Multiple Context Learning Networks for Visual Question Answering

Pufen Zhang
Jiangxi University of Science and Technology

Hong Lan (lanhong@jxust.edu.cn)
Jiangxi University of Science and Technology

Research Article

Keywords: visual question answering, multimodal learning, computer vision, natural language processing, multiple context learning

Posted Date: October 12th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-955099/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License
Multiple Context Learning Networks for Visual Question Answering

Pufen Zhang¹, Hong Lan¹
¹School of Information Engineering, Jiangxi University of Science and Technology, Ganzhou 341000, China.
Correspondence author Hong Lan: lanhong@jxust.edu.cn

Abstract
In recently years, some visual question answering (VQA) methods that emphasize the simultaneous understanding of both the context of image and question have been proposed. Despite the effectiveness of these methods, they fail to explore a more comprehensive and generalized context learning tactics. To address this issue, we propose a novel Multiple Context Learning Networks (MCLN) to model the multiple contexts for VQA. Three kinds of contexts are investigated, namely visual context, textual context and a special visual-textual context that ignored by previous methods. Moreover, three corresponding context learning modules are proposed. These modules endow image and text representations with context-aware information based on a uniform context learning strategy. And they work together to form a multiple context learning layer (MCL). Such MCL can be stacked in depth and which describe high-level context information by associating intra-modal contexts with inter-modal context. On the VQA v2.0 datasets, the proposed model achieves 71.05% and 71.48% on test-dev set and test-std set respectively, and gains better performance than the previous state-of-the-art methods. In addition, extensive ablation studies have been carried out to examine the effectiveness of the proposed method.

Keywords visual question answering, multimodal learning, computer vision, natural language processing, multiple context learning.

1 Introduction

An artificial intelligence agent must be able to understand not just the semantics of words, but also the content of images. This capability is required by many multimodal tasks about image and text modalities [1]-[5]. In this paper, we focus on the visual question answering (VQA) [4]-[5], which aims to identify a correct answer related to the given image-question pair from candidate answers. Compared to other multimodal tasks, VQA is a more difficult task that requires associating visual content in the image with the semantic meaning in the question, together with visual reasoning to make the correct answer.

To process multimodality features of image and question, most VQA methods adopt a combination of Convolutional Neural Networks (CNN) [6] and Recurrent Neural Networks (RNN) [7]. In the early stages, some works [8]-[9] utilized CNNs to extract visual features of global image contents and RNNs to encode global language contents. However, the extracted global features are not fine-grained and contain noisy information. To address this issue and with the development of
attention mechanisms in the field of deep learning, a VQA approach that employs attention mechanisms to further process multimodal features is becoming increasingly popular. Such method can to locate the objects related to answers in an image or the keywords in a question. For example, the question-guided visual attention on image regions was first proposed by [10]. Besides, some works shown that learning textual attention on question words can also improve the VQA performance [11]. Following, a large variety of attention-based variations [12]-[14] including co-attention and stacked attention have been proposed.

Although aforementioned attention-based approaches have dramatically ameliorated the performance of VQA, there are still challenges in VQA, one of which is multimodal context learning. As context reflects the high-order interactive information between entities (image objects or question words) and helps to distinguish the targets from other entities, several attention-based techniques have lately begun to emphasize multimodal context learning. One strategy is to model the image into a relational contextualized graph. For example, ReGAT [15] and v-AGCN [16] structure the contextualized relation graph on the image and use the graph attention network [17] to generate the relation-aware visual representations for capturing the visual context information. Their methods are capable of performing complex relational inference, but the contextualized graph is built upon the relation with language prior, which does not deliver complete contextual information on the image modality. In addition, both models only take into account the context for image modality. Against this disadvantage, more recently, some works have attempted to learn extra textual context. DFAF [22] based on the inter-modality attention flow to complement the intra-modal information and using the intra-modality attention flow to capture the intra-modal contexts. Two modular co-attention models, namely MCAN [19] and MEDAN [20], have been introduced to VQA task to capture more complete context information based on the and encoder-decoder framework. However, these methods only learn the intra-modal context within each modality.

![Figure 1](image.png)

**Figure 1.** A diagram of the proposed context learning method, which simultaneously learns intra- and inter-modal contexts by using a uniform context learning framework. VCL and TCL model the intra-modal contexts in MCL layer, while VTCL models the inter-modal context. For image features $V$ and question feature $T$, the handled features with context information are $V'$ and $T'$, respectively.

Despite the effectiveness of these methods, we still find that a more comprehensive and generalized context learning tactics is rarely explored. On the one hand, a special inter-modal
context, named visual-textual context and expressed as the context dependence under two modal information, is almost ignored by previous methods. On the other hand, how to explore an accordant context learning way modeling multiple contexts and integrate multi-context learning into a unified VQA framework, there is still potential scope for further research. From these insights, multiple contexts are explored in this paper for VQA, which includes two intra-modal contexts (i.e., the visual context in the image modality and textual context in the text modality) and an inter-modal context (i.e., the visual-textual context in the inter-modality). Meanwhile, a novel Multiple Context Learning Networks (MCLN) is proposed for corresponding context learning. Figure 1 shows the proposed multiple context learning strategy. The visual context learning module (VCL), textual context learning module (TCL) and visual-textual context learning module (VTCL) are uniformly modeled base on the key-query attention. Two intra-modal contexts are learned through the VCL and TCL, respectively. Then, the processed image features and question features are connected to form the visual-textual features, which is fed into VTCL to extract the inter-modal context information. After that, by modular composition of the three modules, the multiple context learning layer (MCL) is constructed. In addition, by cascading MCL in depth, the interaction for the intra- and inter-modal contexts can be reached handily, allowing for deeper level and more complex context learning. Finally, the deep Multiple Context Learning Networks Network (MCAN) that consists of cascaded MCA layers is proposed. In addition, we introduce the BERT [21] model to further enhance the context learning ability of MCLN for text modality.

The main contributions of this study can be summarized as follows.

1) We explore multiple contexts for VQA task and propose a uniform context learning strategy to perform multi-context learning upon the detected objects in an image and the words in a question. The proposed MCLN is able to capture the intra- and inter-modal context information.

2) An advanced text encoder based on pre-trained BERT [25] is also introduced for further facilitate the textual context learning, which boosts the performance of MCLN.

3) Evaluation results on the benchmark VQA v2.0 dataset [5] demonstrate that MCLN achieves state-of-the-art performance. In addition, comprehensive ablation studies are conducted to quantitatively and qualitatively prove the roles of different context learning modules in the proposed model, verifying the effectiveness of the proposed MCLN architecture.

3 Multiple Context Learning Networks for VQA

As common practice, identifying a correct answer related to the given image-question pair from a set of candidate answers is a typical formulation of VQA:

\[
\hat{a} = \arg \max_{a \in A} p_\theta(a \mid I, Q) \tag{1}
\]

where the model predicts the correct answer \( \hat{a} \) from the candidate answers \( A \) for a given image \( I \) and question \( Q \) pair, and \( \theta \) is the learned parameters of model. Without loss of generality, the proposed method also follows this convention.

Figure 2 illustrate the overall pipeline of the proposed Multiple Context Learning Networks (MCLN). The proposed MCLN consists of a series of subnetworks: a) Image and Question Representation, which includes image encoder and text encoder. The image encoder detects objects in an image and extracts the object features, which text encoder embeds the question words and encodes the word features; b) Multiple Context Learning that performs multi-context learning and
it is composed of multiple visual and textual context learning layers (VTCL) by the cascaded way; and c) Multimodal Fusion and Answer Prediction that fuses image representation and question representation to predict correct answer.

![Overall flowchart of the MCLN that consists of three subnetworks: (a) Image and Question Representation; (b) Multiple Context Learning; (c) Multimodal Fusion and Answer Prediction.](image)

### 3.1 Image and Question Representation

#### 3.1.1 Image encoder

Inspired by humans using bottom-up attention to process visual information [13], we extract the features of salient objects instead of extracting the global convolutional features of the image. The proposed image encoder encapsulates three main steps: (i) a Faster R-CNN [22] model in conjunction with ResNet-101 [23], which is pre-trained on the Visual Genome dataset [24], to select $K$ most salient objects $O_i \in \mathbb{R}^{2048}$ within a given image $I$, thereby achieving the bottom-up attention. (ii) for each selected object $O_i$, $r_i$ is defined as the mean-pooled convolutional feature from this object region, such that the dimension of the object feature vectors is 2048; and (iii) a linear transformation layer to project the objects feature $r_i$ into a 768-dimensional vector.  

$$O_i = \text{Faster-RCNN}(I), i \in [1, K]$$  

$$r_i = W_i r_i + b_i$$  

Where $W_i \in \mathbb{R}^{2048 \times 768}$ and $b_i \in \mathbb{R}^{768}$ are the parameter of the linear layer. Finally, image $I$ is represented as visual object features set $V = \{v_i \in \mathbb{R}^{768}\}_{i=1}^K$.

#### 3.1.2 Text encoder

Given a question $Q$, the text encoder first tokenizes it into words and uses a maximum of $L$ words to trim the question. Then, the $l$-th word in the question is further embedded into a 300-dimensional vector representation $e_l \in \mathbb{R}^{300}$ by using the pre-trained GloVe word embeddings [25]. The questions shorter than $L$ words are padded at the end with zero vectors. Thus, the $Q$ is initialized as a word embeddings sequence $E = \{e_l \in \mathbb{R}^{300}\}_{l=1}^L$. Here, a one-layer and 768-dimensional long short-term memory networks (LSTM) [7] is utilized as text feature extractor, which scans the word
embeddings sequence $E$ from $e_1$ to $e_L$ and picks up current $e_i$ into its unit to iteratively update hidden state $t_i$.

$$(t_1, ..., t_i, ..., t_L) = LSTM(e_1, ..., e_i, ..., e_L) \quad (3)$$

In contrast to [13] which only uses the final hidden state as the question feature, the hidden state of all words will be preserved and regarded as word features. Finally, the question $Q$ is represented as word features set $T = \{t_i \in \mathbb{R}^{768} \}_{i=1}^{L}$. In addition, motivated by success of BERT in multiple natural semantic tasks [21], we also introduce the BERT model to replace LSTM and fine-tuning it to enhance the context learning of textual features.

### 3.2 Multiple context learning layers

Given that the goal of proposed model is to execute multi-context learning using a uniform strategy, how to capture context of an entity in the entities set (i.e., image object features set $V$ or question word features set $T$) is essential. Such a method should be capable of modeling context dependence of all entities across whole entities set, it requires that the contextual path dependency distance between all entities is 1 and not affected by the data pattern. Bearing in mind the keys, this paper proposes a unified multi context learning strategy based on the key-query attention mechanism [26]. Using visual context learning as an example, Figure 3 shows the design of the visual context learning module (VCL), and the implementation will be discussed in detail below.

![Figure 3. The detailed design of proposed visual context learning module.](image)

For obtained image object features set $V$, which can be denoted as a features matrix $V \in \mathbb{R}^{K \times 768}$ and its $i$-th row represents the feature of $i$-th object. To learn the visual context features, the matrix $V$ would be converted to key matrix $K_V$, query matrix $Q_V$ and value matrix $V_V$ by the linear transformation.

$$K_V = VW^k, Q_V = VW^q, V_V = VW^v \quad (4)$$

where $W^k$, $W^q$ and $W^v$ are the linear transformation parameters that calculate the key, query and value matrices. Then, calculating the dot products between query and key matrices to get the
attention weights matrix $A^V$. 

$$A^V = \text{softmax} \left( \frac{Q^V (K^V)^T}{\sqrt{d}} \right)$$

where $\sqrt{d}$ is a normalization constant. In this way of dot product, the contextual path dependency distance between all objects is set to 1. So $A^V \in \mathbb{R}^{k \times k}$ reflects the context dependence of all objects and stores the attention weights between all objects. For the row attention weights $A^V_i$, which represents the context dependence of $i$-th object in the image. Therefore, the visual context feature for $i$-th object can be obtained by calculating a weighted combination of the $A^V_i$ and all object features, whereas the form of matrices for the visual context feature of all objects can be stated as follows:

$$\text{ATT}(V) = \text{softmax} \left( \frac{Q^V (K^V)^T}{\sqrt{d}} \right) V^V$$

In addition, the multi-head attention based on key-value attention is introduced to perform on $V$.

$$\text{MultiHead}(V) = (\text{head}_1 \parallel \ldots \parallel \text{head}_h) W_t^o$$

$$\text{head}_h = \text{ATT}(V) = \text{softmax} \left( \frac{(V W_h^q)^T (V W_h^k)^T}{\sqrt{d_h}} \right) (V W_h^v)$$

$\text{head}_h$ is $h$-th head key-query attention. $W_h^K \in \mathbb{R}^{768 \times d_h}$, $W_h^Q \in \mathbb{R}^{768 \times d_h}$ and $W_h^V \in \mathbb{R}^{768 \times d_h}$ are the parameter matrices of $\text{head}_h$. The $W_t^o$ is the projection matrix for all heads and $\parallel$ represents concatenation of all heads. $d_h$ is the dimensionality of the output features from each head and usually making $d_h = 768/H$. The multi-head attention allows the model to jointly attend to context information from different representation subspaces, improving the representation capacity of features.

After acquiring the visual context features of all objects using multi-head attention, residual connection followed by layer normalization is applied to integrate the visual context into the object features.

$$\hat{V} = \text{LN}(V + \text{MultiHead}(V))$$

Where $\text{LN} (\cdot)$ represents layer normalization. Aiming to further adjust the object representations, the position wise feed-forward network (FFN) transforms object features $\hat{V}$ with two fully-connected layers. And it can be described as:

$$\text{FFN}(\hat{V}) = (\text{ReLU}(\hat{V} W_1 + b_1)) W_2 + b_2$$

Where the $W_1$, $b_1$, $W_2$ and $b_2$ is the parameters of FFN, and its hidden and output dimensions are $768 \times 4$ and $768$ respectively. ReLu is the rectified linear unit. In addition, the residual connection and layer normalization are also applied after FFN to facilitate optimization.

By performing the above procedure, visual contexts representing high-order interaction and context dependence among all objects can be captured in an image. Such procedure is simplified by the visual context learning module (VCL) as:

$$V' = \text{VCL}(V)$$

The textual context learning module can work in the same manner on the word features $T$, but with a different set of parameters to be learned, thus details are omitted and it is also simplified as:
\[ T' = TCL(T) \] (11)

Since the contextual path dependency distance between all entities is 1 in the proposed context learning method that based on key-query attention, the inter-modal visual-textual context can be easily modeled by feeding the connected visual-textual matrix \( V' \| T' \in \mathbb{R}^{(K+M) \times 768} \) into the visual-textual context learning module (VTCL) with the same context learning mechanism.

\[ V_{out}, T_{out} = VTCL(V' \| T') \] (12)

Then, we combine three context learning modules to form a multiple context learning layer (MCL). Finally, such MCL layer is stacked in depth and the output of the last layer is the input of the next layer, expressed as \( V^N, T^N = MCL_N(V^{N-1}, T^{N-1}) \). So in the MCL layer, the intra-modal contexts will provide complementary information for inter-modal context learning, while the inter-modal context information will facilitate the intra-modal context learning for the next layer. In the way, the deeper and more high-level context information is modeled by associating the intra-modal contexts with the inter-modal context, allowing more comprehensive understanding for image and text contents.

### 3.3 Multimodal fusion and answer prediction

After \( N \) MCL layers, the output image object features \( V^N \) and question word features \( T^N \) not only contain the intra-modal contexts but also capture the inter-modal context dependencies. To distinguish important entities from context information and fuse multimodal features, an attention model with a FFN (its hidden and output dimensions are 768 and 1) is designed for \( V^N \) and \( T^N \) to obtain attended image feature \( v \) and question feature \( q \). Taking image object features \( V^N \) as an example, the attended image features \( v \) is calculated as follows:

\[ v = \sum_{i=1}^{K} \alpha_i v_i^N, \text{where } \alpha_i = \text{softmax}(\text{FFN}(v_i^N)) \] (13)

\( \alpha_i \) is the learned object attention weight for \( i \)-th object. The learned word attention weight \( \beta_q \) and the attended question feature \( q \) can be obtained using an independent attention model by analogy.

After obtaining image feature \( v \) and question feature \( q \), a linear transformation respectively implements on \( v \) and \( q \), and using the addition fusion to get joint representation \( z \). Finally, \( z \) is fed into a fully-connected layer followed by a sigmoid function, generating an answer vector \( p \in \mathbb{R}^{4|A|} \).

\[ z = LN(W_v^T v + W_q^T q) \]
\[ p = \text{sigmoid}(W_z z + b_z) \] (14)

where \( W_v^T \in \mathbb{R}^{1024 \times 768} \), \( W_q^T \in \mathbb{R}^{1024 \times 768} \) and \( W_z \in \mathbb{R}^{4|A| \times 1024} \) are three linear projection matrixes, \( b_z \) is the bias parameter. And sigmoid (\( \cdot \)) is used for classification, \( |A| \) is the number of candidate answers.
4 EXPERIMENTS

4.1 Datasets

VQA v2.0 dataset [5] contains 1.1M questions asked by human and 10 answers are collected for each image-question pair from human annotators. The answer with the highest number of occurrences will be regarded as the correct answer. The dataset is divided into three parts: a training set containing 80 thousand images and 444 thousand questions, a validation set containing 40 thousand images and 214 thousand questions, and a test set containing 80 thousand images and 448 thousand questions. Additionally, based on the answer category, all questions are divided into three types: Yes/No, Number and Other. According to official recommendations, we use following accuracy as the evaluation metric for answering quality.

\[
\text{acc}(\text{answer}) = \min\left(\frac{\text{provided(annotators)}}{3}, 1\right) \quad (15)
\]

4.2 Implementation Details

The proposed MCLN is implemented by PyTorch and all the experiments are conducted on a workstation with NVIDIA 2080 Ti GPU. The hyper-parameters of MCLN model that are used in the experiments are listed as follows. For the pre-trained Faster R-CNN, setting a confidence threshold to the probabilities of detected objects and obtain a dynamic number of objects \(K \in [10, 100]\). The maximum length of tokenized words is \(L=14\). As for textual features, using either a LSTM with a 300-dimensional GloVe word embedding size or a pre-trained BERT [21] model with the 768-dimensional embedding size. In all context learning modules, we set the number of heads \(H = 12\), so the latent dimensionality for each head is \(d_h = d/12 = 64\). Moreover, the number of MCL layers is \(N \in \{1, 2, 3, 4\}\). In this work, the answers that occur less than 8 times in the training and validation dataset are discarded, which produces \(|A|=3129\) candidate answers.

Our model is optimized by the Adam optimizer [27] with 64 batch sizes. For the choice of learning rate \(lr\), the warm-up strategy is employed. Specifically, the initial learning rate is 2.5e-5 and grows by 2.5e-5 at each epoch till it reaches 1e-4 at epoch 4. After 10 epochs, the learning rate is decreased by 1/5 for every epoch up to 12 epochs. Due to the fact that there might exist multiple correct answers for a question in the VQA v2.0 dataset, hereby we utilize the binary cross-entropy loss (BCE) as the loss function to optimize MCLN:

\[
\text{loss} = -\sum_{i=1}^{\text{|A|}} (y_i \log(p_i) + (1-y_i)\log((1-p_i))) \quad (16)
\]

Where \(y_i\) is the provided soft score, \(p_i\) is the predicted score by the MCLN model and corresponds to the \(i\)-th element in the answer vector \(p\).
4.3 Ablation Studies

On the VQA v2.0 validation dataset, the proposed MCLN will be evaluated from three different aspects: 1) the effectiveness of the three context learning modules in the proposed MCL network architecture; 2) the impact of different layers on the performance of MCLN; 3) the effect of BERT for MCLN.

4.3.1 Effect of the Context learning module

As mentioned, we design three modules to perform corresponding context learning. To demonstrate the effectiveness of the proposed context learning modules, we use the MCLN model with LSTM and quantitatively ablate the context modules to exercise this ablation study. The results of different ablated versions of model are reported in Table 1. It can be found that adding any module will help to improve performance, while ablation of any module will hurt performance. And the best performance only occurs when the three modules are combined. Further noteworthy is the result about VTCL. The model 4 only with VTCL module obtains 62.07% overall accuracy, which outperforms model 1 more than 7.47% and higher performance is gained compared to model 2 and model 3. Furthermore, a dramatic damage, 9.92%, has emerged when VTCL module is ablated in model 7 compared with model 8, validating the visual-textual context that ignored by previous methods particularly matter to VQA. From these results, all context learning module plays a significant role in improving the VQA performance, proving the effectiveness of our context learning modules.

<table>
<thead>
<tr>
<th>model</th>
<th>module</th>
<th>All</th>
<th>Y/N</th>
<th>Num</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>without all</td>
<td>54.60</td>
<td>69.79</td>
<td>36.02</td>
<td>47.50</td>
</tr>
<tr>
<td>2</td>
<td>Only VCL</td>
<td>55.13</td>
<td>69.82</td>
<td>36.09</td>
<td>47.99</td>
</tr>
<tr>
<td>3</td>
<td>Only TCL</td>
<td>55.53</td>
<td>69.82</td>
<td>36.44</td>
<td>49.72</td>
</tr>
<tr>
<td>4</td>
<td>Only VTCL</td>
<td>62.07</td>
<td>79.79</td>
<td>42.67</td>
<td>53.73</td>
</tr>
<tr>
<td>5</td>
<td>TCL+VTCL</td>
<td>65.17</td>
<td>82.88</td>
<td>44.68</td>
<td>57.15</td>
</tr>
<tr>
<td>6</td>
<td>VCL+VTCL</td>
<td>62.72</td>
<td>79.33</td>
<td>43.29</td>
<td>55.23</td>
</tr>
<tr>
<td>7</td>
<td>VCL+TCL</td>
<td>55.76</td>
<td>71.01</td>
<td>36.37</td>
<td>49.29</td>
</tr>
<tr>
<td>8</td>
<td>Full modules</td>
<td>65.68</td>
<td>83.40</td>
<td>45.57</td>
<td>57.53</td>
</tr>
</tbody>
</table>

4.3.2 Effect of VTCL layers

Next, we stack the full-module MCL layer in depth to evaluate the effect of MCL layers. Figure 4 shows the performance of the MCLN models with different number of MCL layers \( N \in \{1, 2, 3, 4\} \). Regarding the performance, we observed two phenomena on the overall accuracy: 1) as increasing \( N \), the performances of MCLN models steadily improve and finally saturate at a certain number \( N = 3 \); 2) The one-layer model does not perform as well as the deeper models, but there is a quite improvement over \( N=1 \) while a subtle decrease over \( N=3 \) when \( N=4 \). Deeper MCL layers capture
more complex context information, enabling a more comprehensive understanding of image and question contents, which improves the performance. What fascinates me is why the performance starts to degrade after four layers. Comparing the shallow model and deep model, the number of layers increases and the parameters of model rise as well. As a result, the model with four MCL layers difficult to be optimized and suffer from a larger risk to over-fit the training set due to the higher number of parameters, thereby harming the performance.

![Figure 4](image)

**Figure 4.** The overall and per-type accuracies of the MCLN with different MCL layers.

### 4.3.3 Effect of BERT model

Since BERT [21] is trained on large text corpus and models the text context by stacking 12 Transformer blocks, it provides better generalization and representation for textual context features. So the contextualized text encoder based on BERT is introduced to enhance the textual context learning and the LSTM is replaced when processing the question. The results of fine-tuning the weight of BERT with different learning rate are shown in Table 2. In the case of MCL layer is one, by using the BERT with learning $lr \times 0.001$ (N=1, BERT), 65.21% overall accuracy is gained, and by increasing learning rate of BERT, it achieves the best performance at $lr \times 0.1$. With this learning rate, by increasing the MCL layers, the performance grows and also saturates at three MCL layers, proving that BERT is compatible with MCLN and conducive to enhancing text context learning.

<table>
<thead>
<tr>
<th>Model</th>
<th>$lr$</th>
<th>All</th>
<th>Y/N</th>
<th>Num</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>N=1, LSTM</td>
<td></td>
<td>65.93</td>
<td>84.22</td>
<td>46.24</td>
<td>57.22</td>
</tr>
<tr>
<td>N=1, BERT</td>
<td>0.001</td>
<td>65.21</td>
<td>82.61</td>
<td>45.74</td>
<td>57.13</td>
</tr>
<tr>
<td>N=1, BERT</td>
<td>0.01</td>
<td>66.12</td>
<td>84.08</td>
<td>45.97</td>
<td>57.80</td>
</tr>
<tr>
<td>N=1, BERT</td>
<td>0.1</td>
<td>66.61</td>
<td>85.09</td>
<td>46.14</td>
<td>57.98</td>
</tr>
<tr>
<td>N=2, BERT</td>
<td>0.1</td>
<td>67.52</td>
<td>85.18</td>
<td>49.09</td>
<td>58.97</td>
</tr>
<tr>
<td>N=3, BERT</td>
<td>0.1</td>
<td>67.86</td>
<td>85.35</td>
<td>49.75</td>
<td>59.27</td>
</tr>
<tr>
<td>N=4, BERT</td>
<td>0.1</td>
<td>67.80</td>
<td>85.73</td>
<td>49.94</td>
<td>58.94</td>
</tr>
</tbody>
</table>
4.4 Visualize Analysis

Assuming that, after processing the object and word features through multiple MCL layers, the keywords in a question and the relevant image objects related to the answer can be well distinguished by the multimodal fusion and answer prediction networks according to the multiple context information. Thus, to intuitively illustrate the effectiveness of MCL layers, we selected three models to visualize the learned attention weights by Eq (13): the MCLN-w/o without the MCL layer, the MCLN-1 with one-layer MCL and the MCLN-3 with three MCL layers. The top-2 object attention weights and all word attention weights are visualized in Figure 5. As can be seen from the visualization of attention weights, MCL contributes to focus on relevant objects and keywords. For the correctly predicted example by three models, although the MCLN-w/o can locate the elephants, the MCLN-1 and MCLN-3 can focus on the target objects more fine-grained and the target objects have higher attention weight value. The incorrectly predicted examples by the MCLN-w/o model are more complex and require a comprehensive understanding to the contexts of image and question. Due to the lack of MCL, MCLN-w/o is not able to accurately locate keywords and relevant objects according to the context information, resulting in wrong answers. For example, MCLN-w/o mainly pays attention to the “people” but ignores other keywords (e.g., “many” and “board”), which means that the context dependency of “people on board” cannot be captured. In addition, compared with MCLN-1, more reasonable attention weights are usually learned by the MCLN-3. As shown in the last two questions, MCLN-3 mainly attend to the keyword “people”, “on” and “board” in the second question, and the keyword “person”, “white” and “holding” in the third question. In other words, the deep-level textual contexts like “people on board” and “white person holding” are captured by MCLN-3.

![Visualizations of the learned attention weights. The colors ranging from clear to red on image objects or words denote the attention weights from 0 to 1.](image-url)
4.5 Comparative Experiments with State-of-the-art Models

The proposed model is compared with several state-of-the-art (SOTA) models to verify its advantage. Compared methods include three models without context learning (BUTD [13], MFH [28], Counter [29]), two models with visual context learning (ReGAT [15], v-AGCN [16]) and three models that only learning intra-modal contexts (DFAF [18], MCAN [19], MEDAN [20]). Table 3 shows the compared experiment results on the VQA v2.0 test-dev and test-std sets. We can see that the proposed MCLN can obtain higher performance than the state-of-the-art VQA methods. Compared with three methods that ignore the context learning, on the test-dev set, 4.94%, 1.5% and 2.17% overall accuracy are improved by the MCLN-LSTM. ReGAT [15] explores two types of visual object relations on the image and only models the visual context in the image modality, and which combines various relation learning models and the Counter model. By comparison, the single MCLN-LSTM is not only higher than ReGAT in test-std, but also has an advantage in the number of models. Compared with DFAF [18], who separately learns the inter-modal attention features of vision and text, the proposed MCLN-LSTM captures intra- and inter-modal interactions simultaneously for vision and text features through the visual-textual context learning module. As shown in Table 3, MCLN-LSTM is 0.04 and 0.29 points higher than DFAF on test-dev and test-std. By using the encoder-decoder architecture, recent MCAN [19] and MEDAN [20] achieve the state-of-the-art performance. Even though the overall accuracy of the proposed MCLN-LSTM model is lower than MCAN and MEDAN, the advantage of MCLN model is still obvious. When BERT is introduced as the text encoder, the MCLN-BERT model overperforms these two models on the overall accuracy. This suggests that MCLN is more general that can naturally employ advanced context learning methods.

<table>
<thead>
<tr>
<th>Model</th>
<th>Test-dev</th>
<th>Test-std</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUTD [13]</td>
<td>65.32</td>
<td>56.05</td>
</tr>
<tr>
<td>MFH [28]</td>
<td>68.09</td>
<td>58.97</td>
</tr>
<tr>
<td>Counter [29]</td>
<td>70.22</td>
<td>60.49</td>
</tr>
<tr>
<td>ReGAT [15]</td>
<td>70.27</td>
<td>60.33</td>
</tr>
<tr>
<td>MCAN [19]</td>
<td>70.63</td>
<td>60.72</td>
</tr>
<tr>
<td>MEDAN [20]</td>
<td>70.26</td>
<td>60.72</td>
</tr>
<tr>
<td>MCLN-LSTM</td>
<td>71.05</td>
<td>61.08</td>
</tr>
<tr>
<td>MCLN-BERT</td>
<td>71.48</td>
<td>71.48</td>
</tr>
</tbody>
</table>
5 Conclusion

This paper presents the Multiple Context Learning Networks (MCLN), a novel framework to model multiple context learnings for visual question answering. MCLN exploits three types of contexts: the visual context and textual context in the intra-modality, and the visual-textual context in the inter-modality, to learn a context-aware representation through the corresponding learning module. The core idea of the context learning module is to establish the contextual dependency of all entities in an entity set by using key-query attention mechanism. The proposed approach is simple but extremely effective. MCLN is better able to learn complex contexts and improve VQA performance by composing multiple context learning modules to create MCL layer and stacking such layers in depth. Furthermore, the results of the experiments show that the BERT model is compatible with proposed MCLN. Extensive ablation studies have shown that the proposed method is effective.

References


Acknowledgments
This work was supported in part by the National Natural Science Foundation of China under Grant 61762046 and the Natural Science Foundation of Jiangxi Province under Grant 20161BAB212048.

Author contributions
Hong Lan led and supervised the research. Pufen Zhang suggested the idea. Pufen Zhang finished the experiment, wrote the first draft of the paper and obtained the graphical and tables results. All authors discussed the numerical results and contributed to manuscript completion.

Competing interests
The authors declare no competing interests.

Additional information
Correspondence and requests for materials should be addressed to Hong Lan.