Visual Question Answering: Datasets, Methods, Challenges and Oppurtunities

Shayan Hassantabar
Princeton University
seyedh@princeton.edu

Abstract

Recent advances in machine perception motivated the researchers to tackle more complex problems in this area. One of the most challenging and interesting tasks that recently attracted the attention of researchers from various fields of AI is the Visual Question Answering*, or VQA. In this task, the input is an image and an open-ended question about the image, and the output is an open-ended answer to the question with respect to the image. In order to achieve good results in this task, the expertise in various fields such as computer vision and natural language processing is needed. Various datasets have been collected for QA task. In addition, several methods have been proposed in the literature in the past recent years. In this survey, we explain the most famous datasets, as well as the state-of-the-art methods for VQA task. We compare the results of different methods, and identify the promising approaches for future research in this area. Identifying the shortcomings of current methods, as well as the future opportunities for VQA are also parts of this survey.*

1. Introduction

In the recent years, we have witnessed a significant progress in various fields of AI, such as computer vision, as well as language understanding. These progresses motivated researches to address a more challenging problem: visual question answering. This problem, combines both aforementioned fields of AI, image understanding as well as language understanding. This task attracted attention from machine learning and vision communities greatly. Essentially, this task is defined as follows: an image along with a question about that image are the input to the AI system, and the intelligent system is supposed to output a correct answer to the question with respect the input image.

Since this task is more open and more complex as compared to traditional machine perception tasks, such as classification and segmentation, this introduced a number of new challenges. As a result, this been a motivation for researchers in the related areas to approach this problem from various perspectives. These efforts include collecting curated datasets, as well as designing various methods based on different approaches. The authors of [14] identified a number of challenges which arose in the VQA task. The challenges identified in this paper are categorized into three different sections.The first category, vision and language, deals with scalability of the solution, dealing with inherent concept ambiguity, and handling attributes of the objects. The second category of the challenges has to do with how to use common sense knowledge in answering the questions. The third challenge comes from defining a benchmark dataset and quantifying the performance of different methods.

In this survey, we look at a list of papers which addressed some of these challenges. We go through the most well known datasets for the VQA task. We explain DAQUAR as the first dataset for the VQA, and the VQA 1.0 dataset as the most used and well designed dataset in this area. We explain the shortcomings of the VQA dataset which motivated the design of VQA 2.0. Moreover, we also explain a number of other dataset for VQA task, such as Visual Madlibs, Visual 7W, and CLEVR.

In addition, we also investigate various network architectures which aimed to tackle this challenge. We consider the methods in three different categories: Non-attention approaches, modular networks, and attention-based models. We compare the results from different methods, and show that the results from dense co-attention network (DCN) [16] and ReasonNet [8] are better than those of other methods. We also identify the shortcoming of current methods such as answering questions that require long chain of reasoning, or the ones that require short-term memory such as object counting. We also discuss some of the future opportunities for VQA.

The remainder of the paper is organized as follows. Section 2 explains the most famous datasets for VQA. Section 3 covers some of the most recent methods in this area, as well

as categorizing those solutions. In section 4, we discuss the results from current methods, and we identify some possible future directions of VQA task. Finally, section 5 concludes the paper.

2. Datasets

2.1. DAQUAR

The authors of [13] collected the first large dataset for the visual question answering task. This dataset, which is called DAQUAR, is based on real-world images, and is built on top of the NYU-Depth V2 dataset. It contains 6794 training, and 5674 test question-answer pairs. The question-answer pairs are of two types: *synthetic* and *human*. The synthetic question-answer pairs are based on a few templates. In addition, human question-answer pairs were collected using 5 human subjects.

2.2. VQA 1.0

One the most important datasets for the task of visual question answering is the VQA [2] dataset. This dataset is consisted of 2 parts. The first part includes $123,287$ train and validation images, and $81,434$ test images from MS COCO dataset [11]. The second part of the VQA dataset contains $50k$ abstract scenes. These abstract scenes are added to the dataset to attract the researchers who are interested in high level reasoning required by this task, and not necessarily the low level details of the vision tasks. Fig 1 shows an example of two images in this dataset. For each image, three questions were gathered by asking human subjects. In addition, 10 different subjects provided answers to those questions. Moreover, for each question, 18 candidate responses were created for the task of multiple choice VQA.

The main problem with the VQA dataset was in its inherent bias. Although the authors of [2] showed that the models perform much better in answering the questions when given the image, language priors had a very significant effect on the answers of the questions in VQA dataset. In fact, the simple baseline of always predicting "yes" as the answer was achieving the accuracy of 70.81% on VQA dataset for the "yes/no" questions. In addition, the baseline method which only used a LSTM representation for the question, and did not have the image as input, achieved the overall accuracy of 48.76%. This bias in the dataset, and the huge impact of the language priors on the answers motivated the design of the second version of the VQA dataset.

2.3. VQA 2.0

VAQ 2.0 [6] has been created to address the problem mentioned above. As the results showed, turned out that the

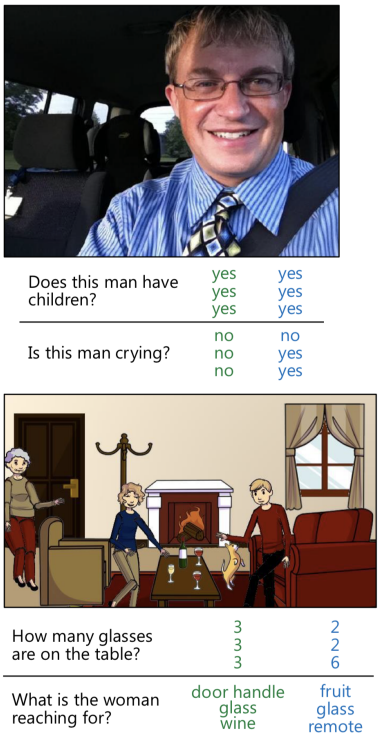

Figure 1. An example of the real images and abstract scenes in VQA dataset, alongside the question-answer pairs. The green answers were given by looking at the image, while the blue answers were given without looking at the image [2].

models were ignoring most of the visual information. In VQA 2.0, for each (I, Q, A), where the three tuple are the image, question, and answer, another image I' were found, where question Q makes sense for the image I', however the answer for the question will be something different, A'. Understandably, for some of the images finding this counter example image-question pair was not possible. However, by adding these new images to the dataset, the imapct of language priors on the models were mitigated to a great degree, as compared to first version of VQA dataset. For example, always answering "yes" to yes/no questions would result in accuracy of 61.20% in this case.

2.4. Visual Madlibs

There are several other datasets for the VQA tasks as well. Visual Madlibs dataset [20] has been collected using fill-in-the-blank templates, which aimed to collect a wide range of descriptions for the visual content in the image. Fig 2 shows an example of an image from Visual Madlibs dataset, alongside its thorough descriptions. The descriptions in this dataset go beyond just the objects that are present in the image, and are basically more detailed than a generic description of the image as a whole. This

dataset also contains a multiple choice question answering task for the images. In order to collect this dataset, a subset of 10738 images from MS COCO dataset, which were human-centric, were used.

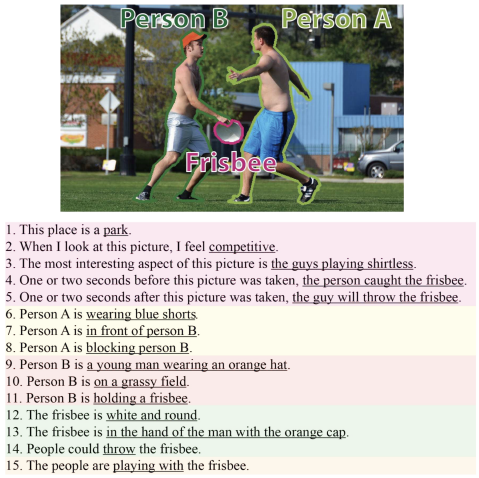

Figure 2. An example of the image from Visual Madlibs dataset [20].

2.5. Visual7W

Visual7W [21] is a QA dataset which has dense annotations and localization of objects in the image. Fig 3 shows an example of images in this dataset alongside other information provided. The visual7W dataset contains seven types of questions: *what*, *where*, *when*, *who*, *why*, *how*, and *which*. As compared to VQA, the questions in this dataset are richer, and the answers are longer in average. In this dataset, to obtain the object groundings in the image, AMT workers were asked to draw these bounding boxes. In total, there are $561,459$ object groundings in this dataset. Furthermore, to make the QA pairs make diverse, this dataset does not contain binary questions.

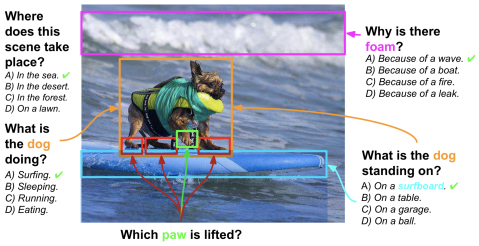

Figure 3. An example of the image from Visual7W [21].

2.6. CLEVR

CLEVR [9] has been created with the aim of better understanding the visual reasoning capabilities of VQA methods. In order to achieve this goal, the authors decided that it is better to use synthetic images, and to use automatically generated questions. The ground-truth of object locations, as well as their attributes are alongside each image. The images in the dataset are consisted of three shapes (cube, sphere, and cylinder), of two different sizes, two different materials, and in 8 different colors. Scenes in each image are represented as a collection of objects, their attributes, and their positions relative to each other. Images are generated by sampling the scene graph space (consisted of all possible combinations of the object in the image). In addition, questions are in the form of functional programs that can be executed on the image's scene graph. Furthermore, to address the problem of choosing the best functional programs to consider as questions, the authors used question families. Question families contain templates to construct functional programs. In addition, they also come with a way of expressing the functional programs in natural language. Figure 4 shows an example of a synthetic image in this dataset.

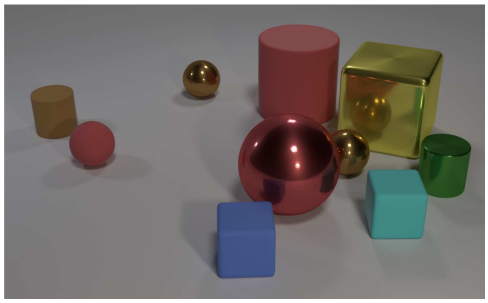

Figure 4. An example of the image from CLEVR [9].

3. Methods

In this section we go through several recent architectures that have been proposed for the VQA task. We categorize these proposal into three main sections: non-attention models, modular networks, and attention-based models. In the following, we discuss each section in more details.

3.1. Non-attention Models

3.1.1 Norm I+ deeper LSTM

One of the first architectures that have been proposed in the VQA paper [2] is shown in fig 5. In this architecture, the image embedding is done using the ℓ_2 normalized activations from last layer of VGGNet, which leads to 4096-dim image embedding. The question is embedded using an LSTM with two hidden layers. The image and question embeddings are combined to get a single embedding, and then passed to an MLP. The results for this architecture showed an overall accuracy of 57.75% on open-eneded answers challenge of VQA dataset. Note that the human

accuracy for the same challenge is 83.30%, which means a great room for improvement.

3.1.2 Learning by Asking Questions (LBA)

In this work [15], the authors investigate the problem of active learning in the context of VQA. In the learning by asking setting, the learner has a limited budget of the number of questions that can ask, and there is an oracle to answer the questions asked by the learner. Therefore, the main difference of this problem with standard VQA training is that most questions are not observed in training time, and the learner must decide which question to ask about the image. The LBA architecture is consisted of three parts: question proposal module, question answering module, and question selection module. Question proposal module generates a diverse set of questions, and it is consisted of two sub-components: question generation model, and question relevance model. The former has to do with generation of a question, while the latter filters the irrelevant questions. The question selection module aims to select the most informative question, using the current state of the question answering module (current accuracy), as well as the difficulty of each of the question proposals. This module keeps track of how fast the answering module is improving (the improvement in accuracy). The training is divided into three parts: initialization phase, online learning by asking phase, and an offline phase. In the initialization phase the modules are pre-trained on a small bootstrap set. The offline training is for evaluating the quality of the generated questions. The model is trained on the union of bootstrap set, and the set of questions generated in the LBA online phase. This method has been evaluated on CLEVR dataset. The results showed the effectiveness of the questions generated in the LBA phase.

3.2. Modular Networks

There are a few approaches that used modular design of the network to address the VQA challenge.

3.2.1 Neural Module Networks (NMN)

The authors of [1] proposed an architecture that is constructed using several pre-defined modules, which will be jointly trained for the VQA task. In this design, question is analyzed with a semantic parser with the aim of determining the basic computational units that are needed. This is the mapping from questions to *layouts*, which specifies both the modules needed to answer the question and the connections between them. Fig 7 gives an example of layout designed by this model. The final model combines the result from neural module network with a LSTM question encoder. In addition to the neural module network, this paper also collected a synthetic dataset, SHAPES, which includes different shapes in various positions. Fig 6 shows an example of an image in this dataset, and a layout to answer a particular question. The NMN approach achieved accuracy of 90.8% on this dataset. In addition, NMN was also trained and tested on VQA dataset, which resulted in overall accuracy of 58.7%.

3.2.2 End-to-End Module Networks (N2NMN)

One of the problems with the NMN is the fact that it uses off-the shelf parsers. In addition, NMN does not try to learn the needed modules. End-to-end module networks (N2NMN) [7] aims to address these problems by both learning to parse the language into linguistic structures, and composition into proper layouts. In N2NMN, the first part is the predection of layout policy using a deep representation of the question. This will result in a modular neural network, alongside a series of attentive actions which extracts parameters for these neural modules. It is worth mentioning that the functionality of each module is not restricted based on its name, such as *find* or *describe*, and these modules are just functions with a set of parameters. The main difference between this work and NMN is in the textual component. While hard coded textual components are used in NMN, such as `describe` *["shape"]*, in N2NMN, the textual components are extracted using soft attention over question words. The textual component for module m, $x_{txt}^{(m)}$, is obtained using following formula:

$$x_{txt}^{(m)} = \sum_{i=1}^{T} \alpha_i^{(m)} \omega_i$$

which essentially means predicting an attention map $\alpha_i^{(m)}$ over the T question words. Then, multiply the values by ω_i, which is the word embedding vector for word i in the question, and sum the values to compute the textual component for module m.

The prediction of layout to answer a particular question is done by predicting a probability distribution over the space of all possible layouts. To do so, the authors formulized the layout prediction problem as a sequence-to-sequence learning problem from questions to modules. The problem is solved using attention recurrent neural network. During training, this method jointly trains the layout policy as well as parameters in each neural module. N2NMN is evaluated on VQA, and in the best case, it achieved the accuracy of 64.9%.

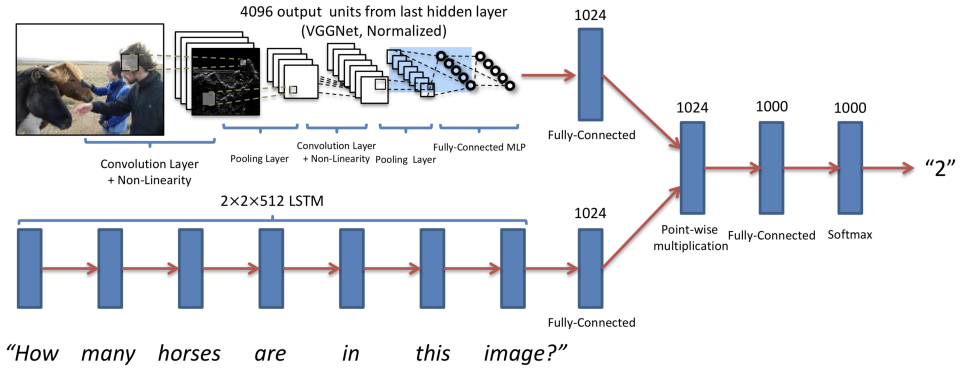

Figure 5. The architecture proposed in VQA paper [2].

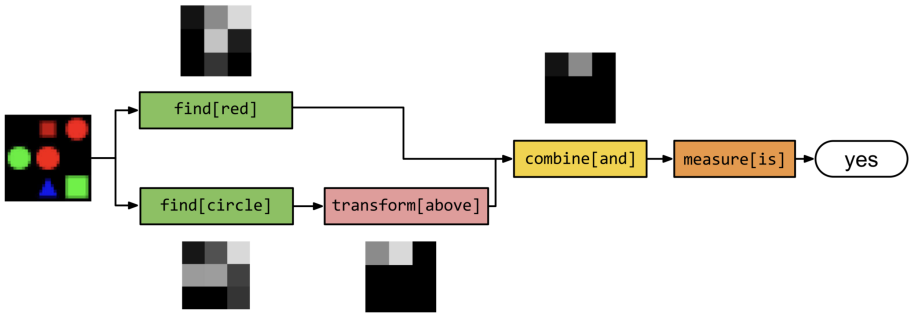

Figure 6. An example of an image in SHAPES dataset, and a layout to answer *Is there a red shape above a circle?* [1].

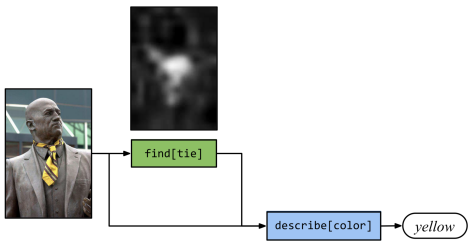

Figure 7. An example of a layout designed by NMN model [1].

3.2.3 Multimodal Learning and Reasoning for VQA (ReasonNet)

This work [8] proposes a modular design, which is the multimodal and multifaceted representation of the question image pair. ReasonNet is consisted of a number of modules: question-specific visual attention module, object-specific visual attention module, face-specific visual attention module, object-classification module, scene classification module, and face analysis classification module. In ReasonNet, the question representation is used to learn an attention probability distribution over the visual feature tensor. Furthermore, ReasonNet uses a fully convolutional network (FCN) for object detection. The output of the object detection module is a set of bounding boxes. ReasonNet uses residual network to classify the image inside each bounding box. In order to obtain the multimodal understanding, this method learns the interaction of each representation. This is formulated as below:

$$g = ||_{r_h \in H}(r_h^T W_h^s r^q + b_h^s) \qquad H = \{r^v, r^o, r^c, r^s, r^f, r^a\}$$

Note that W_h^s is a learned bi-linear tensor, H is the set of representations from various modules that have been mentioned above, and $||$ shows the concatenation of vectors. ReasonNet achieves accuracy of 67.9% on VQA dataset, and 64.61% on the VQA 2.0 dataset.

3.3. Attention-based Models

There are a number of attention-based architectures in the literature as well. These methods are based on generating spatial maps to highlight image regions that are relevant to answering the question. This is shown in figure 8. In the following we discuss some of these methods.

3.3.1 Where to Look

The authors in this paper [17] aim to combine the text features and the image features in order to find the relevant

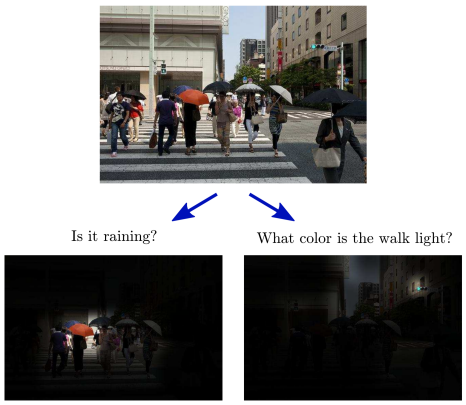

Figure 8. An example that shows the goal of attention-based methods [17].

regions of the image. To do so, there is a *region selection layer* in the architecture. In this part of the system, the first step is to find the relevance. To do so, this layer first projects the image features and the text features into a shared space. Subsequently, inner product is computed for each question-answer pair and all regions. With a feed-forward pass, the relevance weighting for each region is computed. The authors used 100 regions per image, which means 100 region weights for a question-answer pair. The region selection layer directly concatenates the text features with the image features for each region. This will lead to 100 different feature vectors. Then, the weighted average feature is computed, and passed through a 2-layer MLP. Figure 9 shows an overview of this architecture. For the language representation, Stanford Parser [4] is used to bin the question into semantic bins. There are 4 bins in total, which are related to type of question, subject of question, nouns in the question, and all the remaining words in the question. In addition, each bin contains 300-dimensional representation, which is the result of word2vec encoding. This is concatenated with a bin for the words in the candidate answer to produce a 1500 dimensional embedding for question-answer pair. Moreover, image features are fed directly into region selection layer. Overall, this method achieved the accuracy of 62.43% on VQA dataset.

3.3.2 Question-guided Spatial Attention (SMem-VQA)

Many of the questions in the VQA challenge can be answered using the information from various spatial regions and comparing their contents and locations. This motivated the authors of [19] to use a memory network with spatial attention for the VQA task. In this method, the the words in the question are used to compute attention over the visual memory. The visual memory contains extracted image features, which are the result of processing the image by pre-trained GoogleNet. The authors used bag-of-words question representation to guide the attention. To do so, the correlation matrix is computed as follows:

$$C = V.(S.W_A + b_A)^T$$

where V is word vector, and S is the visual features, and W_A contains the attention embedding weights. In addition, two different embeddings are used in this work: attention embedding W_{att}, and evidence embedding W_E. The spatial attention weights W_{att} are calculated by taking maximum over the word dimension T, of the correlation matrix C:

$$W_{att} = \text{softmax}(max_{i=1...T}(C_i))$$

The evidence embedding projects visual features to produce high activations for certain concepts. The selected visual evidence vector S_{att} is computed based on the W_{att} and W_E according to following formula:

$$S_{att} = W_{att}.(S.W_E + b_E)$$

To give the answer for a given question, and image pair, the sum of this evidence vector S_{att} and question embedding Q is used as follows:

$$P = \text{softmax}(W_P.f(S_{att} + Q) + b_P)$$

where P is the prediction and and f is the ReLU activation function.

SMem-VQA can also have a two-hop model to promote deeper inference. In the second hop, the final answer is predicted according to following formula, which shows the use of output from the first hop as an input feature to the second hop.

$$P = \text{softmax}(W_P.f(S_{att2} + O_{hop1}) + b_P)$$

SMem-VQA is evaluated on VQA dataset, and the two-hop model achived the accuracy of 57.99%.

3.3.3 Human attention in VQA

The authors of [3] aimed to answer an interesting question: how close are the attention maps generated from VQA models to human attention regions. Figure 10 shows an example of human attention regions. To answer the aforementioned question, the authors designed a set of game inspired interfaces to collect human attention maps. The basic approach is based on asking AMT workers to de-blur the related regions of the image. In total, they managed to collect human attention maps for 58475 train and 1374 val question-image pairs in the VQA dataset. In order to compare the spatial attention generated by models to the ones generated by human, first, the authors scale both maps to same size, rank

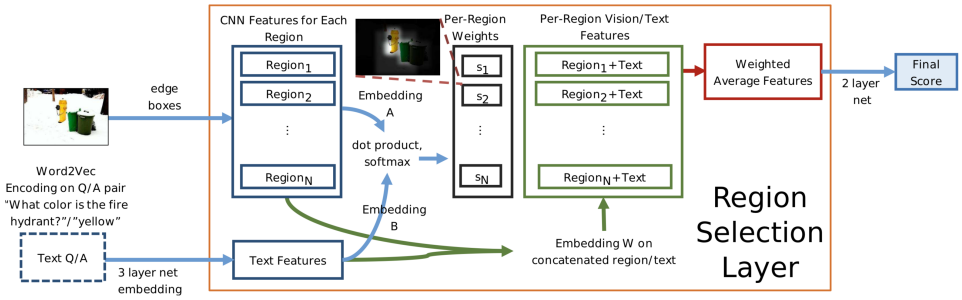

Figure 9. The region selection layer proposed in [17].

pixels based on their spatial attention, and compute the correlation between these two ranked lists. The results showed that the correlation between humans is around 0.63. Based on their result, among various models, the one proposed in [12] shows the most correlation to human attention, which was around 0.26. This showed there is still a huge room for improvement of spatial attention generated by VQA models.

Figure 10. An example of human attention regions [3].

3.3.4 Hierarchical Question-Image Co-Attention

The authors of [12] argued the importance of *question attention* alongside *visual attention*. This paper proposes an architecture which aims to jointly reason about visual attention as well as question attention. The proposed solution is a hierarchical architecture which co-attends the question and the image in three different levels: word level, phrase level, and question level. This means that for each question, its word level, phrase level, and question level embeddings are extracted. Subsequently, at each level, co-attention is applied to both image and question. The authors proposed two co-attention mechanism: parallel co-attention, where question and image attentions are generated simultaneously, as well as alternating co-attention , which refers to sequentially alternating between image and question attentions. The proposed model is evaluated on VQA and COCO-QA datasets. The results showed the accuracy of 62.1% on the open-ended answers for the VQA dataset.

3.3.5 A Strong Baseline for VQA

The authors of [10] investigated different combination of hyper-parameters in an attention-based VQA method to set the best baseline based on common approach in this area. These different configurations are different word embedding size, LSTM state size, attention size, classifier size, to name but a few. Although the architecture here is not novel, but with optimizing the configuration, the authors managed to set a strong baseline, which showed the state-of-the-art at the time. The accuracy of the model proposed in this paper on the VQA dataset is 64.6%, which showed 0.4% improvement over state-of-the-art at the time. Furthermore, the accuracy achieved on VAQ 2.0 dataset is 59.67%, which showed 0.5% improvement over the best previously reported results.

3.3.6 Dense Symmetric Co-Attention

The authors of [16] propose the *dense co-Attention network* architecture, or DCN. Figure 11 shows the golabl structure of DCN. This architecture is consisted of stack of dense co-attention layers, which fuses the language and visual features repeatedly. This is followed by an answer prediction layer. The questions and answers are encoded using a bi-directional LSTM. Visual features are extracted from a pre-trained ResNet. Specifically, the visual features are extracted from the outputs of four conv layers before the last four pooling layers. The main part of DCN is the dense co-attention layer. Essentially, this layer takes the image and question embedding as input, and outputs the updated versions (as shown in figure 11). The co-attention mechanism is dense in a sense that it considers every interaction between any word and any region. In DCN, one attention map is created on regions per each word, and one attention map is created on words per each region. Multiplicative attention is used to obtain attended feature representations of the

Table 1. Summary of the best case accuracy results on VQA dataset

Method	Accuracy
Norm I + deeper LSTM	57.75%
NMN	58.7%
N2NMN	64.9%
ReasonNet	67.9%
Wehre to Look	62.43%
SMem-VQA	57.99%
HieCoAtt	62.1%
Strong Baseline	64.50%
DCN	66.88%

question and image, $\hat{Q}_l$ and $\hat{V}_l$. Subsequently, the image and question representation are fused using the following formula:

$$q_{(l+1)n} = \text{ReLU}(W_{Q_l} \begin{bmatrix} q_{ln} \\ \hat{v}_{ln} \end{bmatrix} + b_{Q_l}) + q_{ln}$$

$$v_{(l+1)t} = \text{ReLU}(W_{V_l} \begin{bmatrix} v_{lt} \\ \hat{q}_{lt} \end{bmatrix} + b_{V_l}) + v_{lt}$$

Where q_{ln} is the layer l representation of $n-th$ question word. v_{lt} is defined similarly for the $t-th$ image region. Ultimately, the answer prediction layer predicts the answer based on V_L and Q_L, which are the output of last dense co-attention layer. This method has been evaluated on both VQA and VQA 2.0, and achieved the accuracies of 66.88% and 66.87%, respectively.

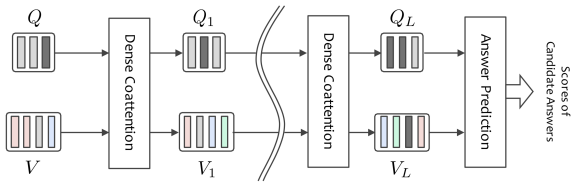

Figure 11. Overview of the DCN architecture [16].

4. Discussion and Future Directions

4.1. Comparison

As the results explained earlier shows, the current state-of-the-art on VQA is the ReasonNet architecture [8], and on VQA 2.0 is the DCN [16]. Generally, the co-attention approaches seem to be promising in the future. The quality of the results generated by these models depend on the fact that how dense is the co-attention between image and question, as well as the fusion method between question and image representations. By improving on these two fronts, DCN achieved better results compared to those of [12]. Table 1 shows the summary of the results on the VQA dataset.

4.2. Shortcomings of Current Methods

Despite all the improvements that we have seen in the recent years, most of these methods suffer from some possible shortcomings. The first problem of current methods is to answer a question which requires a long chain of reasoning. Furthermore, all these system seem to have difficulty in answering questions which require short-term memory such as integer equality questions. Questions about counting the number of specific object in the image is another example which can be very challenging for the most of the models that we discussed in this paper. Recently, there have been efforts to address these challenges as well. The authors of [18] formulate the counting as a sequential decision process, and solved it using reinforcement learning approach. Furthermore, this approach also identifies the objects that contribute to each count. The future models can improve upon current methods, by building on the existing methods such as co-attention or modular networks, and also addressing the challenges mentioned here, maybe by using a solution tailored to these challenges.

One of the interesting approaches discussed here is the LBA [15] method. Although this approach seem to be promising, the problem with this work is the fact that it only uses synthetic images in CLEVR dataset. Developing a real-world version of LBA, by replacing the synthetic images by real-world scenes can be an interesing future work.

4.3. Future Opportunities for VQA

VQA task can be a part of an agent which is in an interactive environment, and should answer the questions from humans. For example, the agent can be presented with the question "Are there any apples in the fridge?". The *Interactive Question Answering* proposed in [5] is a challenging problem. In this task, the agent should be able to navigate through the environment, acquire understanding of the environment, interact with the environment, and be able to plan and execute a series of actions. A visual question answering system can be part of an intelligent agent which is designed to address the interactive question answering challenge. The authors of [5] collected an interactive question answering dataset, which is based on a simulated environment. This dataset can facilitate future research in the area of interactive question answering.

5. Conclusion

VQA is an interesting and challenging task for the researchers. In this survey, we looked at various datasets, and several methods for this task. We compared the results, and identified a number future work that can be done in this area.

References

[1] J. Andreas, M. Rohrbach, T. Darrell, and D. Klein. Neural module networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 39–48, 2016.

[2] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2425–2433, 2015.

[3] A. Das, H. Agrawal, L. Zitnick, D. Parikh, and D. Batra. Human attention in visual question answering: Do humans and deep networks look at the same regions? *Computer Vision and Image Understanding*, 163:90–100, 2017.

[4] M.-C. De Marneffe, B. MacCartney, C. D. Manning, et al. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC*, volume 6, pages 449–454. Genoa Italy, 2006.

[5] D. Gordon, A. Kembhavi, M. Rastegari, J. Redmon, D. Fox, and A. Farhadi. Iqa: Visual question answering in interactive environments. *arXiv preprint arXiv:1712.03316*, 2017.

[6] 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*, volume 1, page 9, 2017.

[7] R. Hu, J. Andreas, M. Rohrbach, T. Darrell, and K. Saenko. Learning to reason: End-to-end module networks for visual question answering. *CoRR, abs/1704.05526*, 3, 2017.

[8] I. Ilievski and J. Feng. Multimodal learning and reasoning for visual question answering. In *Advances in Neural Information Processing Systems*, pages 551–562, 2017.

[9] 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 *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 1988–1997. IEEE, 2017.

[10] V. Kazemi and A. Elqursh. Show, ask, attend, and answer: A strong baseline for visual question answering. *arXiv preprint arXiv:1704.03162*, 2017.

[11] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[12] J. Lu, J. Yang, D. Batra, and D. Parikh. Hierarchical question-image co-attention for visual question answering. In *Advances In Neural Information Processing Systems*, pages 289–297, 2016.

[13] M. Malinowski and M. Fritz. A multi-world approach to question answering about real-world scenes based on uncertain input. In *Advances in neural information processing systems*, pages 1682–1690, 2014.

[14] M. Malinowski and M. Fritz. Towards a visual turing challenge. *arXiv preprint arXiv:1410.8027*, 2014.

[15] I. Misra, R. Girshick, R. Fergus, M. Hebert, A. Gupta, and L. van der Maaten. Learning by asking questions. *arXiv preprint arXiv:1712.01238*, 2017.

[16] D.-K. Nguyen and T. Okatani. Improved fusion of visual and language representations by dense symmetric co-attention for visual question answering. *arXiv preprint arXiv:1804.00775*, 2018.

[17] K. J. Shih, S. Singh, and D. Hoiem. Where to look: Focus regions for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4613–4621, 2016.

[18] A. Trott, C. Xiong, and R. Socher. Interpretable counting for visual question answering. *arXiv preprint arXiv:1712.08697*, 2017.

[19] H. Xu and K. Saenko. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *European Conference on Computer Vision*, pages 451–466. Springer, 2016.

[20] L. Yu, E. Park, A. C. Berg, and T. L. Berg. Visual madlibs: Fill in the blank description generation and question answering. In *Computer Vision (ICCV), 2015 IEEE International Conference on*, pages 2461–2469. IEEE, 2015.

[21] Y. Zhu, O. Groth, M. Bernstein, and L. Fei-Fei. Visual7w: Grounded question answering in images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4995–5004, 2016.