Question Type Guided Attention in Visual Question Answering

Yang Shi[1] ⋆, Tommaso Furlanello[2], Sheng Zha[3], and Animashree Anandkumar[3 4]

[1] University of California, Irvine
shiy4@uci.edu
[2] University of Southern California
furlanel@usc.edu
[3] Amazon AI
{zhasheng},{anima}@amazon.com
[4] California Institute of Technology

Abstract. Visual Question Answering (VQA) requires integration of feature maps with drastically different structures. Image descriptors have structures at multiple spatial scales, while lexical inputs inherently follow a temporal sequence and naturally cluster into semantically different question types. A lot of previous works use complex models to extract feature representations but neglect to use high-level information summary such as question types in learning. In this work, we propose Question Type-guided Attention (QTA). It utilizes the information of question type to dynamically balance between bottom-up and top-down visual features, respectively extracted from ResNet and Faster R-CNN networks. We experiment with multiple VQA architectures with extensive input ablation studies over the TDIUC dataset and show that QTA systematically improves the performance by more than 5% across multiple question type categories such as "Activity Recognition", "Utility" and "Counting" on TDIUC dataset compared to the state-of-art. By adding QTA on the state-of-art model MCB, we achieve 3% improvement in overall accuracy. Finally, we propose a multi-task extension to predict question types which generalizes QTA to applications that lack question type, with a minimal performance loss.

Keywords: Visual question answering, Attention, Question type, Feature selection, Multi-task

1 Introduction

The relative maturity and flexibility of deep learning allow us to build upon the success of computer vision [17] and natural language [13, 20] to face new complex and multimodal tasks. Visual Question Answering(VQA) [4] focus on providing a natural language answer given any image and any free-form natural language question. To achieve this goal, information from multiple modalities must be integrated. Visual and lexical inputs are first processed using specialized encoding modules and then integrated through differentiable operators. Image features are usually extracted by convolution neural networks [7], while recurrent neural networks [13, 26] are used to extract

⋆ Work partially done while the author was working at Amazon AI

question features. Additionally, attention mechanism [30–32] forces the system to *look at* informative regions in both text and vision. Attention weight is calculated from the correlation between language and vision features and then is multiplied to the original feature.

Previous works explore new features to represent vision and language. Pre-trained ResNet [12] and VGG [24] are commonly used in VQA vision feature extraction. The authors in [27] show that post-processing CNN with region-specific image features [3] such as Faster R-CNN [22] can lead to an improvement of VQA performance. Along with generating language feature from either sentence-level or word-level using LSTM [13] or word embedding, Lu *et al*. [19] propose to model the question from word-level, phrase-level, and entire question-level in a hierarchical fashion.

Through extensive experimentation and ablation studies, we notice that the role of "raw" visual features from ResNet and processed region-specific features from Faster R-CNN is complementary and leads to improvement over different subsets of question types. However, we also notice that trivial information in VQA dataset: question/answer type is omitted in training. Generally, each sample in any VQA dataset contains one image file, one natural language question/answer and sometimes answer type. A lot of work use the answer type to analyze accuracy per type in result [4] but neglect to use it during learning. TDIUC [15] is a recently released dataset that contains question type for each sample. Compared to answer type, question type has less variety and is easier to interpret when we only have the question.

The focus of this work is the development of an attention mechanism that exploits high-level semantic information on the question type to guide the visual encoding process. This procedure introduces information leakage between modalities before the classical integration phase that improves the performance on VQA task. Specifically, We introduce a novel VQA architecture **Question Type-guided Attention**(QTA) that dynamically gates the contribution of ResNet and Faster R-CNN features based on the question type. Our results with QTA allow us to integrate the information from multiple visual sources and obtain gains across all question types. A general VQA network with our QTA is shown in Figure 1.

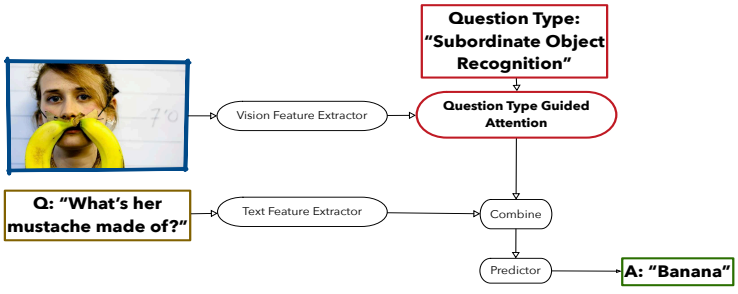

Fig. 1: General VQA network with QTA

The contributions of this paper are:(1) We propose question type-guided attention to balance between bottom-up and top-down visual features, which are respectively extracted from ResNet and Faster R-CNN networks. Our results show that QTA systematically improves the performance by more than 5% across multiple question type categories such as "Activity Recognition", "Utility" and "Counting" on TDIUC dataset. By adding QTA to the state-of-art model MCB, we achieve 3% improvement in overall accuracy. (2) We propose a multi-task extension that is trained to predict question types from the lexical inputs during training time that do not require ground truth labels during inference. We get more than 95% accuracy for the question type prediction while keeping the VQA task accuracy almost same as before. (3) Our analysis reveals some problems in the TDIUC VQA dataset. Though the "Absurd" question is intended to help reduce bias, it contains too many similar questions, specifically, questions regarding color. This will mislead the machine to predict wrong question types. Our QTA model gets 17% improvement on simple accuracy compared to the baseline in [15] when we exclude absurd questions in training.

2 Related Works

VQA task is first proposed in [4]. It focuses on providing a natural language answer given any image and any free-form natural language question. Collecting data and solving the task are equally challenging as they require the understanding of the joint relation between image and language without any bias.

Datasets VQA dataset v1 is first released by Antol *et al*. [4]. The dataset consists of two subsets: real images and abstract scenes. However, the inherent structure of our world is biased and it results in a biased dataset. In another word, a specific question tends to have the same answer regardless of the image. For example, when people ask about the color of the sky, the answer is most likely blue or black. It is unusual to see the answer be yellow. This is the bottleneck when we give a yellow color sky and ask the machine to answer it. Goyal *et al*. [10] release VQA dataset v2. This dataset pairs the same question with similar images that lead to different answers to reduce the sample bias. Agrawal *et al*. [2] also noticed that every question type has different prior distributions of answers. Based on that they propose GVQA and new splits of the VQA v1/v2. In the new split, the distribution of answers per question type is different in the test data compared to the training data. Zhang *et al*. [33, 34] also propose a method to reduce bias in abstract scenes dataset at question level. By extracting representative word tuples from questions, they can identify and control the balance for each question. Vizwiz [11] is another recently released dataset that uses pictures taken by blind people. Some pictures are of poor quality, and the questions are spoken. These data collection methods help reduce bias in the dataset.

Johnson *et al*. [14] introduce Compositional Language and Elementary Visual Reasoning (CLEVR) diagnostic dataset that focuses on reasoning. Strub *et al*. [25] propose a two-player guessing game: guess a target in a given image with a sequence of questions and answers. This requires both visual question reasoning and spatial reasoning.

The Task Driven Image Understanding Challenge dataset(TDIUC) [15] contains a total of over 1.6 million questions in 12 different types. It contains images and annota-

tions from MSCOCO [18] and Visual genome [16]. The key difference between TDIUC and the previous VQA v1/v2 dataset is the categorization of questions: Each question belongs to one of the 12 categories. This allows a task-oriented evaluation such as per question-type accuracies. They also include an "Absurd" question category in which questions are irrelevant to the image contents to help balance the dataset.

Feature Selection VQA requires solving several tasks at once involving both visual and textual inputs: visual perception, question understanding, and reasoning. Usually, features are extracted respectively with convolutional neural networks [7] from the image, and with recurrent neural networks [13, 26] from the text.

Pre-trained ResNet and VGG are commonly used in VQA vision feature extraction. The authors in [27] show that post-processing CNN with region-specific image features [3] can lead to an improvement of VQA performance. Specifically, they use pre-trained Faster R-CNN model to extract image features for VQA task. They won the VQA challenge 2017.

On the language side, pre-trained word embeddings such as Word2Vec [20] are used for text feature extraction. There is a discussion about the sufficiency of language input for VQA task. Agrawal *et al*. [1] have shown that state-of-art VQA models converge to the same answer even if only given half of the question compared to if given the whole sentence.

Generic Methods Information of both modalities are used jointly through means of combination, such as concatenation, product or sum. In [4], authors propose a baseline that combines LSTM embedding of the question and CNN embedding of the image via a point-wise multiplication followed by a multi-layer perceptron classifier.

Pooling Methods Pooling methods are widely used in visual tasks to combine information for various streams into one final feature representation. Common pooling methods such as average pooling and max pooling bring the property of translation invariance and robustness to elastic distortions at the cost of spatial locality. Bilinear pooling can preserve spatial information, which is performed with the outer product between two feature maps. However, this operation entails high output dimension($O(MN)$ for feature maps of dimension M and N). This exponential growth with respect to the number of feature maps renders it too costly to be applied to huge real image datasets. There have been several proposals for new pooling techniques to address this problem:

– Count sketch [5] is applied as a feature hashing operator to avoid dimension expanding in bilinear pooling. Given a vector $a \in \mathcal{R}^n$, random hash function $f \in \mathcal{R}^n$: $[n] \to [b]$ and binary variable $s \in \mathcal{R}^n$: $[n] \to \pm 1$, the **count sketch** [5] operator $cs(a, h, s) \in \mathcal{R}^b$ is:

$$cs(a, f, s)[j] = \sum_{f[i]=j} s[i]a[i], \quad j \in 1, \cdots, b \tag{1}$$

Gao *et al*. [9] use convolution layers from two different neural networks as the local descriptor extractors of the image and combine them using count sketch. "α-pooling" [23] allows the network to learn the pooling strategy: a continuous transition between linear and polynomial pooling. They show that higher α gives larger gain for fine-grained image recognition tasks. However, as α goes up, the computation complexity increases in polynomial order.

– Fukui *et al*. [8] use count sketch as a pooling method in VQA tasks and obtains the best results on VQA dataset v1 in VQA challenge 2016. They compute count sketch approximation of the visual and textual representation at each spatial location. Given text feature $v \in \mathcal{R}^L$ and image features $I \in \mathcal{R}^{C \times H \times W}$, Fukui *et al*. [8] propose **MCB** as:

$$\begin{aligned} &MCB(I[:,h,w] \otimes v)[t_1,h,w] \\ &\quad = (cs(I[:,h,w],f,s) \star cs(v,f,s))[t_1,h,w] \\ &\quad = IFFT1(FFT1(cs(I[:,h,w],f,s))[t_1,h,w] \circ FFT1(cs(v,f,s))[t_1]) \\ &\quad h \in \{1,\cdots H\}, w \in \{1,\cdots W\}, t_1 \in \{1,\cdots,b\} \end{aligned} \tag{2}$$

$\otimes$ denotes outer product. $\circ$ denotes element-wise product. $\star$ denotes convolution operator. This procedure preserves spatial information in the image feature.

Attention Focusing on the objects in the image that are related to the question is the key to understand the correlation between the image and the question. Attention mechanism is used to address this problem. There are soft attention and hard attention [31] based on whether the attention term/loss function is differentiable or not. Yang *et al*. [32] and Xu *et al*. [30] propose word guided spatial attention specifically for VQA task. Attention weight at each spatial location is calculated by the correlation between the embedded question feature and the embedded visual features. The attended pixels are at the maximum correlations. Wang *et al*. [28] explore mechanisms of triplet attention that interact between the image, question and candidate answers based on image-question pairs.

3 Question Type Guided Visual Attention

Question type is very important in predicting the answer regardless whether we have the corresponding image or not. For example, questions starting with "how many" will mostly lead to numerical answers. Agrawal *et al*. [1] have shown that state-of-art VQA models converge to the same answer even if only given half of the question compared to if given the whole sentence. Besides that, inspired by [27], we are curious about combining bottom-up and top-down visual features in VQA task. To get a deep understanding of visual feature preference for different questions, we try to find an attention mechanism between these two. Since question type is representing the question, we propose Question Type-guided Attention(QTA).

Given several independent image features $F_1, F_2, \cdots F_k$, such as features from ResNet, VGG or Faster R-CNN, we concatenate them as one image feature: $F = [F_1, F_2, \cdots F_k] \in \mathcal{R}^M$. Assume there are N different question types, QTA is defined as $F \circ WQ$, where $Q \in \mathcal{R}^N$ is the one-hot encoding of the question type, and $W \in \mathcal{R}^{M \times N}$ is the hidden weight. We can learn the weight by back propagation through the network. In other words, we learn a question type embedding and use it as attention weight.

QTA can be used in both generic and complex pooling models. In Figure 2, we show a simple concatenation model with question type as input. We describe it in detail

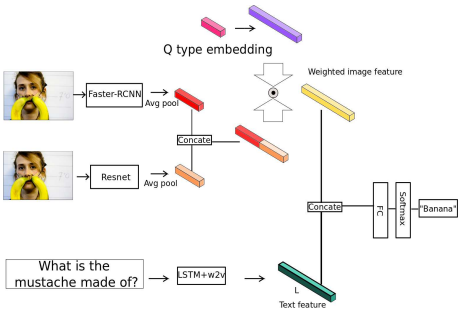

Fig. 2: Concatenation model with QTA structure for VQA task(CATL-QTA^W in Section 4)

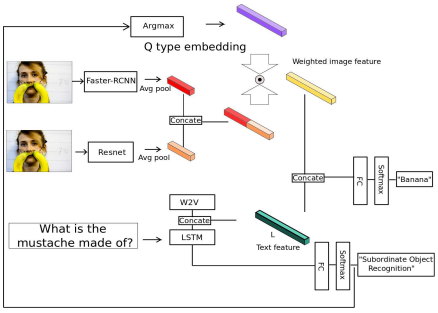

Fig. 3: Concatenation model with QTA structure for multi-task(CATL-QTA-M^W in Section 4)

in Section 4. To fully exploit image features in different channels and preserve spatial information, we also propose MCB with question type-guided image attention in Figure 4.

One obvious limitation of QTA is that it requires question type label. In the real world scenario, the question type for each question may not be available. In this case, it is still possible to predict the question type from the text, and use it as input to the QTA network. Thus, we propose a multi-task model that focuses on VQA task along with the prediction of the question type in Figure 3. This model operates in the setting where true question type is available only at training time. In Section 5, we also show through experiment that it is a relatively easy task to predict the question type from question text, and thus making our method generalizable to those VQA settings that lack question type.

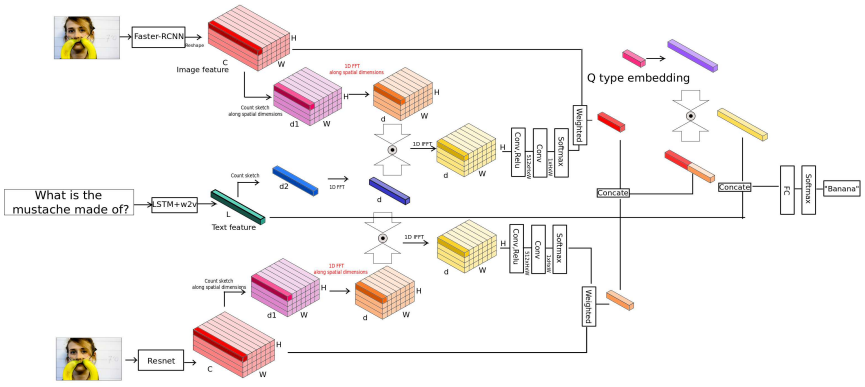

Fig. 4: MCB model with QTA structure(MCB-QTA in Section 4)

4 Experiments

In this section, we describe the dataset in Section 4.1, evaluation metrics in Section 4.2, model features in Section 4.3, and model structures are explained in Section 4.4.

4.1 Dataset

Our experiments are conducted on the Task Driven Image Understanding Challenge dataset(TDIUC) [15], which contains over 1.6 million questions in 12 different types. This dataset includes VQA v1 and Visual Genome, with a total of 122429 training images and 57565 test images. The annotation sources are MSCOCO (VQA v1), Visual genome annotations, and manual annotations. TDIUC introduces absurd questions that force an algorithm to determine if a question is valid for a given image. There are 1115299 total training questions and 538543 total test questions. The total number of samples is 3 times larger than that in VQA v1 dataset.

4.2 Evaluation Metrics

There are total 12 different question types in TDIUC dataset as we mentioned in Section 2. We calculate the simple accuracy for each type separately and also report the arithmetic and harmonic means across all per question-type(MPT) accuracies.

4.3 Feature Representation

Image feature We use the output of "pool" of a 152-layer ResNet as an image feature baseline. The output dimension is $2048 \times 14 \times 14$. Faster R-CNN [22] focuses on object detection and classification. Teney *et al*. [27] use it to extract object-oriented features for VQA dataset and show better performance compared to the ones using ResNet feature. We fix the number of detected objects to be 36 and extract the image features based on their pre-trained Faster R-CNN model. As a result, the extracted image feature is a 36×2048 matrix. To fit in MCB model, which requires spatial representation, we reshape it into a $6 \times 6 \times 2048$ tensor.

Text feature We use common word embedding library: 300-dim Word2Vec [20] as a pre-trained text feature: we sum over the word embeddings for all words in the sentence. A two-layer LSTM is used as an end-to-end text feature extractor. We also use the encoder of google neural machine translation(NMT) system [29] as a pre-trained text feature and compare it with Word2Vec. The pre-trained NMT model is trained on UN parallel corpus 1.0 in MXnet [6]. Its BLEU score is 34. The output dimension of the encoder is 1024.

4.4 Models

Baseline models Baseline models are based on a one-layer MLP: A fully connected network classifier with one hidden layer with ReLu non-linearity, followed by a softmax layer. The input is a concatenation of image and text feature. There are 8192 units in the hidden state.

Table 1: Baseline models

Name	Image feature	Text feature	Model
CAT1	ResNet/Faster R-CNN vector feature	Skipthought/NMT/Word2Vec pre-trined feature	MLP
CAT1L	ResNet/Faster R-CNN vector feature	End-to-end 2-layer LSTM's last hidden state	MLP
CATL	Concatenation of ResNet and Faster R-CNN vector features	End-to-end 2-layer LSTM's last hidden state	MLP
CAT2	Concatenation of ResNet and Faster R-CNN vector features	NMT pre-trined feature	MLP

To compare different image and text feature, we have **CAT1**, **CAT1L** and **CATL**. To check the complementarity of different features between ResNet and Faster R-CNN and show how they perform differently across question types, we set up baseline **CAT2**. In LSTM, the hidden state length is 1024. The word embedding dimension is 300. Detailed definitions are in Table 1.

To further exam and explain our QTA proposal, we use more sophisticate feature integration operators as a strong baseline to compare with. **MCB-A**, as we mentioned in Section 2, is proposed in [8]. **RAU** [21] is a framework that combines the embedding, attention and predicts operation together inside a recurrent network. We reference results of these two models from [15].

QTA models From the baseline analysis, we realize that ResNet and Faster R-CNN features are complementary to each other. Using question type as guidance for image feature selection is the key to make image feature stronger. Therefore, we propose QTA networks in MLP model(**CATL-QTA**) and MCB model(**MCB-QTA**). The out dimension of the count sketch in the MCB is 8000. The structures are in Figure 2, 4. The descriptions are in Table 2.

To check whether the model benefits from the QTA mechanism or from added question type information itself, we design a network that only uses question type embedding without attention. **CAT-QT** and **CATL-QT** are the two proposed network using Word2Vec and LSTM lexical features.

As mentions in Section 3, we propose a multi-task network for QTA in case we don't have question type label at inference. **CATL-QTA-M** is a multi-task model based on CATL-QTA. The output of LSTM is connected to a one-layer MLP to predict question type for the input question. The prediction result is then fed into QTA part through argmax. The Multi-task MLP is in Figure 3.

5 Results and Analysis

We first focus in Sections 5.1 and 5.2 on results concerning the complementarity of different features across question category types. For the visual domain, we explore the use of Faster R-CNN and ResNet features, while for the lexical domain we use NMT,

Table 2: QTA models

Name	Image feature	Text feature	Model
CATL-QTA	QTA weighted pre-trained vector features from ResNet and Faster R-CNN	End-to-end 2-layer LSTM's last hidden state	MLP
MCB-QTA	QTA weighted pre-trained spatial features from ResNet and Faster R-CNN	End-to-end 2-layer LSTM's last hidden state	MCB
CAT-QT	Concatenation of ResNet and Faster R-CNN vector features	Concatenation of Word2Vec pre-trined feature and a 1024-dim question type embedding	MLP
CATL-QT	Concatenation of ResNet hidden state and Faster R-CNN vector features	Concatenation of end-to-end 2-layer LSTM's last and a 1024-dim question type embedding	MLP
CATL-QTA-M	QTA weighted pre-trained spatial features from ResNet and Faster R-CNN	End-to-end 2-layer LSTM's last hidden state	Multi-task MLP

LSTM and pre-trained Word2Vec features. We then analyze the effect of question type both as input and with QTA in VQA tasks in Section 5.3. Finally, in the remaining subsections, we extend the basic concatenation QTA model to MCB style pooling; introduce question type as both input and output during training such that the network can produce predicted question types during inference; and study more in depth the effect of the question category "Absurd" on the overall model performance across categories.

5.1 Faster R-CNN and ResNet Features

Table 3 reports our extensive ablation analysis of simple concatenation models using multiple visual and lexical feature sources. From the results in the second and third columns, we see that overall the model with Faster R-CNN features outperform the one using ResNet features when using NMT features. We show in column 4 that the features sources are complementary, and their combination is better across most categories (in bold) with respect to the single source models in columns 2 and 3. In columns 5,6; 7,8 and 9,10 we replicate the same comparison between ResNet and R-CNN features using more sophisticate models to embed the lexical information. We reach more than 10 % accuracy increase, from 69.53 % to 80.16 % using a simple concatenation model with an accurate selection of the feature type.

5.2 Pre-trained and Jointly-trained Text Feature Extractors

The first four columns in Table 3 show the results of models with text features from NMT. To fully explore the text feature extractor in VQA system, we substitute the NMT pre-trained language feature extractor with a jointly-trained two layer LSTM model. The improved performance of jointly-training text feature extractor can be appreciated by comparing the results of the 1-4 and 5-10 columns in Table 3. For example, comparing second column and fifth column in Table 3, we get 6% improvement using LSTM while keeping image feature and network same.

We obtain the best model by concatenating the output of the LSTM and the pre-trained NMT/Word2Vec feature, as shown in Table 3. It gives us 10% improvement for "Utility and Affordances" when we look at the fifth and seventh column. We find the

Table 3: Benchmark results of concatenation models on TDIUC dataset using different image features and pre-trained language feature. 1: Use ResNet feature and SkipGram feature 2: Use ResNet feature and NMT feature 3: Use Faster R-CNN feature and NMT feature 4: Use ResNet feature and end-to-end LSTM feature 5: Use Faster R-CNN feature and end-to-end LSTM feature. N denotes that additional NMT embedding is concatenated to LSTM output. W denotes that additional Word2Vec embedding is concatenated to LSTM output(Following tables also use the same notation)

Columns	1	2	3	4	5	6	7	8	9	10
Accuracy(%)	**CAT1**[1] [15]	**CAT1**[2]	**CAT1**[3]	**CAT2**	**CAT1L**[4]	**CAT1L**[5]	**CAT1L**[4N]	**CAT1L**[5N]	**CAT1L**[4W]	**CAT1L**[5W]
Scene Recognition	72.19	68.51	68.81	**69.06**	91.62	**92.27**	91.16	**92.33**	91.57	**92.45**
Sport Recognition	85.16	89.67	92.36	**93.15**	90.94	**93.84**	89.62	**93.52**	90.77	**94.05**
Color Attributes	43.69	32.90	34.35	**34.99**	45.62	**49.43**	44.07	**47.78**	47.33	**49.47**
Other Attributes	42.89	38.05	**39.76**	39.67	40.89	**43.49**	39.60	**42.35**	41.92	**45.19**
Activity Recognition	24.16	39.34	45.75	**46.87**	42.95	**49.25**	40.12	**44.11**	42.13	**49.25**
Positional Reasoning	25.15	25.63	27.16	**28.02**	26.22	**29.35**	24.17	**27.50**	25.72	**28.59**
Sub. Object Recognition	80.92	83.94	85.67	**86.78**	82.20	**85.06**	81.85	**84.47**	82.52	**85.05**
Absurd	96.96	94.98	94.77	**95.82**	**90.87**	87.10	**95.38**	93.28	**93.59**	91.95
Utility and Affordances	24.56	25.93	**27.78**	27.16	15.43	**25.93**	**25.31**	18.52	16.05	**17.28**
Object Presence	69.43	77.21	77.90	**78.29**	89.40	**91.14**	90.13	**91.95**	91.08	**91.81**
Counting	44.82	48.46	52.18	**52.57**	45.95	**50.27**	44.26	**49.24**	44.93	**51.30**
Sentiment Understanding	53.00	43.45	46.49	**47.28**	46.49	**48.72**	41.85	**42.81**	44.89	**46.01**
Overall (Arithmetic MPT)	55.25	55.67	57.57	58.31	59.05	62.15	58.96	60.66	59.38	**61.80**
Overall (Harmonic MPT)	44.13	45.37	47.99	48.44	44.09	**51.66**	46.84	46.84	44.42	47.70
Overall Accuracy	69.53	71.41	72.44	73.05	77.55	78.66	78.35	79.94	78.94	**80.16**

use of Word2Vec is better than NMT feature in last four columns in Table 3. We think the better performance of Word2Vec with respect to the NMT encoder, might be due to the more similar structure of single sentence samples of Word2Vec training set with those from classical VQA dataset with respect to those used for training NMT models.

Accuracy(%)	**CATL**	**CATL-QTA**	**CATL**^{W}	**CATL-QTA**^{W}
Scene Recognition	93.18	93.45	93.31	93.80
Sport Recognition	94.69	95.45	94.96	95.55
Color Attributes	54.66	56.08	57.59	60.16
Other Attributes	48.52	50.30	52.25	54.36
Activity Recognition	53.36	58.43	54.59	60.10
Positional Reasoning	32.73	31.94	33.63	34.71
Sub. Object Recognition	86.56	86.76	86.52	86.98
Absurd	95.03	100.00	98.01	100.00
Utility and Affordances	29.01	23.46	29.01	31.48
Object Presence	93.34	93.48	94.13	94.55
Counting	50.08	49.93	52.97	53.25
Sentiment Understanding	56.23	56.87	62.62	64.38
Overall (Arithmetic MPT)	65.62	66.34	67.46	69.11
Overall (Harmonic MPT)	55.95	54.60	57.83	60.08
Overall Accuracy	82.23	83.62	83.92	85.03

Table 4: QTA in concatenation models on TDIUC dataset

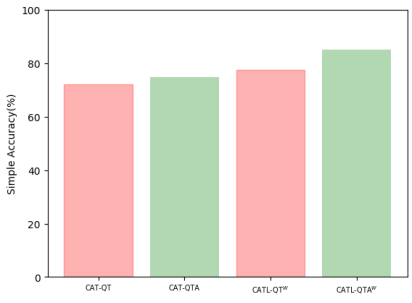

Fig. 5: Evaluation of different ways to utilize information from question type

5.3 QTA in concatenation models

We use QTA in concatenation models to study the effect of QTA. The framework is in Figure 2. We compare the network using a weighted feature with the same network using an unweighted concatenated image feature in Table 4. As we can see, the model

using the weighted feature has more power than the one using the unweighted feature. 9 out of 12 categories get improved results. “Color” and “Activity Recognition” get around 2% and 6% accuracy increases.

To ensure that the improvement is not because of the added question type information but the attention mechanism using question type, we show the comparison of QTA with QT in Figure 5. With same text feature and image feature and approximately same number of parameters in the network, QTA is 3-5% better than QT.

We show the effect of QTA on image feature norms in Figure 6. By weighing the image features by question type, we find that our model relies more on Faster R-CNN features for “Absurd” question samples while it relies more on ResNet features for “Color” questions.

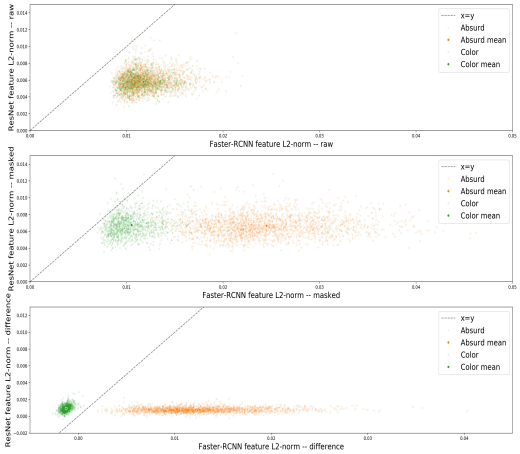

Fig. 6: Effects of weighting by QTA. Top: raw feature norms, Middle: feature norms weighted by QTA, Bottom: differences of norms after weighting vs before weighting. For color questions, the feature norms shift towards ResNet features, while for absurd questions they shift towards Faster-RCNN features.

The best setting we get in concatenation model is using a weighted image feature concatenated with the output of the LSTM and Word2Vec feature(CATL-QTA^W). It gets 5% improvement compared to complicated deep network such as RAU and MCB-A in Table 5.

5.4 QTA in pooling models

To show how to combine QTA with more complicated feature integration operator, we propose MCB-QTA structure. Even though MCB-QTA in Table 5 doesn’t win with simple accuracy, it shows great performance in many categories such as “Object Recognition” and “Counting”. Accuracy in “Utility and Affordances” is improved by 6% compared to our CATL-QTA model. It gets 8% improvement in “Activity recognition”

Table 5: Results of QTA models on TDIUC dataset compared to state-of-art models

Accuracy(%)	**CATL-QTA**^{W}	**MCB-QTA**	**MCB-A** [15]	**RAU** [15]
Scene Recognition	93.80	93.56	93.06	**93.96**
Sport Recognition	95.55	**95.70**	92.77	93.47
Color Attributes	60.16	59.82	**68.54**	66.86
Other Attributes	54.36	54.06	**56.72**	56.49
Activity Recognition	60.10	**60.55**	52.35	51.60
Positional Reasoning	34.71	34.00	**35.40**	35.26
Sub. Object Recognition	86.98	**87.00**	85.54	86.11
Absurd	**100.00**	100.00	84.82	96.08
Utility and Affordances	31.48	**37.04**	35.09	31.58
Object Presence	**94.55**	94.34	93.64	94.38
Counting	53.25	**53.99**	51.01	48.43
Sentiment Understanding	64.38	65.65	**66.25**	60.09
Overall (Arithmetic MPT)	69.11	**69.69**	67.90	67.81
Overall (Harmonic MPT)	60.08	**61.56**	60.47	59.00
Overall Accuracy	**85.03**	84.97	81.86	84.26

compared to state-of-art model MCB-A and also gets the best Arithmetic and Harmonic MPT value.

5.5 Multi-task analysis

In this part, we will discuss how we use QTA when we have questions without specific question types. It is quite easy to predict the question type from the question itself. We use a 2-layer LSTM followed by a classifier and the test accuracy is 96% after 9 epochs. The problem is whether we can predict the question type while keeping the same performance for VQA task or not. As described in Figure 3, we use the predicted question type as input of the QTA network in a multi-task setting. We get 84.33% test simple accuracy for VQA task as shown in Table 9. When we compare it to MCB-A or RAU in Table 5, though accuracy gets a little affected for most of the categories, we still get 2% improvement in "Sports Recognition" and "Counting".

We fine-tune our model on VQA v1 using a pre-trained multi-task model that was trained on TDIUC. We use the question type predictor in the multi-task model as the input of QTA. Our model's performance is better than MCB in Table 6 with an approximately same number of parameters in the network.

5.6 Findings on TDIUC dataset

To further analyze the effects of the question type prediction part in this multi-task framework, we list the confusion matrix for the question type prediction results in Table 7. "Color" and "Absurd" question type predictions are most often bi-directionally confused. The reason for this is that among all absurd questions, more than 60% are questions start with "What color". To avoid this bias, we remove all absurd questions and run our multi-task model again. In this setting, our question type prediction did much better than before. Almost all categories get 99% accuracy as shown in Table 8.

Table 6: Results of test-dev accuracy on VQA v1. Models are trained on the VQA v1 train split and tested on test-dev

	Accuracy(%)
Element-wise Sum [8]	56.50
Concatenation [8]	57.49
Concatenation + FC [8]	58.40
Element-wise Product [8]	58.57
Element-wise Product + FC [8]	56.44
MCB(2048 $\times$ 2048 $\rightarrow$ 16K) [8]	59.83
CATL-QTA-M + FC	**60.32**

We also compare our QTA models' performance without absurd questions in Table 9. In CATL-QTA network, removing absurd questions doesn't help much because in test we feed in the true question type labels. But it is useful when we consider the multi-task model. From fourth and fifth columns, we see that without absurd questions, we get improved performance among all categories. This is because we remove the absurd questions that may mislead the network to predict "color" question type in the test.

Table 7: Confusion matrix for test question types prediction in CATL-QTA-M using TDIUC dataset. 1. Other Attributes 2. Sentiment Understanding 3. Sports Recognition 4. Position Reasoning 5. Object Utilities/Affordances 6. Activity Recognition 7. Scene Classification 8. Color 9. Object Recognition 10.Object Presence 11.Counting 12. Absurd

Target	Predicted												Acc(%)
	1	2	3	4	5	6	7	8	9	10	11	12	95.66
1	77.76	0.00	0.89	3.20	0.00	0.08	0.42	1.15	0.12	0.00	0.00	16.38	
2	0.80	60.51	1.77	8.83	0.00	2.25	2.57	0.00	1.44	0.96	0.16	20.71	
3	0.31	0.00	73.08	0.37	0.00	0.17	0.00	0.03	0.02	0.00	0.01	26.01	
4	2.95	0.02	0.01	89.52	0.00	0.01	0.02	0.19	1.88	0.03	0.03	5.35	
5	12.50	0.63	3.12	45.62	0.00	0.00	3.12	0.00	11.25	0.00	0.00	23.75	
6	0.79	0.00	14.56	1.76	0.00	13.18	0.00	0.00	2.21	0.00	0.07	67.43	
7	0.04	0.00	0.04	0.40	0.00	0.01	99.40	0.02	0.00	0.00	0.06	0.03	
8	0.32	0.00	0.18	0.13	0.00	0.00	0.00	86.10	0.00	0.00	0.00	13.28	
9	0.01	0.00	0.00	0.31	0.00	0.00	0.00	0.00	98.96	0.01	0.00	0.71	
10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	
11	0.00	0.00	0.00	0.01	0.00	0.00	0.02	0.00	0.02	0.05	99.90	0.00	
12	0.35	0.00	0.18	0.41	0.00	0.03	0.00	3.18	0.40	0.00	0.00	95.46	

6 Conclusion

We propose a question type-guided visual attention (QTA) network. We show empirically that with the question type information, models can balance between bottom-up and top-down visual features and achieve state-of-the-art performance. Our results show that QTA systematically improves the performance by more than 5% across multiple

Table 8: Confusion matrix for test question types prediction in CATL-QTA-M using TDIUC dataset without absurd questions. Numbers represent same categories as in Table 7

Target	Predicted												Acc(%)
	1	2	3	4	5	6	7	8	9	10	11	12	99.50
1	98.39	0.00	0.07	0.15	0.00	0.13	0.08	0.63	0.55	0.00	0.00	N/A	
2	0.16	84.03	3.67	0.00	0.00	3.35	5.59	0.00	0.48	0.00	2.72	N/A	
3	0.00	0.08	97.31	0.00	0.00	2.37	0.01	0.00	0.10	0.02	0.11	N/A	
4	1.01	0.00	0.00	98.07	0.00	0.01	0.00	0.51	0.41	0.00	0.00	N/A	
5	8.64	3.70	14.81	0.00	0.00	59.26	7.41	1.23	4.94	0.00	0.00	N/A	
6	0.45	0.15	31.42	0.00	0.00	67.39	0.04	0.04	0.45	0.00	0.07	N/A	
7	0.02	0.03	0.00	0.00	0.00	0.03	99.86	0.02	0.00	0.00	0.04	N/A	
8	0.06	0.00	0.00	0.13	0.00	0.04	0.07	99.70	0.00	0.00	0.00	N/A	
9	0.06	0.00	0.13	0.01	0.00	0.02	0.00	0.00	99.76	0.01	0.00	N/A	
10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	N/A	
11	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01	99.98	N/A	
12	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	

Table 9: Results of test accuracy when question type is hidden with/without absurd questions in training. We compare them with similar QTA models. * denotes training and testing without absurd questions

	CATL-QTA^{W}	**CATL^{W*}**	**CATL-QTA^{W*}**	**CATL-QTA-M**	**CATL-QTA-M***	**CAT1^{1*}** [15]
Scene Recognition	93.80	93.46	93.62	93.74	93.82	72.75
Sport Recognition	95.55	94.97	95.47	94.80	95.31	89.40
Color Attributes	60.16	57.84	58.63	57.62	59.73	50.52
Other Attributes	54.36	53.90	53.44	52.05	56.17	51.47
Activity Recognition	60.10	57.38	59.43	53.13	58.61	48.55
Positional Reasoning	34.71	33.98	34.63	33.90	34.70	27.73
Sub. Object Recognition	86.98	86.62	86.74	86.89	86.80	81.66
Absurd	100.00	N/A	N/A	98.57	N/A	N/A
Utility and Affordances	31.48	27.78	34.57	24.07	35.19	30.99
Object Presence	94.55	93.87	94.22	94.57	94.60	69.50
Counting	53.25	52.33	52.20	53.59	55.30	44.84
Sentiment Understanding	64.38	64.06	65.81	60.06	61.31	59.94
Overall (Arithmetic MPT)	69.11	65.11	66.25	66.92	66.88	57.03
Overall (Harmonic MPT)	60.08	55.89	58.51	55.77	58.82	50.30
Simple Accuracy	85.03	79.79	80.13	84.33	80.95	63.30

question type categories such as “Activity Recognition”, “Utility” and “Counting” on TDIUC dataset. We consider the case when we don’t have question type for test and propose a multi-task model to overcome this limitation by adding question type prediction task in the VQA task. We get around 95% accuracy for the question type prediction while keeping the VQA task accuracy almost same as before.

Acknowledgements We thank Amazon AI for providing computing resources. Yang Shi is supported by Air Force Award FA9550-15-1-0221.

References

1. Agrawal, A., Batra, D., Parikh, D.: Analyzing the behavior of visual question answering models. EMNLP (2016)
2. Agrawal, A., Batra, D., Parikh, D., Kembhavi, A.: Don't just assume; look and answer: Overcoming priors for visual question answering. CVPR 2018
3. Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., Zhang, L.: Bottom-up and top-down attention for image captioning and VQA, http://arxiv.org/abs/1707.07998
4. Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C.L., Parikh, D.: VQA: Visual Question Answering. In: International Conference on Computer Vision (ICCV) (2015)
5. Charikar, M., Chen, K., Farach-Colton, M.: Finding frequent items in data streams. In Proceedings of ICALP (2002)
6. Chen, T., Li, M., Li, Y., Lin, M., Wang, N., Wang, M., Xiao, T., Xu, B., Zhang, C., Zhang, Z.: Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems. Neural Information Processing Systems, Workshop on Machine Learning Systems 2015 (2015)
7. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: Decaf: A deep convolutional activation feature for generic visual recognition, http://arxiv.org/abs/1310.1531
8. Fukui, A., Park, D.H., Yang, D., Rohrbach, A., Darrell, T., Rohrbach, M.: Multimodal compact bilinear pooling for visual question answering and visual grounding. EMNLP (2016)
9. Gao, Y., Beijbom, O., Zhang, N., Darrell, T.: Compact bilinear pooling. Computer Vision and Pattern Recognition (CVPR) (2016)
10. Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., Parikh, D.: Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
11. Gurari, D., Li, Q., Stangl, A.J., Guo, A., Lin, C., Grauman, K., Luo, J., Bigham, J.P.: Vizwiz grand challenge: Answering visual questions from blind people, https://arxiv.org/abs/1802.08218
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
13. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)
14. Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C.L., Girshick, R.B.: CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning, http://arxiv.org/abs/1612.06890
15. Kafle, K., Kanan, C.: An analysis of visual question answering algorithms. In: ICCV (2017)
16. Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L., Shamma, D.A., Bernstein, M.S., Li, F.: Visual genome: Connecting language and vision using crowdsourced dense image annotations, https://arxiv.org/abs/1602.07332
17. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 25, pp. 1097–1105. Curran Associates, Inc. (2012)
18. Lin, T., Maire, M., Belongie, S.J., Bourdev, L.D., Girshick, R.B., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: common objects in context. In ECCV (2014)
19. Lu, J., Yang, J., Batra, D., Parikh, D.: Hierarchical question-image co-attention for visual question answering. In NIPS (2016)

20. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems pp. 3111–3119
21. Noh, H., Han, B.: Training recurrent answering units with joint loss minimization for VQA, https://arxiv.org/abs/1606.03647
22. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks, http://arxiv.org/abs/1506.01497
23. Simon, M., Gao, Y., Darrell, T., Denzler, J., Rodner, E.: Generalized orderless pooling performs implicit salient matching. In: International Conference on Computer Vision (ICCV) (2017)
24. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition, http://arxiv.org/abs/1409.1556
25. Strub, F., de Vries, H., Mary, J., Piot, B., Courville, A.C., Pietquin, O.: End-to-end optimization of goal-driven and visually grounded dialogue systems. In: International Joint Conference on Artificial Intelligence (IJCAI) (2017)
26. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Proceedings of the 27th International Conference on Neural Information Processing Systems. pp. 3104–3112. NIPS'14, MIT Press, Cambridge, MA, USA (2014), http://dl.acm.org/citation.cfm?id=2969033.2969173
27. Teney, D., Anderson, P., He, X., van den Hengel, A.: Tips and tricks for visual question answering: Learnings from the 2017 challenge, http://arxiv.org/abs/1708.02711
28. Wang, Z., Liu, X., Chen, L., Wang, L., Qiao, Y., Xie, X., Fowlkes, C.: Structured triplet learning with pos-tag guided attention for visual question answering. IEEE Winter Conf. on Applications of Computer Vision (2018)
29. Wu, Y., Schuster, M., Chen, Z., Le, Q.V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Łukasz Kaiser, Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., Dean, J.: Google's neural machine translation system: Bridging the gap between human and machine translation http://arxiv.org/abs/1609.08144
30. Xu, H., Saenko, K.: Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. European Conference on Computer Vision (2016)
31. Xu, K., Ba, J., Kiros, R., Courville, A., Salakhutdinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. International Conference on Machine Learning (2015)
32. Yang, Z., He, X., Gao, J., Deng, L., Smola, A.: Stacked attention networks for image question answering, https://arxiv.org/abs/1511.02274
33. Zhang, P.: Towards Interpretable Vision Systems. Ph.D. thesis, Virginia Polytechnic Institute and State University (2017)
34. Zhang, P., Goyal, Y., Summers-Stay, D., Batra, D., Parikh, D.: Yin and Yang: Balancing and answering binary visual questions. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2016)