Testing Image Understanding through Question-Answering

Vaibhav Singh, Sankalp Dayal, Kevin Tsai
{vaibhvs, sankalpd, kevin259}@stanford.edu

Abstract

In recent years, neural networks have produced some very exciting results in diverse computer vision tasks, such as image recognition, image captioning, artistic style transfer, auto-coloring, segmentation and image generation. In this paper we ask a fundamental question: "How well do these networks really understand images?" We explore this question through through two approaches: first through the task of Visual Question Answering (VQA), where a model is trained on images and associated question answer pairs in natural language. Next, we measure effectiveness of transfer-learning to image recognition. We use a model pre-trained on image-recognition task and retrain its weights during VQA training and then test it again on image recognition task.

1. Introduction

Neural networks have been used with remarkable success in both Vision and Natural Language processing fields. Although they grew out of the same underlying principles of deep learning, these two fields have evolved somewhat in silos. Visual Question Answering (VQA) is one of the few task where we see the interplay of the two.

Architecture of VQA systems proposed in recent years exhibit a recurring theme at broader level, typically consisting of a module to extract features from the input image, a module to learn some representation of input question, a module for attention mechanism to identify the relevant regions of image (yang et al, 2016, Xiong et al., 2016) and question (Lu et al., 2017), and a module for Softmax output. These methods typically used either Long Short Term Memory (LSTM) or Gate Recurrent Units (GRU) for the memory component. Memory for the LSTM/GRU cells reside in the cell states, stored as weights. Xiong et al. also proposed a new variant of GRU, the Attention GRU, for handling complex queries sensitive to both position and ordering of input facts. The gating mechanisms introduced in the recurrent unit cells not only address the vanishing or exploding gradient problems, but it also protects the cell against irrelevant perturbations.

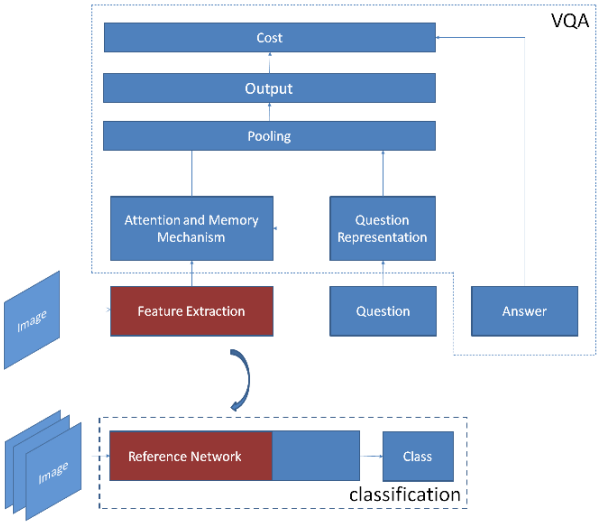


Figure 1: High-level View of Visual Question Answering System and Cross-Transfer Learning. The blue arrow defines typical transfer learning flow where representations learned on one tasks is used in another task. The orange arrow defines plugging the transfer learning representation back to the original task.

Due to recurrent update mechanism, this memory captures sequential dependencies well, but these very mechanism also makes it to vulnerable to biases present in the training data itself. In a recent study by Goyal et al. 2017, performances of winning entries of VQA 2016 challenge were observed to degrade on more balanced VQA v2.0 dataset.

In this paper, we seek to investigate the aspects of these models that make it susceptible to biases in the training data. We also aim to investigate the memory aspects of these models and potential methods to make it generalized better. Finally, we also report the end-to-end learning on the CNN components by re-evaluating it on the image classification task. We define cross transfer learning as the term for this task.

The following section on related work presents a brief summary of primary methods we have analyzed. The Ex-

periments section outlines the baseline architecture and proposed variations. The details of primary metrics of the dataset and results are provided in the subsequent sections.

2. Related Work

Yang, Zichao, et al. suggest that the conventional architecture of CNN feeding the whole image in one. step to RNN/LSTM downstream does not allow the network to have fine-grained learning. In most VQA scenarios, one or a few small parts of the image usually answers the question. The authors propose an architecture that evaluate sub-parts of the image in phases and identify probabilities against the question. This process allows for the network to learn where to focus. Additionally, the authors also tried CNN in addition to LSTM for the question module, with CNNs filtered on unigram, bigram, and trigram.

Richard Socher, et al. attempted to address certain limitations around information flow of the standard GRU implementation by an episodic memory module and infusion layer, in what they call Dynamic Memory Network+ (DMN+). The attention mechanism in the episodic memory module creates a contextual vector-based on previous memory states and the question.

The authors state that the unidirectional flow of traditional VQA networks allows knowledge to be propagated only forward. The infusion layer addresses this by implementing two sets of GRUs, one set going forward and the other going back.

Lu, Jiasen, et al. suggest that while most VQA works focuses on attention in the image, attention in the question, specifically which words to look at, is just as important. The authors introduce two primary concepts in their paper: question hierarchy and image/question co-attention.

In the question module, the authors apply a question hierarchy which is similar to Yang, Zichao, et al. The hierarchies are also processed by convolution into unigram, bigram, and trigrams.

The second concept introduced by the authors is co-attention of image and question. The difference of this part of the process is the affinity matrix C is learned. The authors also suggest the image/question co-attention to be learned in parallel or alternating.

Another approach worth mentioning is Multi-Modal Compact Bi-Linear pooling used by Akira Fukui et al., 2016. There are two fundamental concepts at the heart of this implementation. First, as per the convolution theorem element-wise product is frequency domain is equivalent to convolution in time domain and second, interaction of features using elementwise multiplication or concatenation can be significantly enhanced by using outer products (Bi-Linear pooling, Lin et al. 2015, Gao et al. 2016). Bilinear pooling yields to richer interaction because each vector component interacts with every other element, however because of higher dimensionality reasons it can quickly become less that pragmatic. Akira et al. solve this problem by introducing Compact Bi-Linear pooling. In this approch first the features are projected to lower dimensional space and then outer product computation is done using element-wise product in frequency domain(Pham and Pagh,2013). Compact Bi-Linear pooling was winning approach in VQA challenge 2016.

3. Methods

3.1. Visual Question Answering

For VQA tasks we considered two base implementations: Stacked Attention Network (SAN), Yang et al. 2014, and DMN+, Xiong et al., 2016. Both implementations have a visual input module, a question module and an answer module.

3.1.1 Stacked Attention Network

SAN was one of the earlier models introduced for VQA tasks. In a SAN, output of visual input module and question module are fed to the SAN. This follows the general topology that consists of an image module, a question module and an attention module.

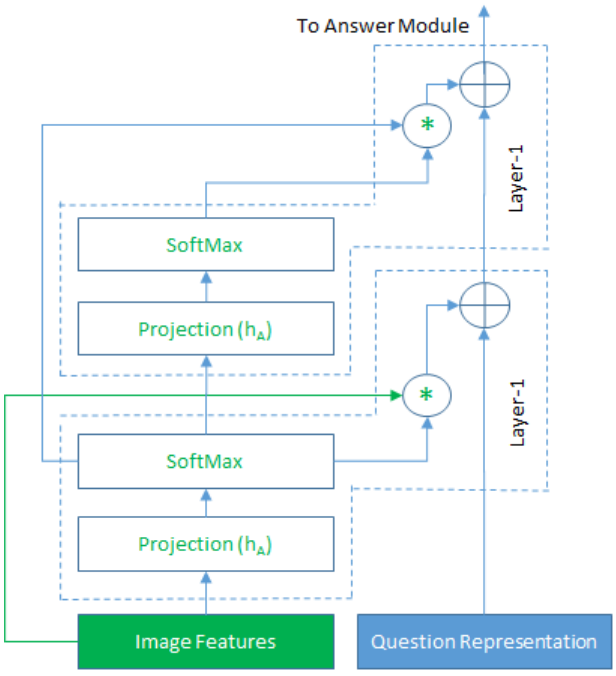


Figure 2: Attention Computation in Stacked Attention Network. Each layer is marked by broken lines. 2-Layer SAN was found to perform better in practice.

Image features in Figure-2 are derrived from pre-trained VGG19 model (Last pooling layer (512x14x14); 14x14 is the number of regions and 512 features for each region is selected).

The question representation used by Yang et al. is based on CNN. Authors reported experiments with both CNN and LSTM-based representations, of which CNN based implementation was found to perform better. Hence we chose CNN-based representation for our baseline experiments. The architecture used for this representation is similar to the one outlined by Kim et al. 2014 for sentiment analysis:

In the attention module, the image feature vector is fed to a single layer neural network with softmax output to calculate the attention distribution.

$$h_A = \tanh(W_{I,A^{v_I}} \oplus (W_{Q,A^{I_{v_Q}}} + b_A)), \quad (1)$$

$$p_I = \text{softmax}(W_P h_A + b_P) \quad (2)$$

where $v_I \in \mathrm{R}^{dXm}$, d is the image representation dimension (512) and m is the number of image regions (196), $v_Q \in \mathrm{R}^d$ is a d embedding dimensional vector. Suppose $W_{I,A}$,$W_{Q,A}$ $\in \mathrm{R}^{kXd}$ and $W_P \in \mathrm{R}^{1Xk}$, then $p_I \in \mathrm{R}^m$ is an m dimensional vector, which corresponds to the attention probability of each image region given the question vector v_Q. $\oplus$ is the addition of a matrix and a vector (broadcasting).

An attention weighted image vector $\tilde{v_I}$ is subsequently derived from element-wise multiplication of v_I and probabilities per region p_I.

$$\tilde{v_I} = \sum_i p_i v_i \quad (3)$$

And refined query vector u is computed as:

$$u = \tilde{v_I} + v_Q \quad (4)$$

Construction of eq-(1)-(4) may be repeated multiple times to generate a deeper attention abstraction and corresponding query vector as shown in figure-(2).

It should be noted here that the VGG net weights are used only once for feature extraction of images are not cross-trained on the VQA task. We found in methods proposed by other authors that cross-training the pre-trained weights of image module is helpful (Xiong et al. 2016, Goyal et al., 2107) for VQA task.

3.1.2 Dynamic Memory Network

Input Module In DMN+, we feed the raw input image to a pre-trained VGG-19 model for image input module. We use the output of the last pooling layer of VGG-16 which has a dimension of 512x14x14. This gives 196 local regional vectors of size 512. Each of these local regional vectors are multiplied by projected on the input weight matrix to give

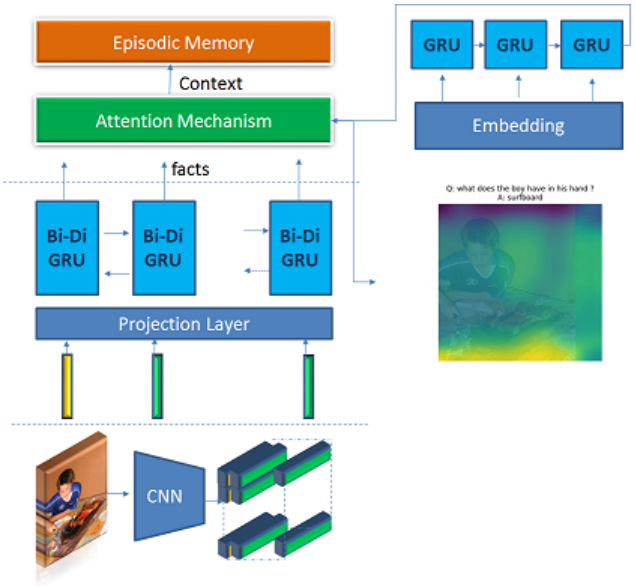


Figure 3: Dynamic Memory Network [+] Architecture.

feature embeddings f_i. The traversal to create embedding is done from left to right, row by row. These individual facts represent different regions of the input image. In order to derive sequence sensitive representation of these facts, facts in original reverse order are fed into GRUs. The output of these GRUs is then summed to derive a fact representation that can capture dependencies in both orders.

$$\overrightarrow{f_i} = GRU_{fwd}(f_i, \overrightarrow{f_{i-1}}) \quad (5)$$

$$\overleftarrow{f_i} = GRU_{bwd}(f_i, \overleftarrow{f_{i+1}}) \quad (6)$$

$$\overleftrightarrow{f_i} = \overleftarrow{f_i} + \overrightarrow{f_i} \quad (7)$$

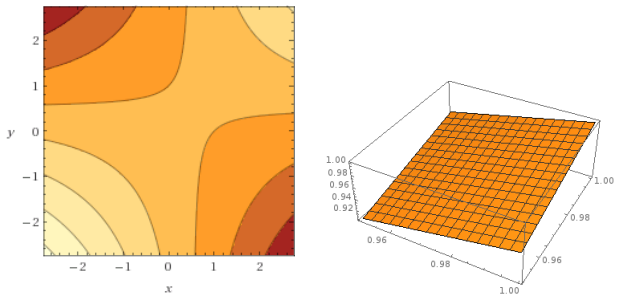
Forward and backward GRUs are collectively shown as bidirectional GRUs in figure 1. Specific unrolling of CNN features and use of bidirectional GRU allows information propagation from neighboring image patches, capturing spatial information (Xiong et al., 2016).

Question Representation For question representation Xiong et al. have used **positional encoding** introduced by Sukhbataar et al. (2015) As per this encoding scheme, the sentence representation is produced by:

$$f_i = \sum_{M}^{j=1} l_j \odot w_j^i \quad (8)$$

where $\odot$ is element-wise multiplication and l_j is a column vector with following structure:

$$l_{jd} = (1j/M)(d/D)(12j/M) \quad (9)$$



(a) Contour Plot (b) Positional Transform

Figure 4: Transform to map word embedding for sentence words such that it is unique for all possible orders of words.

Here, d is the embedding index and D is the dimension of the embedding. M is the total number of words in the sentence. At first sight, this equation seems rather esoteric but with a minor rearrangement we can see both the underlying simplicity and the rationale for using this expression.

$$l_{jd} = (1j/M)(1 - d/D) + (d/D)(j/M) \quad (10)$$

Preceding equation reveals the underlying hyperbolic plane, which simply serves the purpose of unique mapping of each point in the D-M plane to a unique value (as also shown in the following figure). It is not difficult to observe that any other transfer function which establishes similar bijective relationship should serve the same purpose.

An alternative representation is a **GRU-based question representation**. Xiong et al. have cited efficiency reason for using positional encoding vs the GRU-based implementation. There is an important distinction to be made here. DMN+ original architecture was targeted towards both the natural language fact settings and visual question answering setting. In case of neural language fact setting, use of GRU in input and question answer layer certainly leads to efficiency issues. Since it would involve encoding multiple sentences for both facts and question. However, in VQA setting we only need to encode the questions from text. This is the reason why in out implementation we took the liberty of replacing the positional encoding modules with a GRU for question representation.

Attention Mechanism and **Episodic Memory** are perhaps the most crucial components of a VQA system. Attention Mechanism and Episodic Memory work in conjunction; an intuitive notion of attention suggests that attention should be function of memory, question and facts (representation of the input image). How the model allows these inputs to interact results in its own unique attention mechanism.

Xiong et al. described the following set for interactions in their DMN-based VQA system.

$$z_i^t = [\overleftrightarrow{f_i} \odot q; \overleftrightarrow{f_i} \odot m^{t-1}; |\overleftrightarrow{f_i} - q|; |\overleftrightarrow{f_i} - m^{t-1}|] \quad (11)$$

where $\overleftrightarrow{f_i}$ is the ith fact, m is the previous episode memory, q is the original question, $\odot$ is the element-wise product, $|.|$ is the element-wise absolute value, and ; represents concatenation of the vectors.

Compact Bi-Linear Pooling (Akira et al.), Hierarchical Co-attention (Jiasen et al.) and the original Dynamic Memory Network (Kumar et al.), each use a different set of interactions approaching the same objective. Interactions from the preceding expression used further to compute attention gate g_i^t:

$$Z_i^t = W^{(2)} tanh(W^{(1)} z_i^t + b^{(1)}) + b^{(2)} \quad (12)$$

$$g_i^t = \frac{exp(Z_i^t)}{\sum_{k=1}^{M_i} exp(Z_k^t)} \quad (13)$$

Attention gate g_i^t is used furthermore to generate a context vector which is used for episodic memory update. Dynamic memory plus architecture uses a variant of Gate Recurrent Units called **Attention GRU**:

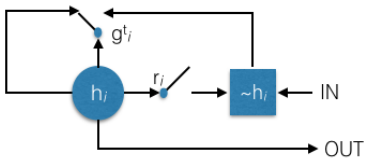


Figure 5: Attention GRU. Notice the update gate vector

For our experiment we have used attention-based GRU over another alternative presented by Xiong et al., attention GRU method was reported to perform better by the authors.

Episodic Memory Module is GRU layer that uses context vector as an input. Sukhbataar et al. also report using different GRU weights for different memory passes.

Another alternative episodic memory implementation described by Xion et. al is ReLU-based. In this implementation, the new episode memory state is obtained by

$$m^t = ReLU(W^t[m^{t-1}; c^t; q] + b) \quad (14)$$

For efficiency reasons in our expetiments we have used same GRU-based episodi memory implementation with fixed weights across iterations. (referred to as **tied weights**). The final output of the memory network is passed to the answer module (softmax layer).

3.2. Cross Transfer Learning

Settings for cross-transfer learning follows the architecture presented in Figure-1 with DMN architecture, with

one difference: resnet-40 architecture trained on CIFAR-10 is used for image feature extraction. Our implementation of resnet-40 network was achieved validation accuracy of 90% on CIFAR-10. We chose CIFAR-10 so that weights post VQA training could be plugged back in an re-evaluated easily.

4. Dataset

Our data sources primarily came from Virginia Techs Visual Question Answering (VQA) Challenge, version 1.0 and version 2.0; see References section. The v1.0 set includes 204,721 COCO images and 614,163 questions.

Unlike a binary scenario where one answer is correct and the other is wrong, VQA datasets has possible multiple correct answers, and likely but incorrect answers. As an example, a picture may have a child, a woman, and a frisbee, but answers to the questions is the child throwing the frisbee or is the woman holding the frisbee will be different based on what is actually happening in the picture. We know one of the two can be true or neither is true, but we cannot have both being true. Hence, of the 614,163 questions, there are 6,141,630 ground truths (10x) and 1,842,489 plausible but incorrect answers (3x).

An initial analysis we performed was to visualize the position of words in the question set. We performed a simple word count on the questions using Apache Spark to collect the top 50 most common words. We then used PCA to compress the 300-dimension GloVe Global Vectors into 2-dimensions. This results are shown in the following Figure 6. The 2-D representation of 300-D original captured 58% variance. One question this word distribution brings

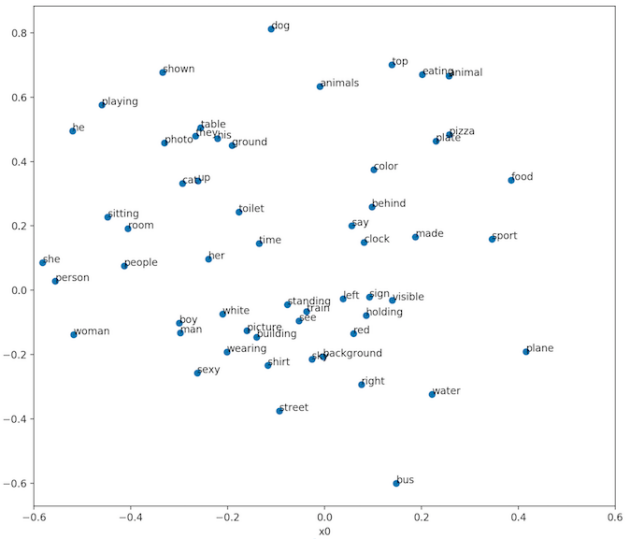


Figure 6: 2-D PCA Representation of GloVe Top-50

up is how much of the language prior plays into the results and how much is the actual contribution of vision. Since ultimately the model will respond based on probability, the closeness of terms can possibly affect the outcome regardless of the picture itself. As in the previous example of the child and frisbee, a question with the verb playing will have a much higher probability than with the verb burning. This is expected, as word vectors are learned to predict words in proximity. In effect, the nature of the data set has an internal skew.

Skew in a binary data set such as malignant/benign in the case of cancer or abnormal/normal in the case of system failure anomalies are common, where the former cases are much less frequent than the latter cases. The VQA v2.0 attempts to address this (Goyal, et al., 2016) by asking the same question on a picture with the ground truth changed. For example, in the case of the child, woman, and frisbee, the question is the child throwing a frisbee? is asked of two nearly-identical pictures, one where the child is indeed throwing a frisbee, and the other where the woman is holding onto the frisbee and the child is not throwing the frisbee.

The v2.0 data set uses the same 204,721 COCO images, but instead has 1,105,904 questions, with 11,059,040 ground truth answers. When we used the v2.0 dataset, we saw a significant drop in accuracy compared to v1.0. This suggests that the model relies heavily on language prior and question, which is also seen in other research; see Results section.

5. Experiments

For VQA baseline we consider used Stacked Attention Network introduced by Yang et al. 2014. We trained the network on VQA v1.0 data set. Dynamic memory network architecture (Xiong et al.) was used for VQA objective and cross transfer learning analysis. Conclusion of these experiments is discussed in the next section.

5.1. Visual Question Answering

Since image data and natural language both are key inputs to VQA tasks we decide to perform four main experiments:

1. DMN+ (glove 300 init) on VQA v1.0 dataset. This is the configuration reported by Xiong et al. to perform best on VQA 1.0 data set.

2. DMN+ (random init) on VQA 1.0 dataset. In order to see the effect of pre-trained vs random word embedding on VQA task.

3. DMN+ (glove 100 init), on VQA 2.0 dataset to analyze of word vector dimensionality reduction as any influence on the strong language prior as cited Goyal et al. for VQA v1.0 dataset.

4. DMN+ (glove 300 init) on VQA 2.0 to assess the baseline performance of dynamic memory plus architecture in VQA2.0 dataset.

5.2. Cross Transfer Learning

The experiment for cross-transfer learning involved plugging back post VQA training Resnet-40 for image classification task on CIFAR-10 dataset and analyze the difference in performance.

To test for the impact of language probabilities skewing our results, our third experiment was to use the VQA v2.0 dataset introduced by Goyal, et al., 2016. This dataset adds complimentary pictures for a given question to cancel out the positive case bias.

6. Results

6.1. VQA

The experimental results of DMN+ implementation on VQA 1.0 and VQA 2.0 for different instialization settings are presented in Table2. In VQA dataset each question is answered by mutliple people and the answers may not be the same. Hence the correct answer is considered as the human consensus. To get the accuracy, answer a is only fully correct if at least 3 people provide the same answer. If the answer is given be less than 3 people,accuracy is given as fraction of number of people gave the same answer and 3. The results are obtained as overall accuracy and accuracy depending on answer type. The dataset has 3 types of possible answers types, Yes/No, Number and Other.

Methods	Dataset	All	Yes/No	Number	Other
SAN(2, CNN)	VQA1.0	52.3	79.3	36.6	46.1
DMN+ glove 300 init	VQA1.0	52.64	77.3	29.34	31.37
DMN+ random init	VQA1.0	54.27	78.33	38.24	31.46
DMN+ glove 300 init	VQA2.0	47.78	66.25	34.34	29.22
DMN+ glove 100 init	VQA2.0	47.34	64.3	35.24	28.32

Table 1: Results for different Methods on VQA 1.0 and 2.0

In above table, SAN(2,CNN) is SAN with CNN as network for question representation and 2 is number of attention layers as shown in figure 2. For DMN+ glove 300, vector representation of words was initialized with 300 d glove representation and for glove 100 with 100 d glove representation. DMN+ Random init is method when word representation was randomly initialized.

6.1.1 Visualizing Attention

To visualize what the model is locating in the regions that are relevant to the potential answers we introduced a method to visualize attention on the original image. The yellow part of image is the attention region which is obtained using the attention weights. Attention image is created from Attention wegiths by reverse traversing the pixels in snake like fashion. Then this attention image is upsampled and blurred. This is then overlayed on original image. Some overlayed images are shown in figure 7 and 8. More images are showing in appendix.

6.2. Cross Transfer Learning

ResNet performance in our experiments drop drastically when resnet weights post VQA training are plugged back in for image classification task on CIFAR-10 dataset.

Architecture	Dataset	pre-VQA	post-VQA
Resnet-40	CIFAR-10	89.7	10.7

Table 2: Results for Cross-Transfer Learning Experiments

7. Conclusion

7.1. VQA

Visual question answering on V1.0 suffers from a strong language prior. The drop in Yes/No category of accuracy clearly indicates that when we moved to more balanced dataset, language priors no longer help. This finding is inline with observation made by Goyal et al. on VQA v1.0 and v2.0 dataset.

Another interesting observation in our experiment was that random word-embedding performed better than glob 300 initialization. This finding indicates that word vector trained on context do not necessarily help in question representation. It appears that architecture which derives attention on the question words as well (similar to Jiasen et al.) could possibly be sensitive to word vector initialization.

7.2. Cross Transfer Learning

After weight/bias visualization of Resnet layers participating in VQA task we notice a shift in modal weight distribution.

Since conv/residual unit weight undergo such shift while the final stage of resnet which did not participate in VQA training stay at the original value, the Resnet weights over all move to sub-optimal position in classification loss manifold and hence we see this catastrophic loss in the classification performance.

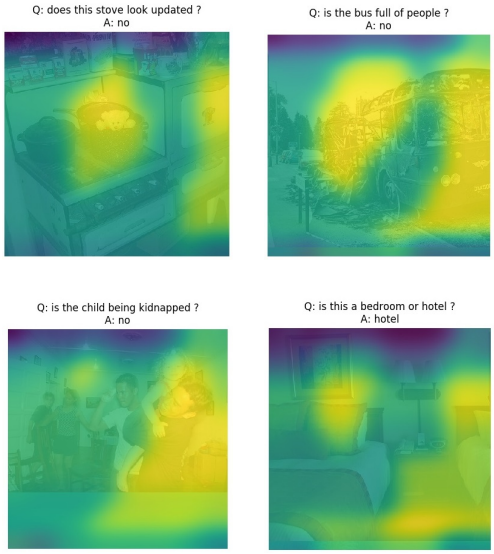


Figure 7: Correctly Predicted

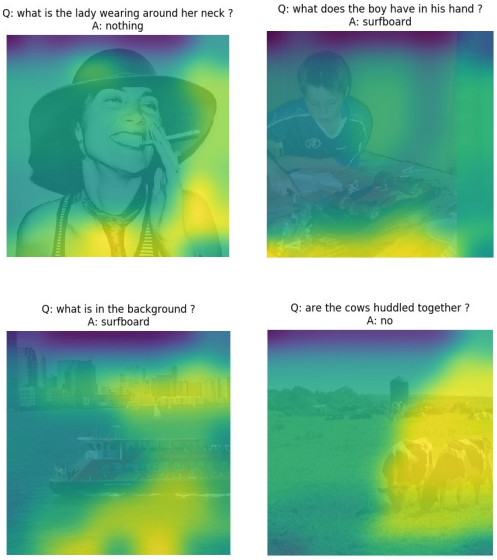


Figure 8: Incorrectly Predicted

7.3. Future Work

More extensive set of experiments are required to rectify the sensitivity to the language prior (responsible for drop in performance on v2.0 dataset). A better cross-transfer learning setting could be training of VQA and classification task similar to training of Generator and Discriminator networks in Generative Adversarial Network paradigm.

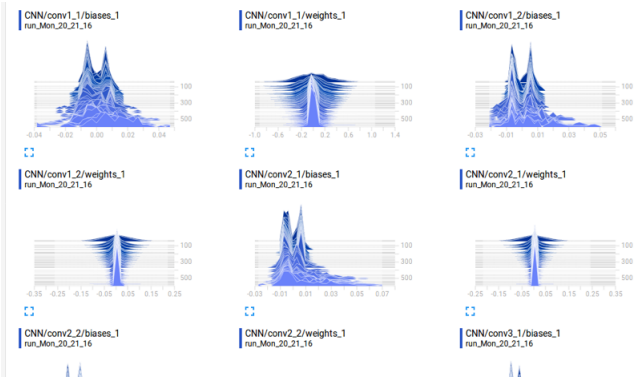


Figure 9: Shift of Resnet-40 CONV Weights and Biases of selected layers during VQA training

References

[1] Stacked Attention Networks for Image Question Answering", Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng and Alex Smola. To appear in CVPR 2016.

[2] Dynamic Memory Networks for Visual and Textual Question Answering Caiming Xiong, Stephen Merity, Richard Socher

[3] Hierarchical Question-Image Co-Attention for Visual Question Answering, Jiasen Lu, Jianwei Yang, Dhruv Batra, Devi Parikh

[4] Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering, CVPR 2017, ash Goyal and Tejas Khot and Douglas Summers-Stay and Dhruv Batra and Devi Parikh

[5] Convolutional Neural Networks for Sentence Classification Yoon Kim

[6] On the difficulty of training recurrent neural networks, Razvan Pascanu, Tomas Mikolov, Yoshua Bengio

[7] Memory Networks, Jason Weston, Sumit Chopra Antoine Bordes

[8] Introduction to LSTM, Sepp Hochreiter, Jurgen Schmidhuber

[9] Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell1 Marcus Rohrbach

[10] Show, Attend and Tell: Neural Image Caption Generation with Visual Attention Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, Yoshua Bengio

Github References:

[1] https://github.com/DeepRNN/visual_question_answering.git

[2] https://github.com/zcyang/imageqa-san.git (Base SAN implementation)

8. Appendix: Results

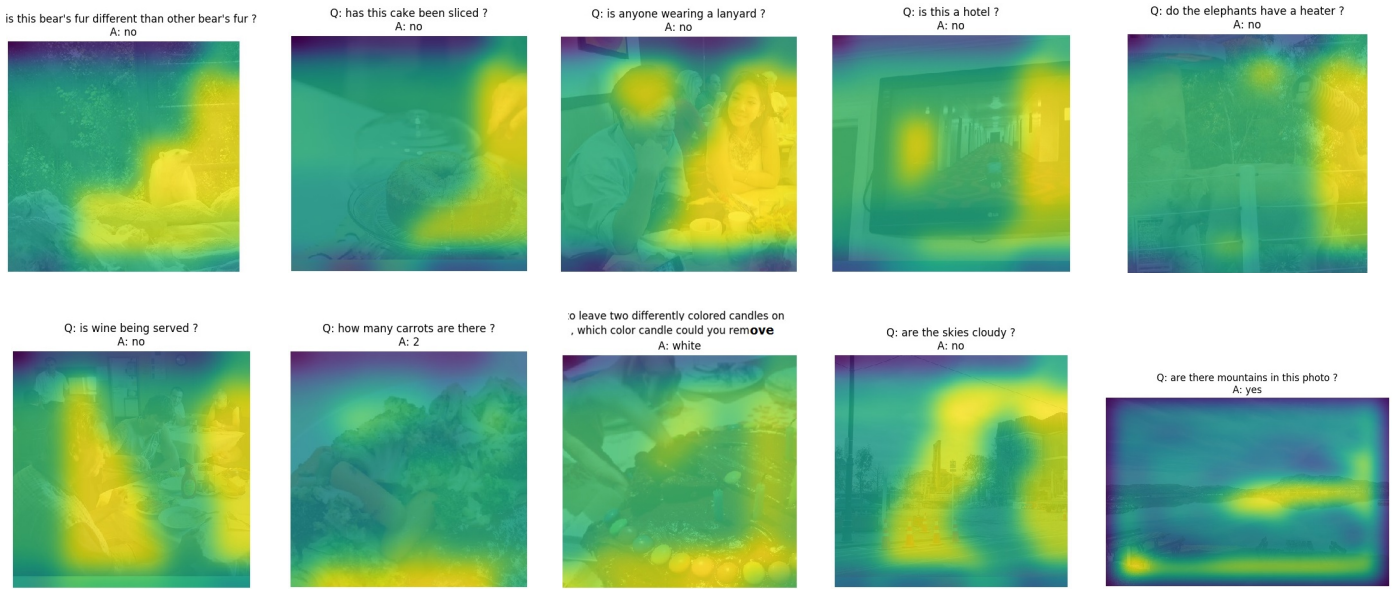


Figure 10: Correctly Predicted

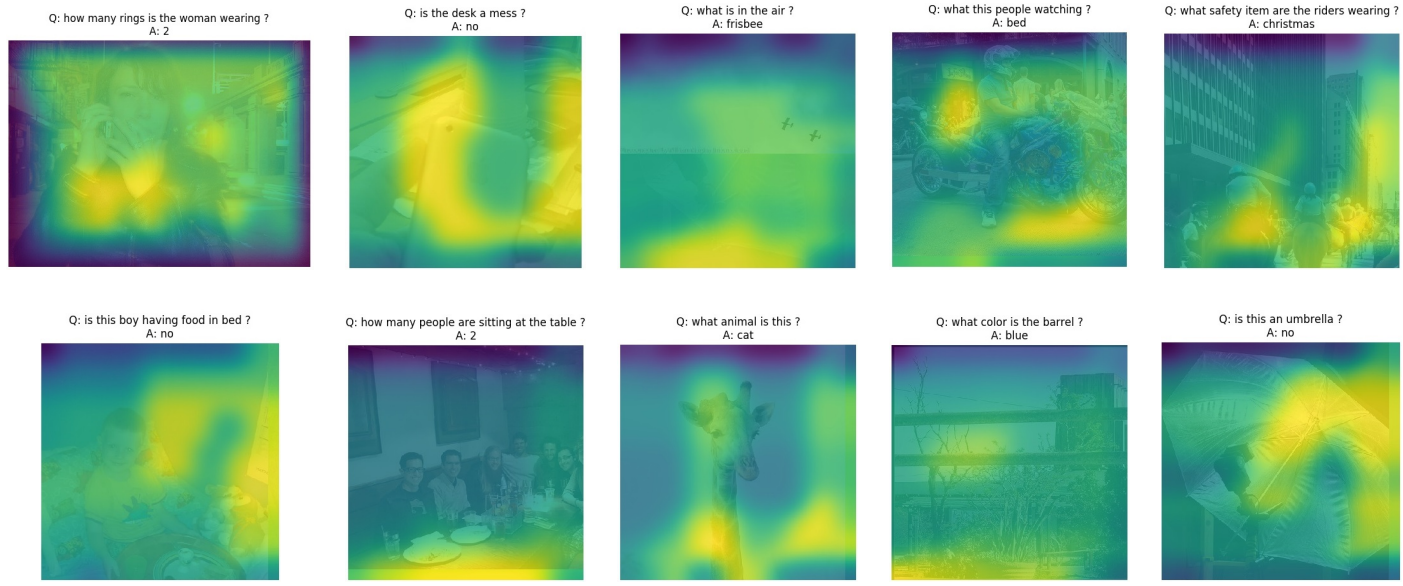


Figure 11: Incorrectly Predicted