Visual Question Answering (VQA)

Parth Shah (20759634), Priya Patel (20787652), Prateeti Deb Chaudhuri (20802005)

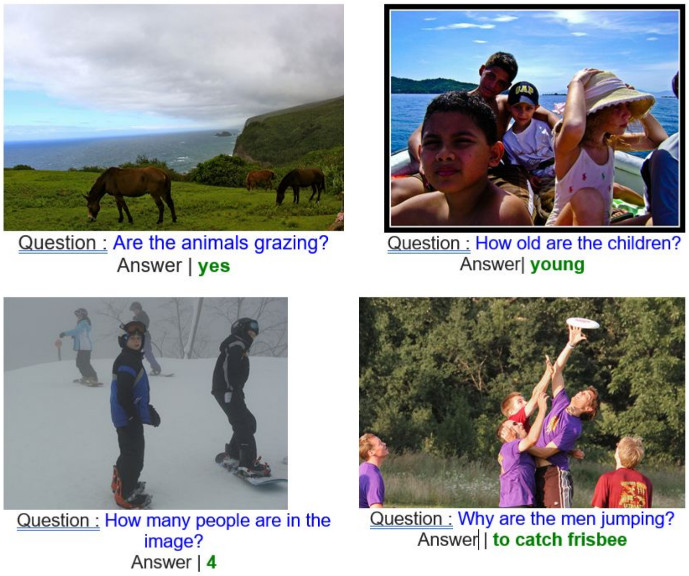
Abstract:

The Visual Question Answering (VQA) systems predict the answer of the question related to an image. We have performed Visual Question Answering on COCO-VQA dataset which has open-ended questions and open-ended answers associated with an image. We studied prior work related to VQA and built a base model which uses the VGG16 architecture for image and GloVe Embedding for question processing. We further improved upon the base model by using attention features of an image. It helps the algorithm to focus on the specific region of an image related to a question. Additionally, we explored the different hyperparameters and we compare our results with state-of-the-art models.

1. INTRODUCTION

Visual Question Answering is a rather important problem in computer vision and natural language processing. It uses algorithms which allows the user to ask text-based questions about images. A large set of AI capabilities are required to answer for open-ended questions like “What kind of cheese is there on the pizza?” which can be an example of fine-grained recognition or “What is the number of the car? which can be an example of object detection or “Is the person is laughing?” which can be an example of activity recognition or “Is this burger a vegetarian burger?” which can be an example of knowledge base reasoning or “Does this person have a 20/20 vision?”, “Is this person expecting some company?” which can be an example of common sense reasoning [2]. Potential applications for VQA can be varied but one of the most important application is to provide aid to blind or visually challenged people so that they can get an insight of the images on the web or social media. VQA is also a Visual Turing Test which rigorously evaluates a computer vision system to assess whether it is capable of human-level semantic analysis of images.

We chose to build a comparatively simple model proposed in the paper [2] to get the ball rolling & finds pitfalls and ways to improve performance. VQA requires two types of information generally (text and images). The images are given as input to the VQA system along with the questions regarding the images which are free-form, open-ended natural language questions. Then the model’s goal is to produce a natural language answer about the input. Then we turned our focus towards using attention network in the model. The aim of this project is to build a VQA model which will explore the previous methods of implementing VQA along with understanding the architecture and characteristics of a more successful network. This evaluation of the model against other VQA implementations is done using an evaluation metric which was used in the VQA Challenge.



Question : Are the animals grazing?
Answer | **yes**

Question : How old are the children?
Answer| **young**

Question : How many people are in the image?
Answer | **4**

Question : Why are the men jumping?
Answer| | **to catch frisbee**

Fig 1: Examples of training questions and their correct answer from the COCO-VQA dataset[12]

2. LITERATURE REVIEW

The paper was written by Aishwarya Agrawal et al. called VQA: Visual Question Answering [2] has provided the COCO-VQA dataset containing over 250K images, 760K questions, and around 10M answers. They have demonstrated a wide variety of questions and answers in their dataset, as well as the diverse set of AI capabilities in computer vision, natural language processing, and commonsense reasoning required to answer these questions accurately. The questions were human annotated and open-ended and not task-specific. For some application domains, it would be useful to collect task-specific questions. For instance, questions may be gathered from subjects who are visually impaired, or the questions could be focused on one specific domain (say sports).

They have explored the difficulty of the VQA dataset for the MS COCO images using several baselines and novel methods like - per Q-type prior, nearest neighbor (baselines) and for the methods they have used a two-channel vision (image) and language (question) model that culminates with a softmax over K possible outputs [2]. They have chosen the top K = 1000 most frequent answers as possible outputs.This answer set covers 82.67% of training and validation answers [2]. The

Image Channel provides an embedding for the image and Question Channel provides an embedding for the question. Two different types of embedding are used for Image Channel - I and norm I. And three different types of embedding have been used for the Question Chanel like Bag-of-Words Question (BoW Q), LSTM Q and deeper LSTM Q.

The image and question embeddings are combined to obtain a single embedding - Multi-Layer Perceptron (MLP). The accuracy of their best model (deeper LSTM Q + norm I, which was selected using VQA test-dev accuracies) is 58.16% for open-ended questions and 63.09% for multiple choice questions [2]. So, as a result, the accuracy of multiple choice questions are better than that for open-ended questions. However, all methods are significantly worse than human performance. They had also organized an annual challenge and workshop to facilitate systematic progress in this area - the VQA Challenge, the first instance of the which was will be held at CVPR 2016 [2].

The second paper, which we reviewed was written by Kushal and Christopher [3] focuses on critically reviewing the existing state of VQA in terms of problem formulation, existing datasets, evaluation metrics, and algorithms like DAQUAR, Visual 7W, FM -IQA, VQA, etc. They have enforced particular emphasis on exploring whether the current VQA benchmarks are suitable for evaluating if a system is capable of robust image understanding. The datasets explored in this paper are - DAQUAR, COCO-QA, The VQA Dataset, FM-IQA, Visual7W, and Visual Genome. For a dataset to be ideal, it has to be large enough to capture the variability among questions, images, and concepts that are occurring in real-world scenarios. So, if an algorithm can perform well on such a dataset then it can answer definitively for a large variety of questions [3]. However, if the dataset consists of easily exploitable biases then an algorithm can perform well on the dataset without actually solving the VQA problem. They have concluded after researching the various datasets that for evaluation of open-ended responses becomes simpler when the responses are one-word answers. This happens in 87% of COCO-VQA questions, 100% of COCO-QA questions and 90% of DAQUAR questions. However, when the responses are more than one word then the possibility of multiple correct answers arises. This has occurred in FM-IQA, Visual7W, and Visual Genome, e.g., 27% of Visual7W answers have three or more words [3].

They have also reported different types of techniques for VQA like Baseline models, attention modes, Bayesian models, etc. and compared their results. Particularly, it discusses the major problem that occurs when biases and other problems like multi-word answers to some questions affect these existing datasets. It concludes with the possible solution that VQA needs a dataset such that any VQA algorithm which will perform well on this dataset will do well in VQA in general.

The third and final paper which we reviewed was written by Peter Anderson et. al. on VQA called the Tips and Tricks for Visual Question Answering: Learnings from the 2017 Challenge [4], focuses on presenting a state of the art model for VQA. This paper has won the first prize in the 2017 VQA Challenge. It identifies the tips and tricks of successful performance of deep neural networks which is depending on the choices of architectures and hyperparameters, by exploring different choices representing more than 3000 GPU hours and using attention module architecture [4]. They have used image features from bottom-up attention that provide region-specific features. They have used a sigmoid output that allows multiple correct answers per question, instead of a common single-label softmax [4].

3. PROBLEM DEFINITION

The main goal of our system is classification. Predicting the answer given a question related to an image. It will use the combine features of input image and question and succeeded by the multinomial classifier which classifies into the answer. It will have three major modules, First, it will take different types of images as an input to CNN and map the different aspect of the image contents to a module-specific class representation space. Second, The LSTM or GRU module will generate the feature representation of the question asked for the image. In the end, The LSTM/GRU question representation and the CNN image features are fused via an element-wise multiplication, and then passed through fully-connected layers to generate a softmax distribution over output answer classes [2].

A. Dataset

We have used newly-released COCO-VQA dataset [12]. It consists of approximately 83K training images, 41K validation images, and 81K testing images of MS COCO dataset, this is best suited for our project as it provides a broad category of object. Each image contains 3 questions and each question has 10 answers answered by 10 different human annotators. We have used train-test-validation data split from [1] same as MS COCO dataset.

B. Questions

The COCO-VQA dataset have diverse range of questions. The questions are collected such that an image is required to answer about the question. They still do have questions that asks for commonsense reasoning to answer it. Majority of the question length is between 4 to 10 words [2].

The dataset contains open-ended task that results in a wide variety of answers. For binary questions answers just in “yes” or “no” is sufficient, on other hands, common reasoning question requires short phrases and multiple answers can also be correct. For example “what is color of the sky?” answer light-blue, sky-blue, blue all are correct [2]. we are evaluating predicted answers with the dataset's answers as it contains 10 different answers per each question collected from different human annotators.

C. Evaluation Matrix:

The annotators are generated for the VQA dataset as ten answers per question. These are used with a variation of the accuracy metric, which is given by Accuracy

$$VQA = min(n/3,\ 1)$$

where the total number of annotators is denoted by n which has the same answer as that of the algorithm. Using this metric, if the algorithm agrees with three or more annotators then it is awarded a full score for a question i.e. it is given 100% accuracy for that question.[2] Moreover, most of the answers are single word answers (89.32%).

4. METHODS AND APPROACH

We started by converting all the answers to the uniform format i.e. we removed the articles, punctuation marks and transformed words into digits (eg: first -> 1). We use transfer learning which is based on the idea the we transfer the knowledge gained while learning from one process to another related process that might take advantage of it. for example, if we have learned how to ride the bicycle: i.e. how to keep balance, how to keep moving then while learning to ride moped or motorcycle, we don't have again start from scratch and learn how to keep balance. we can just focus on new features of motorcycle. Same analogy is applied here. Then we created the simple base model to explore the result and set up the project. We iterated on it by implementing the attention model.

A. Base Model

We have started building the base model for the VQA Task. Here, we are using VGG16 model architecture and its pre-trained weights on ImageNet Classification Challenge for extracting the features from the Images.

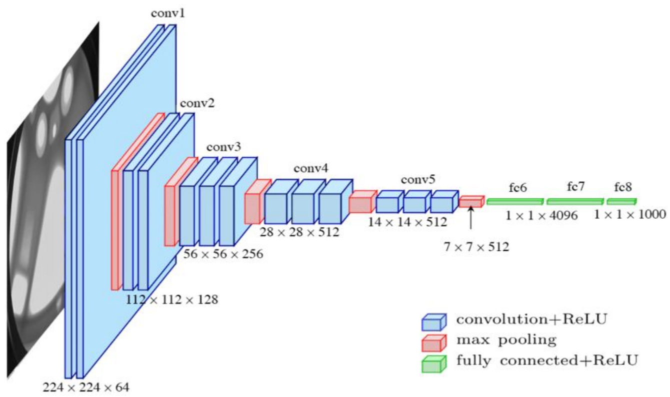


Fig 2: VGG16 Model Architecture [13]

The Fig 2 shows the VGG16 architecture. Its has 5 convolution blocks and total 16 layers. All the layers of VGG16 are frozen so that doesn't learn anything new as it is already pre-trained. Last 2 layers of the VGG16 model were removed as we want to extract the feature vector of the image only i.e. we only use the output of the last convolution-block. The last 2 layers are used to classify the image into 1000 objects and we don't want that. VGG16 model is used because it is small, versatile and simple to use. It has a top-1 error of 24.70% and top-5 error of 7.3 % [6].

Moreover, We can't process the text as it is. Firstly, We need to convert question into tokens of words and then tokens into the digits. The simplest approach is to convert all the words into the vector using one-hot encoding. However, it doesn't represent the complex semantics between words. Hence, we have used the pre-trained GloVe (Global Vector) embedding to process the text. It reduces a given token into a 300-dimensional representation.

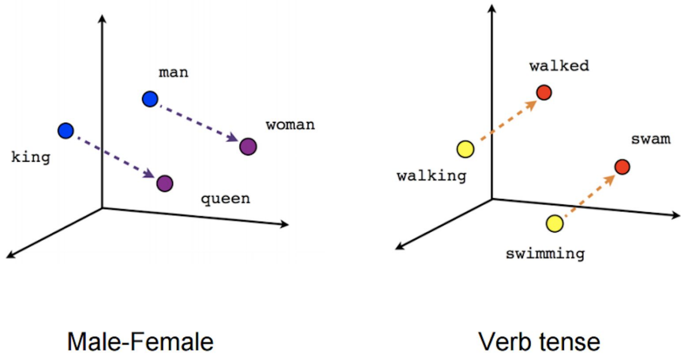


Fig 3: GloVe Embedding semantic relationships[11]

The GloVe is model for distributed word representation. The model used here is an unsupervised learning algorithm used for obtaining vector representation for words which is achieved by mapping these words to a meaningful space and the distance between the words is related to semantic similarity [10]. We are using model pre-trained on a large corpus of text. The GloVe Embedding captures the similarities between the word. For example, the distance between the king & queen and men & women is the same as can be seen from the graph for the GloVe embedding [10].

We will have three major modules, First, it will take different types of images as an input to VGG16 model and pass the extracted features to fully connected layer to map the different aspect of the image contents to a module-specific class representation space. Second, The LSTM module will generate the feature representation of the question asked for the image passed by GloVe embedding. In the end, The LSTM question representation and the CNN image features are fused via an element-wise multiplication, and then passed through fully- connected layers to generate a sigmoid distribution over output answer classes. The fig 4 shows the architecture of the base model developed for VQA task

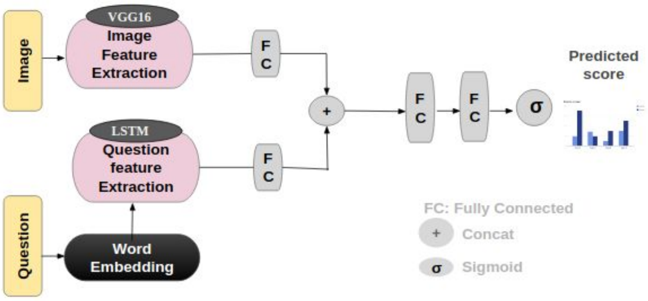


Fig 4: Base Model Architecture for VQA task [11]

The base model got 52.87% on validation dataset. The accuracy seems low so we tried to improve on the model. We all know that certain regions in the image and certain words in the question are more important than others. we tried to apply this analogy to the VQA task. From the literature review, we knew that complex attention models are having state-of-the-art result for VQA [4]. Hence, we moved our focus towards the attention network which helps the algorithm to decide which parts of the image to focus on. It uses the output from one model to focus on specific sections of another neural network. Hence, we built the next model using the attention network and called it attention model.

B. Attention Model

For the attention model, we are using preprocessed image features of MS-COCO dataset. Peter Anderson et al. [5] developed a bottom-up-attention model using Faster R-CNN object detection method and applied it to MS-COCO dataset such that output of an image forms a set 36-object labels and their bounding boxes. These features help the attention network to focus on certain regions of an image.

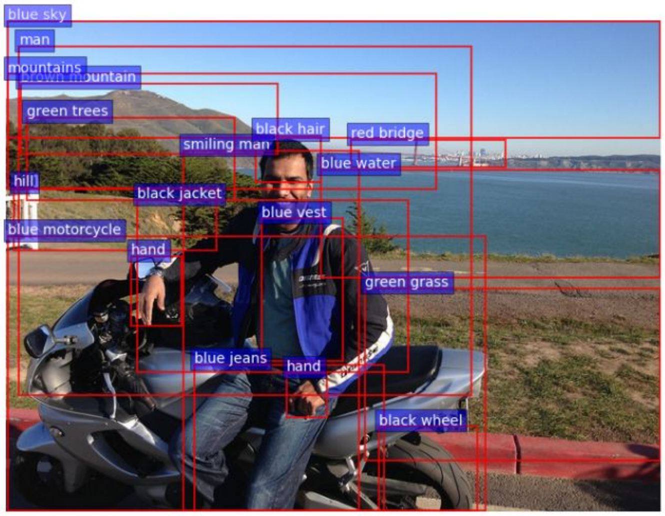


Fig 5: An example of features extracted from the image using bottom-up-attention Model [7]

We are passing the Question to GRU via GloVe embedding, then we are giving the text and image features to attention mechanism in the model. A softmax function is used to normalize the attention weights for all the locations. The image features from all locations are then weighted by the normalized values and summed to obtain a single vector representing the attended image. The attended image and text features from GRU are passed to Fully connected layers and the results are combined using element-wise multiplication. The resultant vector is again passed through 2 fully connected layers before applying sigmoid function on it.

The word-tokens from the question helps the algorithm to decide to which part of the image to focus on as here the image features also have 36 relevant object labels present in the image. At the end, the softmax uses only one answer as truth from whereas the sigmoid can have more than one correct answer per question. As the data can have disagreement between the answers for the given question related to an image, we have used sigmoid to produce the probability of the answers.

Here, we have used GRU instead of LSTM as used in the base model. The use of GRU doesn't result in significant result drop or improvement. However, it has fewer parameters then LSTM. Hence, it took less training time. Additionally, the attention mechanism used here is a simple & one-directional. We haven't explored more complex attention mechanism such as bidirectional attention and stacked attention mechanism as proposed in [5].

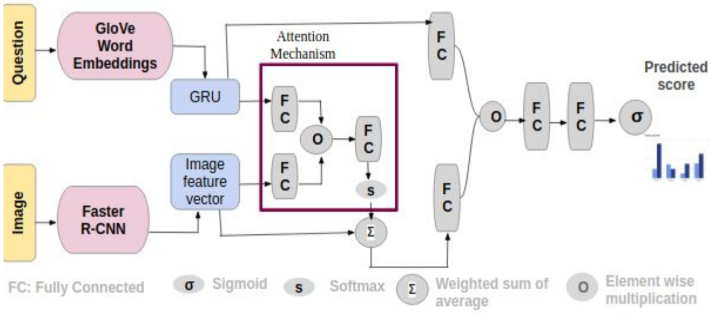


Fig 6: Attention Model Architecture for VQA Task

5. RESULTS AND DISCUSSION:

The results of our models (base model and attention model) along with a comparison of the corresponding results of the language base model and the state-of-the-art model are shown in the table 1.

The accuracy of the language only base model might look surprising as it completely ignores the image and only focuses on the question, still, it got the accuracy of 45.06% as claimed in [2]. The accuracy can be validated by the fact that some questions are about facts that never changes depending on the situation. (eg. What is color of Apple? It can be answered without even looking at the image “Red”).

Table 1: Comparison of different models developed for VQA

Model	Validation Accuracy (%)	Validatio n Accuracy at #epoch	Training time per epoch on NVIDIA Tesla K80 (mins)
Language only Model [2]	45.06	-	-
Base Model	52.87	18th	10.43 min
Attention Model	63.46	12th	8.29 min
State-of-the-Art	71.84	-	-
Inter-Human Agreement [2]	83.30	-	-

VGG16 has a top-1 error of 24.70%. Hence, it is not surprise that our base model got 52.87% of accuracy out of 76.30% possible accuracy score as the language model will also have introduced some error. The accuracy achieved in the attention model can also be visualized in a graphical manner to gather more understanding of the results, as shown below:

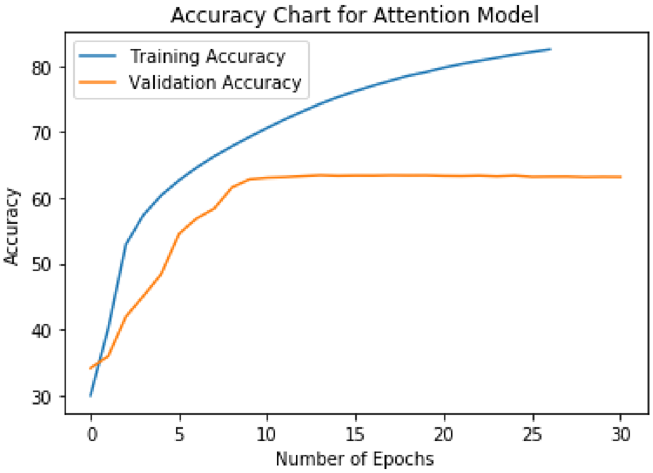


Fig 6: Accuracy chart of Attention Model

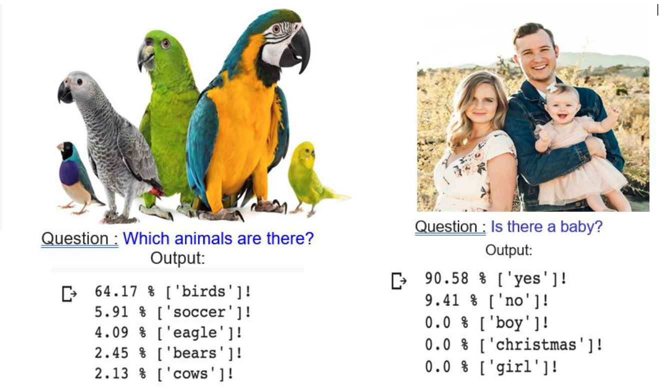
The interhuman agreement on the answers is at around 83% [2]. The can be due to 2 things. Firstly, it might be possible that annonater might have mistakenly selected the wrong answer and. secondly, we are not matching singular and plural words or similar words, etc.

Training time for the models are considerably less compared to other model developed from scratch as using transfer learning technique helps in that regards.. In the base model, we have used pretrained VGG16 weights on ImageNet Classification and in attention model, we have used bottom-up-attention features of images as proposed by Anderson et al. Moreover, In both models, We have also used pretrained GloVe embedding trained on large corpus of text for converting word token to 300 dimension vector. Thus, transfer learning aids significantly to speed up the learning process.

It can be concluded from these results that it is difficult to predict the effects of hyperparameters. Even if a network’s capacity is increased by some fancy tweak, it may be again redundant with other improvements [4]. In the paper, they have used the gated tanh activation function in the attention model. However, we have used ReLU and didn’t get any significant performance drop. Also, it can be seen that even when we are using a fixed number of objects per image(K=36) and not using extra data from Visual Genome (Visual Genome is another dataset for VQA, as is done in the he papers we mentioned above in literature review section), our results are similar compared to that of the paper.

Moreover, it can be comprehended that experimenting with reduced training data does not translate into improved performance rather than training on the whole of the dataset using the same architectures and parameters. Using such a large dataset has its own disadvantage. On large Data set it takes more time to hypertune parameters and oversee the effect of changing the architecture.

Our model is inefficient in some areas like counting some objects in any image. The accuracy, in that case, ~39% in the base model and ~49.7% in attention model. Additionally, the model also captures the language bias as it is more likely to answer yes in an binary yes/no question. The model also lacks the common sense reasoning that normal humans are better at it in such scenarios. All of these demerits can be improved by using different techniques to tackle each sub-problem.



Question : Which animals are there?
Output:

```
64.17 % ['birds']!
5.91 % ['soccer']!
4.09 % ['eagle']!
2.45 % ['bears']!
2.13 % ['cows']!
```

Question : Is there a baby?
Output:

```
90.58 % ['yes']!
9.41 % ['no']!
0.0 % ['boy']!
0.0 % ['christmas']!
0.0 % ['girl']!
```

Fig 7: Demo of an VQA Application

The [Fig 7] shows the example of VQA task on the developed attention model. Given an image and question as an input to the algorithm, it predicts the answer. Here, we have choose to report Top-5 answer with its probabilities.

6. CONCLUSION:

This report presents two models for VQA, one base model and one attention model and compared their results with a language base model and state-of-the-art model on the COCO-VQA dataset with open-ended questions. From this comparison, it can be concluded that the focusing on certain regions and words helps to achieve good accuracy for the VQA task.

However, If the models were trained on task-specific datasets then it may help enable practical VQA applications. Moreover, instead of GloVe embedding, as was used here, different other types of embedding can be explored like FastText word embedding or ELMo (Deep Contextualized word representations). Also, the attention model can be made more complex, like bilinear attention networks, to explore further options. Knowledge from external sources can be embedded into the system to help it to answer the factual question like Fire extinguisher seen in the image can resolve the fire? These can be considered as options for future enhancements for the project.

REFERENCES:

[1] T. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. Zitnick and P. Dollár, "Microsoft COCO: Common Objects in Context", arXiv.org, 2019. [Online]. Available: https://arxiv.org/abs/1405.0312

[2] A. Agrawal, J. Lu, A. Stanislaw, M. Mitchell, C. L. Zitnick, D. Batra, and D. Parikh, “VQA: Visual Question Answering,” *arXiv.org*, 27-Oct-2016. [Online]. Available: https://arxiv.org/abs/1505.00468.

[3] K. Kafle, C. Kanan, “Visual Question Answering: Datasets, Algorithms, and Future Challenges,” *arXiv.org*, 05-Oct-2016.[Online].Available: https://arxiv.org/abs/1610.01465.

[4] D. Tenney, P. Anderson, X. He, and A. van den Hengel, “Tips and Tricks for Visual Question Answering: Learnings from the 2017 Challenge,” *arXiv.org*, 09-Aug-2017. [Online]. Available: https://arxiv.org/abs/1708.02711

[5] J. Singh, V. Ying and A. Nutkiewicz, "Attention on Attention: Architectures for Visual Question Answering (VQA)", arXiv.org, 2019. [Online]. Available: http://arxiv.org/abs/1803.07724

[6] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition", arXiv.org, 2019. [Online]. Available: https://arxiv.org/abs/1409.1556

[7] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould and L. Zhang, "Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering", arXiv.org, 2019. [Online]. Available: https://arxiv.org/abs/1707.07998

[8] P. Wang, Q. Wu, C. Shen, A. Hengel and A. Dick, "FVQA: Fact-based Visual Question Answering", arXiv.org, 2019. [Online]. Available: https://arxiv.org/abs/1606.05433

[9] K. Zeng, T. Chen, C. Chuang, Y. Liao, J. Niebles and M. Sun, "Leveraging Video Descriptions to Learn Video Question Answering", arXiv.org, 2019. [Online]. Available: https://arxiv.org/abs/1611.04021

[10] J. Pennington, "GloVe: Global Vectors for Word Representation", Nlp.stanford.edu, 2019. [Online]. Available: https://nlp.stanford.edu/projects/glove/

[11] Singh, Manjeet. “Word Embedding.” *Medium*, Data Science Group, IITR, 14 Oct. 2017, http://medium.com/data-science-group-iitr/word-embedding-2d05d270b285
.

[12] VQA Challenge Website. http://visualqa.org

[13] M. Ferguson, R. Ak, Y. Lee and K. Law, "Automatic localization of casting defects with convolutional neural networks", 2017 IEEE International Conference on Big Data (Big Data), 2017.