

# Gated Attention for Visual Question Answering

Rakesh Radhakrishnan Menon\*  
rrmenon@cs.umass.edu

Arjun Suresh Karuvally\*  
akaruvally@cs.umass.edu

## Abstract

Visual Question Answering is a task that has been gaining a lot of attention in the last few years. This paper presents *gated attention networks* that attempts to solve this task by using an attention mechanism that is based on multiplicative interactions between the query and the image to answer questions. Our approach is shown to learn attention embeddings that focus on the subject of the query and semantically related word embeddings. We also test the effectiveness of our approach on the Visual Question Answering dataset against some baselines.

## 1 Introduction

Visual Question Answering (VQA) is an exciting problem that combines natural language processing with computer vision methods. Visual QA systems are also important for real-world applications such as helping the visually impaired understand what kind of scene they maybe having in front of them<sup>1</sup>. Much of the recent attention, that this field has been receiving, has come about because of the availability of large datasets such as the VQA dataset(Antol et al., 2015), DAQUAR (Malinowski and Fritz, 2014) and MS COCO (Lin et al., 2014). Some of the typical tasks in these datasets includes answers in the form of : a word, a phrase, a yes/no answer, choosing out of several possible answers.

The task is particularly challenging because the system has to understand about different parts of the query and the image while also having to create an effective link between the

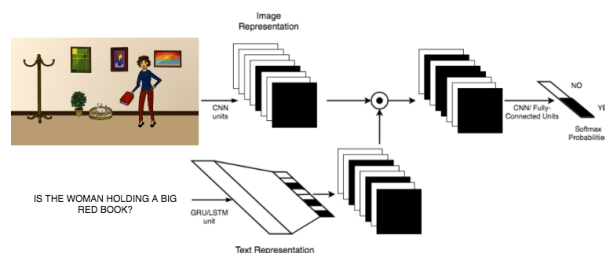


Figure 1: Our Gated-Attention architecture with multi-modal fusion for Visual Question Answering.

two modalities in order to perform high-level reasoning. An important component hence, of a Visual Question Answering system is its ability focus on the relevant parts of the image given the query. In the past, a concatenation of the image and query representations has been used to perform this task (Antol et al., 2015). More recent work has looked towards applying multi-layer soft attention mechanisms for performing reasoning at multiple-levels (Yang et al., 2016).

In this project, we propose the use of *gated* attention in order to perform multi-modal fusion for high-level understanding of images for answering yes-no questions. Our work differs from the work of (Yang et al., 2016) in that we do not compute an additive encoding of the two modalities to produce the attention vector for focusing on the image. Rather, we employ a mechanism where the query embedding itself becomes the attention vector and further perform multiplicative interactions between the attention vector and the feature maps that result from convolutions of the context image. Such an attention mechanism has been shown in the past to perform very well for language grounding (Chaplot et al., 2018) and text comprehension (Dhingra et al., 2017).

Equal contribution. Arranged by last names.

<sup>1</sup><https://itunes.apple.com/us/app/vizwiz/id439686043?mt=8>

In the next few sections, we go through some of the related work in this field (Section 2), followed by detailing our proposed *gated attention* approach (Section 3) and then explain our experimental setup in Section 4 along with some preliminary results.

## 2 Related Work

Prior attempts at VQA have tried to use a stacked-attention mechanism (Yang et al., 2016) that produces attention vectors by taking as input a concatenated representation of both the image and the query embedding. The attention vectors are further used for attending to different parts of the image over different layers in the convolution in order to produce relevant answers to the queries. A representative image of the architecture has been provided in Figure 3. Experimental results showed that the method outperformed previous baselines on 4 Visual QA datasets including COCO, DAQUAR and VQA 1.0. More recently, (Kazemi and Elqursh, 2017) used a very similar architecture to (Yang et al., 2016) but was able to obtain much better performances on the VQA 1.0 and VQA 2.0 datasets. Another work that takes a slightly different approach towards the use of attention for visual question answering was proposed in (Lu et al., 2016) called Hierarchical Co-Attention networks.

In (Lu et al., 2016), the author(s) propose different co-attention approaches that either jointly learn attentions for words and images, or separately for the two modalities. In the parallel(joint) co-attention model, the author(s) compute a similarity function between the query and image embeddings and then follow it with another fully connected layer to produce attention values for each of query and image embeddings. The alternating co-attention model, as the name suggests, computes the attention required for one-modality given the other. Furthermore, the author(s) propose the use of a hierarchical approach for predicting answers, which involves combining the embeddings from multiple encoding layers of the image and query encoders. The final encoder resembles a form of an LSTM which tracks both image and query embeddings. A figure which shows how the hierarchical structure is created has been shown in Figure 2. The

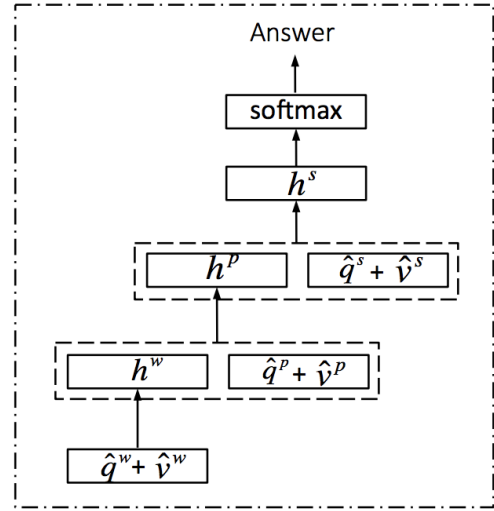


Figure 2: Hierarchical Encoder for Answer Prediction. Figure from (Lu et al., 2016).

method was able to give state-of-the-art results on the VQA dataset at the time.

Stacked Attention Networks were later introduced by (Yang et al., 2016), which proposed the use of multiple layers of attention in order to perform multi-step reasoning. To this end, the author(s) used a multi-layer LSTM to obtain query embeddings and subsequently computed the attention for the image embedding at different layers using a multi-layer perceptron that took as input the query embedding and the image embedding(for layers<sub>i</sub>1). A figurative representation can be seen in Figure 3. More recently in (Kazemi and Elqursh, 2017), produced an improvement over the method in (Yang et al., 2016). In this approach, the image is embedded using a convolutional neural network based on ResNet (He et al., 2016). The input is tokenized and embedded using multi-layer LSTM. The image embeddings and input query embeddings are used to get multiple attention distributions over image glimpses obtained from attention. This concatenated attention and the input query embeddings is fed into a final fully connected network to produce the distribution over all the answer classes. It is seen in both the papers that attention has a very positive effect on the performance on the visual question answering task.

*Gated Attention* models have had successes in text comprehension (Dhingra et al., 2017) where the

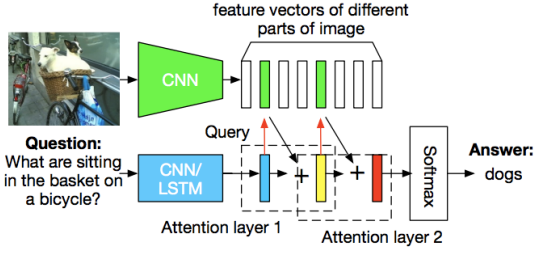


Figure 3: Stacked Attention Network for Visual QA. Image from (Yang et al., 2016).

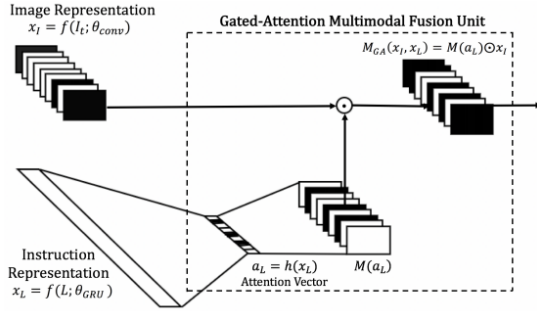


Figure 4: Gated Attention unit architecture. Image taken from (Chaplot et al., 2018)

authors integrate it with a multi hop architecture to create a Gated Reader. In the Gated-Attention (GA) mechanism, multiplicative interactions are performed between the query embedding and the context embeddings to create an effective representation that focuses on the required sub-text of the document to answer queries. The authors were able to achieve state-of-the-art results on three benchmarks for this task – the CNN & Daily Mail news stories and the Who Did What dataset. Recently, (Chaplot et al., 2018) has been able to extend the gated-attention mechanism for the task of language grounding in which autonomous agents need to extract the semantically meaningful representation of language and map it to visual elements and actions in the environment. Note that in case, the context embedding would be the feature maps from a convolutional neural network output. A representative figure of the gated-attention unit used in this paper has been shown in Figure 4. The proposed approach in this paper was further shown to outperform baseline models on multi-tasking and zero-shot task generalization.

### 3 Gated Attention for VQA

Our approach towards solving the task is best explained through the following example. Imagine the feature maps in the convolutional network capturing different semantics of the image such as, one feature map could be “recognizing yellow objects in the image” and the second could be “recognizing round objects in the image” among many such feature maps. In such a situation, suppose our question is “Is there a lime in the scene?”, the query could provide attention scores that activate the mentioned feature maps. This assumption that the feature maps capture better semantic features of the image when compared to flattened embeddings forms the basis of our “gated attention” approach.

Through our method, we look to learn feature maps that capture semantic representations from the image while also learning to embed natural language queries in a meaningful way to exploit information from the image. To this end, we apply the multi-modal fusion GA unit as proposed in (Chaplot et al., 2018) for mapping the semantic meaning of the query to the relevant portion of the input image. To describe the underlying equations in the GA unit, let us consider the notations from Figure 4. Here,  $x_L$  is the representation of the query and  $x_I$  is the representation of the image that has been provided for context. The query representation is further transformed into attention scores  $a_L$ , followed by a softmax to give the attention vector ( $M(a_L)$ ) that assigns probabilities to the different feature maps of the given context image. The attention vector and the features maps are multiplied element-wise to give the a representation for the “relevant” part of the image required for answering the query,  $M_{GA}(x_I, x_L)$ . Finally the output of the GA unit is passed through some fully connected layers and decoded to produce the output answer (via softmax classification). The equations for the GA unit have been summarized below,

$$a_L = h(x_L)$$

$$M_{GA}(x_I, x_L) = M(a_L) \odot x_I$$

A figurative representation of how our model looks like can also be found in 1.

### 3.1 Model details

The model is a deep neural network model that is broken down into four parts. Each part is concerned with a different functionality in the network. The parts are:

- **Image processor:** Image processor is a deep convolution network that handles the processing of image into feature maps. The output of the image processor is considered as the output of the final layer of convolutional network.
- **Query processor:** Query processor consists of a network that is concerned with converting the input query into a hidden dimensional representation. The encoded input query is processed sequentially, with an embedding layer that maps a word to its corresponding embeddings and an GRU/LSTM layer that processes the variable length sequence and maps it into a fixed dimensional space. The output of the query processor is an attention scores obtained by passing the fixed dimensional GRU/LSTM output to a linear layer. The output of the linear layer has dimension equal to that of the number of channels in the final layer of the image processor network. This is created to correspond to attention distribution over all the channels in the output of the network.
- **Multimodal fusion:** This part is concerned with combining the outputs of Image processor and query processor into a single output. To do this, the attention score obtained from the query processor is convolved with the output from image processor. The logic for how they are combined is as described in the previous section. The output will be a single stream obtained from both the image and query.
- **Final Decision network:** The final part of the model would be computing the scores of each possible answer which is achieved using a fully connected network. The output of this layer encodes all the possible answers in the dataset. This enables to convert the problem of VQA into a classification problem(prediction of which answer in list of answers given an input query and image).

The final scores obtained from the model for each score is converted to a probability distribution using softmax and the cross entropy of the distribution with the target is taken as the loss function. The target is a distribution that is obtained from the answers from different people. To enable easy training of network, early stopping is implemented based on validation loss.

## 4 Experiments

### 4.1 Dataset

VQA(Antol et al., 2015) is one of the widely used datasets for the task of Visual Question Answering. Questions and answers are generated from crowd-sourced workers. There are 10 questions for each image and each question has 10 answers that are obtained from unique workers. There are different types of answers in the dataset like:

- Binary yes/no
- Single word answers
- Multiple word choices
- Phrase based answers

We use the version 2 of the dataset which contains 20,000 images, 60,000 questions and 600,000 answers. The dataset is public and can be downloaded from <http://www.visualqa.org/download.html>. For this work, we will only be focusing on the yes-no type questions since our aim is to show how gated attention functions for visual question answering and the kind of attributes that are learned by the attention mechanism and the word embeddings of query words. Reducing the dataset to yes-no questions alone, we have about 40% of the complete dataset. Furthermore, There is a slight skew in distribution of yes and no. 'Yes' answer is about 52% of dataset and 'No' answer spans 48% of the reduced dataset. As it can be noticed here, the JSON files are not in a format that can be directly used by our training procedures. To enable this, we preprocess this dataset into a form that can be easily used.

After this, we split validation set obtained from VQA into a validation and test set. We randomly sample 1000 points from the validation set obtained to create a test set. The preprocessed training set is used entirely for training our networks.

## 4.2 Baselines

We propose the use of multiple baselines for checking how effective our proposed attention mechanism on the VQA dataset. For the baseline model, we replicate the model that was used in the VQA dataset paper (Antol et al., 2015). This architecture involves a VGG-16 network for the image processing module and a stacked 2-layer LSTM for the word embedding module. The output answer is obtained through a function that takes as input a concatenation of the image and query module embeddings (embeddings for image and query module are of length 512 each). The variants of this model that we use include:

1. **M1** : Random network with no training of any weights.
2. **M2** : Random network with pre-trained VGG-16 weights for the image processing module alone.
3. **M3** : Trained network with pre-trained VGG-16 weights (available in PyTorch) for the image processing module.

## 4.3 Evaluation

We use the standard evaluation metric for VQA to measure the performance of model and gain insight into the training of the proposed model. The metric takes the answer from the model and a score is computed as  $\text{score} = \min(\text{number of human who provided that answer}/3, 1)$ . An answer is considered as correct if it matches answers by at least 3 annotators. If it does not match any 10 possible answers, the score given is zero (Antol et al., 2015). This gives a quantitative measure of performance of models.

## 4.4 Ablation Studies

### 4.4.1 LSTM vs GRU

While LSTM models have been known to have a resounding success in many sequence-to-sequence models, recent work suggests the use of Gated Recurrent Units (GRU) for many natural language sequence to sequence tasks. The main attribution for this result has been because of the ability of the GRU to be more selective about the information it retains within different time steps of computation. To compare the two sequence models, we apply them both for our task of visual question answering. The general architecture we used for our

LSTMs was the same as the one used in our baselines except here we use gated attention (**M4**) with a sigmoidal activation for getting the attention map scores. While for the GRU architecture, we have used a single layer GRU and it’s final output (after reading in all words in the query) as our query embedding (**M5**). The results after about 20 epochs of training are shown in Table 1. The results suggest

Table 1: LSTM vs GRU Ablation.

Model	accuracy (%)
M4	64.03
M5	69.23

that GRU has an advantage over LSTM when using it for query embedding for the task of visual question answering. GRU model got 5% more accuracy than LSTM based model.

### 4.4.2 Learning Rate

An important consideration to be made for most deep learning models is the tuning of hyperparameters for learning good representations and getting good performances. To this end, we apply a learning rate ablation study on a small GRU network for a range of learning rates between  $1e-6$  and  $7e-5$ . The small GRU network (**M6**) that we use in this scenario, has an extremely shortened image processing module that uses 4 layers of convolution with lesser number of channels (32) and filters sizes of  $8 \times 8$  (stride=4),  $4 \times 4$  (stride=2),  $4 \times 4$  (stride=2) and  $4 \times 4$  (stride=2). The query processing module is also altered to provide a low-dimensional embedding for attention (32). The results under different learning rates can be found in Table 2.

Table 2: Learning rate Ablation. Best accuracy in **bold**.

Model	accuracy (%)
M6, $1e-6$	<b>69.1</b>
M6, $2e-6$	66.63
M6, $5e-6$	68.76
M6, $7e-6$	68.5
M6, $1e-5$	68.5
M6, $2e-5$	67.9
M6, $5e-5$	66.76
M6, $7e-5$	65.3



The best accuracy was achieved for the learning rate of  $1e^{-6}$ .

#### 4.4.3 Sigmoid vs Softmax

Once, we have obtained the query embeddings, there are two kinds of questions we can ask regarding attention, (a) Do we want the attention between words to affect each other? and (b) Do we need the attention to just check how important a particular word is with respect to its embedding. To answer these two questions, we use an experiment that uses the sigmoid activation (**M5**) and the softmax activation (**M7**) to compare the models. The relation (a) corresponds to the softmax activation while relation (b) corresponds to the sigmoid activation. These relations have been deduced by taking into account the functional types. The results can be seen in Table 3.

Table 3: Sigmoid vs Softmax Ablation.

Model	accuracy (%)
M5	69.23
M7	70.33

It is seen that softmax layer has slightly higher accuracy compared to sigmoid. This suggests that the task requires the attention between different words to affect each other.

#### 4.5 Comparison of all Models

In this section, we would like to highlight the results from all the models that have been experimented over so far from all the models that we have trained. Additionally, we also mention the score that has been obtained by some of the current state-of-the-art methods in (Kazemi and Elqursh, 2017) and (Lu et al., 2016). We also present some of the sample outputs from our network where it predicted both the correct answers as well as the wrong answers in Table .

### 5 Analyses

#### 5.1 Learned Word Representations

Word embedding analysis has been used in literature to suggest the effectiveness of various learning models. The argument to this is that models learn meaningful word embeddings according to the task it has been exposed to. In order to analyze the effectiveness our model, the embeddings

Table 4: Accuracy results for all the Models. We have highlighted the top 3 results in this table in **bold**.

Model	accuracy (%)
M1	50.529
M2	54.133
M3	56.12
M4	64.133
M5	69.23
M6	69.1
M7	<b>70.33</b>
Hierarchical Co-Attention	<b>71.80</b>
Strong-Baseline VQA	<b>77.45</b>

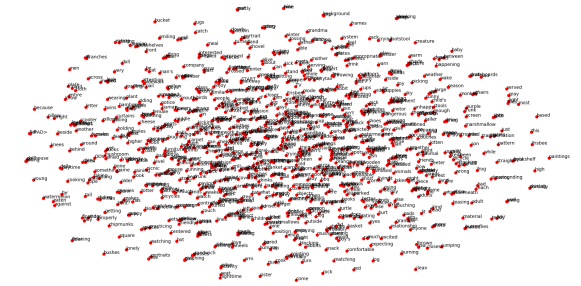


Figure 5: Word embeddings learned by our model for words in the vocabulary.

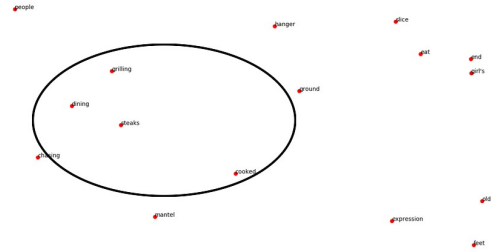







Figure 6: Zoomed in version of the embedding space that shows words related to cooking having embeddings near to each other

of words in vocabulary are mapped to a two dimensional space using t-SNE (Maaten and Hinton, 2008). This lower dimensional data can be easily visualized using a scatter plot. The visualization of embeddings learned by our model is shown in Figure 5.

It is observed that words have meaningful embeddings in that some words that have semantic similarity based on context occur near to one another.

Table 5: Sample results for VQA questions from our best network.

Image	Question	Predicted Answer	Correct Answer
	Does a young girl live in this house?	YES	YES
	Are they waving at each other?	YES	YES
	Will the curtain catch fire?	NO	NO
	Is the man looking at the sleeping dog?	NO	NO
	Is the woman going to fall?	YES	NO

It should be noted here that by projecting to a lower dimensional space, there is loss of information and it is not possible for the t-SNE method to capture all the words which are close to one another in the lower dimensional space it is projecting to.

Figure 6 shows words that have the context of cooking occurring next closer to one another that suggest that the model is able to learn meaningful representations by training it in the task of visual question answering.

## 5.2 Sentence Representation

Observing that the model learned a reasonable word representation, we study how different input queries are embedded by the GRU into the latent space. To study this, we input custom sentences into the model and take the embeddings obtained from the input query processing network - the GRU unit. The embeddings obtained are projected into a lower dimensional space for visualization using t-SNE.

From Figure 7 it can be observed that input queries containing the same subject (like woman,

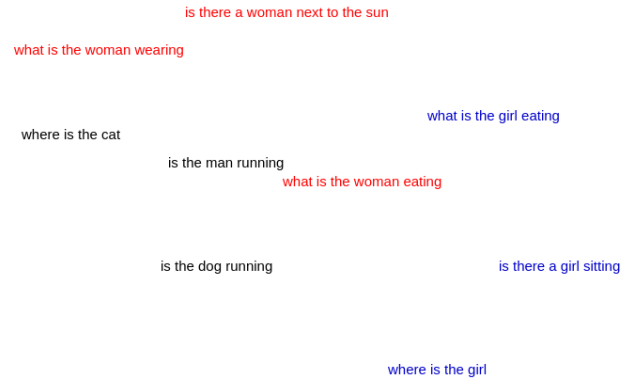


Figure 7: Sentence embeddings taken after query processing. Red sentences are the sentences with woman as subject. Blue sentences have girl. Black sentences are sentences in that has subject neither girl nor woman.

girl) occur near one another in the latent space suggesting the model was able to capture information about the subject referred to in the query.

## 6 Conclusion and Future Work

In this paper, we have introduced a new attention mechanism for the task of Visual Question Answering. With some ablations and analyses, we have shown how some of our proposed *gated attention* mechanism is capable of giving really good results on the yes-no question answering task. The semantically related word embeddings learned by our model is indicative of the attention mechanism’s ability to grasp concepts from a really hard multi-modal setting.

In the future, we would like to perform a multi-layered *gated attention* attention approach similar to the architectures in (Kazemi and Elqursh, 2017; Yang et al., 2016) as it has been shown to allow learning of multiple steps of reasoning. Further, we could also try to incorporate pre-trained word embeddings, like GLoVe (Pennington et al., 2014), Word2Vec (Mikolov et al., 2013) or even fastText (Bojanowski et al., 2017), as a starting point for our network to learn. Unlike the image features, using pre-trained word embeddings can help a lot more since it is directly linked to the attention scores that the network learns and hence having deep semantic understanding of different words will help. Similarly on the image embedding layer side, we could look towards using deeper networks like ResNets.

## References

- Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2425–2433.
- Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.
- Devendra Singh Chaplot, Kanthashree Mysore Sathyendra, Rama Kumar Pasumarthi, Dheeraj Rajagopal, and Ruslan Salakhutdinov. 2018. Gated-attention architectures for task-oriented language grounding. *AAAI 2018*.
- Bhuwan Dhingra, Hanxiao Liu, Zhilin Yang, William Cohen, and Ruslan Salakhutdinov. 2017. *Gated-attention readers for text comprehension*. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1832–1846. Association for Computational Linguistics.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.
- Vahid Kazemi and Ali Elqursh. 2017. Show, ask, attend, and answer: A strong baseline for visual question answering. *arXiv preprint arXiv:1704.03162*.
- Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.
- Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical question-image co-attention for visual question answering. In *Advances In Neural Information Processing Systems*, pages 289–297.
- Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605.
- Mateusz Malinowski and Mario Fritz. 2014. 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.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. *Glove: Global vectors for word representation*. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.
- Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. 2016. Stacked attention networks for image question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 21–29.