Cascade Reasoning Network for Text-based Visual Question Answering

Fen Liu*†
South China University of Technology
Guangzhou, China
sefliu@mail.scut.edu.cn

Guanghui Xu*†
South China University of Technology
Guangzhou, China
sexuguanghui@mail.scut.edu.cn

Qi Wu‡
University of Adelaide
Adelaide, Australia
qi.wu01@adelaide.edu.au

Qing Du
South China University of Technology
Guangzhou, China
duqing@scut.edu.cn

Wei Jia
CVTE
Guangzhou, China
jiawei@cvte.com

Mingkui Tan‡
South China University of Technology
Guangzhou, China
mingkuitan@scut.edu.cn

ABSTRACT

We study the problem of text-based visual question answering (T-VQA) in this paper. Unlike general visual question answering (VQA) which only builds connections between questions and visual contents, T-VQA requires reading and reasoning over both texts and visual concepts that appear in images. Challenges in T-VQA mainly lie in three aspects: 1) It is difficult to understand the complex logic in questions and extract specific useful information from rich image contents to answer them; 2) The text-related questions are also related to visual concepts, but it is difficult to capture cross-modal relationships between the texts and the visual concepts; 3) If the OCR (optical character recognition) system fails to detect the target text, the training will be very difficult. To address these issues, we propose a novel Cascade Reasoning Network (CRN) that consists of a progressive attention module (PAM) and a multimodal reasoning graph (MRG) module. Specifically, the PAM regards the multimodal information fusion operation as a stepwise encoding process and uses the previous attention results to guide the next fusion process. The MRG aims to explicitly model the connections and interactions between texts and visual concepts. To alleviate the dependence on the OCR system, we introduce an auxiliary task to train the model with accurate supervision signals, thereby enhancing the reasoning ability of the model in question answering. Extensive experiments on three popular T-VQA datasets demonstrate the effectiveness of our method compared with SOTA methods. The source code is available at https://github.com/guanghuixu/CRN_tvqa.

CCS CONCEPTS

• **Information systems** → **Multimedia and multimodal retrieval**; • **Computing methodologies** → **Knowledge representation and reasoning**.

KEYWORDS

Text-based VQA; Multimodal Information; Progressive Attention; Reasoning Graph

ACM Reference Format:
Fen Liu, Guanghui Xu, Qi Wu, Qing Du, Wei Jia, and Mingkui Tan. 2020. Cascade Reasoning Network for Text-based Visual Question Answering. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3394171.3413924

1 INTRODUCTION

Many practical images contain rich text information, such as product descriptions and advertising images. Texts in such images generally convey valuable information and thus are of critical importance in visual understanding tasks such as visual question answering (VQA) [21]. Existing VQA methods [1, 11, 18, 27] tend to directly capture the relationships among visual concepts via complex visual attention mechanisms. However, these methods may lack the ability to read texts in the image and suffer from severe performance degradation when answering text-related visual questions [38].

Compared with common VQA [12, 30], the text-based visual question answering (T-VQA) is more practical as it has the ability to aid visually impaired users in better understanding the information of their surrounding physical world, such as identifying time, temperatures and brand names [16]. In this sense, Singh *et al.* [38] proposed a new dataset TextVQA that mainly contains text-related questions, and they introduced a benchmark method (LoRRA) for text-based VQA (T-VQA). LoRRA is equipped with an external OCR (optical character recognition) model to recognize texts in images. M4C [19] proposed enriching the feature embedding of texts (OCR tokens) and exploring a multimodal transformer layer to fuse visual and text information. Both LoRRA and M4C realize that texts play an important role in answering text-related questions.

*Both authors contributed equally to the paper.
†Also with Pazhou Laboratory, Guangzhou, China.
‡Corresponding author.

MM '20, October 12–16, 2020, Seattle, WA, USA

ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3413924

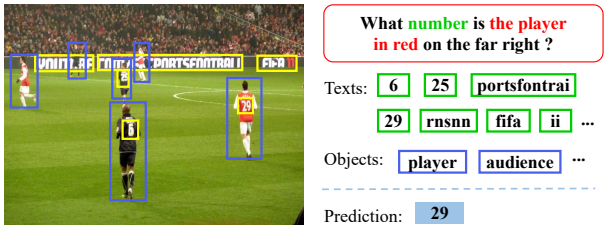


Figure 1: Illustration of T-VQA challenges. To answer the question on the right side, we first need to correctly understand the complex logic in the question and the visual contents (e.g., understand "the player in red" and read the number), which is nontrivial when the questions and/or image contents are complex. For example, the OCR system may fail to recognize the small number in the T-shirt. Moreover, we need to establish a reasoning map to match the texts and visual concepts, which is very challenging.

Nevertheless, several challenges still exist in T-VQA. The rich visual content (including texts and visual concepts) provide indispensable knowledge for answering the question and thus need to be considered carefully. On the one hand, while images contain rich information, texts and visual concepts detected from the image are not always related to the question. As shown in Figure 1, quite a few texts and objects can be detected, but not all are related to the question. It takes considerable effort to understand the complex logic of the question and extract specific useful information from the redundant image content. On the other hand, it is crucial to exploit both recognized texts and visual concepts in an input image for T-VQA. Consider the question - 'What number is the player in red on the far right?'. Once texts and objects are obtained, a correct match between '29' and 'the player in red on the far right' is crucial to 'copy' the exact text as the answer. In other words, only correctly building semantic relationships between texts and visual concepts can enable the model to give the correct answer. Last but not least, the performance of the T-VQA model is highly dependent on the accuracy of OCR systems. That is, even if the model has performed reasoning and found the correct place in the image to 'copy' the text, false detection of the OCR system still provides an incorrect supervision signal during training (*see also* Figure 3).

In this paper, we propose a cascade reasoning network (CRN) to address the above three challenges in the T-VQA task. Our CRN mainly consists of a progressive attention module (PAM) and a multimodal reasoning graph (MRG). It is difficult to identify specific useful information from complex image content at once. In the PAM, we fuse multimodal features in turn and update them in a progressive reasoning manner. To capture cross-modal relationships between the updated features (i.e., text and visual features), we design a heterogeneous graph, where the nodes are the updated features, and the edge is the relative position between every two nodes. In addition, we design a question-guided attention operation to fuse the node representation itself and the cross-modal information. As mentioned above, since the immaturity of the existing OCR detection technology, even if the reasoning is correct, it is difficult for the model to output the same prediction as the ground-truth. In this sense, we further propose an auxiliary task to reduce the model's dependence on the OCR system and improve the model reasoning ability. Specifically, if the prediction is highly similar to the ground truth but not exactly the same, the model still obtains a positive training signal, while existing methods ignore this. Last, we show that all these components improve the model performance on several T-VQA datasets.

In summary, our main contributions are as follows:

(1) We propose a novel cascade approach to fuse multimodal information in turn. The results of previous attention can provide guidance for the next reasoning phase.
(2) To fully exploit multimodal information, we also propose a graph-based reasoning module that models the relationships between texts and visual concepts.
(3) Equipped with the auxiliary task, our model learns a deeper relationship between predictions and ground-truths. Extensive experiments show that our method is superior to several state-of-the-art methods.

2 RELATED WORKS

Visual Question Answering. VQA answers a given question asked about an image. It has attracted increasing interest since it was first introduced by Malinowski *et al.* [29]. To better model the relationship of each image region and question word, most studies [14, 22, 32, 47] employ an attention mechanism to learn attended image features for a given question. Moreover, the co-attention mechanism [28, 50] has been used to learn attention for both the image and the natural language question. A bottom-up attention mechanism [1, 46] is further proposed to focus on objects and other salient image regions. Huang *et al.* [20] propose to model object realtionship using graph convolutional networks.

Text-based VQA. VQA involving reading is gradually attracting more attention because text content conveys rich semantic information and abounds in real scenes, such as billboards, banners and displays. Recently, TextVQA [38] and ST-VQA [6] were introduced to facilitate studies of more general methods and pose difficult challenges for current VQA models. Specifically, TextVQA focused on reasoning about texts in natural everyday scenes, while ST-VQA introduced three tasks of increasing difficulty, differentiated according to degrees of prior knowledge. OCR-VQA [31] is another large-scale dataset for T-VQA, most questions of which are related to book cover information.

To address text-based VQA, Multi-Output Model (MOM) [21] incorporates an OCR sub-network that extracts texts and bounding boxes. Singh *et al.* [38] proposed LoRRA which uses a copy mechanism based on pointer generator networks to copy a word in context as the answer. Inspired by the success of Transformer [43] and BERT [10], M4C [19] treats all entities from each modality homogeneously with a transformer architecture and predicts answers with a pointer-augmented multi-step decoder.

Relational Reasoning and Graph Networks. Relational reasoning has been widely explored in computer vision tasks [15], from region classification [9] to visual question answering [8, 48, 49], and further promoted after graph neural networks appeared [36]. Here we only review related works that use graph networks in VQA and related tasks. Santoro *et al.* [35] propose relation networks (RNs) to

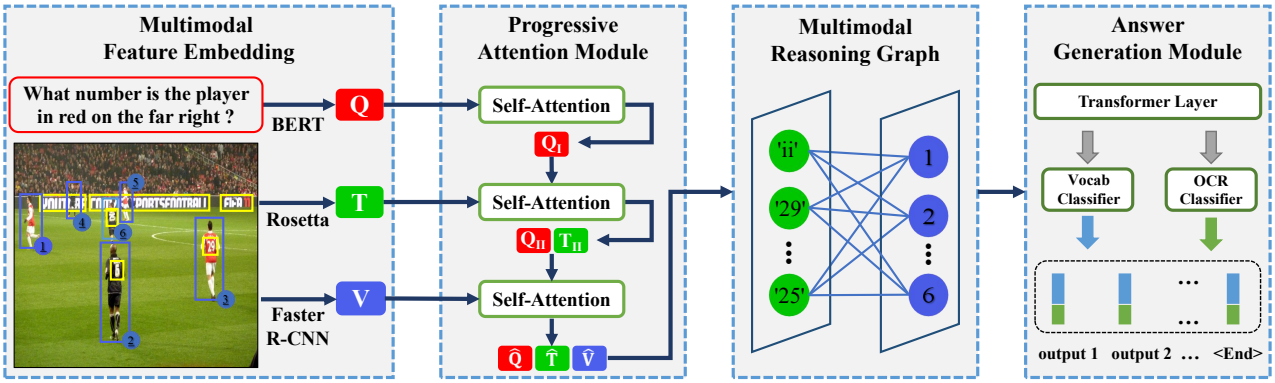


Figure 2: An overview of the cascade reasoning network. Given an input image and question, we use the Rosetta OCR system [7] to recognize and represent texts (T), extract visual features (V) by Faster-RCNN [34], and extract question features (Q) using BERT [10]. To answer the input question, we feed all features (Q, T, V) into the progressive attention module in turn and update informative features gradually. Following that, we use a multimodal reasoning graph to dynamically model cross-modal relationships from the updated features. Last, we apply an answering module to predict answers word by word.

consider relations across all pairs of objects, conditioned on a question. To build contextualized representations for objects in a visual scene, recent studies have introduced graph networks to iteratively pass messages among linked object nodes [9, 18]. Wang *et al.* [44] propose a language-guided graph attention network composed of both node and edge attentions to explicitly represent objects, with intra- and inter-class relationships. Li *et al.* [26] model explicit and implicit inter-object relations via a graph attention mechanism to learn question-adaptive relation representations. Compared with the above multi-modality interaction modeling methods, we exploit relative positions to establish cross-modal relationships.

3 CASCADE REASONING NETWORK

In this paper, we focus on text-based visual question answering (T-VQA), which seeks to answer questions according to the texts and visual concepts in images. To this end, we propose a cascade reasoning network (CRN) to fuse multimodal information, as shown in Figure 2. Specifically, after extracting multimodal features with pre-trained models (*see* Sec. 3.1), the progressive attention module (in Sec. 3.2) prioritizes combining question and text information and then combines corresponding visual concepts for further reasoning. The multimodal reasoning graph (in Sec. 3.3) uses topology to explicitly reflect the interaction of texts and visual concepts. Last, the answering module generates answers word by word in an auto-regressive manner (*see* Sec. 3.4). We present CRN in detail in this section.

3.1 Multimodal Feature Embedding

As shown in Figure 2, understanding the complex logic of question and extracting text and visual features are prerequisites for reasoning. In this section, we formally introduce how to extract multimodal features with the help of the pre-trained models.

Following M4C [19], a question of K words is fed into a pre-trained BERT model [10] to obtain question embedding feature $\mathbf{Q} = \{\mathbf{q}_i\}_{i=1}^{K}$, where $\mathbf{q}_i \in \mathbb{R}^d$ is the embedding of the i-th question word, and d is the feature dimension.

Given an image as input, we first use Faster R-CNN [34] to detect visual objects and use an OCR system [7] to recognize texts (OCR tokens). For visual objects, the detection model extracts N appearance features $\{\mathbf{v}_i^a\}_{i=1}^{N}$ and the corresponding bounding boxes $\{\mathbf{v}_i^b\}_{i=1}^{N}$. Here, we use a 4-d vector to encode the top-left position and bottom-right position of the i-th bounding box, i.e., $\mathbf{v}_i^b = [x_i^{tl}, y_i^{tl}, x_i^{br}, y_i^{br}]$. We first project the above features into d-dimensional space (as $\mathbf{q}_i \in \mathbb{R}^d$) and apply layer normalization [4] to ensure that feature representations of our model are on the same scale. To fuse appearance and location information, we calculate the visual object features $\mathbf{V} = \{\mathbf{v}_i\}_{i=1}^{N}$ as

$$\mathbf{v}_i = \mathbf{W}_1 \mathbf{v}_i^a + \mathbf{W}_2 \mathbf{v}_i^b, \tag{1}$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ are learnable parameters. For texts recognized in the input images, rich representations help answer the text-related questions. Following M4C [19], we use four different types of text features, including 1) FastText feature $\mathbf{t}^f$, which is a pretrained word embedding for the token; 2) PHOC (pyramidal histogram of characters) feature $\mathbf{t}^p$, capturing what characters are present in the token; 3) appearance feature $\mathbf{t}^a$ extracted from Faster R-CNN; and 4) the corresponding bounding box $\mathbf{t}^b$. Based on the rich representations of OCR tokens, we calculate text features $\mathbf{T} = \{\mathbf{t}_i\}_{i=1}^{M}$ by

$$\mathbf{t}_i = \mathbf{W}_3 \mathbf{t}_i^f + \mathbf{W}_4 \mathbf{t}_i^p + \mathbf{W}_5 \mathbf{t}_i^a + \mathbf{W}_6 \mathbf{t}_i^b, \tag{2}$$

where $\mathbf{W}_3 \backsim \mathbf{W}_6$ are learnable parameters, and M is the number of OCR tokens. Thus far, we have completed the multimodal feature embedding and obtained question feature $\mathbf{Q} \in \mathbb{R}^{d \times K}$, visual feature $\mathbf{V} \in \mathbb{R}^{d \times N}$, and text feature $\mathbf{T} \in \mathbb{R}^{d \times M}$ as shown in Figure 2.

3.2 Progressive Attention Module

To fuse multimodal features, we design a progressive attention module (PAM), which is built on transformer layers. Transformer [43] and BERT [10] have achieved great success in natural language processing (NLP) and VQA. To clarify our method more clearly, we briefly review the main idea of the Transformer.

Self-attention mechanism. With the help of a self-attention mechanism, Transformer allows input entities to connect and interact with each other freely. Given a feature vector $\mathbf{f} \in \mathbb{R}^d$, the updating process of the feature vector adopting self-attention mechanism is formulated as

$$\widehat{\mathbf{f}} = \text{softmax}(\frac{(\mathbf{W}_Q\mathbf{f})\ (\mathbf{W}_K\mathbf{f})^\top}{\sqrt{d}})\mathbf{W}_V\mathbf{f}, \tag{3}$$

where $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V$ are learnable parameters of self-attention, $\frac{1}{\sqrt{d}}$ is a scaling factor, and $\widehat{\mathbf{f}} \in \mathbb{R}^d$ is the updated feature. In practice, a standard transformer layer consists of L self-attention layers and employs residual connection [17]. For convenience, we denote the above updating process as $\Psi(\mathbf{f}; \mathbf{W})$, where $\mathbf{W}$ is the learnable parameter of the transformer layer.

As a progressive reasoning method, the PAM first understands the complex logic of questions and then finds the relevant visual contents to answer the questions. Cascading multiple attention layers allows the next attention to take advantage of the previous attention and find more useful information for answering questions. Now, we introduce the three levels (Q-QT-QTV) of PAM in detail.

Question Level. Intuitively, a comprehensive understanding of a question is the first step to answering it. To represent rich information of a given question, we apply a stack of L_1 transformer layers with the parameters $\mathbf{W}_\text{I}$. Specifically, $\mathbf{Q}_\text{I} = \Psi(\mathbf{Q};\ \mathbf{W}_\text{I})$ is obtained by applying self-attention to $\mathbf{Q}$. In this reasoning phase (PAM_1), the model focuses on understanding the complex logic in questions and mining potential information.

Question-Text Level. For T-VQA, most questions are concerned with the texts in images. Since texts provide clear guidance for answering text-related questions, we prioritize fusing the text and question information. In practice, we also apply a stack of L_2 transformer layers over the list of question features $\mathbf{Q}_\text{I}$ and text features $\mathbf{T} = \{\mathbf{t}_i\}_{i=1}^M$. In this phase ($\text{PAM}_2$), the model fuses not only the information of the same entities (e.g., $\mathbf{t}_i$) but also the features of different entities (i.e. $\mathbf{Q}_\text{I}$ and $\mathbf{T}$). PAM_1 uses a self-attention method to find the keywords in the input question, and PAM_2 tries to reason about the answer by matching the keywords in the question with the texts detected in the image (*see also* (d) in Figure 5). We apply the attention operation to update features as

$$[\mathbf{Q}_\text{II}, \mathbf{T}_\text{II}] = \Psi([\mathbf{Q}_\text{I}, \mathbf{T}];\ \mathbf{W}_\text{II}), \tag{4}$$

where $[,]$ is a concatenation operation and $\mathbf{W}_\text{II}$ is the learnable parameter in PAM_2.

Question-Text-Visual Level. Note that visual concepts (objects) also provide a wealth of information to answer text-related questions. However, everything has two sides. Such rich visual information requires our model to mine useful information during the reasoning process and reduce or even eliminate the negative effects of redundant information. In this sense, the attention results of the previous two reasoning phases provide cues for selecting visual features that are more relevant to both the question and the text. Thus, we also apply a stack of L_3 transformer Ψ_III over three entities (questions, visual objects and texts) as follows:

$$[\widehat{\mathbf{Q}},\ \widehat{\mathbf{T}},\ \widehat{\mathbf{V}}] = \Psi([\mathbf{Q}_\text{II}, \mathbf{T}_\text{II}, \mathbf{V}];\ \mathbf{W}_\text{III}), \tag{5}$$

where $[,]$ is a concatenation operation and $\mathbf{W}_\text{III}$ is the parameter of PAM_3. As a result, we obtain the updated features $\widehat{\mathbf{Q}} = \{\widehat{\mathbf{q}}_i\}_{i=1}^K$,$\widehat{\mathbf{T}} = \{\widehat{\mathbf{t}}_i\}_{i=1}^M$, and $\widehat{\mathbf{V}} = \{\widehat{\mathbf{v}}_i\}_{i=1}^N$, where $\widehat{\mathbf{q}}_i, \widehat{\mathbf{t}}_i, \widehat{\mathbf{v}}_i \in \mathbb{R}^d$.

3.3 Multimodal Reasoning Graph

Since both the texts (OCR tokens) and visual concepts (objects) are significant for solving T-VQA, we seek to model the relationships between them to obtain more informative features. Note that, both the texts and objects are visual contents and exist in images as a whole. However, using two different models (object detector and OCR system) to extract features separately results in text features and object features being independent and scattered. To alleviate this issue, we use relative positions to re-establish the relationships between the texts and objects in one image. In this way, the model is able to understand the texts around visual objects.

Graph construction. To capture the relationships between the visual objects and texts recognized in images, we build a directed heterogeneous graph $\mathcal{G}=\{\mathcal{N}, \mathcal{E}\}$, where $\{\mathcal{N}\}$ and $\{\mathcal{E}\}$ denote the node set and edge set, respectively. Note that, our node set consists of two types of nodes: the visual object nodes $\mathcal{N}_v = \{\widehat{\mathbf{v}}_i\}_{i=1}^N$ and the text nodes $\mathcal{N}_t = \{\widehat{\mathbf{t}}_i\}_{i=1}^M$. The edge set $\mathcal{E}$ also includes two kinds of edges: 1) $\mathcal{E}_{v_i \to t_j}$ is the edge from the i-th object to the j-th text; 2) $\mathcal{E}_{t_i \to v_j}$ is the edge from the i-th text to the j-th object. We then model the edge representation by the distance between objects and texts in spatial location. Specifically, we denote the relationship between $\widehat{\mathbf{v}}_i$ and $\widehat{\mathbf{t}}_j$ as $\boldsymbol{e}_{v_i \to t_j} = [\frac{x_j^1 - x_i^c}{w_i}, \frac{y_j^1 - y_i^c}{h_i}, \frac{x_j^2 - x_i^c}{w_i}, \frac{y_j^2 - y_i^c}{h_i}, \frac{w_j * h_j}{w_i * h_i}]$, where $\left[x_i^c, y_i^c, w_i, h_i\right]$ is the center coordinate, width and height of the i-th object, and $\left[x_j^1, y_j^1, x_j^2, y_j^2, w_j, h_j\right]$ is the top-left coordinate, bottom-right coordinate, width and height of the j-th text, respectively. In this way, the MRG is constructed, where $\mathcal{G} = \{(\mathcal{N}_v, \mathcal{N}_t), (\mathcal{E}_{v\to t}, \mathcal{E}_{t\to v})\} = \{(\widehat{\mathbf{V}}, \widehat{\mathbf{T}}), (\{\boldsymbol{e}_{v\to t}\}, \{\boldsymbol{e}_{t\to v}\})\}$.

Question-guided attention. Inspired by the bottom-up attention mechanism [1], we design a question-guided attention operation to fuse multimodal information between the text and visual objects. Given the question feature $\widehat{\mathbf{Q}} = \{\widehat{\mathbf{q}}_i\}_{i=1}^K$ and edge feature $\boldsymbol{e}_{t\to v} \in \mathbb{R}^{5\times M\times N}$, we first transform them into the same embedding space. We then fuse the embedded features to derive the attention transition matrix $\mathbf{A} \in \mathbb{R}^{N\times M}$ as follows:

$$\mathbf{q} = \sum_{k=1}^{K} \text{softmax}(\mathbf{W}_7\widehat{\mathbf{q}}_k) \cdot \widehat{\mathbf{q}}_k, \tag{6}$$

$$\mathbf{A}_{i,:} = \text{softmax}(\mathbf{W}_8\ \boldsymbol{e}_{t\to v_i} + \mathbf{W}_9\ \mathbf{q}), \tag{7}$$

where $\mathbf{W}_7 \sim \mathbf{W}_9$ is learnable parameters, $\mathbf{A}_{i,:} \in \mathbb{R}^M$ denotes the correlation between the i-th visual object and all text. Taking the updating of the visual nodes $\mathbf{v}_i$ as an example, we obtain the updated feature by following operations:

$$\begin{aligned}
\widehat{\boldsymbol{e}}_{t\to v_i} &= \sum_{j=1}^{M} \mathbf{A}_{i,j}\ \mathbf{W}_{10}\ \boldsymbol{e}_{t_j\to v_i}, \\
\mathbf{n}_{t\to v_i} &= \mathbf{W}_{11}\ \widehat{\mathbf{T}}\ \mathbf{A}_i, \\
\check{\mathbf{v}}_i &= \mathbf{W}_{12}\ [\widehat{\mathbf{v}}_i, \widehat{\boldsymbol{e}}_{t\to v_i}, \mathbf{n}_{t\to v_i}],
\end{aligned} \tag{8}$$

where $\mathbf{W}_{10} \sim \mathbf{W}_{12}$ are learnable parameters, $\widehat{\boldsymbol{e}}_{t\to v_i} \in \mathbb{R}^d$ is the transition edge feature, and $\mathbf{n}_{t\to v_i} \in \mathbb{R}^d$ denotes the transition node feature. By fusing the node itself ($\widehat{\mathbf{v}}_i$) and cross-modal features, we obtain the updated visual node $\check{\mathbf{V}} = \{\check{\mathbf{v}}_i\}_{i=1}^N$. Similarly, we can update the text node $\check{\mathbf{T}} = \{\check{\mathbf{t}}_i\}_{i=1}^M$. In practice, we do not observe a significant improvement by stacking more layers in MRG; while the parameters and computational cost shall increase significantly. Therefore, we use a single layer MRG in this paper.

3.4 Answer Generation Module

The candidate answer set consists of two parts: 1) The fixed answer set consists of C words that appear frequently in the training set; 2) The dynamic answer set consists of M OCR tokens detected and recognized from input images. Thus we develop an answer generation module (AGM), which consists of an L_4-layer transformer and two separate sub-classifier (ϕ_1 and ϕ_2), corresponding to the fixed answer set and the dynamic answer set respectively, as shown in Figure 2. In practice, the AGM makes the final prediction word by word for T times, since the answer may contain multiple tokens.

In the sequence-to-sequence task [41], the learned word embedding of previous outputs $\mathbf{O} \in \mathbb{R}^{d\times T}$ and the updated features ($\check{\mathbf{V}}$, $\check{\mathbf{T}}$) are the inputs of the transformer layer. Here, we use the prefix language modeling (LM) technique [33] to ensure that the input entries only use **previous** predictions, and avoid peeping at subsequent answering processes. Formally, we use a transformer with the parameter $\mathbf{W}_{\text{IV}}$ to obtain updated features and especially the predicted output $\widetilde{\mathbf{O}} \in \mathbb{R}^{d\times T}$ by

$$[\widetilde{\mathbf{V}}, \widetilde{\mathbf{T}}, \widetilde{\mathbf{O}}] = \Psi([\check{\mathbf{V}},\ \check{\mathbf{T}},\ \text{LM}(\mathbf{O})]; \mathbf{W}_{\text{IV}}). \tag{9}$$

At each step t, we obtain the prediction scores $\widetilde{\mathbf{y}}_t \in \mathbb{R}^{C+M}$ by concatenating the prediction scores of the fixed answer set and dynamic answer set:

$$\widetilde{\mathbf{y}}_t = [\phi_1(\widetilde{\mathbf{O}}_t); \phi_2(\widetilde{\mathbf{T}}, \widetilde{\mathbf{O}}_t)], \tag{10}$$

where $[;]$ is a concatenation operation. Specifically, given the current output embedding $\widetilde{\mathbf{O}}_t$, the classifier ϕ_1 is a two-layer feed-forward network to predict the scores of words in the fixed answer set. For the dynamic answer set, the OCR tokens recognized in each image may be different, so it cannot be simply regarded as a classification task (using a fixed classifier such as ϕ_1). More intuitively, since $\widetilde{\mathbf{O}}_t$ is the "target" answer embedding (predicted by our model) at the current step, we use ϕ_2 to dynamically calculate the similarity between $\widetilde{\mathbf{O}}_z$ and the text (OCR token) embeddings $\widetilde{\mathbf{T}}$. By cascading the output of the two classifiers at step t, we apply the argmax operation to find the word with the highest score as the predicted answer and obtain its corresponding word embedding, which constitutes $\mathbf{O}_t$ for the next prediction.

Specifically, the first token $\mathbf{O}_0$ is the word embedding of `<Begin>` and the answering process will stop if the token `<End>` is predicted. Following [3, 24] in the sequence translation task, we use the ground-truth as the previous output embedding $\mathbf{O}$ during training. In the inference process, $\mathbf{O}$ is initialized with `<Begin>` and `<PAD>`, and our model predicts answers T times autoregressively. See also the Algorithms in the supplementary section for details of the complete training and inference process.

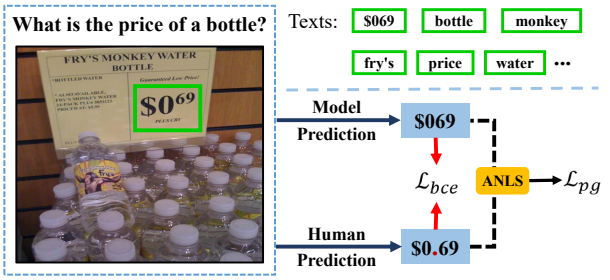


Figure 3: More discussion on the T-VQA challenge. Since the decimal point does not appear in the input image, it is difficult for the model to generate the target answer as human. However, existing methods only use a classification loss (e.g., $\mathcal{L}_{bce}$) to train the model, and thus cannot handle such an issue. To address this, we propose to directly improve the ANLS (metric) score and achieve it via a policy gradient loss $\mathcal{L}_{pg}$.

3.5 Training Loss

Following previous works [19, 38], we use binary cross-entropy loss to train our model. To ease the model's dependence on the OCR system, we introduce an auxiliary task in which if the model's prediction is highly similar to the target, the model can obtain an auxiliary signal. Specifically, we use the reinforcement learning technology to enhance the reasoning ability of our model. In this way, the model learns not only the semantic information but also the character composition of the predicted answer. Formally, we define the overall training loss $\mathcal{L}$ as follows:

$$\mathcal{L} = \mathcal{L}_{bce} + \alpha\, \mathcal{L}_{pg}, \tag{11}$$

where $\mathcal{L}_{bce}$ and $\mathcal{L}_{pg}$ are the binary cross-entropy loss and the policy gradient loss, and α is a trade-off parameter.

Binary cross-entropy loss. Since an answer can exist on both fixed and dynamic answer sets, it can be converted into a multi-label classification problem. Thus we use a binary cross-entropy loss function ($\mathcal{L}_{bce}$), which is more suitable for multi-label classification. Formally, given the prediction $\widetilde{\mathbf{Y}} = \{\widetilde{\mathbf{y}}_t\}_{t=1}^T$, the loss $\mathcal{L}_{bce}$ can be defined as

$$\mathcal{L}_{bce} = -\sum_{t=1}^{T} \mathbf{y}_t^\top \log(\sigma(\widetilde{\mathbf{y}}_t)), \tag{12}$$

where σ is a sigmoid function and $\mathbf{Y} = \{\mathbf{y}_t\}_{t=1}^T$ is the ground-truth.

Policy gradient loss. For the T-VQA task, the performance of a T-VQA model depends highly on the OCR system. As shown in Figure 3, since the false detection of target texts by the OCR system, it is difficult for the model to obtain the correct answer like a human. Note that, some predicted answers are highly similar to the ground-truth, it means that the T-VQA model has achieved the correct reasoning and found the right place to 'copy' the target text. Thus, those predictions cannot simply be regarded as negative samples. However, previous works do not consider this issue, thereby leading to incorrect training feedback for the model. It is necessary to distinguish the reason why the model cannot obtain the correct answer, whether the reasoning ability of the model is insufficient,

or the model is limited by the existing OCR detection technology. As shown in Figure 3, since there is no corresponding ground-truth in the candidate answer set, regardless of the model's prediction, the model can only obtain negative training feedback.

In this sense, **ANLS**[1] (average normalized Levenshtein similarity), a metric of the T-VQA task, is more suitable for this situation, since it focuses on character comparisons between the predicted answer and the ground truth. Thus, we use the ANLS score as the auxiliary signal to train our model. Formally, given the prediction score $\widetilde{\mathbf{Y}}$ and ground-truth $\mathbf{Y}$, we define a function φ to calculate the ANLS score $S_{\text{ANLS}} = \varphi(\widetilde{\mathbf{Y}}, \mathbf{Y})$.

Since the ANLS calculator $\varphi(\cdot)$ is non-differentiable, we use the policy gradient method [42] to iteratively update the model parameters θ. To this end, we formalize the prediction process to a Markov Decision Process. Specifically, at the t-th prediction step, we take all previous $t-1$ generated tokens as the `state` s_t, and take the current predicted token as the `action` a_t. Here, the action probability is $P(a_t) = \text{softmax}(\widetilde{\mathbf{y}}_t)$ and $a_t^* = \text{argmax}_a(P(a_t))$. For a given prediction sequence $\tau = (s_1, a_1, ..., s_T, a_T)$, we use the ANLS score (S_{ANLS}) as the `reward` $R(\tau)$. Then, the goal of our model is to maximize an expectation reward $\mathbb{E}[R(\tau)]$, represented by solving the following optimization problem:

$$\min_{\theta} -\mathbb{E}_{P(\tau;\theta)}[R(\tau)]. \tag{13}$$

To solve the above optimization problem via gradient descent, following policy gradient methods [37, 45], we obtain the gradients by differentiating the following loss function:

$$\mathcal{L}_{pg} = -\mathbb{E}_{P(\tau;\theta)}[\log P(\tau;\theta)R(\tau)], \tag{14}$$

$$\approx -\frac{1}{B}\sum_{b=1}^{B}\log P(\tau_b;\theta)R(\tau_b), \tag{15}$$

where B is the batch size.

4 EXPERIMENTS

We verify the effectiveness of our method on three popular T-VQA datasets, i.e., TextVQA [38], ST-VQA [6] and OCR-VQA [31]. In the following, we first briefly introduce three datasets and evaluation metrics in Sec. 4.1. We then compare our method with the SOTA methods (e.g., M4C [19]) on three benchmarks in Sec. 4.2. After that, we investigate the effect of each component in Sec. 4.3. Last, we demonstrate our proposed method by providing some visualization results and analysis in Sec. 4.4. More implementation details can be found in the supplementary material.

4.1 Datasets and Evaluation Metrics

TextVQA is a recently proposed dataset for T-VQA that requires reading texts in images to answer questions [38]. Collected from the Open Images v3 dataset [23], TextVQA includes 28,408 images with 45,336 questions, and the average question length is 7.18 words. For each image, there are 1-2 questions and each question has 10 answers.

ST-VQA is a parallel dataset for the T-VQA task, which also requires reading and reasoning about scene text [6]. It comprises 23,038 images and 31,791 question/answer pairs collected from six different datasets to reduce dataset bias. In particular, each image of ST-VQA contains at least 2 text pieces, which increases the difficulty of answering questions. It is worth noting that most questions of the ST-VQA can be unambiguously answered with the text recognized in the image. This means that the performance depends on the accuracy of the OCR system. Moreover, ST-VQA introduces three novel tasks, namely strongly contextualized (Task 1), weakly contextualized (Task 2) and open vocabulary (Task 3), which gradually increase in difficulty.

OCR-VQA is a large-scale dataset for the T-VQA task. It contains 207,572 images with 1,002,146 template questions, which are concerned with the information on book covers, such as titles, author names, genres (types) of books, years and editions. Each question has a single ground-truth answer, and the average answer length is 3.31 words. Compared with the aforementioned T-VQA dataset, the book cover background is messy and the texts may change greatly, including font size, language type and fancy characters.

Evaluation metrics. Following TextVQA [38], we use a common VQA accuracy (**Acc**) [2] as an evaluation metric. In particular, ST-VQA proposes a new evaluation metric, **ANLS** (average normalized Levenshtein similarity) [6], which focuses on character comparisons between the predicted answer and the ground truth answer. Note that if the ANLS score is below the threshold of 0.5, it is truncated to 0 before averaging. We also report ANLS on TextVQA.

Table 1: Comparison on the TextVQA dataset. "LoRRA" is the benchmark on the TextVQA dataset and "M4C" is the SOTA method for the T-VQA task.

Model	Acc (%)		ANLS
	Val	Test	Val
LoRRA [38]	27.40	27.63	0.368
DCD [51]	28.96	31.44	-
MM-GNN [13]	31.44	31.10	-
MSFT VTI [39]	32.92	32.46	-
M4C [19]	39.40	39.01	0.529
Ours	**40.39**	**40.96**	**0.547**

4.2 Overall Results

In this section, we report our experimental results on TextVQA [38], ST-VQA [6] and OCR-VQA [31] with two evaluation metrics, namely, Acc (accuracy) and ANLS. Higher values are better for both metrics.

TextVQA. Quantitative results on TextVQA are shown in Table 1. We compare with three baselines including LoRRA, DCD [51] and M4C, where LoRRA is the benchmark on TextVQA, DCD is the TextVQA 2019 challenge winner, and M4C is the SOTA method for the T-VQA task. Our model outperforms all considered methods both on the validation set and test set. Our method significantly outperforms LoRRA by approximately 13% in Acc, and 0.18 in ANLS. Moreover, our model also significantly outperforms DCD in terms of Acc. Our method surpasses the SOTA method M4C by 1% and 2% in terms of Acc on the validation set and test set, respectively.

ST-VQA. As shown in Table 2, we report the ANLS of our method on three tasks on the ST-VQA dataset, where ANLS is the official evaluation metric for this dataset. Impressively, our model achieves SOTA results for three tasks and is significantly better than other

[1] For more details, see ST-VQA [5] and Levenshtein edit distance [25].

Table 2: Comparison on the ST-VQA dataset. "VTA" is the ST-VQA Challenge[2] winner and "M4C" is the SOTA method for the T-VQA task.

Model	ANLS		
	Task 1	Task 2	Task 3
SAN+STR [5]	0.135	0.135	0.135
MM-GNN [13]	-	0.203	0.282
VTA [5]	0.506	0.279	0.282
M4C [19]	-	-	0.462
SMA [40]	0.508	0.310	0.466
Ours (w/o Policy)	0.554	0.443	0.470
Ours	**0.678**	**0.482**	**0.483**

Table 3: Comparison on the OCR-VQA dataset. "BLOCK" and its variants are the benchmarks on the OCR-VQA dataset. "M4C" is the SOTA method for the T-VQA task.

Model	Acc (%)	
	Val	Test
BLOCK [31]	-	42.0
BLOCK+CNN [31]	-	41.5
BLOCK+CNN+W2V [31]	-	48.3
M4C [19]	63.5	63.9
Ours	**64.09**	**64.48**

methods on task 1 and task 2. Compared with the ST-VQA challenge champion VTA, our model achieves significant improvement on all three tasks. Our model also outperforms M4C[3] on task 3. Compared with the SOTA method SMA, our model is advantageous, especially on task 1 and task2. The results on the ST-VQA tasks are able to reflect the generalization ability of the model more realistically, since the test images are collected from six different image datasets (including ImageNet, VizWiz, and COCO-text).

OCR-VQA. As shown in Table 3, we report the accuracy of our model on the OCR-VQA dataset. Compared with the baseline method BLOCK [31], our model achieves significant improvements, where Acc improves by approximately 16%. Our method also outperforms M4C both on the validation and test sets.

4.3 Ablation Studies

In this section, we conduct ablation experiments to demonstrate the effectiveness of our model on the TextVQA dataset.

As shown in Table 4, we report the ablation results of our method in terms of the two aforementioned metrics. In rows 1-3, we mainly evaluate the PAM, where each row corresponds to a reasoning phase of PAM. The performance of our model will decrease by 0.4% if the **Q** (PAM_1) is not executed. It is worth noting that **QT** (PAM_2) plays a key role in our method. As shown in row 2, skipping the "QT" results in a great performance decrease on the validation and test sets. Indeed, texts recognized in the image are crucial for answering text-related questions. Furthermore, removing the **QTV** (PAM_3) also leads to performance degradation, and the accuracy of the

[2]https://rrc.cvc.uab.es/?ch=11&com=tasks

[3]M4C did not report its results on task 1 and task 2.

Table 4: Ablation study on the TextVQA dataset. The last row is our whole model and "×" denotes without a module or operation of "Ours".

#	PAM			MRG	Policy	Acc (%)		ANLS
	Q	QT	QTV			Val	Test	Val
1	×	√	√	√	√	39.97	40.58	0.541
2	√	×	√	√	√	39.78	40.19	0.544
3	√	√	×	√	√	40.33	40.23	0.542
4	√	√	√	×	√	40.19	39.82	0.537
5	√	√	√	√	×	39.55	38.89	0.535
Ours	√	√	√	√	√	**40.39**	**40.96**	**0.547**

Table 5: Ablation study with respect to different text (OCR token) features.

Text Feature	Acc (%)	ANLS
FastText	36.72	0.493
FastText+bbox	36.72	0.499
FastText+bbox+FRCN	39.62	0.539
FastText+bbox+FRCN+PHOC (**Ours**)	**40.39**	**0.547**

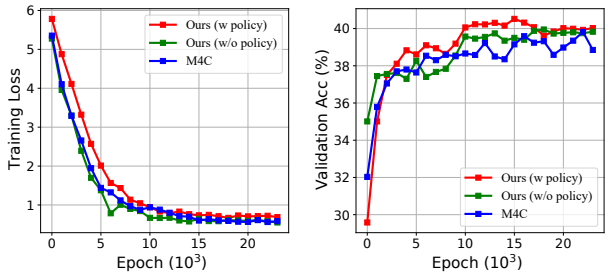


Figure 4: The training loss and validation accuracy under different epochs on the TextVQA dataset.

model in the test set is reduced by 0.7%. Without MRG (multimodal reasoning graph), the performance of our model on the test set decreases by 1%. Impressively, removing policy gradient loss leads to 1-2% performance degradation on the validation set and test set. As shown in the last row, our model achieves the best results. See more ablation studies about the PAM in the supplementary section.

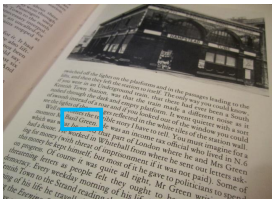
As shown in Table 5, we conduct more ablation experiments for different types of text features, which are mentioned in Sec. 3.1. By introducing "FRCN", the appearance features of the recognized text, our model improves the accuracy by 3% and achieves a higher ANLS score. As shown in (c) and (f) of Figure 5, "FRCN" provides the appearance information of the text (e.g., size and color), which is essential for understanding the text. Moreover, " PHOC " provides the character composition information of the text, which is useful for solving questions related to characters (*see* (d) in Figure 5).

Moreover, we compare our method with M4C [19] in terms of the training loss and validation accuracy. As shown in Figure 4, the training loss of our method tends to converge after 15~20k iterations. For the validation accuracy, our method ("w policy") achieves the best performance. Moreover, our method without the

(a) *What is in red text?*
GT: sport
M4C: allure homme sport
Ours: sport

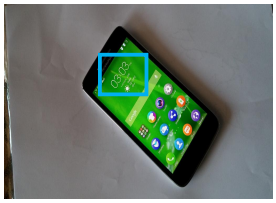
(b) *What volume is on the far right?*
GT: 6
M4C: 5
Ours: 6

(c) *What are the titles of these dvds?*
GT: south park
M4C: south fourth park
Ours: south park



(d) *What was Mr. Green's first name?*
GT: basil
M4C: charles
Ours: basil

(e) *What date is on the lower left poster?*
GT: 2013.10.27
Ours: 2013.10.27(sun)
ANLS: 0.66



(f) *What time is it on this phone?*
GT: 03:03
Ours: 03.03
ANLS: 0.8

(g) *What is the brand of this glass?*
GT: air pacific
Ours: airpacific
ANLS: 0.91

(h) *What brand of beer is this?*
GT: skoj
Ours: koj
ANLS: 0.75

Figure 5: Visualization results on the TextVQA validation set. For better visualization, the blue box in each image shows the texts most relevant to the input question. The prediction results of our model and M4C (the SOTA model for the T-VQA task) are represented in blue and green. In the last row, we report the ANLS score of our prediction. In particular, we use red fonts to identify parts that are inconsistent with the ground-truth.

policy gradient loss ("w/o policy") still outperforms M4C with the help of the PAM and the MRG.

4.4 Visualization Analysis

In this section, we show some visualization examples of our model on the TextVQA dataset in Figure 5. A complete understanding of the texts is helpful for answering text-related questions. Such understanding includes not only semantic information, but also some basic attributes of the text, such as color in (a). For (b), our model has more advantages in answering questions related to location, and M4C is easily misled by the salient text. Note that, the texts recognized in images are redundant and interfere with each other, such as (c) and (d). In this case, our model is more robust than M4C. With the progressive attention module, our model gradually extracts useful information to answer questions.

In the last row of Figure 5, we show some failure cases of the model, aiming to intuitively explain the motivation of the $\mathcal{L}_{pg}$. For (e), the prediction of our model is more detailed than the ground-truth. For (f), the model identifies the number on the phone but mistakes the colon as a comma. In (g), the prediction is one space less than the ground-truth, which is caused by the OCR system detecting the target texts as one word. Due to the angle of the image in (h), it is difficult for the model to see all the letters of the whole brand. The failure of these samples is largely due to the immaturity of the existing detection technology. Even so, our model still reasons correctly and then predicts answers highly similar to the targets.

We also provide more visualization results in the supplementary material.

5 CONCLUSIONS

In this paper, we propose a novel cascade reasoning network to solve the T-VQA task. The CRN mainly consists of a progressive attention module (PAM) and a multimodal reasoning graph (MRG). The PAM uses a progressive approach to gradually fuse multimodal information, while MRG uses a heterogeneous graph to model the rich semantic relationship between the texts and visual objects in the image. We also found that the performance of the model depends on the OCR system. To mitigate this effect, we design an auxiliary task to help the model learn richer feedback information from the failed samples for model training. Our model achieves SOTA performance on three T-VQA benchmark datasets. Quantitative analysis of extensive ablation studies also proves that our model has great potential.

ACKNOWLEDGMENTS

This work was partially supported by Science and Technology Program of Guangzhou, China under Grants 202007030007, Key-Area Research and Development Program of Guangdong Province 2019B010155001, National Natural Science Foundation of China (NSFC) 61836003 (key project), Guangdong 2017ZT07X183, Fundamental Research Funds for the Central Universities D2191240.

REFERENCES

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 6077–6086.

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual Question Answering. In *The IEEE International Conference on Computer Vision (ICCV)*.

[3] Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised Neural Machine Translation. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net. https://openreview.net/forum?id=Sy2ogebAW

[4] Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer Normalization. *CoRR* abs/1607.06450 (2016). arXiv:1607.06450 http://arxiv.org/abs/1607.06450

[5] Ali Furkan Biten, Rubèn Tito, Andres Mafla, Lluis Gomez, Marçal Rusiñol, Minesh Mathew, CV Jawahar, Ernest Valveny, and Dimosthenis Karatzas. 2019. ICDAR 2019 Competition on Scene Text Visual Question Answering. *15th International Conference on Document Analysis and Recognition (ICDAR)* (2019).

[6] Ali Furkan Biten, Ruben Tito, Andres Mafla, Lluis Gomez, Marçal Rusiñol, Ernest Valveny, CV Jawahar, and Dimosthenis Karatzas. 2019. Scene Text Visual Question Answering. *The IEEE International Conference on Computer Vision (ICCV)* (2019).

[7] Fedor Borisyuk, Albert Gordo, and Viswanath Sivakumar. 2018. Rosetta: Large scale system for text detection and recognition in images. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 71–79.

[8] Remi Cadene, Hedi Ben-Younes, Matthieu Cord, and Nicolas Thome. 2019. Murel: Multimodal relational reasoning for visual question answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1989–1998.

[9] Xinlei Chen, Li-Jia Li, Li Fei-Fei, and Abhinav Gupta. 2018. Iterative Visual Reasoning Beyond Convolutions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *The North American Chapter of the Association for Computational Linguistics (NAACL)-HLT*.

[11] Xuanyi Dong, Linchao Zhu, De Zhang, Yi Yang, and Fei Wu. 2018. Fast parameter adaptation for few-shot image captioning and visual question answering. In *Proceedings of the 26th ACM International Conference on Multimedia*. 54–62.

[12] Chuang Gan, Yandong Li, Haoxiang Li, Chen Sun, and Boqing Gong. 2017. Vqs: Linking segmentations to questions and answers for supervised attention in vqa and question-focused semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*. 1811–1820.

[13] Difei Gao, Ke Li, Ruiping Wang, Shiguang Shan, and Xilin Chen. 2020. Multi-Modal Graph Neural Network for Joint Reasoning on Vision and Scene Text. *Computing Research Repository (CoRR)* abs/2003.13962 (2020).

[14] Peng Gao, Zhengkai Jiang, Haoxuan You, Pan Lu, Steven CH Hoi, Xiaogang Wang, and Hongsheng Li. 2019. Dynamic Fusion With Intra-and Inter-Modality Attention Flow for Visual Question Answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 6639–6648.

[15] Yong Guo, Yin Zheng, Mingkui Tan, Qi Chen, Jian Chen, Peilin Zhao, and Junzhou Huang. 2019. NAT: Neural Architecture Transformer for Accurate and Compact Architectures. In *Advances in Neural Information Processing Systems (NeurIPS)*. 735–747.

[16] Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. 2018. Vizwiz grand challenge: Answering visual questions from blind people. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 3608–3617.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 770–778. https://doi.org/10.1109/CVPR.2016.90

[18] Ronghang Hu, Anna Rohrbach, Trevor Darrell, and Kate Saenko. 2019. Language-Conditioned Graph Networks for Relational Reasoning. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2019).

[19] Ronghang Hu, Amanpreet Singh, Trevor Darrell, and Marcus Rohrbach. 2019. Iterative Answer Prediction with Pointer-Augmented Multimodal Transformers for TextVQA. arXiv:1911.06258 [cs.CV]

[20] Deng Huang, Peihao Chen, Runhao Zeng, Qing Du, Mingkui Tan, and Chuang Gan. 2020. Location-Aware Graph Convolutional Networks for Video Question Answering.. In *the Association for the Advance of Artificial Intelligence (AAAI)*. 11021–11028.

[21] Kushal Kafle, Brian Price, Scott Cohen, and Christopher Kanan. 2018. DVQA: Understanding data visualizations via question answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 5648–5656.

[22] Vahid Kazemi and Ali Elqursh. 2017. Show, ask, attend, and answer: A strong baseline for visual question answering. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017).

[23] Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Andreas Veit, et al. 2017. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://github. com/openimages* 2 (2017), 3.

[24] Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Phrase-Based & Neural Unsupervised Machine Translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 5039–5049.

[25] Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, Vol. 10. 707–710.

[26] Linjie Li, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019. Relation-aware Graph Attention Network for Visual Question Answering. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2019).

[27] Fei Liu, Jing Liu, Richang Hong, and Hanqing Lu. 2019. Erasing-based Attention Learning for Visual Question Answering. In *Proceedings of the 27th ACM International Conference on Multimedia*. 1175–1183.

[28] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical question-image co-attention for visual question answering. In *Advances In Neural Information Processing Systems (NeurIPS)*. 289–297.

[29] Mateusz Malinowski and Mario Fritz. 2014. A Multi-World Approach to Question Answering about Real-World Scenes based on Uncertain Input. In *Advances in Neural Information Processing Systems (NeurIPS)*.

[30] Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B. Tenenbaum, and Jiajun Wu. 2019. The Neuro-Symbolic Concept Learner: Interpreting Scenes, Words, and Sentences From Natural Supervision. In *International Conference on Learning Representations (ICLR)*.

[31] Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. 2019. OCR-VQA: Visual question answering by reading text in images.

[32] Liang Peng, Yang Yang, Zheng Wang, Xiao Wu, and Zi Huang. 2019. CRA-Net: Composed Relation Attention Network for Visual Question Answering. In *Proceedings of the 27th ACM International Conference on Multimedia*. 1202–1210.

[33] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683* (2019).

[34] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NeurIPS)*. 91–99.

[35] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. 2017. A simple neural network module for relational reasoning. In *Advances in Neural Information Processing Systems (NeurIPS)*. 4967–4976.

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

[37] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[38] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. 2019. Towards vqa models that can read. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 8317–8326.

[39] Anonymous submission. 2019. MSFT VTI. https://evalai.cloudcv.org/web/challenges/challenge-page/244/.6,7.

[40] Anonymous submission. 2020. Structured Multimodal Attentions for TextVQA. https://rrc.cvc.uab.es/?ch=11&com=evaluation&task=1.

[41] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. *Computing Research Repository (CoRR)* abs/1409.3215 (2014). arXiv:1409.3215 http://arxiv.org/abs/1409.3215

[42] Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. 2000. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems (NeurIPS)*. 1057–1063.

[43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems (NeurIPS)*.

[44] Peng Wang, Qi Wu, Jiewei Cao, Chunhua Shen, Lianli Gao, and Anton van den Hengel. 2019. Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1960–1968.

[45] Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning* 8, 3-4 (1992), 229–256.

[46] Chenfei Wu, Jinlai Liu, Xiaojie Wang, and Xuan Dong. 2018. Object-Difference Attention: A Simple Relational Attention for Visual Question Answering. In *Proceedings of the 26th ACM International Conference on Multimedia*. 519–527.

[47] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. 2016. Stacked Attention Networks for Image Question Answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
[48] Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B Tenenbaum. 2020. Clevrer: Collision events for video representation and reasoning. In *International Conference on Learning Representations (ICLR)*.
[49] Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Josh Tenenbaum. 2018. Neural-symbolic vqa: Disentangling reasoning from vision and language understanding. In *Advances in neural information processing systems (NeurIPS)*. 1031–1042.
[50] Zhou Yu, Jun Yu, Yuhao Cui, Dacheng Tao, and Qi Tian. 2019. Deep Modular Co-Attention Networks for Visual Question Answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 6281–6290.
[51] Yanan Li Donghui Wang Yuetan Lin, Hongrui Zhao. 2018. DCD presentation. https://url.cn/5iQYM5n.