 network (He et al., 2016) as the local features x^loc (i.e. each x^loc is a 1024-dimensional vector and $K = 196$). We adopt the same embedding process as Perez et al. (2017) to embed question tokens. For all datasets, we both use Adam as optimizer, set batchsize as 64 and learning rate as $3e - 4$.

Table 1: Ablation study of RA-GCN on the impacts of FiLM transformation and node-specific kernel paramerters

FiLM transformation	node-specific parameters	VQA-CP v2	GQA	Clevr
✓	✓	42.3	62.4	90.0
✓	×	40.3	59.3	87.6
×	✓	32.5	43.2	65.6
×	×	31.3	44.7	63.2

Table 2: Number of iterations. Impact of the number of steps in the iterative process on VQA-CP v2 test split.

Model	1 step	2 steps	3 steps	4 steps
RA-GCN	39.8	41.0	42.3	41.6

4.2 Model validation

Ablation study

For all the ablation studies, we set iterative reasoning steps as three since such setting make us achieve the state-of-the-art performance in VQA-CP v2 dataset.

In Table 1, we compare four ablated instances of RA-GCN module. First, we validate the benifits of FiLM transformation. Removing it from RA-GCN module will lead huge loss of its accuray acrossing all the datasets. As our network does not include multimodal fusion module, removing the FiLM transformation will let our model lack question information. It make sense that the performance will decreases a lot while lacking FiLM layer. Second, we validate the benifits of FiLM transformation. When the FiLM transformation layer is added, using the node-specific kernel parameters will leads to higher accuracy on every datasets.

Number of reasoning steps

In Table 2, we perform an analysis ot the iterative process. We train four different RA-GCN networks on the VQA-CP v2 train split, each with a different number of iterations over the RA-GCN module. Performance is reported on test split. Networks with two and three steps respectively provide a gain of $+1.2$ and $+2.5$ in overall accuracy over the network with a single step. An intersting aspect of the iteractive process of RA-GCN is that network with 4 steps reports a decrease in overall accuracy over the network with 3 steps while the amount of parameters increase. The reason behinds this maybe is a GCNs with multiple convolutional layers will suffer from an over-smoothing problem (Luan et al., 2019).

4.3 State of the art comparison

VQA-CP v2 In Table 3, we compare our model to the most recent contributions on the VQA-CP v2 dataset and our RA-GCN model achieves a new remarkably overall state-of-the-art performance. We observe that RA-GCN provides a substantial gain over other methods. Given the different distribution between train and val splits, models that only focus on linguistic biases to answer the question are systematically penalized on their test splits. This property of VQA-CP v2 implies that our RA-GCN method is less prone to question-based overfitting.

Among these methods in the Table 3, LCGN and LCGS models are trained by ourself as they have not been originly evaluated on the VQA-CP v2 dataset. The LCGN method surports realtional reasoning and improves performance across GQA and Clevr datasets. The LCGS approach combines a graph learner and spatial graph convolutions, which is the mostly similar to our method. So we train them using Bottom-up region representations to fully demonstrate our method's effectivness. Interestingly, our model surpasses MUREL, ReGAT and LCGN methods, which correspond to some of the latest development in relational reasoning models for VQA. This tends to indicate our RA-

Table 3: State-of-the-art comparison on the VQA-CP v2 dataset. Results on test split. All these models were trained on the same training set.

Model	Bottem up	Accuracy
LCGS (Norcliffe-Brown et al., 2018b)	✓	39.4
MuRel (Cadene et al., 2019)	✓	39.54
ReGAT (Li et al., 2019)	✓	40.42
LCGN (Hu et al., 2019)	✓	40.21
RA-GCN (ours)	✓	42.3

Table 4: State-of-the-art comparison on the GQA dataset. Results on val split. All these models were trained on the same training set.

Model	object features from detection	Accuracy
CNN+LSTM (Hudson & Manning, 2019)	✓	49.2
MAC (Hudson & Manning, 2018)	×	57.5
LCGN (Hu et al., 2019)	✓	63.8
RA-GCN (ours)	✓	62.4

GCN model has better relational reasoning capability. Moreover, our model greatly improves over LCGS model where the region features are refined with spatial graph convolutions. This show the competitive advantages of our RA-GCN.

GQA In Table 4, we compare our model to the contributions on the GQA dataset. CNN+LSTM and Bottom-Up are simple fusion approaches between the text and the image, using the released GQA object detection features respectively. The MAC model is a multi-step attention and memory model with specially designed control, reading and writing cells, and is trained on the same object detection features as our model. Our approach outperforms the MAC model that performs multi-step inference. This shows our RA-GCN provides much stronger visual reasoning ability on GQA dataset. Finally, even though we did not extensively tune the hyperparameters of our model, our overal score on the val split is highly competitive with state-of-the-art methods.

Clevr One of the core aspect of VQA models lies in their ability to handle different datasets, especially with cross datasets of natural images and synthetic images that many methods do not generalize well. In Table 5, we compare our model to the contributions on the Clevr dataset. Although our model only achieve 90.0 score which is much less than LCGN model's 97.9, there may be two factors in our model affecting our accuray. The first factor is our RA-GCN module is originly designed to handle graph inputs, not the convolutional grid features provided in Clevr dataset. The size of convolutional grid features is $14 * 14 * 1024$, which leads to 196 objects if we treat the grid features as object features. Due to the limitaion of our gpu resources, we can not train our model to handle 196 objects per image at the same time. So we use a convolution filter with stride 2 to downsample the grid features to size $7 * 7$. For above reasons, it is not fair to compare our result with other methods. However, our model still achieves 90.0 scores, meaning our RA-GCN model is able to perform visual reasoning well.

5 Conclusion

In this paper, we introduced RA-GCN, a resoning-aware graph convolutional network for Visual Question Answering task. Our system is based on the concept that FiLM layer can infer neural networks reasoning ability and node-sensitive transformation matrix can augments the expressive power of GCNs. To the best of our knowledge, RA-GCN is the first GCN designed for visual reasoning tasks.

We validated our approach on three challenging datasets: VQA-CP v2, GQA, and Clevr. We exhibted various ablation studies to validate our RA-GCN model. Our final RA-GCN network achieves

Table 5: State-of-the-art comparison on the Clevr dataset. Results on val split. All these models were trained on the same training set.

Model	convolutional grid features	Accuracy
LCGN (Hu et al., 2019)	✓	97.9
MAC (Hudson & Manning, 2018)	✓	98.9
RA-GCN (ours)	✓	90.0

state-of-the-art performance in VQA-CP v2, and is very competitive with state-of-the-art performance on GQA. Additonally, our model is capable of handling both datasets of natural images and synthetic images.

REFERENCES

Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. Don't just assume; look and answer: Overcoming priors for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4971–4980, 2018.

Davide Boscaini, Jonathan Masci, Emanuele Rodolà, and Michael M. Bronstein. Learning shape correspondence with anisotropic convolutional neural networks. In *NIPS*, 2016.

Remi Cadene, Hedi Ben-Younes, Matthieu Cord, and Nicolas Thome. Murel: Multimodal relational reasoning for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1989–1998, 2019.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *NIPS*, 2016.

Haoqi Fan and Jiatong Zhou. Stacked latent attention for multimodal reasoning. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1072–1080, 2018.

Hongyang Gao and Shuiwang Ji. Graph u-nets. *arXiv preprint arXiv:1905.05178*, 2019.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

Ronghang Hu, Anna Rohrbach, Trevor Darrell, and Kate Saenko. Language-conditioned graph networks for relational reasoning. *arXiv preprint arXiv:1905.04405*, 2019.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

Drew A Hudson and Christopher D Manning. Compositional attention networks for machine reasoning. *arXiv preprint arXiv:1803.03067*, 2018.

Drew A Hudson and Christopher D Manning. Gqa: a new dataset for compositional question answering over real-world images. *arXiv preprint arXiv:1902.09506*, 2019.

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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2901–2910, 2017.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105, 2012.

Guohao Li, Hang Su, and Wenwu Zhu. Incorporating external knowledge to answer open-domain visual questions with dynamic memory networks. *ArXiv*, abs/1712.00733, 2017.

Linjie Li, Zhe Gan, Yu Cheng, and Jingjing Liu. Relation-aware graph attention network for visual question answering. *arXiv preprint arXiv:1903.12314*, 2019.

Qing Li, Jianlong Fu, Dongfei Yu, Tao Mei, and Jiebo Luo. Tell-and-answer: Towards explainable visual question answering using attributes and captions. In *EMNLP*, 2018.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *NIPS*, 2016.

Pan Lu, Lei Ji, Wenjun Zhang, Nan Duan, Ming Zhou, and Jianyong Wang. R-vqa: Learning visual relation facts with semantic attention for visual question answering. *ArXiv*, abs/1805.09701, 2018.

Sitao Luan, Mingde Zhao, Xiao-Wen Chang, and Doina Precup. Break the ceiling: Stronger multi-scale deep graph convolutional networks. *arXiv preprint arXiv:1906.02174*, 2019.

Mateusz Malinowski, Carl Doersch, Adam Santoro, and Peter W. Battaglia. Learning visual question answering by bootstrapping hard attention. In *ECCV*, 2018.

Federico Monti, Davide Boscaini, Jonathan Masci, Emanuele Rodolà, Jan Svoboda, and Michael M. Bronstein. Geometric deep learning on graphs and manifolds using mixture model cnns. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5425–5434, 2017.

Hyeonseob Nam, Jung-Woo Ha, and Jeonghee Kim. Dual attention networks for multimodal reasoning and matching. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2156–2164, 2016.

Will Norcliffe-Brown, Efstathios Vafeias, and Sarah Parisot. Learning conditioned graph structures for interpretable visual question answering. In *NeurIPS*, 2018a.

Will Norcliffe-Brown, Stathis Vafeias, and Sarah Parisot. Learning conditioned graph structures for interpretable visual question answering. In *Advances in Neural Information Processing Systems*, pp. 8334–8343, 2018b.

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. *arXiv preprint arXiv:1709.07871*, 2017.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J Black. Generating 3d faces using convolutional mesh autoencoders. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 704–720, 2018.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pp. 91–99, 2015a.

Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015b.

Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2008.

Robik Shrestha, Kushal Kafle, and Christopher Kanan. Answer them all! toward universal visual question answering models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10472–10481, 2019.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Hang Su, Varun Jampani, Deqing Sun, Orazio Gallo, Erik Learned-Miller, and Jan Kautz. Pixel-adaptive convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11166–11175, 2019.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Alejandro Romero, Pietro Lió, and Yoshua Bengio. Graph attention networks. *CoRR*, abs/1710.10903, 2018.

Qi Wu, Chunhua Shen, Peng Wang, Anthony J. Dick, and Anton van den Hengel. Image captioning and visual question answering based on attributes and external knowledge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40:1367–1381, 2016.

Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 670–685, 2018.

Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alexander J. Smola. Stacked attention networks for image question answering. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21–29, 2015.

Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Exploring visual relationship for image captioning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 684–699, 2018.

Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris N Metaxas. Semantic graph convolutional networks for 3d human pose regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3425–3435, 2019.

Bohan Zhuang, Lingqiao Liu, Chunhua Shen, and Ian Reid. Towards context-aware interaction recognition for visual relationship detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 589–598, 2017.