Visual Question Answering in Bangla to assist individuals with visual impairments extract information about objects and their spatial relationships in images

Sheikh Ayatur Rahman
sheikh.ayatur.rahman@g.bracu.ac.bd
Brac University

Albert Boateng
albert.boateng@g.bracu.ac.bd
Brac University

Sabiha Tahseen
sabiha.tahseen@g.bracu.ac.bd
Brac University

Md. Sabbir Hossain
md.sabbir.hossain1@g.bracu.ac.bd
Brac University

Annajiat Alim Rasel
annajiat@gmail.com
Brac University

ABSTRACT

With millions of speakers worldwide, Bangla is one of the most spoken languages in the world and as a result, a large number of people rely on Bangla as their primary medium of communication. Among them, a lot of individuals have visual impairments including but not limited to central vision loss, peripheral vision loss, and blurry vision. This poses several challenges to these individuals - one of them being extracting information from images. While techniques such as image captioning have been proposed to address this issue, visual question answering (VQA) offers a much in-depth and robust way to understand an image. However, there has been a paucity of research into developing a VQA system in Bangla to assist individuals with visual impairments. To reduce this gap, we have introduced a VQA system in Bangla designed to assist visually impaired individuals. VQA is a multifaceted problem, and in this paper we focus on finding the spatial relationships between objects. We have broken down this problem into three sub-tasks: object detection, object counting, and finally relative positioning for the detected objects. The system takes in questions from the user, understands which sub-task to perform and then returns the answer. We have have leveraged several pre-trained models such as Bangla-BERT, EfficientDet-D7, InceptionResNetV2, and MiDas v2.1. The major aspects of this paper are the introduction of a procedurally generated dataset to train models to identify what action to perform based on the prompt of the user and using image segmentations to identify the relative spatial position between objects in all three spatial dimensions.

CCS CONCEPTS

• **Computing methodologies**; • **Artificial Intelligence**; • **Natural language processing**; • **Information extraction**;

ICCTA 2023, May 10–12, 2023, Vienna, Austria

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

KEYWORDS

Object Detection, Image Segmentation, Monocular Depth Estimation, Visual Question Answering, Bangla NLP, Spatial Positioning

ACM Reference Format:
Sheikh Ayatur Rahman, Albert Boateng, Sabiha Tahseen, Md. Sabbir Hossain, and Annajiat Alim Rasel. 2023. Visual Question Answering in Bangla to assist individuals with visual impairments extract information about objects and their spatial relationships in images. In *Proceedings of ICCTA 2023.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Many solutions have been proposed to enable individuals with visual impairments extract information from images. One such solution from the field of deep learning is using image captioning to help such individuals understand what is happening in an image. However, while image captioning can successfully provide a brief overview of an image, it falls short when it comes to answering more specific and in-depth questions about an image. Examples of include, “How many people are there in the image?”, “What object is there on top of the table?”, “What object is there to the right of the television?”, and “What is the person beside the bookshelf doing?”. Visual Question Answering provides an answer to this problem by allowing users to query information about an image and helping to provide a more thorough understanding of the image to the users which is very beneficial to users with visual impairments [9].

Despite the promise of Visual Question Answering in addressing this problem, there has been inadequate research done into developing a VQA system in the Bangla language. In Bangladesh alone, a predominantly Bangla speaking country, 650,000 people above the age of 30 suffer from blindness [4]. We believe that by offering an effective VQA system in Bangla, we can vastly improve the ability of the visually impaired to retrieve context-specific information from images.

We break down the task into two parts. In the first part, we perform question featurization. Here, we generate context-sensitive word embeddings from the BERT model [3] which are then fed into a BiLSTM layer [14] [12]. The output from the BiLSTM layer is then fed into two fully connected layers which then classifies which task the system has to perform.

In the second part of the task, we divide the task further into three sub-tasks which are object detection, object counting, and finding relative position of objects in all three dimensions. After performing the required sub-task, the answer is returned to the user. The models we have used while performing these subtasks are: EfficientDet-D7 [18], Mask R-CNN Inception ResNet V2 1024x1024 [16], and MiDaS v2.1 small [13].

2 LITERATURE REVIEW

Various efforts have been made in the past to make the architectures in each of the three steps outlined above efficient and feasible. [5] hypothesized that the outer product of the question and image vectors would produce better results than performing concatenation or element-wise addition or multiplication on them. Furthermore, [11] introduced a co-attention model so that the model that understand which words in the question are more relevant to the answer it is looking for. This approach builds on top of previous proposals that focused on visual attention where the models tried to understand which parts of the image were more helpful in answering the question. Another method introduced in [6] used question-guided convolutions to capture the visual spatial relationships that are lost in current approaches when trying to merge the image and text vectors.

The vast majority of research on VQA has been conducted using questions and answers in English. The morphological and synctactic complexity of Bangla varies quite significantly from that of English [15] [1] and these differences need to be captured when making VQA effective for the Bangla language. As opposed to English, VQA datasets in Bangla are not readily available - which is why we address the issue of VQA in Bangla in a manner different from the methods listed above.

3 BACKGROUND ARCHITECTURES

1. BERT model

Bidirectional Encoder Representations from Transformers, or BERT, was introduced in 2018 by [3]. BERT utilizes the transformer architecture, more specifically the encoders, by stacking encoders together. BERT is more powerful than previous models as it can understand context in a sentence in a truly bidirectional way.
BERT offers a much more powerful understanding of language and context which has given state of the art results in multiple Natural Language Processing (NLP) tasks.
For our research, we have used the BERT-Bangla-Base [2] model which is specifically trained on a large corpus of Bangla text.

2. EfficientDet-D7

EfficientDet-D7 [18] is a part of the family of EfficientDet architectures. It provides state of the art results in object detection on the Common Objects in Context (COCO) [10] test-dev dataset with an AP score of 55.1.

3. Mask R-CNN Inception ResNet V2 1024x1024

Mask R-CNN Inception ResNet V2 1024x1024 [16] is based on the family of Inception networks introduced in [17] which incorporates residual networks which were introduced in [7] which helps it to achieve state-of-the-art results.

4. MiDaS v2.1 small MiDaS v2.1 small belongs to the MiDaS group of architectures introduced in [13] for performing monocular depth estimation on images. These architectures were pretrained on a diverse range of images over extended periods of time. They find the inverse depth of the images, meaning that a higher value is assigned to pixels that are closer to the camera than the ones that are farther away.

4 METHODOLOGY

2.1 Understanding user prompts

2.1.1 *Task overview*

Given a question from the user, the system has to respond to four categories of questions - object detection, object counting, positioning questions, and positioning and counting questions, which is a combination of the previous two tasks.

For this project, we have 80 classes of objects all belonging to the Common Objects in Context (COCO) dataset [10] with all of their labels translated to Bangla by native Bangla speakers. Given a question from the user, the system performs a classification task. This classification task consists of 1041 classes - 1 class for object detection task, 80 classes for object counting with each object category having its own class. 480 classes are for detecting objects in a particular direction relative to another object. Since we have 6 directions - left, right, above, below, front, and behind - for 80 objects, we get 480 classes.

Table 1: Breakdown of question formats in the dataset

Question Format	Question Category
ছবিতে কি কি জিনিস আছে? (What are the things in the image?)	Detection
ছবিতে কয়টি obj দেখা যাচ্ছে? (How many obj can be seen in the image?)	Object Counting
obj-এর direction কি আছে? (What is there in the direction of obj)	Relative Positioning
obj-এর direction কয়টি জিনিস আছে? (How many things are there in the direction of obj?)	Relative Positioning and Counting

2.1.2 *Dataset*

Due to the scarcity of VQA datasets in Bangla, we used our own procedurally generated dataset. To accomplish this, we defined a set of question formats containing some parameters depending on the question type. An example of such a format is "ছবিতে কয়টি object আছে?" (Translation: ”How many [object] are there in the image?”). By replacing the 'object' token with a randomly selected from our 80 object types, we can generate a question for the dataset. The corresponding answers can be generated using a similar format. Given we already know what a sub-task a particular question format is referring to, we can simply generate answers using the corresponding parameters. In the above example, the answer would be generated by simply setting it to 'counting_object' where the object token is the same random object type we had previously selected for the question. Examples of question-answer formats is given in Table 1 along with English translations.

2.1.3 *Word Embeddings*

Word embeddings are vector representations of tokens that can be used as input into various Natural Language Processing (NLP) tasks. Word embeddings produced by BERT offer a way into preserving the semantic meaning of a token in a given context while also offering a token representation that can be fed into deep learning models. Since our data is in Bangla, we have used the Bangla-BERT-Base model [2], which is the BERT model trained on a large corpus of Bangla text. Preserving semantic meaning ensures our model performs well against a wide variety of queries.

2.1.4 *BiLSTM layer*

LSTMs [8] are a type of Recurrent Neural Network that can detect long range dependencies in sequential data like sentences. BiLSTM is a special type of LSTM, which can detect depedencies in a bidirectional manner.
The summary of this model is provided in Table 3.

Table 2: Model Summary

Layer (type)	Output Shape	Param
input_2 (InputLayer)	[(None, 20, 768)]	0
bidirectional_1 (Bidirectional)	(None, 20, 1024)	5246976
flatten$_1(Flatten)$	(None, 20480)	0
dense_2 (Dense)	(None, 1024)	20972544
dense_3 (Dense)	(None, 1041)	1067025

Total parameters: 27,286,545
Trainable parameters: 27,286,545
Non-trainable parameters: 0

2.2 Finding relative positions of objects:

In this sub-task, we have to find the direction in which an object is with respect to another object. There are 6 possible answers to this question: left, right, above, below, front, behind. These 6 answers can be grouped into 3 categories: left and right as horizontal relative position, above and below as vertical relative position, and front and behind as the relative depth of the objects in the image. There are two key assumptions made when finding the solution to this problem:

I) The question the user is asking is from the perspective of the camera which took the picture
II) Three answers from the three categories mentioned above can be true at the same time.

Given that there can be multiple answers to a question, our system needs to be able to find the most relevant answer which will help users visualize the relative positions between a pair of objects more easily. We offer a method to do this, later on.
In order to perform this sub-task, we found the semantic segmentations of the image. Through semantic segmentation, we can find each and every pixel that belongs to a given object. We have obtained the image segmentations using the pre-trained model Mask R-CNN Inception ResNet V2 [16]. By using the information given to us by semantic segmentation, we propose the following techniques to find relative position of objects as follows:

2.2.1) *Horizontal Relative Position*

Every pixel in the semantic segmentation map of the input image can be represented as a tuple (x, y, z) where x and y are the x-coordinate and y-coordinate of the pixel respectively and z is a boolean value which indicates whether the pixel belongs to the object of interest or not. For this task, our input is two objects - A and B - where we are trying to find the relative position of B with respect to A. We first obtain the segmentations of A and B in the image. Then, we calculate the mean of the unique x-coordinate values of all pixels belonging to B which is denoted as ${M^x}_B$. We find the same quantity for object A and denote it as ${M^x}_A$. If ${M^x}_B$ is greater than ${M^x}_A$, our answer is 'right'. If ${M^x}_B$ is smaller than ${M^x}_A$, our answer is 'left'. If both quantities are the same, we say that are aligned along the horizontal axis.

The intuition behind this operation is that it can simply be seen as an extension to finding the difference between 2 pixels. For example, if A and B were just individual pixels instead of objects, a positive x-coordinate difference between B and A would indicate B is to the left of A while a negative one would indicate B is to the right of A. Thus, by finding the differences between the means of all unique x-coordinates between the objects, we can get an idea of the spatial relationship between the objects in the horizontal axis.

2.2.2) *Vertical Relative Position*

The method used for this task is the same as previous one except for two differences:

I) We use y-coordinates of the pixels instead of x-coordinates to perform our calculations
II) If the mean of the unique y-coordinates of object B is greater than that of A, then B is below A and vice versa. If they are equal, we say that they are aligned in the vertical direction.

2.2.3) *Finding the relative depth of two objects*

For this task, we have to find how far the objects are from the camera taking the picture. Since images have 2 spatial dimensions, the approach used above does not work. As such, we use the MiDaS v2.1 small model to generate a depth map of the image. In the depth map, each pixel can be represented as a 3-tuple (x, y, d) where x is the x-coordinate of the pixel, y is the y-coordinate of the pixel and d is the depth of the pixel. A higher d indicates the pixel is closer to the camera than a lower d. Formally, we sum up all values of d of every pixel where z = 1 for both object A and object B. We then divide this sum by the total number of pixels in the segmentations of the object for object A and object B. This gives us the average depth of A and B. If the average depth of B is greater than that of A, the answer is 'front' (B is in front of A) and if the average depth of B is lesser than that of A, the answer is 'behind'.

By using the techniques above, a range of queries relating to spatial relationships in images can be answered.

2.3 Finding the best answer to questions about spatial relationships

While answers pertaining to questions about spatial dimensions can be answered in terms of either horizontal direction (left or right), vertical direction (above or below), or in terms of image depth (front or behind), in most circumstances one of these answers is more important to our understanding of the image than the other two.

To determine the best answer, we propose a simple heuristic to determine the best answer. We need three quantities we had calculated previously:

I) The absolute value of the difference between the mean values of the unique x-coordinates of object A and object B.

II) The absolute value of the difference between the mean values of the unique y-coordinates of object A and object B

III) The absolute value of difference in the mean depths of object A and B in the image

We take the maximum of these three quantities and choose the dimension which corresponds to this maximum. There is one important detail to keep in mind about quantities (I) and (II). If the width of the image is greater than the height, quantity (II) has to be scaled by a factor of width / height. Conversely, if the height is greater than the width, quantity (I) has to be scaled by a factor of height / image. This is done to ensure that comparison between the dimensions is fair.

Quantity (III) does not require any such scaling since it is not dependent on the height and width of the image.

The idea behind this technique is that the spatial dimension where the difference between the objects is the most pronounced is the one that gives the most relevant answer to the question of spatial dimension between a pair of objects.

Fig 6. The objects of interest are the pizza and the wine glass. The values of quantities (I), (II), and (III) were 66.0, 323.8, and 642.3 respectively. Thus the answer is 'pizza is behind the wine glass'. Image is from COCO dataset [10].

2.4 Object Detection and Object Counting

These tasks are fairly straightforward with the help of modern deep learning architectures. To perform object detection on images using a EfficientDet-D7 architecture, as discussed above, pre-trained on the COCO dataset. We have used a threshold score of 0.3. After performing object detection, object counting is a trivial problem as we return the number of instances that detected as belonging to a specific class.

5 RESULT ANALYSIS

Understanding User Prompts

In this task, we had to classify questions depending on what sub-task it required us to perform. To make our model flexible and understand questions that may vary from the training data, we obtained word embeddings of our tokenized questions. We then passed them through a BiLSTM layer and finally through two fully connected layers to find the answer.

We have used the question format specified in Table 1 to generate 60,896 pairs of questions and answers for the model to train on. The model was trained for 10 epochs with a batch size of 64. A training accuracy of 99.06% was obtained.

Additionally, we also wanted to test how robust and flexible our model was. As such, we obtained a testing partition, where we modified the original formats such as using Bangla synonyms for certain terms, changing punctuation, changing the sequence of words, and omitting certain words in the questions to generate 20,000 modified questions. On this data, our model achieved an accuracy of 88.91%.

Relative Positioning

To test the efficacy of the techniques introduced to find the spatial relationships between objects, we collected 150 object pairs across 33 images from the COCO dataset [10] for each of the three categories - horizontal relative position (left or right), vertical relative position (above or below), and finding relative depths of objects (front or behind) and we obtained accuracies of 99.3%, 99.3%, and 97.3% respectively.

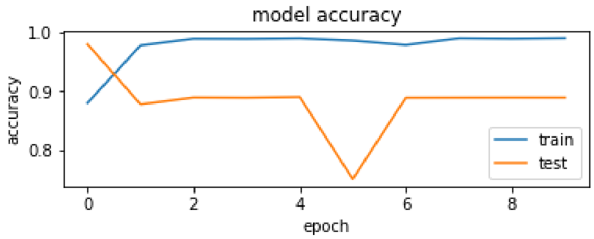

Fig 2. Training and testing graphs of question classification model.

All of the results are summarized in Table 4.

Table 3: Breakdown of accuracy of each of the tasks

Task	Train Accuracy	Test Accuracy
Question Classification	99.06%	88.91%
Horizontal Relative Positioning	-	99.3%
Vertical Relative Positioning	-	99.3%
Finding Relative Depth	-	97.3%

6 DISCUSSION

From the training and testing metrics of the question classification task, we can see that the model generalizes reasonably well to questions of a different format to the ones used to train it. This performance was achievable due to the large amount of training data we generated procedurally as well the word embeddings generated by the Bangla-BERT model. By providing more data, and a more diverse range of question formats, the performance of the model should improve even further.

For the relative positioning task, we have used simple heuristics to convert the numeric outputs provided by the models used to categories that can be easily understood by humans. We see from the accuracies that these methods performs well purely from a spatial perspective. There are some limitations to this method. Firstly, this method will not provide consistent results if certain transformations are applied to the image. For example, if the image is rotated, these methods will be unable to detect it. Secondly, these methods only capture spatial relationships between objects and not their semantic relationships. For example, relationships like 'the person is riding the horse' cannot be detected by this system. Future research can be focused on answering these issues.

7 CONCLUSION

In this paper, we introduced a visual question answering system which allows individuals with visual impairments to better understand the spatial relationships between the objects. For this, we leveraged various state of the art models. We also constructed a procedurally generated dataset to train a model identify what action to perform based on user prompts as well as use image segmentations to find the spatial relationships between objects.

REFERENCES

[1] Shafi Ahmed, Md Humaion Kabir Mehedi, Moh Rahman, and Jawad Bin Sayed. Bangla music lyrics classification. pages 142–147, 09 2022.
[2] Abhik Bhattacharjee, Tahmid Hasan, Wasi Ahmad, Kazi Samin Mubasshir, Md Saiful Islam, Anindya Iqbal, M. Sohel Rahman, and Rifat Shahriyar. Banglabert: Language model pretraining and benchmarks for low-resource language understanding evaluation in bangla. *Findings of the Association for Computational Linguistics: NAACL 2022*, 2022.
[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
[4] Brendan Dineen, Rupert Bourne, S Ali, D Huq, and G Johnson. Prevalence and causes of blindness and visual impairment in bangladeshi adults: Results of the national blindness and low vision survey of bangladesh. *The British journal of ophthalmology*, 87:820–8, 07 2003.
[5] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 457–468, Austin, Texas, November 2016. Association for Computational Linguistics.
[6] Peng Gao, Hongsheng Li, Shuang Li, Pan Lu, Yikang Li, Steven C. H. Hoi, and Xiaogang Wang. Question-guided hybrid convolution for visual question answering. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, pages 485–501, Cham, 2018. Springer International Publishing.
[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
[8] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9:1735–80, 12 1997.
[9] Kushal Kafle and Christopher Kanan. Visual question answering: Datasets, algorithms, and future challenges. *Computer Vision and Image Understanding*, 163:3–20, 2017. Language in Vision.
[10] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.
[11] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 289–297, Red Hook, NY, USA, 2016. Curran Associates Inc.
[12] Rakin Mostafa, Md Humaion Kabir Mehedi, Md Alam, and Annajiat Alim Rasel. Bidirectional lstm and nlp based sentiment analysis of tweets, 12 2022.
[13] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3), 2022.
[14] Mike Schuster and Kuldip Paliwal. Bidirectional recurrent neural networks. *Signal Processing, IEEE Transactions on*, 45:2673 – 2681, 12 1997.
[15] Safwan Shaheer, Ishmam Hossain, Sudipta Sarna, Md Humaion Kabir Mehedi, and Annajiat Alim Rasel. Evaluating question generation models using qa systems and semantic textual similarity, 03 2023.
[16] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-first AAAI conference on artificial intelligence*, 2017.
[17] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.
[18] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10781–10790, 2020.