Answer Them All! Toward Universal Visual Question Answering Models

Robik Shrestha[1] Kushal Kafle[1] Christopher Kanan[1,2,3]
[1]Rochester Institute of Technology [2]PAIGE [3]Cornell Tech
{rss9369, kk6055, kanan}@rit.edu

Abstract

Visual Question Answering (VQA) research is split into two camps: the first focuses on VQA datasets that require natural image understanding and the second focuses on synthetic datasets that test reasoning. A good VQA algorithm should be capable of both, but only a few VQA algorithms are tested in this manner. We compare five state-of-the-art VQA algorithms across eight VQA datasets covering both domains. To make the comparison fair, all of the models are standardized as much as possible, e.g.*, they use the same visual features, answer vocabularies, etc. We find that methods do not generalize across the two domains. To address this problem, we propose a new VQA algorithm that rivals or exceeds the state-of-the-art for both domains.*

1. Introduction

Visual Question Answering (VQA) requires a model to understand and reason about visuo-linguistic concepts to answer open-ended questions about images. Correctly answering these questions demands numerous capabilities, including object localization, attribute detection, activity classification, scene understanding, reasoning, counting, and more. The first VQA datasets contained real-world images with crowdsourced questions and answers [36, 9]. It was assumed that this would be an extremely difficult problem and was proposed as a form of Visual Turing Test to benchmark performance in computer vision. However, it became clear that many high performing algorithms were simply exploiting biases and superficial correlations, without really understanding the visual content [24, 3]. For example, answering 'yes' to all yes/no questions in VQAv1 [9] results in an accuracy of 71% on these questions [25]. Later natural image VQA datasets endeavored to address this issue. By associating each question with complementary images and different answers, VQAv2 [16] reduces some forms of language bias. TDIUC [24] analyzes generalization to multiple kinds of questions and rarer answers. CVQA [5] tests concept compositionality and VQACPv2 [4] tests performance when train and test distributions differ.

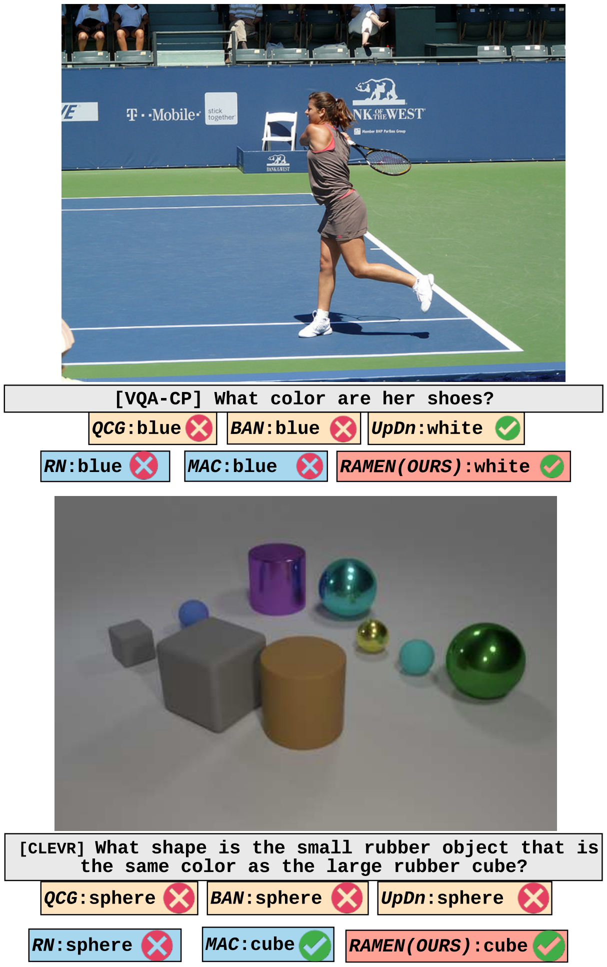

Figure 1: Many VQA algorithms do not transfer well across natural and synthetic datasets. We argue it is necessary to do well on both domains and present an algorithm that achieves this goal.

While later natural image datasets have reduced bias, the vast majority of questions in these datasets do not rigorously test reasoning skills. Several synthetic datasets [20, 7] were created as a remedy. They contain simple visual scenes with

Table 1: Comparison of datasets used in this paper.

Dataset	Num. of Images	Num. of QA Pairs	Question Source	Image Source
VQAv1	204K	614K	Human	Natural
VQAv2	204K	1.1M	Human	Natural
TDIUC	167K	1,6M	Both	Natural
C-VQA	123K	369K	Human	Natural
VQACPv2	219K	603K	Human	Natural
CLEVR	100K	999K	Synthetic	Synthetic
CLEVR-H	32K	32K	Human	Synthetic
CoGenT-A	100K	999K	Synthetic	Synthetic
CoGenT-B	30K	299K	Synthetic	Synthetic

challenging questions that test multi-step reasoning, counting, and logical inference. To properly evaluate an algorithm's robustness, the creators of these datasets have argued algorithms should be tested on both domains [20, 7].

However, almost all recent papers report their performance on only one of these two domains. The best algorithms for CLEVR are not tested on natural image VQA datasets [19, 21, 37, 44, 53], and vice versa [10, 6, 28, 39, 13]. Here, we test five state-of-the-art VQA systems across eight datasets. We found that most methods do not perform well on both domains (Fig. 1), with some suffering drastic losses in performance. We propose a new model that rivals state-of-the-art methods on all of the evaluated datasets.

Our major contributions are:

1. We perform a rigorous comparison of five state-of-the-art algorithms across eight VQA datasets, and we find that many do not generalize across domains.
2. Often VQA algorithms use different visual features and answer vocabularies, making it difficult to assess performance gains. We endeavor to standardize the components used across models, *e.g.*, all of the algorithms we compare use *identical* visual features, which required elevating the methods for synthetic scenes to use region proposals.
3. We find that most VQA algorithms are not capable of understanding real-word images *and* performing compositional reasoning. All of them fare poorly on generalization tests, indicating that these methods are still exploiting dataset biases.
4. We describe a new VQA algorithm that rivals state-of-the-art methods on all datasets and performs best overall.

2. Related Work

2.1. VQA Datasets

Many VQA datasets have been proposed over the past four years. Here, we briefly review the datasets used in our experiments. Statistics for these datasets are given in Table 1. See [25] and [51] for reviews.

VQAv1/VQAv2. VQAv1 [9] is one of the earliest, open-ended VQA datasets collected from human annotators. VQAv1 has multiple kinds of language bias, including some questions being heavily correlated with specific answers. VQAv2 [16] endeavors to mitigate this kind of language bias by collecting complementary images per question that result in different answers, but other kinds of language bias are still present, *e.g.*, reasoning questions are rare compared to detection questions. Both datasets have been widely used and VQAv2 is the de facto benchmark for natural image VQA.

TDIUC [24] attempts to address the bias in the *kinds* of questions posed by annotators by categorizing questions into 12 distinct types, enabling nuanced task-driven evaluation. It has metrics to evaluate generalization across question types.

CVQA [5] is a re-split of VQAv1 to test generalization to concept compositions not seen during training, *e.g.*, if the train set asks about 'green plate' and 'red light,' the test set will ask about 'red plate' and 'green light.' CVQA tests the ability to combine previously seen concepts in unseen ways.

VQACPv2 [4] re-organizes VQAv2 such that answers for each question type are distributed differently in the train and test sets, *e.g.*, 'blue' and 'white' might be the most frequent answers to 'What color...' questions in the train set, but these answers will rarely occur in the test set. Since it has different biases in the train and test sets, doing well on VQACPv2 suggests that the system is generalizing by overcoming the biases in the training set.

CLEVR [20] is a synthetically generated dataset, consisting of visual scenes with simple geometric shapes, designed to test 'compositional language and elementary visual reasoning.' CLEVR's questions often require long chains of complex reasoning. To enable fine-grained evaluation of reasoning abilities, CLEVR's questions are categorized into five tasks: 'querying attribute,' 'comparing attributes,' 'existence,' 'counting,' and 'integer comparison.' Because all of the questions are programmatically generated, the **CLEVR-Humans** [21] dataset was created to provide human-generated questions for CLEVR scenes to test generalization to free-form questions.

CLEVR-CoGenT tests the ability to handle unseen concept composition and remember old concept combinations. It has two splits: CoGenT-A and CoGenT-B, with mutually exclusive shape+color combinations. If models trained on CoGenT-A perform well on CoGenT-B without fine-tuning, it indicates generalization to novel compositions. If models fine-tuned on CoGenT-B still perform well on CoGenT-A, it indicates the ability to remember old concept combinations. The questions in these datasets are more complex than most in CVQA.

Using VQAv1 and VQAv2 alone makes it difficult to gauge whether an algorithm is capable of performing robust compositional reasoning or whether it is using superficial correlations to predict an answer. In part, this is due to the limitations of seeking crowdsourced questions and answers, with humans biased towards asking certain kinds of questions more often for certain images, *e.g.*, counting questions are most often asked if there are two things of the same type in a scene and almost never have an answer of zero. While CVQA and VQACPv2 try to overcome these issues, synthetic datasets [20, 7, 22] minimize such biases to a greater extent, and serve as an important litmus-test to measure *specific* reasoning skills, but the synthetic visual scenes lack complexity and variation.

Natural and synthetic datasets serve complementary purposes, and the creators of synthetic datasets have argued that both should be used, *e.g.*, the creators of SHAPES, an early VQA dataset similar to CLEVR, wrote 'While success on this dataset is by no means a sufficient condition for robust visual QA, we believe it is a necessary one' [7]. While this advice has largely been ignored by the community, we **strongly believe** it is necessary to show that VQA algorithms are capable of tackling VQA in both natural and synthetic domains with little modification. Otherwise, an algorithm's ability to generalize will not be fully assessed.

2.2. VQA Algorithms

Many algorithms for natural image VQA have been proposed, including Bayesian approaches [23, 36], methods using spatial attention [52, 33, 40, 6], compositional approaches [7, 8, 18], bilinear pooling schemes [29, 14], and others [50, 41, 26]. Spatial attention mechanisms [6, 33, 38, 14, 10] are one of the most widely used methods for natural language VQA. Attention computes relevance scores over visual and textual features allowing models to process only relevant information. Among these, we evaluate UpDn [6], QCG [41], and BAN [28]. We describe these algorithms in more detail in Sec. 4.

Similarly, many methods have been created for synthetic VQA datasets. Often, these algorithms place a much greater emphasis on learning compositionality, relational reasoning, and interpretability compared to algorithms for natural images. Common approaches include modular networks, with some using ground-truth programs [21, 37], and others learning compositional rules implicitly [18, 19]. Other approaches have included using relational networks (RNs) [48], early fusion [34], and conditional feature transformations [44]. In our experiments, we evaluate RN [48] and MAC [19], which are explained in more detail in Sec. 4.

Although rare exceptions exist [18], most of these algorithms are evaluated only on natural or synthetic VQA datasets and not both. Furthermore, several algorithms that claim specific abilities are not tested on datasets designed to test these abilities, *e.g.*, QCG [41] claims better compositional performance, but it is not evaluated on CVQA [5]. Here, we evaluate multiple state-of-the-art algorithms on both natural and synthetic VQA datasets, and we propose a new algorithm that works well for both.

3. The RAMEN VQA Model

We propose the Recurrent Aggregation of Multimodal Embeddings Network (RAMEN) model for VQA. It is designed as a conceptually simple architecture that can adapt to the complexity of natural scenes, while also being capable of answering questions requiring complex chains of compositional reasoning, which occur in synthetic datasets like CLEVR. As illustrated in Fig. 2, RAMEN processes visual and question features in three phases:

1. **Early fusion of vision and language features.** Early fusion between visual and language features and/or early modulation of visual features using language has been shown to help with compositional reasoning [34, 44, 12]. Inspired by these approaches, we propose early fusion through concatenation of spatially localized visual features with question features.

2. **Learning bimodal embeddings via shared projections.** The concatenated visual+question features are passed through a shared network, producing spatially localized bimodal embeddings. This phase helps the network learn the inter-relationships between the visual and textual features.

3. **Recurrent aggregation of the learned bimodal embeddings.** We aggregate the bimodal embeddings across the scene using a bi-directional gated recurrent unit (bi-GRU) to capture interactions among the bimodal embeddings. The final forward and backward states essentially need to retain all of the information required to answer the question.

While most recent state-of-the-art VQA models for natural images use attention [6] or bilinear pooling mechanisms [28], RAMEN is able to perform comparably without these mechanisms. Likewise, in contrast to the state-of-the-art models for CLEVR, RAMEN does not use pre-defined modules [37] or reasoning cells [19], yet our experiments demonstrate it is capable of compositional reasoning.

3.1. Formal Model Definition

The input to RAMEN is a question embedding $\boldsymbol{q} \in \mathbb{R}^d$ and a set of N region proposals $\boldsymbol{r}_i \in \mathbb{R}^m$, where each $\boldsymbol{r}_i$ has

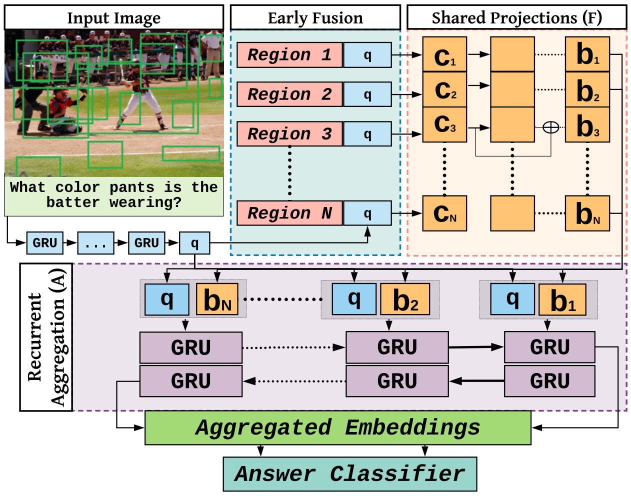

Figure 2: Our recurrent aggregation of multimodal embeddings network (RAMEN).

both visual appearance features and a spatial position. RAMEN first concatenates each proposal with question vector, which is followed by batch normalization, *i.e.*,

$$\boldsymbol{c}_i = BatchNorm\left(\boldsymbol{r}_i \oplus \boldsymbol{q}\right), \tag{1}$$

where $\oplus$ represents concatenation.

All N of the $\boldsymbol{c}_i$ vectors are then passed through a function $F\left(\boldsymbol{c}_i\right)$, which mixes the features to produce a bimodal embedding $\boldsymbol{b}_i = F\left(\boldsymbol{c}_i\right)$, where $F\left(\boldsymbol{c}_i\right)$ was modeled using a multi-layer perceptron (MLP) with residual connections.

Next, we perform late-fusion by concatenating each bimodal embedding with the original question embedding and aggregate the collection using

$$\boldsymbol{a} = A\left(\boldsymbol{b}_1 \oplus \boldsymbol{q}, \boldsymbol{b}_2 \oplus \boldsymbol{q}, \ldots, \boldsymbol{b}_N \oplus \boldsymbol{q}\right), \tag{2}$$

where the function A is modeled using a bi-GRU, with the output of A consisting of the concatenation of the final states of both the forward and backward GRUs. We refer to $\boldsymbol{a}$ as the RAMEN embedding, which is then sent to a classification layer that predicts the answer. While RAMEN is simpler than most recent VQA models, we show it is competitive across datasets, unlike more complex models.

3.2. Implementation Details

Input Representation. We represent question words as 300 dimensional embeddings initialized with pre-trained GloVe vectors [43], and process them with a GRU to obtain a 1024 dimensional question embedding, *i.e.*, $\boldsymbol{q} \in \mathbb{R}^{1024}$. Each region proposal $\boldsymbol{r}_i \in \mathbb{R}^{2560}$ is made of visual features concatenated with spatial information. The visual features are 2048 dimensional CNN features produced by the bottom-up architecture [6] based on Faster R-CNN [47]. Spatial information is encoded by dividing each proposal into a 16×16 grid of (x, y)-coordinates, which is then flattened to form a 512-dimensional vector.

Model Configuration. The projector F is modeled as a 4-layer MLP with 1024 units with swish non-linear activation functions [45]. It has residual connections in layers 2, 3 and 4. The aggregator A is a single-layer bi-GRU that has a 1024 dimensional hidden state, so the concatenation of forward and backward states produces a 2048 dimensional embedding. This embedding is projected through a 2048 dimensional fully connected swish layer, followed by an output classification layer that has one unit per possible answer in the dataset.

Training Details. RAMEN is trained with Adamax [30]. Following [28], we use a gradual learning rate warm up ($2.5 * epoch * 10^{-4}$) for the first 4 epochs, $5 * 10^{-4}$ for epochs 5 to 10, and then decay it at the rate of 0.25 for every 2 epochs, with early stopping used. The mini-batch size is 64.

4. VQA Models Evaluated

In this section, we will briefly describe the models evaluated in our experiments.

Bottom-Up-Attention and Top-Down (UpDn) [6] combines bottom-up and top-down attention mechanisms to perform VQA, with the bottom-up mechanism generating object proposals from Faster R-CNN [47], and the top-down mechanism predicting an attention distribution over those proposals. The top-down attention is task-driven, using questions to predict attention weights over the image regions. This model obtained first place in the 2017 VQA Workshop Challenge. For fair comparison, we use its bottom-up region features for all other VQA models.

Question-Conditioned Graph (QCG) [41] represents images as graphs where object-level features from bottom-up region proposals [6] act as graph nodes and edges that encode interactions between regions that are conditioned on the question. For each node, QC-Graph chooses a neighborhood of nodes with the strongest edge connections, resulting in a question specific graph structure. This structure is processed by a patch operator to perform spatial graph convolution [31]. The main motivation behind choosing this model was to examine the efficacy of the proposed graph representations and operations for compositional reasoning.

Bilinear Attention Network (BAN) [28] fuses visual and textual modalities by considering interactions between all region proposals (visual channels) with all question words (textual channels). Unlike dual-attention mechanisms [38], BAN handles interactions between all channels. It can be considered a generalization of low-rank bi-

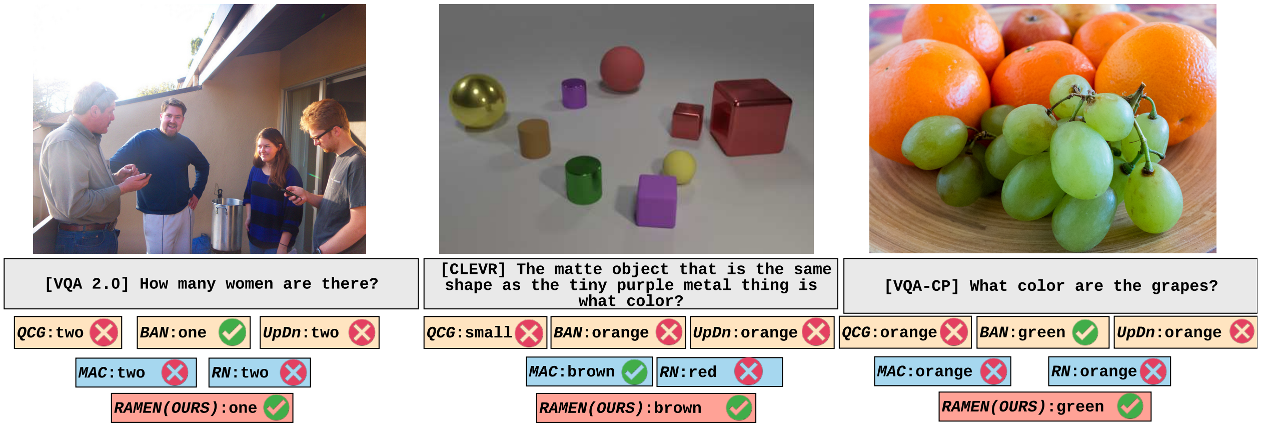

Figure 3: Some example predictions from our model RAMEN compared to other existing methods.

linear pooling methods that jointly represent each channel pair [33, 29]. BAN supports multiple glimpses of attention via connected residual connections. It achieves 70.35% on the test-std split of VQAv2, which is one of the best published results.

Relation Network (RN) [48] takes in every pair of region proposals, embeds them, and sums up all N^2 pair embeddings to produce a vector that encodes relationships between objects. This pairwise feature aggregation mechanism enables compositional reasoning, as demonstrated by its performance on CLEVR dataset. However, RN's computational complexity increases quadratically with the number of objects, making it expensive to run when the number of objects is large. There have been recent attempts at reducing the number of pairwise comparisons by reducing the number of input objects fed to RN [35, 2].

The **Memory, Attention and Composition (MAC)** network [19] uses computational cells that automatically learn to perform attention-based reasoning. Unlike, modular networks [7, 18, 8] that require pre-defined modules to perform pre-specified reasoning functions, MAC learns reasoning mechanisms directly from the data. Each MAC cell maintains a control state representing the reasoning operation and a memory state that is the result of the reasoning operation. It has a computer-like architecture with read, write and control units. MAC was evaluated on the CLEVR datasets and reports significant improvements on the challenging counting and numerical comparison tasks.

4.1. Standardizing Models

Often VQA models achieve state-of-the-art performance using visual features that differ from past models, making it difficult to tell if good performance came from model improvements or improvements to the visual feature representation. To make the comparison across models more meaningful, we use the same visual features for all algorithms across all datasets. Specifically, we use the 2048-dimensional 'bottom-up' CNN features produced by the region proposal generator of a trained Faster R-CNN model [15] with a ResNet-101 backend. Following [49], we keep the number of proposals fixed at 36 for natural images, although performance can increase when additional proposals are used, e.g., others have reported that using 100 proposals with BAN can slightly increase its performance [28]. This Faster R-CNN model is trained for object localization, attribute recognition, and bounding box regression on Visual Genome [32]. While CNN feature maps have been common for CLEVR, state-of-the-art methods for CLEVR have also been shifting toward region proposals [53]. For datasets that use CLEVR's images, we train a separate Faster R-CNN for multi-class classification and bounding box regression, because the Faster R-CNN trained on Visual Genome did not transfer well to CLEVR. To do this, we estimate the bounding boxes using 3D coordinates/rotations specified in the scene annotations. We keep the number of CLEVR regions fixed at 15. We also augment these features with a 512 dimensional vector representing positional information about the boxes as described in Sec. 3.2 for TDIUC, CLEVR, CLEVR-Humans and CLEVR-CoGenT. Following [6], we limit the set of candidate answers to those occurring at least 9 times in the training+validation set, resulting in vocabularies of 2185 answers for VQAv1 and 3129 answers for VQAv2. Following [4, 5], we limit the answer vocabulary to the 1000 most frequent training set answers for CVQA and VQACPv2. For VQAv2, we train the models on training and validation splits and report results on test-dev split. For the remaining datasets, we train the models on their training splits and report performance on validation splits.

Maintaining Compatibility. UpDn, QCG and BAN are all designed to operate on region proposals. For both MAC and RN, we needed to modify the input layers to accept

Table 2: Overall results from six VQA models evaluated using same visual features across all datasets. We highlight the top-3 models for each dataset, using darker colors for better performers. To study the generalization gap, we present the results before fine-tuning for CLEVR-CoGenT and CLEVR-Humans. For VQAv2, we train models on the train and validation splits and report results on test-dev questions. For CLEVR-CoGenT-B, we report results on a sub-split of validation split. For the other datasets, we train models on the train split and report results on validation splits.

Dataset/Algorithm	**UpDn**	**QCG**	**BAN**	**MAC**	**RN**	**Ours**
VQAv1	60.62	59.90	**62.98**	54.08	51.84	61.98
VQAv2	64.55	57.08	**67.39**	54.35	60.96	65.96
TDIUC	68.82	65.57	71.10	66.43	65.06	**72.52**
CVQA	57.01	56.45	57.36	50.99	48.11	**58.92**
VQACPv2	38.01	38.32	**39.31**	31.96	26.70	39.21
CLEVR	80.04	46.73	90.79	**98.00**	95.97	96.92
CLEVR-Humans	54.51	28.12	**60.23**	50.20	57.65	57.87
CLEVR-CoGenT-A	82.47	59.63	92.50	**98.04**	96.45	96.74
CLEVR-CoGenT-B	72.22	53.45	79.48	**90.41**	84.68	89.07
Mean	64.18	51.69	69.00	66.05	65.26	**71.02**

bottom-up features, instead of convolutional feature maps. This was done so that the same features could be used across all datasets and also to upgrade RN and MAC so that they would be competitive on natural image datasets where these features are typically used [6]. For MAC, we replace the initial 2D convolution operation with a linear projection of the bottom-up features. These are fed through MAC's read unit, which is left unmodified. For RN, we remove the initial convolutional network and directly concatenate bottom-up features with question embeddings as the input. The performance of both models after these changes are comparable to the versions using learned convolutional feature maps as input, with MAC achieving 98% and RN achieving 95.97% on the CLEVR validation set.

5. Experiments and Results

5.1. Main Results

In this section, we demonstrate the inability of current VQA algorithms to generalize across natural and synthetic datasets, and show that RAMEN rivals the best performing models on all datasets. We also present a comparative analysis of bias-resistance, compositionality, and generalization abilities for all six algorithms. Table 2 provides our main results for all six algorithms on all eight datasets. We use the standard metrics for all datasets, *i.e.*, we use simple accuracy for the CLEVR family of datasets, mean-per-type for TDIUC, and '10-choose-3' for VQAv1, VQAv2, CVQA, and VQACPv2. Some example outputs for RAMEN compared to other models are given in Fig. 3.

Generalization Across VQA Datasets. RAMEN achieves the highest results on TDIUC and CVQA and is the second best model for VQAv1, VQAv2, VQACPv2 and all of the CLEVR datasets. On average, it has the highest score across datasets, showcasing that it can generalize across natural datasets and synthetic datasets that test reasoning. BAN achieves the next highest mean score. BAN works well for natural image datasets, outperforming other models on VQAv1, VQAv2 and VQACPv2. However, BAN shows limited compositional reasoning ability. Despite being conceptually much simpler than BAN, RAMEN outperforms BAN by 6% (absolute) on CLEVR and 10% on CLEVR-CoGenT-B. RAMEN is within 1.4% of MAC on all compositional reasoning tests. UpDn and QCG perform poorly on CLEVR, with QCG obtaining a score below 50%.

Generalization Across Question Types. We use TDIUC to study generalization across question types. TDIUC has multiple accuracy metrics, with mean-per-type (MPT) and normalized mean-per-type (N-MPT) compensating for biases. As shown in Table 3, all methods achieve simple accuracy scores of over 82%; however, both MPT and N-MPT scores are 13-20% lower. Lower MPT scores indicate that all algorithms are struggling to generalize to multiple tasks. RAMEN obtains the highest MPT score of 72.52% followed by BAN at 71.10%. For all algorithms, 'object presence,' 'object recognition,' and 'scene recognition' are among the easiest tasks, with all of the methods achieving over 84% accuracy on them; however, these tasks all have relatively large amounts of training data (60K - 657K QA pairs each). All of the methods performed well on 'sports recognition' (31K QA pairs), achieving over 93%, but all performed poorly on a conceptually similar task of 'activity recognition' (8.5K QA pairs), achieving under 62% accuracy. This showcases the inability to generalize to question types with fewer examples. To emphasize this, TDIUC pro-

Table 3: Performance comparison on TDIUC using three different metrics. MPT measures task generalization and N-MPT measures generalization to rare answers. We highlight the top-3 models, emboldening the winner.

Metric / Algorithm	UpDn	QCG	BAN	MAC	RN	Ours
MPT	68.82	65.67	71.10	66.43	65.06	**72.52**
N-MPT	38.93	37.43	40.65	39.02	35.75	**46.52**
Simple Accuracy	82.91	82.05	84.81	82.53	84.61	**86.86**

Table 4: Performance on CLEVR's query types.

	Exist	Query Attribute	Compare Attribute	Equal Integer	Greater Than	Less Than	Count
UpDn	83.07	90.08	79.87	65.65	80.43	85.76	64.03
QCG	66.11	31.11	51.47	59.76	69.35	70.57	44.19
BAN	94.72	90.56	98.44	72.35	81.35	86.39	86.47
MAC	99.18	99.59	99.33	85.44	96.82	97.55	95.46
RN	98.40	98.19	97.81	77.30	93.40	84.27	90.90
RAMEN	98.90	98.93	99.30	79.40	93.41	88.53	94.10

vides the Normalized MPT (N-MPT) metric that measures generalization to rare answers by taking answer frequency into account. The differences between normalized and unnormalized scores are large for all models. RAMEN has the smallest gap, indicating a better resistance to answer distribution biases, while BAN has the largest gap.

Generalization to Novel Concept Compositions. We evaluate concept compositionality using CVQA and CLEVR-CoGenT-B. As shown in Table 2, scores on CVQA are lower than VQAv1, suggesting all of the algorithms struggle when combining concepts in new ways. MAC has the largest performance drop, which suggests its reasoning cells were not able to compose real-world visuo-linguistic concepts effectively.

To evaluate the ability to generalize to new concept compositions on the synthetic datasets, we train the models on CLEVR-CoGenT-A's train split and evaluate on the validation set without fine-tuning. Following [44], we obtain a test split from the validation set of 'B,' and report performance without fine-tuning on 'B.' All algorithms show a large drop in performance. Unlike the CVQA results, MAC's drop in performance is smaller. Again, RAMEN has a comparatively small decrease in performance.

Performance on VQACPv2's Changing Priors. All algorithms have a large drop in performance under changing priors. This suggests there is significantly more work to be done to make VQA algorithms overcome linguistic and visual priors so that they can more effectively learn to use generalizable concepts.

Counting and Numerical Comparisons. For CLEVR, counting and number comparison ('equal integer,' 'greater than,' and 'less than') are the most challenging tasks across algorithms as shown in Table 4. MAC performs best on these tasks, followed by RAMEN. Algorithms apart from MAC and QCG demonstrate a large ($> 4.8\%$) discrepancy between 'less than' and 'greater than' question types, which require similar kinds of reasoning. This discrepancy is most pronounced for RN (9.13%), indicating a difficulty in linguistic understanding. BAN uses a counting module [54]; however, its performance on CLEVR's counting task is still 9% below MAC. All of the algorithms struggle with counting in natural images too. Despite TDIUC having over 164K counting questions, all methods achieve a score of under 62% on these questions.

Other CLEVR Tasks. As shown in Table 4, RAMEN is within 0.03-1.5% of MAC's performance on all tasks except number comparison. UpDn and QCG are the worst performing models on all query types. Except for QCG, all of the models find it easy to answer queries about object attributes and existence. Models apart from UpDn and QCG perform well on attribute comparison questions that require comparing these properties. Surprisingly, BAN finds attribute comparison, which requires more reasoning, easier than the simpler attribute query task. We present results on CLEVR-Humans without fine-tuning to examine how well algorithms handle free-form language if they were only trained on CLEVR's vocabulary. BAN shows the best generalization, followed by RAMEN and RN.

Table 5: Ablation studies comparing early versus late fusion between visual and question features, and comparing alternate aggregation strategies.

	VQAv2	CLEVR
Without Early Fusion	61.81	77.48
Without Late Fusion	65.64	96.63
Aggregation via Mean Pooling	63.01	92.45
Without Ablation	65.96	96.92

5.2. Ablation Studies

Results from several ablation studies to test the contributions of RAMEN's components are given in Table 5. We found that early fusion is critical to RAMEN's performance, and removing it causes an almost 20% absolute drop in accuracy for CLEVR and a 4% drop for VQAv2. Removing late fusion has little impact on CLEVR and VQAv2.

We also explored the utility of using a bi-GRU for aggregation compared to using mean pooling, and found that this caused a drop in performance for both datasets. We believe that the recurrent aggregation aids in capturing interactions between the bimodal embeddings, which is critical for reasoning tasks, and that it also helps remove duplicate proposals by performing a form of non-maximal suppression.

5.3. Newer Models

Additional VQA algorithms have been released since we began this project, and some have achieved higher scores than the models we evaluated on *some* datasets. The Transparency By Design (TBD) network [37] obtains 99.10% accuracy on CLEVR by using ground truth functional programs to train the network, which are not available for natural VQA datasets. Neural-Symbolic VQA (NS-VQA) [53] reports a score of 99.80% on CLEVR, but uses a question parser to allocate functional modules along with highly specialized segmentation-based CNN features. They did not perform ablation studies to determine the impact of using these visual features. None of the models we compare have access to these additional resources.

Results on VQAv2 can be significantly improved by using additional data from other VQA datasets and ensembling, *e.g.*, the winner of the 2018 challenge used dialogues from Visual Dialog [11] as additional question answer pairs and an ensemble of 30 models. These augmentations could be applied to any of the models we evaluated to improve performance. VQACPv2 results can also be improved using specialized architectures, *e.g*. GVQA [4] and UpDn with adversarial regularization [46]. However, their performance on VQACPv2 is still poor, with UpDn with adversarial regularization obtaining 42.04% accuracy, showing only 2.98% improvement over the non-regularized model.

6. Discussion: One Model to Rule them All?

We conducted the first systematic study to examine if the VQA systems that work on synthetic datasets generalized to real-world datasets, and vice versa. This was the original scope of our project, but we were alarmed when we discovered none of the methods worked well across datasets. This motivated us to create a new algorithm. Despite being simpler than many algorithms, RAMEN rivals or even surpasses other methods. We believe some state-of-the-art architectures are likely over-engineered to exploit the biases in the domain they were initially tested on, resulting in a deterioration of performance when tested on other datasets. This leads us to question whether the use of highly specialized mechanisms that achieve state-of-the-art results on one specific dataset will lead to significant advances in the field, since our conceptually simpler algorithm performs competitively across both natural and synthetic datasets without such mechanisms.

We advocate for the development of a single VQA model that does well across a wide range of challenges. Training this model in a continual learning paradigm would assess forward and backward transfer [17, 27, 42]. Another interesting avenue is to combine VQA with related tasks like visual query detection [1]. Regardless, existing algorithms, including ours, still have a long way to go toward showcasing both visuo-linguistic concept understanding *and* reasoning. As evidenced by the large performance drops on CVQA and VQACPv2, current algorithms perform poorly at learning compositional concepts and are affected by biases in these datasets, suggesting reliance on superficial correlations. We observed that methods developed solely for synthetic closed-world scenes are often unable to cope with unconstrained natural images and questions. Although performance on VQAv2 and CLEVR are approaching human levels on these benchmarks, our results show VQA is far from solved. We argue that future work should focus on creating one model that works well across domains. It would be interesting to train a dataset on a universal training set and then evaluate it on multiple test sets, with each test set demanding a different skill set. Doing so would help in seeking one VQA model that can rule them all.

7. Conclusion

Our work endeavors to set a new standard for what should be expected from a VQA algorithm: good performance across both natural scenes and challenging synthetic benchmarks. We hope that our work will lead to future advancements in VQA.

Acknowledgements. We thank NVIDIA for the GPU donation. This work was supported in part by a gift from Adobe Research.

References

[1] M. Acharya, K. Jariwala, and C. Kanan. VQD: Visual query detection in natural scenes. In *NAACL*, 2019.

[2] M. Acharya, K. Kafle, and C. Kanan. TallyQA: Answering complex counting questions. In *AAAI*, 2019.

[3] A. Agrawal, D. Batra, and D. Parikh. Analyzing the behavior of visual question answering models. In *EMNLP*, 2016.

[4] A. Agrawal, D. Batra, D. Parikh, and A. Kembhavi. Don't just assume; look and answer: Overcoming priors for visual question answering. In *CVPR*, 2018.

[5] A. Agrawal, A. Kembhavi, D. Batra, and D. Parikh. C-vqa: A compositional split of the visual question answering (vqa) v1.0 dataset. *CoRR*, abs/1704.08243, 2017.

[6] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, 2018.

[7] J. Andreas, M. Rohrbach, T. Darrell, and D. Klein. Deep compositional question answering with neural module networks. In *CVPR*, 2016.

[8] J. Andreas, M. Rohrbach, T. Darrell, and D. Klein. Learning to compose neural networks for question answering. In *NAACL*, 2016.

[9] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. VQA: Visual question answering. In *ICCV*, 2015.

[10] H. Ben-Younes, R. Cadene, M. Cord, and N. Thome. Mutan: Multimodal tucker fusion for visual question answering. In *CVPR*, 2017.

[11] A. Das, S. Kottur, K. Gupta, A. Singh, D. Yadav, J. M. Moura, D. Parikh, and D. Batra. Visual dialog. In *CVPR*, 2017.

[12] H. De Vries, F. Strub, J. Mary, H. Larochelle, O. Pietquin, and A. C. Courville. Modulating early visual processing by language. In *NeurIPS*, 2017.

[13] M. R. Farazi and S. Khan. Reciprocal attention fusion for visual question answering. In *BMVC*, 2018.

[14] A. Fukui, D. H. Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. In *EMNLP*, 2016.

[15] R. Girshick. Fast R-CNN. In *CVPR*, 2015.

[16] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh. Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *CVPR*, 2017.

[17] T. L. Hayes, N. D. Cahill, and C. Kanan. Memory efficient experience replay for streaming learning. In *ICRA*, 2019.

[18] R. Hu, J. Andreas, M. Rohrbach, T. Darrell, and K. Saenko. Learning to reason: End-to-end module networks for visual question answering. In *ICCV*, 2017.

[19] D. A. Hudson and C. D. Manning. Compositional attention networks for machine reasoning. In *ICLR*, 2018.

[20] J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. L. Zitnick, and R. Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*, 2017.

[21] J. Johnson, B. Hariharan, L. van der Maaten, J. Hoffman, L. Fei-Fei, C. L. Zitnick, and R. B. Girshick. Inferring and executing programs for visual reasoning. In *ICCV*.

[22] K. Kafle, S. Cohen, B. Price, and C. Kanan. DVQA: Understanding data visualizations via question answering. In *CVPR*, 2018.

[23] K. Kafle and C. Kanan. Answer-type prediction for visual question answering. In *CVPR*, 2016.

[24] K. Kafle and C. Kanan. An analysis of visual question answering algorithms. In *ICCV*, 2017.

[25] K. Kafle and C. Kanan. Visual question answering: Datasets, algorithms, and future challenges. *Computer Vision and Image Understanding*, 2017.

[26] K. Kafle, M. Yousefhussien, and C. Kanan. Data augmentation for visual question answering. In *INLG*, 2017.

[27] R. Kemker, M. McClure, A. Abitino, T. L. Hayes, and C. Kanan. Measuring catastrophic forgetting in neural networks. In *AAAI*, 2018.

[28] J.-H. Kim, J. Jun, and B.-T. Zhang. Bilinear attention networks. In *NeurIPS*, 2018.

[29] J.-H. Kim, K.-W. On, J. Kim, J.-W. Ha, and B.-T. Zhang. Hadamard product for low-rank bilinear pooling. In *ICLR*, 2017.

[30] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[31] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017.

[32] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. S. Bernstein, and L. Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017.

[33] J. Lu, J. Yang, D. Batra, and D. Parikh. Hierarchical question-image co-attention for visual question answering. In *NeurIPS*, 2016.

[34] M. Malinowski and C. Doersch. The visual QA devil in the details: The impact of early fusion and batch norm on clevr. *arXiv preprint arXiv:1809.04482*, 2018.

[35] M. Malinowski, C. Doersch, A. Santoro, and P. Battaglia. Learning visual question answering by bootstrapping hard attention. In *ECCV*, 2018.

[36] M. Malinowski and M. Fritz. A multi-world approach to question answering about real-world scenes based on uncertain input. In *NeurIPS*, 2014.

[37] D. Mascharka, P. Tran, R. Soklaski, and A. Majumdar. Transparency by design: Closing the gap between performance and interpretability in visual reasoning. In *CVPR*, 2018.

[38] H. Nam, J.-W. Ha, and J. Kim. Dual attention networks for multimodal reasoning and matching. In *CVPR*, 2017.

[39] D.-K. Nguyen and T. Okatani. Improved fusion of visual and language representations by dense symmetric co-attention for visual question answering. In *CVPR*, June 2018.

[40] H. Noh and B. Han. Training recurrent answering units with joint loss minimization for VQA. *arXiv preprint arXiv:1606.03647*, 2016.

[41] W. Norcliffe-Brown, E. Vafeais, and S. Parisot. Learning conditioned graph structures for interpretable visual question answering. In *NeurIPS*, 2018.

[42] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 2019.

[43] J. Pennington, R. Socher, and C. Manning. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543, 2014.

[44] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville. FiLM: Visual Reasoning with a General Conditioning Layer. In *AAAI*, 2018.

[45] P. Ramachandran, B. Zoph, and Q. V. Le. Searching for activation functions. *CoRR*, abs/1710.05941, 2017.

[46] S. Ramakrishnan, A. Agrawal, and S. Lee. Overcoming language priors in visual question answering with adversarial regularization. In *NeurIPS*, pages 1548–1558, 2018.

[47] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015.

[48] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap. A simple neural network module for relational reasoning. In *NeurIPS*, 2017.

[49] D. Teney, P. Anderson, X. He, and A. van den Hengel. Tips and tricks for visual question answering: Learnings from the 2017 challenge. In *CVPR*, 2018.

[50] D. Teney and A. v. d. Hengel. Visual question answering as a meta learning task. In *ECCV*, 2017.

[51] Q. Wu, D. Teney, P. Wang, C. Shen, A. Dick, and A. v. d. Hengel. Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding*, 2017.

[52] Z. Yang, X. He, J. Gao, L. Deng, and A. J. Smola. Stacked attention networks for image question answering. In *CVPR*, 2016.

[53] K. Yi, J. Wu, C. Gan, A. Torralba, P. Kohli, and J. B. Tenenbaum. Neural-Symbolic VQA: Disentangling Reasoning from Vision and Language Understanding. In *NeurIPS*, 2018.

[54] Y. Zhang, J. Hare, and A. Prügel-Bennett. Learning to count objects in natural images for visual question answering. In *ICLR*, 2018.