VISUAL QUESTION ANSWERING SYSTEM: A COMPREHENSIVE ANALYSIS

SREE GOWRI ADDEPALLI, SREE LAKSHMI ADDEPALLI
COURANT INSTITUTE OF MATHEMATICAL SCIENCES, NEW YORK UNIVERSITY

***Abstract*— Question answering systems have been always the heart of natural language processing tasks. Visual Question Answering (VQA) System is a hot research topic both in academia and industry. The growth of Artificial Intelligence especially in Deep Learning, has facilitated us to solve tasks that lie in the junction of Computer Vision, Knowledge Representation and NLP. VQA gives an answer to the question asked regarding the image shown. There are various datasets that have been created to serve this purpose. Through this paper we attempt to analyze the various datasets created for this task and analyze the evolution and performance of various models created using techniques involving deep learning. Finally, we try to get a scope of the future direction of visual question answering research and the possibilities it creates.**

Keywords: Natural Language Processing, Computer Vision, Knowledge Representation, Deep Learning, Neural Networks, Pretrained Models, Convolutional Neural Network, Recurrent Neural Network, Long Short-Term Memory, Transfer Learning, ResNet, GoogLeNet, VGGNet, Word Embeddings.

I. INTRODUCTION

Computer Vision and NLP have processes specific to them for solving tasks related to each respectively. With the era of deep learning many problems like image captioning, visual question answering system, content-based image retrieval etc. use techniques that are common between the two. Building VQA with good accuracy would be a major achievement in the field to improve human machine interaction. This review paper aims to analyze the latest trends in this domain. We aim to bring out the following 1) Analysis and evaluation of various datasets present. 2) Comparison and evaluation of various models and algorithms present to solve this task. 3) A design of our proposed model. Finally, we review the future direction of the VQA system and the future advancement in this field. Throughout the paper references have been made to the referenced article and their authors to understand how the system evolved and their contribution towards the current state of art system.

.

II. DATASETS AND PRIOR WORK

Data is the most important part of any deep learning or machine learning task which sets the base. The deep learning model consists of many hyperparameters which can be tuned during the training phase to get a good model. The better the quality of data the more generalized the model can be.

A. DAQUAR

The Dataset for Question Answering on Real World Images (DAQUAR) was the first significant dataset for Visual Question Answering system. It has images based on NYU DEPTH v2 Dataset. There are a total of 1449 images (795 training, 654 test) and a total of 12468 question answer pairs. Every image has objects and is labelled accordingly to the 894 object classes present. An example is “what is on the bed in the image786 ?” whose answer is rug. So, the ideal template for the question is “what is [object] in the image[image_ID]?” So, the answers to them was collected by 5 human participants who were allowed to answer with constraints that the answers can either be colors, classes or numbers. There are a few drawbacks of using the DAQUAR dataset. The NYU dataset consists of only indoor scenes with not great lighting conditions. The size of the images in this dataset make it unsuitable for developing complex models after training. There also exists a highly reduced dataset which has 37 classes in it. There are set of two evaluation strategies used here. Firstly, accuracy is used as a measure but that is not a good measure for multi word answers. The author then introduces WUPS as a score which generalizes the accuracy measure and handles word level ambiguities which ranges from 0.0 to 1.0. WUPS score is responsible for finding the semantic distance using wordnet for computing distance between the answer and ground truth. If the WUPS score is more than 0.9, it means that the answer generated regrading it is correct.

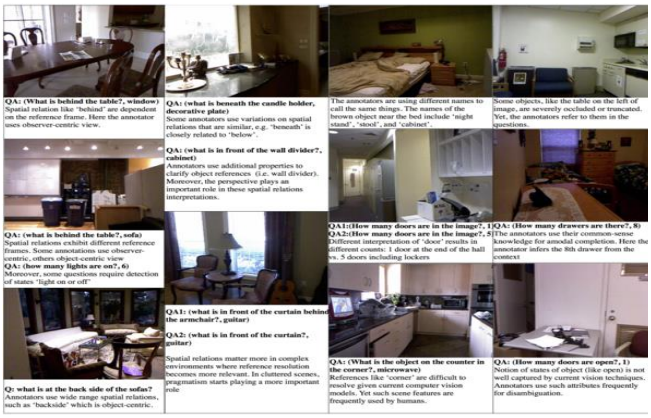


Figure Reference: Challenges in DAQUAR Dataset (https://datasets.d2.mpi-inf.mpg.de/mateusz14visual-turing/challenges.pdf)

	What is on the right side of the cabinet?	How many drawers are there?	What is the largest object?
Neural-Image-QA:	bed	3	bed
Language only:	bed	6	table

Table 7. Examples of questions and answers. Correct predictions are colored in green, incorrect in red.

	What is on the refrigerator?	What is the colour of the comforter?	What objects are found on the bed?
Neural-Image-QA:	magnet, paper	blue, white	bed sheets, pillow
Language only:	magnet, paper	blue, green, red, yellow	doll, pillow

Table 8. Examples of questions and answers with multiple words. Correct predictions are colored in green, incorrect in red.

	How many chairs are there?	What is the object fixed on the window?	Which item is red in colour?
Neural-Image-QA:	1	curtain	remote control
Language only:	4	curtain	clock
Ground truth answers:	2	handle	toaster

Figure Reference: Example of questions from DAQUAR dataset with answers and prediction. (https://www.d2.mpi-inf.mpg.de/sites/default/files/iccv15-neural_qa.pdf)

B. COCO- QA

The COCO-QA is a dataset generated through the COCO images and the question answer pairs were generated through a natural language processing algorithm through the image captions present. The answers generated here are single word. There are total of 117,684 questions of which 78,736 images are from the training set and 38,948 images are from the test set. Here, the evaluation is done using accuracy and WUPS score. As the question answer pairs are algorithmically generated, they are bound to have their own limitations apart from grammatical errors. Moreover, the dataset has questions unevenly generated related to object that constitutes 69.84% questions, color which related to 16.59%, counting which relates to 7.47% and location which is 6.10% of it. A sample question could be of the type when generated from an image caption could be A [object] is buying a [color] object, and the question generated could be what color is the [object] which is bought? The color is the answer for it.

COCO-QA: What does an intersection show on one side and two double-decker buses and a third vehicle,?
Ground Truth: Building

Figure Reference: COCO-QA Image sample (https://tryolabs.com/blog/2018/03/01/introduction-to-visual-question-answering/)

C. VISUAL QA

Visual QA is considered an important dataset after it was released for the Visual QA challenge. The Visual QA dataset supports the task of asking free and open-ended questions. As appeared previously of generating question pairs from image captions does not appear AI -complete and it restricts the users from asking a more general question. From [6], AI-complete tasks should understand multi-modal knowledge beyond a single domain such as natural language processing or computer vision. Moreover, it should have a well-defined

quantitative metric to track progress. This dataset explicitly aims to be able to answer queries for the questions posed by visually impaired users or intelligence analysts. As stated in [6], open-ended questions require advanced AI capabilities like fine grained recognition like what kind of cheese is on pizza? , object detection like How many bikes are there?, activity recognition like whether this man is crying?, knowledge base reasoning like is this a vegetarian pizza? and commonsense reasoning like whether this person has 20/20 vision? The images in the VQA dataset that contain real world images come from COCO dataset and another dataset contains abstract clip art.

There are two major versions regarding this dataset. The V2 dataset 443,757 questions in the training set, 214,354 questions in the validation set, 447,793 questions in the test set. Real-world images are distributed as 82,783 training, 40,504 validation and 81,434 testing. The abstract set of images contain 20,629 training images, 10,696 validation images 22,055 training questions and 11, 328 validation questions. The V1 dataset contains 248,349 questions in the training set, 121,512 questions in the validation set, 244,302 questions in the test set. While it contains the following distribution of real-world images as 82,783 training images, 40,504 validation images and 81,434 testing images. The abstract set of images contain 20,000 training images, 10,000 validation images, 20,000 testing images, 60,000 training questions, 30,000 validation questions and 60,000 testing questions.

This dataset also provides two modalities of answering the question. The algorithm can itself generate a free form open ended answer or select from a multiple choice. The real images contain multiple objects and rich contextual information [6] whereas the abstract scene image dataset was constructed in exploring the high-level reasoning but not the complex and noisy visual recognizers needed to be used in the real images. The splits for the train/val/test set remain the same as that done for the COCO image dataset.

Per image three questions and ten answers were correspondingly collected with the help of people workers. As mentioned in [6] for open ended questions, the answer is 100% accurate if at least 3 workers provided the same answer.

$$\text{accuracy} = \min\left(\frac{\#\text{ humans that provided that answer}}{3}, 1\right)$$

Figure Reference: Accuracy in open ended questions.[6]

For multiple choice questions as per [6] they created 18 answers which included both correct and incorrect answer for each question of which the correct answer is the most frequent answer given by the ten annotators. The three answers collected from annotators without looking at the image is considered the most plausible answer as per [6]. The most popular answers are the ten most popular answers in the dataset. The random answers are randomly selected corrected for other questions.

Figure Reference: VQA image [6]

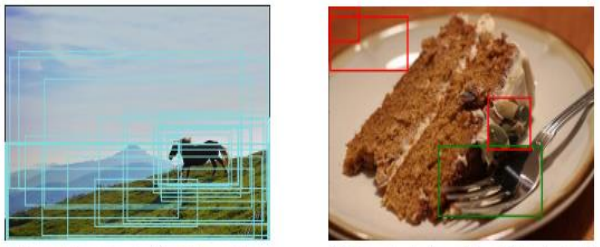
Dataset	Input	All	Yes/No	Number	Other
Real	Question	40.81	67.60	25.77	21.22
	Question + Caption*	57.47	78.97	39.68	44.41
	Question + Image	83.30	95.77	83.39	72.67
Abstract	Question	43.27	66.65	28.52	23.66
	Question + Caption*	54.34	74.70	41.19	40.18
	Question + Image	87.49	95.96	95.04	75.33

Figure Reference: Test Set Accuracy on VQA data.[6]

D. VISUAL7W

Visual7W is a dataset also made from the COCO Dataset and as per [7] is a subset of visual genome and contains 47,300 images and 327,929 questions. It is named after seven categories of question like what, where, who, why, how, which and when. The questions are generated by workers and questions who have votes less than two were discarded. The workers also drew bounding boxes on objects to resolve textual ambiguity and enable answers of visual nature. The dataset has two question types where the telling question have answer on the text level [7], while the pointing question which have 'which' component use an algorithm to select one of the bounding box amongst the alternatives present as per [7]. It doesn't contain any binary questions. Visual 7w presents

multiple choice answers to the user which are the bounding boxes surrounding the images. The answer choices come from the plausible answers where the answers are answered by workers without looking seeing the images.



(a) Example image from the Visual Genome dataset along with annotated image regions. This figure is taken from [35].
Free form QA: What does the sky look like?
Region based QA: What color is the horse?

(b) Example of the pointing QA task in Visual7W [34]. The bounding boxes are the given choices. Correct answer is shown in green
Q: Which object can you stab food with?

Figure Reference: Image from Visual7W dataset [7]

E. VISUAL MADLIBS

Visual Madlibs has two tasks of filling in the blanks when presented with an image and choosing a multiple-choice question. A total of 360,001 details for 10,738 images are present. The images have been collected in Madlibs style and exists 12 type of fill in the blanks. As stated in [10], they consist of General scene, Emotional content, what happened before, what will happen next, the most interesting part, Appearance, activity and location of each person, Appearance, affordance and position of each object, Interaction between people and object.

1. This place is a park.
2. When I look at this picture, I feel competitive.
3. The most interesting aspect of this picture is the guys playing shirtless.
4. One or two seconds before this picture was taken, the person caught the frisbee.
5. One or two seconds after this picture was taken, the guy will throw the frisbee.
6. Person A is wearing blue shorts.
7. Person A is in front of person B.
8. Person A is blocking person B.
9. Person B is a young man wearing an orange hat.
10. Person B is on a grassy field.
11. Person B is holding a frisbee.
12. The frisbee is white and round.
13. The frisbee is in the hand of the man with the orange cap.
14. People could throw the frisbee.
15. The people are playing with the frisbee.

Figure Reference: Example from Visual Madlibs dataset. [10]

F. FM- IQA

The freestyle multilingual Image Question Answering dataset is made from COCO dataset that uses a crowdsourcing server as mentioned in [5] to generate the pairs of questions and answers where the answers can either be words, either phrases or either full sentences. Question Answers pairs are available in Chinese along with English translations. This dataset consists of 158,392 images and 316,193 questions as stated in [5]. The evaluation is done here based on human judges giving a score from 0-2. This approach is not practical for research groups and developing algorithms based on it is difficult.

G. CLEVR DATASET

The CLEVR (A diagnostic dataset for compositional language and elementary visual reasoning as per [21]) dataset is presented by STANFORD to test various ranges in the abilities of visual reasoning. Unlike other datasets, this dataset has biases to minimal and in detail consists of reasoning as per it's requirement. The dataset here has the computer system answer question about the features of an object like the shape, color, size and its spatial and logical relationship. Every question in this dataset has a representation in natural language and functional program. The functional program here is quite responsible for determining the reasoning skill required to answer these questions. It consists of 70,000 images and 699,989 questions in training set, 15,000 images and 149,991 questions in validation set and 15,000 images and 14,988 questions in test set. It also contains answers for all the training and validation questions. Scene graph annotations are present as stated in [21] for train and val images giving ground truth locations, attributes, and relationships for objects. A functional program representation is present for all training and validation images.

Questions in CLEVR test various aspects of visual reasoning including **attribute identification**, **counting**, **comparison**, **spatial relationships**, and **logical operations**.

Q: Are there an **equal number** of **large things** and **metal spheres**?
Q: **What size** is the **cylinder** **that is left of** the **brown metal** thing **that is left of** the **big sphere**?
Q: There is a **sphere** with the **same size as** the **metal cube**; is it **made of the same material as** the **small red sphere**?
Q: **How many** objects are **either** **small cylinders** or **red** things?

Figure Reference: Sample questions from the CLEVR dataset. [21]

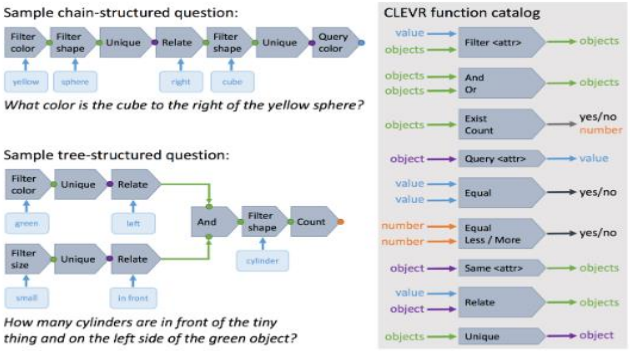


Figure reference: Functional representation to get the reasoning skill required to answer the question.[21]

H. FIGUREQA DATASET

The FigureQA is another VQA dataset specific for visual reasoning for graphical plots and figures. Here the questions are related to each other which require comparison of many elements of the plots within. The answers to the questions are binary in either yes or no. The questions are related to attributes which are quantitative. The questions are about properties like minimum, maximum, greater and less than, medians, curve roughness and area under curve as stated in [23]. The images are comprised of figures like vertical bar graph, horizontal bar graph, line graph, dot line graph and pie chart which are frequently found in analytical documents.

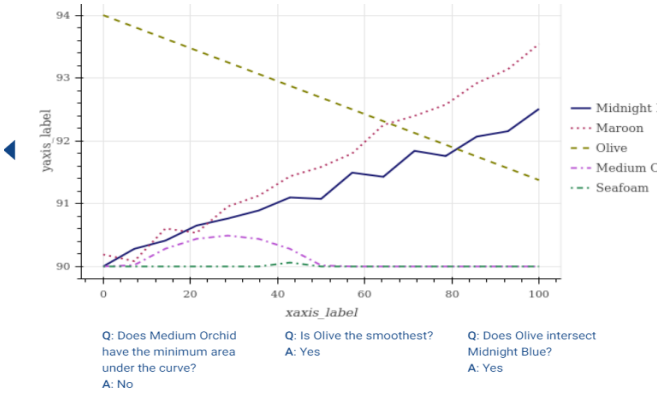


Figure Reference: Sample Question Answer from FigureQA dataset [23].

I. VISUAL GENOME

Visual Genome dataset is one the largest Visual Question Answering System with about 1.7 million Visual Question Answers. It consists of 108,077 images in all with an average of 17 questions per image. The following dataset has no binary questions to include complexity in asking questions. The dataset as mentioned in [7] consists of six major type of 'W' questions – 'What', 'Where', 'When', 'Who', 'How' and 'why'. Instead of asking only quantitative or general questions the workers were also made to ask questions related to specific image regions to improve diversity. The average question length is 6.0 +/- 1.9 and the average answer length is 1.9 +/- 1.3.

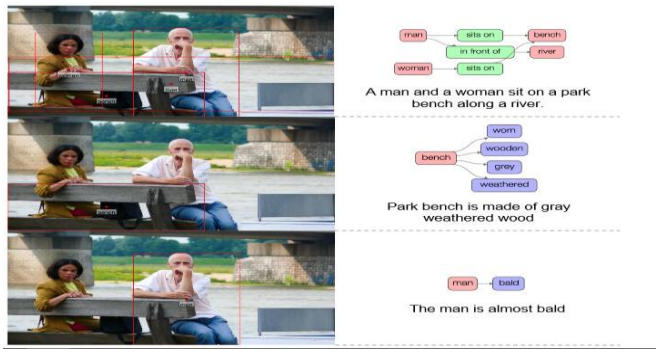


Figure Reference: An image from the dataset [26]

DATASET	No. of Images	No. of Questions	Questions per Image	Q/A generation
DAQUAR	1449	12,468	11.5	Human
Visual7W	47,300	327,939	6.9	Human
Visual Madlibs	10,738	360,001	4.9	Human
COCO-QA	117,684	117,684	9.65	Automatic
FM-IQA	158,392	316,193	7.38	Human
VQA (COCO)	204,721	614,163	6.2	Human
VQA (Abstract)	50,000	150,000	6.2	Human
CLEVR	100,000	864,878	8.64	Automatic
Visual Genome	108,077	1,837,309	17	Human

Table Reference: Evaluation of VQA datasets. [5]

III. DATASET EVALUATION

The datasets can be evaluated as related to the following table above referred from the paper [5].

There are majorly two type of tasks for VQA, open ended questions and multiple-choice questions. The multiple-choice question is easier to analyze for accuracy. The correct answer picked up by the algorithm is used for the evaluation of the algorithm performance. The open ended VQA algorithms usually have a difficult time evaluating performance as the output should be the same as the ground truth answer. Like if the question asked was what are the objects in the photo? If the answer given was trees rather than tree, the algorithm could penalize it hard. There could also be multiple correct answers for it, and this could affect the algorithms performance too. Many evaluation techniques have been proposed for evaluating open ended Visual Question Answering tasks. Wu-Palmer Similarity (WUPS) is one such measure as discussed above which tries to measure the difference in semantic meaning and the value range lies between 0 to 1. A threshold to W47300UPS score was mentioned as 0.9 to ensure that a score above it indicates a correct answer. WUPS has limitations, where two semantically words but having stark contrast words have higher WUPS score which is an anomaly. This is highly visible in colors where if the answer is black and the predicted answer is white, they still have a high WUPS score of 0.91. Another problem with WUPS, it fits well with single word answers rather than sentences which are found as answers in visual7W and VQA datasets. An alternative to this was to collect multiple ground truth answers. For DAQUAR dataset, an average of five human annotators were collected and a minimum and average consensus was collected to measure the answers value against it. The answer needs to match minimum one of the human annotators answer. Like in VQA dataset at least 3 answers must be matched. The authors of FM- IQA recommended usage of human judges for assessing multi word answers which present multiple problems as stated in [7]. The use of human resources is expensive both in terms of time and expenses. Moreover, altering the algorithm to tune it to improve performance would be an issue in terms of human resources. There are two metrics proposed by creators of FM-IQA for evaluating multi word answers for human judges. Firstly, the judges should distinguish whether the answer was human produced or non-human produced? Secondly, another measure is rating an answer with 2 holding fully correct, 1 indicating partially correct and 0 indicating completely wrong. Alternative to handling multi word answers was to give the judges multiple choice answers which makes the evaluation easier as done in visual7w, VQA and visual Genome dataset.

The best evaluation strategy for a VQA system is still open to various interpretations and question. There are pros and cons of every evaluation strategy. Every dataset has its own biases within it on how it was constructed, and we need to build a more efficient to handle these biases and handling multiword answers. The below table referred from [7] shows the pluses and minuses of each evaluation strategy.

	Pros	Cons
Simple Accuracy	• Very simple to evaluate and interpret • Works well for small number of unique answers	• Both minor and major errors are penalized equally • Can lead to explosion in number of unique answers, • especially with presence of phrasal or sentence answers
Modified WUPS	• More forgiving to simple variations and errors • Does not require exact match • Easy to evaluate with simple script	• Generates high scores for answers that are lexically related but have diametrically opposite meaning • Cannot be used for phrasal or sentence answers
Consensus Metric	• Common variances of same answer could be captured • Easy to evaluate after collecting consensus data	• Can allow for some questions having two correct answers • Expensive to collect ground truth • Difficulty due to lack of consensus
Manual Evaluation	• Variances to same answer is easily captured • Can work equally well for single word as well as phrase or sentence answers	• Can introduce subjective opinion of individual annotators • Very expensive to setup and slow to evaluate, especially for larger datasets

Figure Reference: Evaluation metrics of accuracy comparison. [7]

IV. DEEP LEARNING TERMINOLOGY AND PRIOR WORK

Following are the concepts that will be eventually be used in the models discussed below.

A. NEURAL NETWORKS

Neural Networks are based on the design of human brain which are made to recognize patterns. These patterns recognized are numerical contained in vectors and all real-world data, images, sound, text and time series must be translated into it. Neural Networks contain layers of nodes where in each node, computation occurs, and the layers are like switches that turn on and off. A basic neural network consists of one input layer, one output layer and many hidden layers.

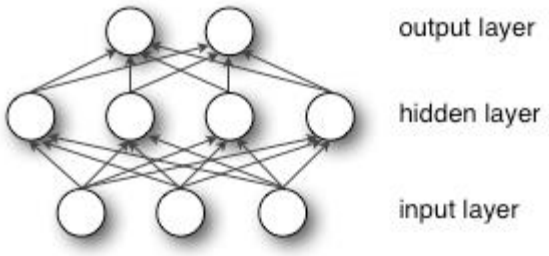


Figure Reference: Neural Network Architecture. (https://skymind.ai/wiki/neural-network)

B. CONVOLUTIONAL NEURAL NETWORK

Convolutional neural networks (CNN) are famous set of deep neural networks that usually take image as an input. Convolutional neural networks are biologically inspired as the connectivity pattern between neurons resembles the organization in human visual cortex. In traditional artificial neural networks, every layer is fully connected to other every node in the next layer. Convolution neural networks are stack of convolutions of each other connected through non- linear activation functions where convolutional filters are used to slide over nodes in each layer to compute the output. Different layers use different filters to compute the output. The three basic components of a CNN are the convolutional layer, the pooling layer which is optional and the output layer. Pooling is done only to reduce the spatial size of the image.

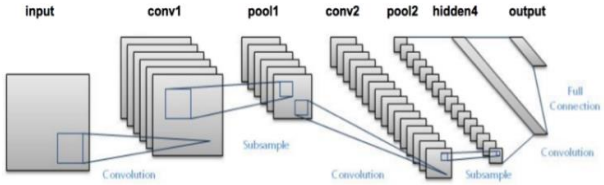


Figure Reference: Example of a CNN. (https://www.analyticsvidhya.com/blog/2017/06/architecture-of-convolutional-neural-networks-simplified-demystified/)

C. RECURRENT NEURAL NETWORKS

RNN or recurrent neural network is a class of deep neural network where the connection among nodes form a directed graph along a temporal sequence. Due to this they exhibit temporal dynamic behavior and use memory or internal states to process sequence of inputs. They perform the same manipulation for each element of the input sequence. The output here depends on the current element as well as the previous computations.

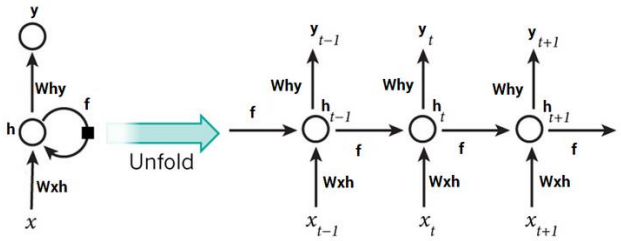


Figure Reference: Recurrent Neural Networks (https://www.analyticsvidhya.com/blog/2017/12/introduction-to-recurrent-neural-networks/)

D. LONG SHORT-TERM MEMORY

Long short-term memory (LSTM) is a type of RNN and can be used as a block in building larger recurrent neural network. LSTMs consists of four main components which are the cell which is primarily responsible for remembering values within the inconsistent time intervals, input gate, output gate along with a forget gate. These three gates act as feedforward neural network as they compute an activation function over it's weighted sum. LSTMs are primary useful in natural language processing tasks and were clearly responsible for improving machine translation, language modelling, text to speech synthesis. They are nowadays being used in combination with other deep neural networks to solve problems across domains like image captioning, visual question answering systems.

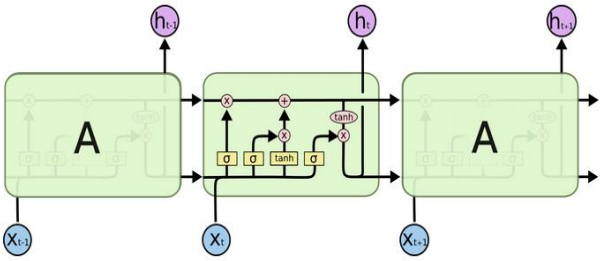
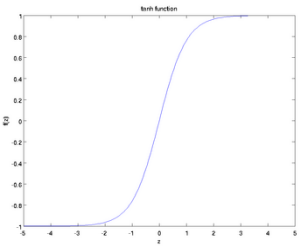


Figure Reference: Long short-term memory representation. (https:/colah.github.io/posts/2015-08-Understanding-LSTMs/)

E. ACTIVATION FUNCTIONS

Activation functions are used to get output of a node with a given set of input nodes. Activation functions are can both be linear and non-linear in fashion. Non-Linear activation functions are crucial because it helps the model generalize while adapting to variety of data. Some of the activation functions which are usually used while building such systems are:

I. HYPERBOLIC TANGENT OR TANH:



$$f(x) = tanh(x) = \frac{2}{1+e^{-2x}} - 1$$

Figure Reference: https://medium.com/the-theory-of-everything/understanding-activation-functions-in-neural-networks-9491262884e0

Tanh is a non linear activation function which is well known for its steep gradient and is very popular choice for back propagation. Tanh also has vanishing gradient problem.

II. RECTIFIED LINEAR UNIT OR RELU:

ReLU is a non-linear activation function and any combination with them is also non-linear. The range of ReLU is from [0, inf). It means that ReLU can blow up activation. This is ideally used when we want to activate only a few neurons in neural network to make the activations sparse and efficient. It has been widely used since it has been proven that it's usage has resulted to improve the performance of deep learning models.

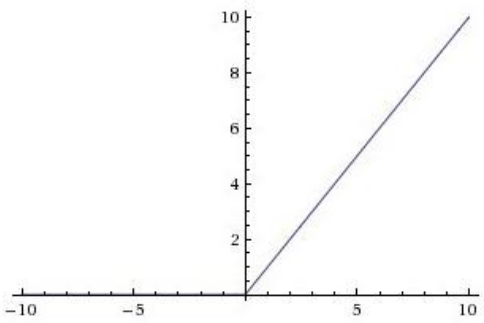


Figure Reference: https://medium.com/the-theory-of-everything/understanding-activation-functions-in-neural-networks-9491262884e0

It is represented as:

f(x) =0 for x<0
f(x) = x for x>=0

III. SOFTMAX:

It is usually used in the last layer that turns into probabilities over different classes that sum up to one. It is generally used for multi class classification as it mathematically transforms any distribution over n classes into a probability distribution in the same range of (0,1) over those n classes.

E. TRANSFER LEARNING

Given the vast amount of compute and time resources required to model deep neural networks, many of these models can serve as a starting point for other tasks and thereby save the computation resources and this process is called transfer learning. As stated in [32] the learned features from the first task is repurposed on the second task if the features are general across the tasks.

F. PRETRAINED MODELS

i) VGGNET

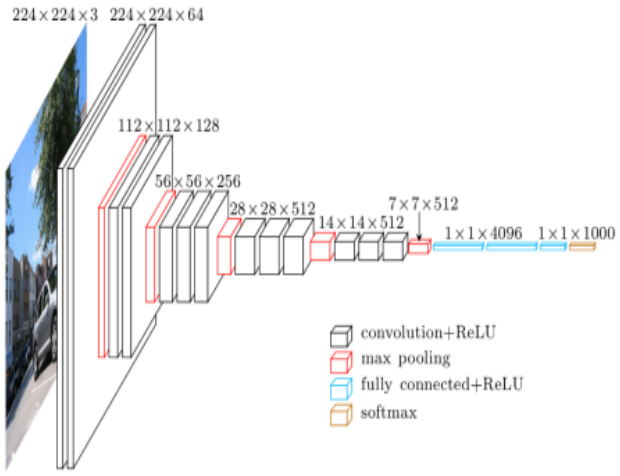


Figure Reference:
https://www.cs.toronto.edu/~frossard/post/vgg16/

VGGNet is Convolutional neural network which is trained on the ImageNet dataset where it performs exceptionally on classification of images and was built by Visual Geometry group. The ImageNet challenge had a goal of classifying the image into a class of 1000 separate objects. This pretrained

classifier with the designed architecture has the capability of generalize images outside the ImageNet dataset using transfer learning through feature extraction or fine tuning. The network has the following design as shown in the picture above. It takes an RGB input image size of 224*224*3. It uses 3*3 convolutional layers which are stacked on the top of each other. Further the volume is reduced by max pooling. It then consists of fully connected layers at the end with 4096 nodes which then use a SoftMax distribution over 1000 classes. The two variants of VGG are VGG 16 and VGG 19. The 16 and 19 stands for the number of weights layers in the network as seen in column D and E respectively below. The model size is 533 MB for VGG 16 and 574 MB for VGG 19. Two drawbacks of VGG architecture is it is very slow to train it and the network architecture weights are quite large.

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Figure Reference: Convolution Neural Network configuration for VGG as shown in [43].

ii) RESIDUAL NETWORK ARCHITECTURE (ResNET)

ResNet is a convolution neural network trained on the ImageNet Dataset. ResNet can train up to thousands of layers yet achieving great performance over it's peers like GoogLeNet and VGGNet in tasks like object detection and face recognition. It is built on constructs known from pyramidal cells in the cerebral cortex of the brain as per [50]. ResNet had a depth of 152 layer which is 8 times deeper than VGGNet but still having lower complexity than other deep neural network layers. ResNet was successful in many competitions like ILSVRC, COCO-2015 competition in ImageNet detection, ImageNet Localization, COCO detection and segmentation as per [53]. ResNet was efficiently trained with 100 layers and 1000 layers as well. ResNet solves the problem of vanishing gradient, where it was found that as network depth increases, accuracy was found to get saturated with then degrading rapidly. The core idea behind the ResNet is the identity shortcut connection which skips one or more layers as per [51].

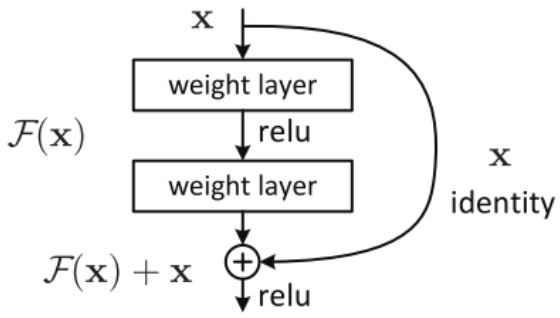


Figure Reference: A residual block [51].

As per [53], the intuition behind residual blocks is that it is easy to optimize the residual mapping function F(x) than optimize the original unreferenced mapping H(x).

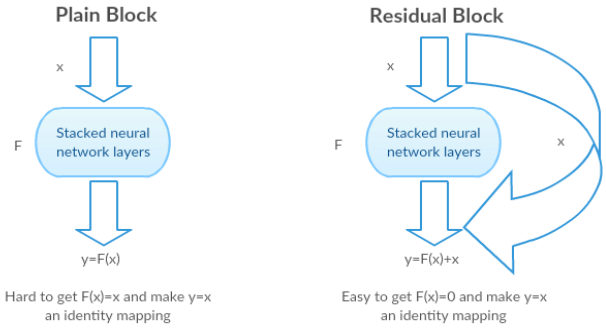


Figure Reference: Identity Mapping in residual blocks [53].

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
		3×3 max pool, stride 2				
conv2_x	56×56	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Figure Reference: Various ResNet Architectures. [53]

Each ResNet block is either 2 or 3 layers deep.

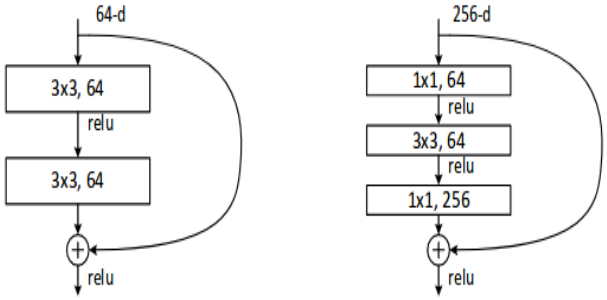


Figure Reference: ResNet 2- and 3-layer block [53].

iii) GOOGLENET

GoogLeNet is another Convolutional neural network architecture alias Inception which received a new state of art performance in ImageNet challenge 2014. As per [54], GoogLeNet architecture improved utilization of computation resources inside the networks. It is inspired from LeNet model but had it's own inception module. It used batch normalization along with image distortions and RMSprop. It used small convolutions for reduction of parameters from 60 million to 4 million with 22-layer deep CNN.

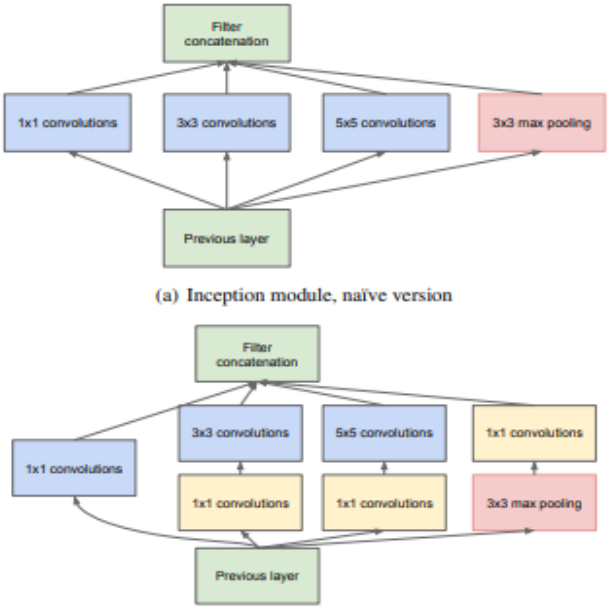


Figure Reference: Inception Module in GoogLeNet architecture.[https://www.cs.unc.edu/~wliu/papers/GoogLeNet.pdf]

type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

Figure Reference: Architecture of GoogLeNet. (https://www.cs.unc.edu/~wliu/papers/GoogLeNet.pdf)

G. GATED RECURRENT UNITS

Gated Recurrent Units (GRU) are a variation of Long short-term memory and equally produce excellent results while GRU solves the problem of vanishing gradient through the usage of update gate and reset gate which decide what information should be passed out to the output gate. The important aspect offered by GRU is that they can keep information from long ago while also having the ability to remove irrelevant information for the prediction.

H. ATTENTION MODELS

Attention models are the latest trends in the world of deep learning where they have shown to present the state of art results especially in natural language processing tasks, machine translation etc. Attention mechanisms on neural networks focus on perception which is like a short-term subset of all memory like mentioned in [38]. So, using attention mechanisms neural networks make a choice on what features to concentrate upon. As mentioned in [38], it is more like assigning credits to features which face two challenges like long term dependencies and massive data instances similar to large images. Like said, in it, neural networks can understand what 'it' is referring to. Thereby focusing on what is relevant and disregard noise. In many visual question answering systems attention-based mechanisms have shown to improve performance and especially when parts of images were given more attention relevant to the question asked, the accuracy improved.

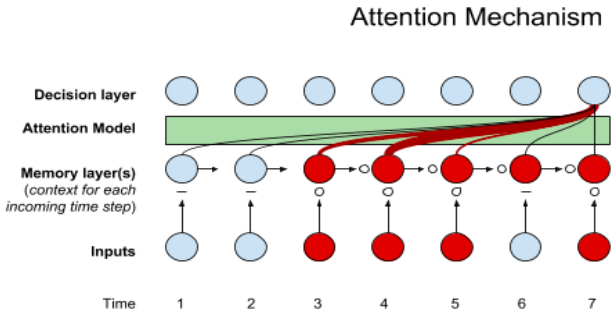


Figure reference: Attention mechanism in neural network (https://skymind.ai/wiki/attention-mechanism-memory-network)

I. WORD EMBEDDINGS

Word embeddings are the state of art in natural language processing replacing bag of words, LDA or LSA as features as they usually improve performance when used. The process is known as a distributed representation for words. It acts like a dense feature in a lower dimension space. The advantage of word vectors are they maintain semantic relationship among words. The two type of word vector architecture are skip-gram and continuous bag of word which reduced computation complexity and included context. Continuous bag of words helps represent a word representation before and after the target word. Skip gram model on the other hand reverses the approach where it predicts the future and past words

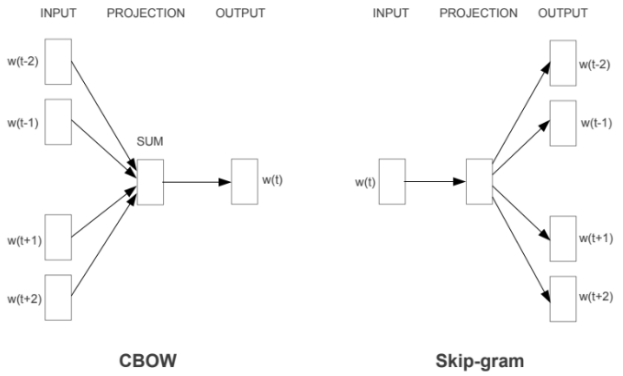


Figure reference: Representation of skip gram and CBOW model [37].

There are three common word embeddings provided for usage while solving NLP tasks. The word2vec are word vectors provided by google that are trained built using the google news corpus and are of 300 dimensions. Glove or global vector for word representation is issued by Stanford NLP team containing 26,42,840 billion tokens from 25, 50, 100,200 to 300 dimensions as referred in [33]. FastText by Facebook provides 3 word embedding models with 300 dimensions each but trained on different contexts. Maximum size of Glove is 5.5 GB, word2vec is 3.5 GB, and Fastext has 8.2 GB. All these take 9, 1 and 9 minutes for processing them respectively as referred in [33].

J. BIDIRECTIONAL LSTM

Bidirectional LSTM will run inputs in both directions, both from past to future and future to present to access information from both past and future unlike in unidirectional LSTM which stores information only from past which has processed inputs only from the past. BiLSTMs show excellent results because they understand context better.

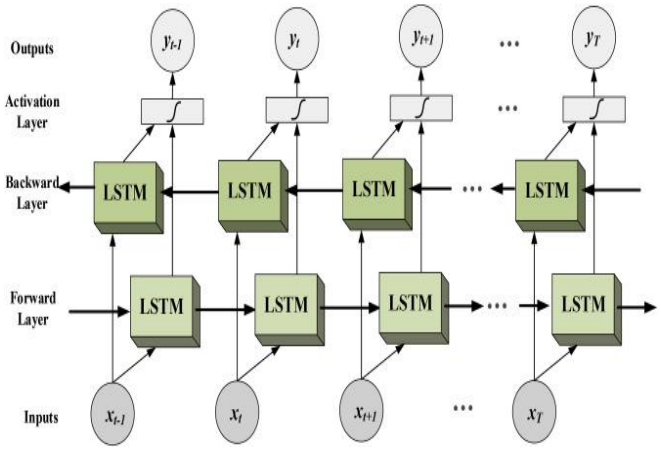


Figure Reference: Representation of Bidirectional LSTM [41]

K. MULTILAYER PERCEPTRON

As per [39], Multilayer perceptron (MLP) are vanilla neural networks when they have a single hidden layer. They are a class of feedforward neural network where an MLP uses nonlinear activation function and utilizes a supervised learning technique, backpropagation for training. MLP is generally used to make good classifiers.

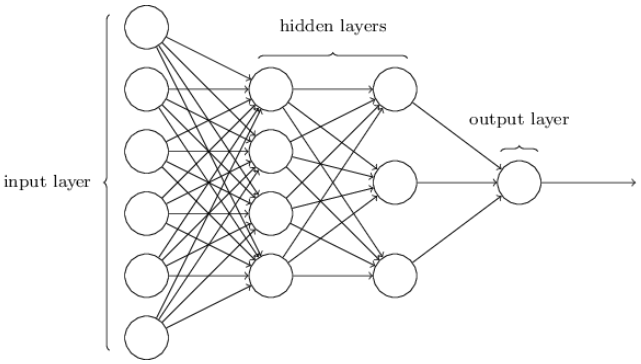


Figure Reference: Example of multilayer perceptron architecture from [https://github.com/ledell/sldm4-h2o/blob/master/sldm4-deeplearning-h2o.Rmd]

L. SKIP THOUGHT VECTORS

As per [48] skip thought vectors are a way of encoding sentences into a fixed length using neural networks that serve as the base of its architecture. It is usually an unsupervised algorithm that uses an encoder-decoder model which tries to

reconstruct sentences surrounding the encoded sentences. This way sentences like word embeddings that have same syntactic and semantic meaning are mapped to similar vector representations. There is a need for fixed vector representation of sentences in tasks like to distinguish between the semantic similarity of sentences, to tell whether a sentence is positive or negative. Below is the architecture of it as per [48] and it has three primary components:

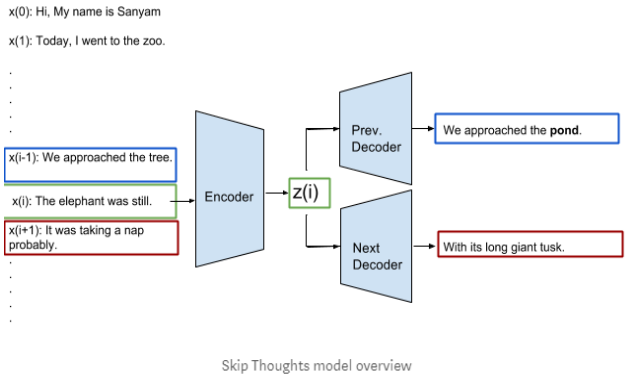


Figure reference: Skip thought model as per [48]

Encoder Network: It takes a sentence to generate a fixed vector representation through a recurrent neural network like a GRU or a LSTM which takes in words sequentially.

i) **Previous Decoder Network:** This takes an embedding and tries to generate a previous sentence through a recurrent neural network like a GRU or LSTM that generates the word sequentially.

ii) **Next Decoder Network:** This takes an embedding as input and tries to generate the next sentence which again uses a Recurrent neural network like a GRU or LSTM to generate the word sequentially.

V. MODELS FOR VQA, PRIOR WORK AND SYSTEM DESCRIPTION.

There are many models or algorithms that have been proposed for visual question answering. Many of these models are based on deep learning because this task was proposed after deep learning started to give out state of art performance. The standard procedure for approaching this task is image featurization which involves extraction of image features, secondly question featurization which involves extracting features of questions and lastly the combination of these features to get an answer for the given question directed to the image. The common technique used to treat this is a classification task. The image feature and question feature are treated as feature inputs to the classification system where each unique answer is treated as a class. There are varieties in models only in difference between integrating the image and question features. There are various non-neural techniques that have been experimented with in an attempt to solve this task. The latest techniques are the ones that use deep learning with attention mechanisms.
Some of the few examples of integrating image and question features are as mentioned in [7]:

i.) Concatenation, elementwise multiplication or element addition and giving them to linear classifier or neural network.
ii.) bilinear pooling is one of schemes in neural network framework.
iii.) A classifier that uses question features to compute spatial attention maps for the visual features or that adaptively scales local features based on their relative importance.
iv.) Bayesian models that exploit the relationships between question-image-answer feature distributions.
v) The VQA task can be considered a series of subproblems based on the questions.

For most of VQA open ended questions, it can generate only those answers for which the model has been trained for during classification. This can be changed to generating multi word answers using LSTM. There have been systems that treat VQA as ranking problem as mentioned in [7]. Here the system is trained with each multiple-choice answer, corresponding image and question to generate a scoring. Post, that the answer with the highest score corresponding to its image and question is chosen as the answer. For many of these algorithms the accuracies depend on tuning hyperparameters.

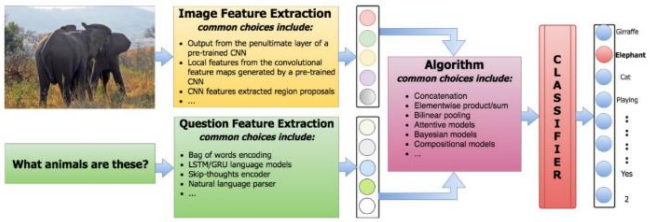


Figure Reference: General model for VQA systems (https://tryolabs.com/blog/2018/03/01/introduction-to-visual-question-answering/**)**

A. NON – DEEP LEARNING APPROACHES.

i) Answer Type Prediction. [57]

In this paper, the authors present the solution to the question answering problem in a Bayesian framework. When this approach is combined with a discriminative model, it achieves state of art performance on four open ended VQA: DAQUAR, COCO-QA, VQA, Visual7W. This method requires each question to be assigned a type during training. Datasets like

COCO-QA have explicitly defined answer categories while for DAQUAR dataset there is an explicit need to create answer categories. In addition, the questions were represented ith skip thought vectors and applied with logistic regression to infer the answer type for the question with 99.7% accuracy. According to [57], the Bayesian framework contains x as a column vector containing image features, while q be a column vector containing question features. For a given question and image, the model estimates the probability of an answer k and question type c as P (A = k, T = c|x, q). Using Bayes rule along with chain of probabilities, it can be expressed as:

$$P(A=k,T=c|\mathbf{x},\mathbf{q}) = \frac{P(\mathbf{x}|A=k,T=c,\mathbf{q})\,P(A=k|T=c,\mathbf{q})\,P(T=c|\mathbf{q})}{P(\mathbf{x}|\mathbf{q})},$$

Figure Reference: [57]

Where P (x|A = k, T = c, q) is the probability of an image feature, given the answer, answer-type and question, P (A = k|T = c, q) the probability of an answer given the question and P(x|q) is the probability of the image feature given the question as mentioned in [57]. To obtain an answer for a question corresponding to an image, the authors could marginalize over all the answer types using

$$P(A=k|\mathbf{x},\mathbf{q}) = \sum_{c\in T} P(A=k,T=c|\mathbf{x},\mathbf{q}).$$

Figure Reference: [57]

The three probabilities in the numerator are modelled as three separate models. The second and third probabilities are modeled using logistic regression while the first probability is modeled as a conditional multivariate gaussian like quadratic discriminant analysis. The authors in addition, tested five baseline models as mentioned in [57],

i) A Logistic regression classifier trained with image features where it knows nothing about the question.
ii) The answer type in the dataset is trained with logistic regression classifier to select among the given question without having access to detailed question information.
iii) A Logistic regression classifier trained only with the question features.
iv) A Logistic regression classifier trained with image features concatenated to the question features.
v) A multilayer perceptron network with SoftMax output layer with image and question features as input. The authors used an MLP which a 4-layer neural network with 6000 units in the first layer, 4000 for the second, 2000 for third and finally a SoftMax output layer with units equal to number of categories. All the hidden layers use ReLU and a drop out of 0.3 was used in the hidden layers to regularize the network.

They also used hybrid models that combine generative and discriminative classifiers can achieve a lower error rate than either alone. The results for the performance are shown on DAQUAR and COCO-QA.

ii) MULTI-WORD QUESTION ANSWERING. [1]

Here, the unpredictability about the conceived world in a Bayesian framework is represented by a combination of discrete reasoning with uncertain predictions by a multi word approach as stated in [1].

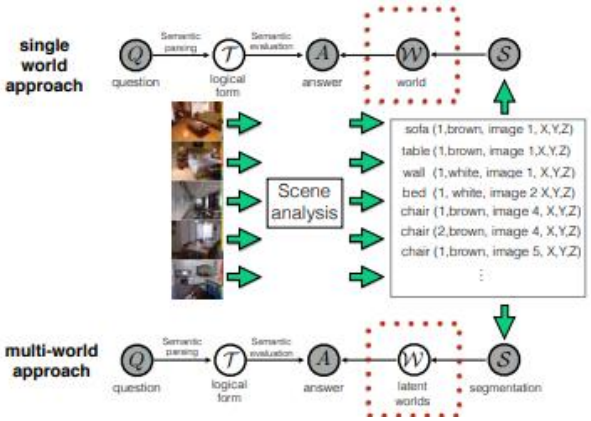


Figure Reference: Multi word question answering approach [1]

Here, the authors work on a perceived world W which now has facts derived from automatic, semantic segmentation S. The authors then run a state of art segmentation algorithm and collected relevant information about objects class, 3D position and color. So now every object is now represented as a n- tuple as predicate (instance id, image id, color, spatial loc) where predicates are like objects with instance-id as the object -id, image-id as the image containing that object , color being the estimated color of the object and the spatial location being the object's position in the image. The authors here propose a multi word approach drawing from the ideas from probabilistic databases, which marginalizes over the possible multi-words derived from the segmentation S which contain multiple interpretation of a visual scene. Therefore, the model gives the probability of an answer A, given a question Q, semantic segmentation S of the image marginalizes over the latent worlds W and logical forms T as represented below as mentioned in [1].

$$P(A \mid Q,S) = \sum_{\mathcal{W}} \sum_{\mathcal{T}} P(A \mid \mathcal{W},\mathcal{T}) P(\mathcal{W} \mid S)\, P(\mathcal{T} \mid Q)$$

Figure Reference: [1].

This model was evaluated on the dataset DAQUAR. The authors conclude that their model brings promising progress despite the complexity in uncertain visual perception, language understanding and program induction. Their model includes ideas from automatic scene analysis, semantic parsing with symbolic reasoning to combine them under a multi-world approach.

B. DEEP LEARNING APPROACHES

i) BASELINE MODELS

Baseline models measure the complexity of a dataset and set a minimum performance so that a better model developed can succeed. As per [7], the simplest baseline would be random guessing or guessing the most frequent answer. Common techniques involve using Convolutional neural networks for image features and using word embeddings for recurrent neural network for questions. Various variants are used in the above to generate aa combination of those features to obtain a baseline model. Following are some of the baseline models examples.

a. IBOWIMG FOR VISUAL QUESTION ANSWERING [58]

[58] presents a baseline model for VQA. They use pretrained GoogLeNet model for image features and the word embedding for each word adds to the text features which is nothing but a bag of words. The text and image features are concatenated after which a SoftMax regression is applied to get the answer classes. The model achieves comparable performance with respect to other models that use LSTMs for representing text features.

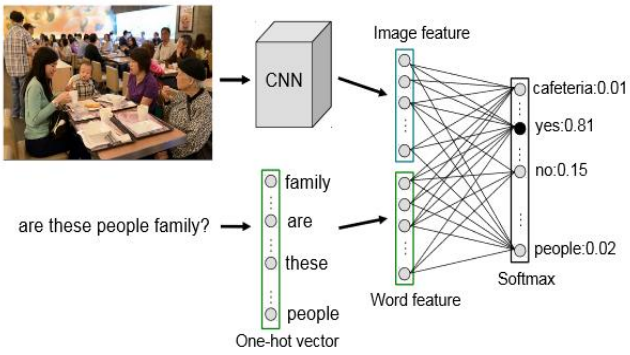


Figure Reference: iBowimg model [58]

b. FULL CONVOLUTION NEURAL NETWORK. [59]

Here, the authors propose a solution that completely uses CNN for extracting both the image and question features and their inter-modal interactions as per [59].

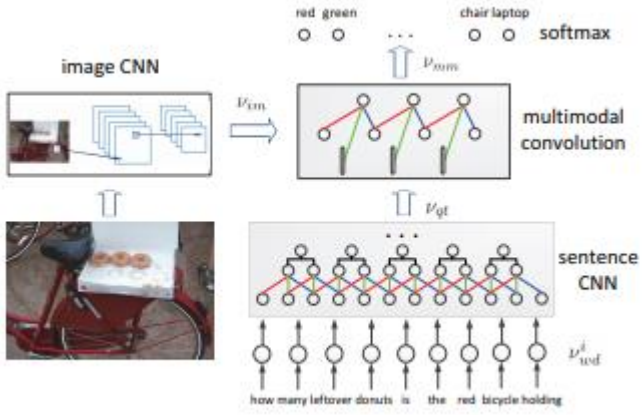


Figure Reference: Full Convolutional Neural network model for Visual Question answering system.[59]

The framework consists of three individual CNNs where one is the image encoding CNN, the other the sentence CNN generating the question representation and lastly a multimodal convolutional layer combining the image and question representation to generate a joint representation which is fed to a SoftMax layer to produce an output answer. The image CNN uses the same architecture as VGGNet and obtains a length vector of 4096 from the second last layer of this network where the last layer without the SoftMax and last layer ReLU of the CNN is removed which is passed through a fully connected layer to get the image representation vector of 400. The sentence CNN for the question involves three layers of convolution with max pooling with the size of convolutional receptive field set to 3 with the kernel looking for words along with it's immediate neighbours. The multimodal CNN, which is a joint representation, has a receptive field size is 2. The final representation of this is given to SoftMax layer which is used for classification. The evaluation on the model is done on the DAQUAR and COCO-QA datasets.

c. ASK YOUR NEURONS [2]

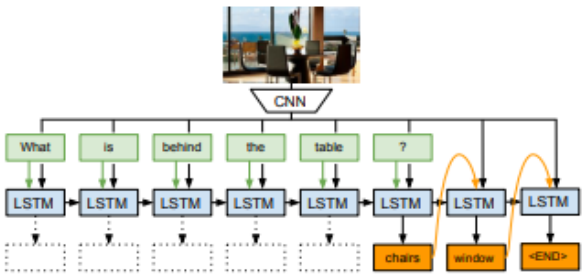


Figure Reference: Ask your Neurons approach [2]

In this model, the model uses CNN and LSTM in its architecture to solve this task. The image features are encoded using a CNN and the questions are encoded using a LSTM which takes a word embedding as well as the image vector. The image vector obtained at the end is represented as the question encoding. A way of generating answer is through passing this through a fully connected layer which in addition is passed through SoftMax for a classification answer. The other way would be to generate it using a decoder LSTM which takes in as input the previously generated word, the image and question encoding where the next word is predicted using SoftMax. Here the LSTM shares the parameter with the encoder. The DAQUAR dataset has been used for evaluation.

d. VIS+ LSTM [60]

The authors propose neural networks which use visual semantic embeddings which do not need object detection and image segmentation as intermediaries as mentioned in [60]. The authors also solely generate one-word answers which makes it easier to treat this task as a classification task. They use VGG -19 for extracting image features. The image features of vector size 4096 dimension are transformed 300 or 500-dimensional vector to match dimension of the word embeddings. The model also proposes to experiment with sending the image as the last word of the question through a different weight matrix and pass it through a reverse LSTM which acts in the sequential manner which is called 2-VIS + BLSTM. The outputs of LSTM are fed into the SoftMax layer to generate answers. The other model tried is IMG+BOW which performs multinomial logistic on the dimensionality reduced image of 4096 image vector and a bag of word vector which is the summation of all the word vectors of the question. They also built another model which the average of all the three models is mentioned above.

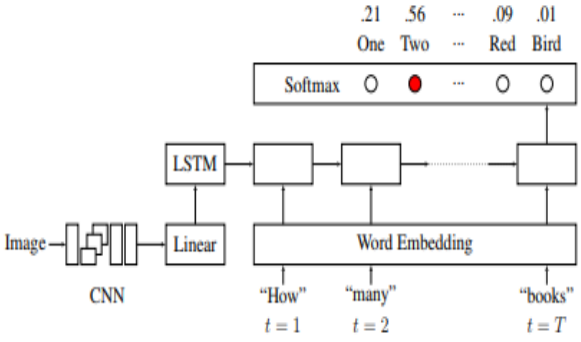


Figure Reference: Architecture of Vis+LSTM [60]

e. DYNAMIC PARAMETER PREDICTION [61]

In this model the authors build a CNN which has a dynamic parameter layer for which weights are calculated on the input questions. This adaptive parameter consists of a GRU which takes a question as input and a fully connected layer which generates a set of candidate weights as output. This becomes an issue if we generate for a huge set of parameters. For this issue the complexity is reduced where the candidate weights given by the parameter prediction network are selected using a predefined hash function as mentioned in [61]. The proposed design for the VQA task is the joint network with the CNN and the parameter prediction network which is trained end to end through backpropagation where the weights are initialized and trained using pre-trained CNN and GRU. This proposed model brings in the state of art performance as mentioned in [61]. The model has been tested on DAQUAR, COCO-QA and VQA datasets.

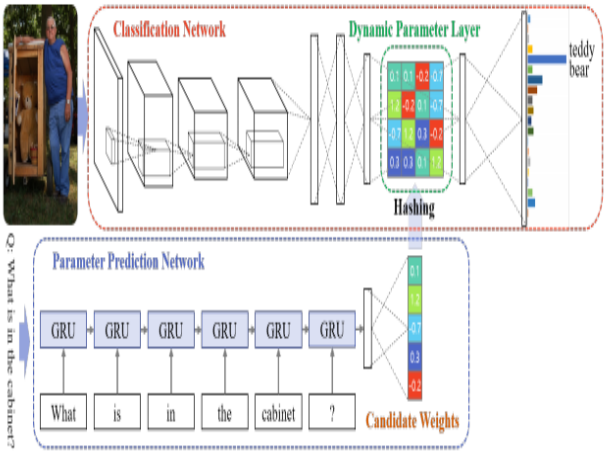


Figure Reference: Architecture of dynamic parameter prediction module. [61]

ii) ATTENTION BASED MODELS

Attention based mechanism is to use features relevant to the input rather than relying on global set of features. Attention mechanisms are currently the latest and are the ones on which the combination with deep learning techniques is giving state of art results in tasks related to NLP and computer vision such as machine translation, image captioning and object recognition. It basically works as a windowing mechanism where the most relevant CNN features in the image and the most relevant text in the question too are considered. The global image and text features may not be too fine grained to answer the local region-specific questions.

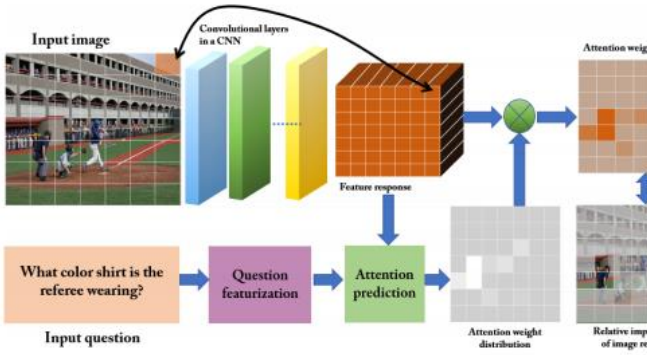


Figure Reference: Attention mechanism in VQA [7]

As shown in the above figure, one way of implementing attention mechanism, is to make a uniform grid above all image locations, with each grid location containing features that are local. Also, secondly, to make bounding boxes and then encode these boxes using a CNN to state the relevance of each box feature to the question. The above figure corresponds to N feature maps which are 3D CNN outputs of a K*N*N tensor of feature responses and a weighted factor is computed according to spatial location and it's relevance to the question which could be used for computation attention weighted image features as presented in [7]. Below are few of the attention models discussed as in [5]:

a) WHERE TO LOOK [64]

In [64], the authors present a model which selects an image region that maps the relevant image region to textual queries. This model they built is tested on the VQA dataset on the multiple-choice task and have achieved to overtake the baselines and already existing work. This model also works highly well for some types of questions like identifying object colors. Below is the architecture of the model as mentioned in [64]:

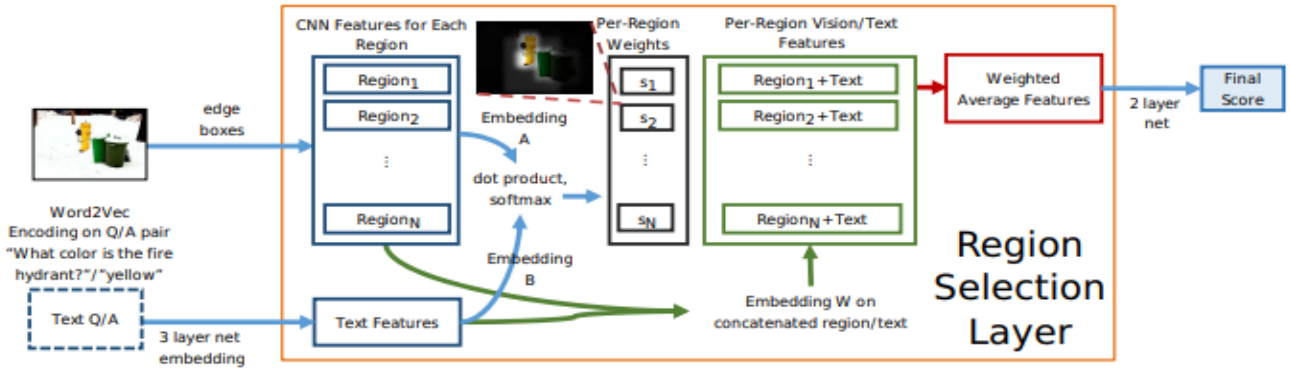


Figure Reference: Architecture of the where to look model [64].

As per [5], the model uses the outputs of the last two layers of VGGNet for getting the image representation. The Question is represented by getting the average word vector representation of every word in the question. These language and image features are concatenated with a dot product which is later softmaxed to get a weighted region-specific scoring. The weighted vision features along with the language features is the input for the two-layer network which gives a final score and does the classification for the answer [64].

b) RECURRENT SPATIAL ATTENTION (R-SA) [65]

In this the authors introduce spatial attention to the VQA task by introducing spatial attention mechanism which was first applied to the image captioning task. Continuing from the above model, as LSTMs have continued to generate state of art results in sequence to sequence tasks, they are used for the encoding of the question, such that after each word is scanned, the attention is computed all over the image again. As per [5], repeated computation of attention based weighted sum of image features is done at time t which goes as another input to the next step of the LSTM. Here the attention weights and the weighted image features are calculated using dense SoftMax layer over the previous state of the LSTM and the image. The decoding stage the model selects the answer from the multiple choice using the log-likelihood of an answer by computing the dot product of the weighted visual feature and the previous state in the LSTM. The model here is trained and the evaluation is done on the visual7W dataset. Cross entropy loss was used while training the model.

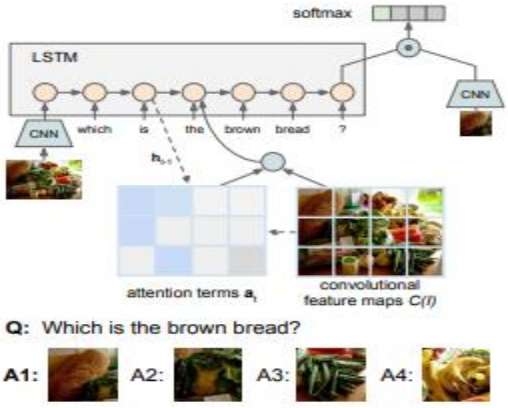


Figure reference: Recurrent spatial attention (R-SA) [65]

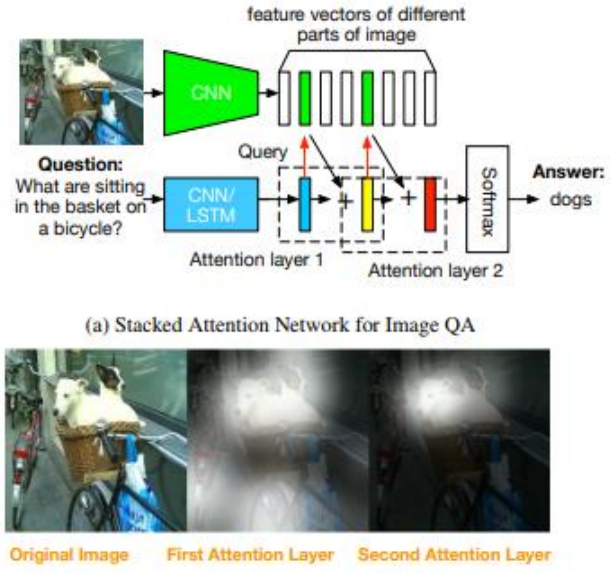


Figure Reference: Stacked attention network and visualization. [66]

c) STACKED ATTENTION NETWORKS (SAN) [66]

Stacked attention networks use semantic representation of the input question and with relation to it search for the answer in the image. The authors here query the image multiple times to result in a better answer using multiple layer stacked attention network. As we can see in the visualization below the answer here is led by layer by layer approach. The authors have experimented on the four image QA data sets like DAQUAR-ALL, DAQUAR-reduced, COCO-QA, VQAs which have bought out state of at performance results. Here, the authors encode the question with an LSTM or a CNN as per [5]. As in the previous model, with the help of question encoding, the attention weighted image is concatenated with question encoding which is used to calculate attention over the original image. This entire process can be repeated for k times like a stacked attention network which helps iteratively remove unimportant regions and help predict the right answer. The authors here used VGGNet to extract image features from the last pooling layer with a dimension of 512 x14 x 14. For some of the dataset, the word embeddings and the dimension of the LSTM was set to be 500 for the question model. After using it with unigram, bigram and trigram the convolution filter size as 128, 256 and 256 respectively, the question vector size was set to 640. The authors then experimented with k=1 or 2 for the stacked attention network as they found increasing k more than two did not help improve performance. Here, all the models are trained using stochastic gradient descent with the batch size fixed to be 100 and the grid search provided them the best learning rate. The authors also used Gradient clipping technique and dropout.

d) HIERARCHICAL CO-ATTENTION (COATT) [68]

The authors in this paper [68] present another facet to solve this task. Instead of only concentrating on where to look or getting the relevant image features to the question, the authors state that it is even more important to formulate what words to listen to or question attention. The authors here present a co-attention model for VQA system which reasons both for image and question attention. They use a one-dimensional CNN in hierarchical manner to understand about the question and the image attention mechanism simultaneously. The authors were able to improve the state of art performance on the VQA and COCO-QA dataset and using the ResNet they were able to further it. The authors main contribution was to experiment with two strategies parallel and alternating co-attention. The hierarchical architecture builds the image question co-attention maps at with three different levels, word level, phrase level and question level. These feature levels are recursively merged from word level to question level to get the final answer prediction as mentioned in [68].. Question level representation is obtained by LSTM while word and phrase level representation are obtained through CNNs.

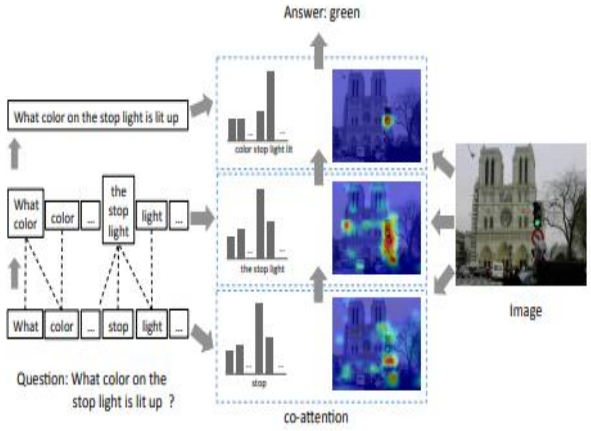


Figure reference: Hierarchical co-attention model [68]

iii) BILINEAR POOLING

For fine grained image recognition, instead of simply concatenating the image feature and question feature by simple concatenation, the outer multiplication between the two modes of information would give much complex information. There are two methods that have used bilinear pooling:

a) MULTIMODAL COMPACT BILINEAR POOLING.[62]

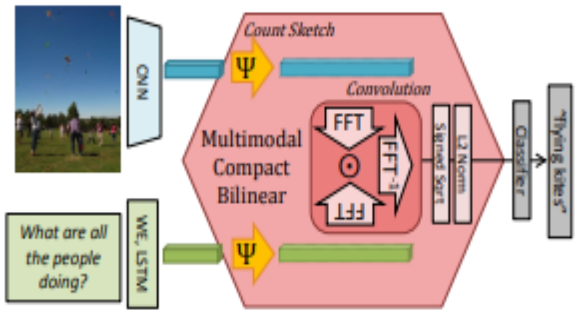


Figure Reference: Multimodal compact bilinear pooling.[62]

In multimodal compact bilinear (MCB) pooling approach, instead of straightly multiplying the outer features or concatenating, which would be a high degree of multiplication, multimodal compact bilinear pooling aims to efficiently and expressively combine the different modes of features. Here, MCB was used once for recognizing the spatial features relevant to the question which is then used again with combination with the attention mechanism to combine the image and text feature. This model was able to achieve a state of art performance on the visual7w and VQA dataset.

b) HADAMARD PRODUCT FOR LOW-RANK BILINEAR POOLING. [63]

The authors state that bilinear pooling was a richer representation over linear pooling which tended to be highly dimensional but tended to be computationally complex even though it was known to improve performance in visual tasks like object recognition, segmentation and visual question answering as mentioned in [63]. The authors here portray a better state of art performance result-oriented model on the VQA dataset that does better than multimodal compact bilinear pooling approach using the proposed low-rank bilinear pooling using Hadamard product for improvising the attention mechanism of multimodal learning as mentioned in [63].

iv) SPECIFIC ANALYSIS OF MODELS ON VQA DATASET [6]

We here analyze the baselines and various models built on the VQA dataset. The authors in [6] have trained their models in VQA train+ val data, with human accuracies on test data and machine accuracies are on dev data.

Following are the baselines as mentioned in [6]:

i) Random – Here randomly an answer from top 1000 answers of the VQA train/val dataset is chosen.

ii) Prior- Here the most popular answer is usually chosen for open -ended and multiple-choice tasks like for example yes, is always an answer for the multiple-choice question.

iii) Per-Q-type prior – Here the most popular answer for each question type is selected for open ended task. For multiple choice tasks the authors pick up the most similar answer for the open-ended question using cosine similarity.

iv) Nearest Neighbor – Here, they pick the nearest question pairs and its corresponding images. From this set of questions, the most frequent ground truth answer is found. For multiple choice per Q type prior is done and the most similar answer is picked for open ended using cosine similarity in the feature space of word2Vec.

The authors in [6] have built a dual channel with vision with image as input and language with question as input which has to output with K classes as possible output using a SoftMax function. The authors chose k=1000 which covered 82.67% of the answers. The different model components are:

i) **Image channel** – This provides a representation for the image as embeddings. The authors experiment with two different embeddings:

1. **I:** The 4096-dimension image embedding are from the activations of the last hidden layer of VGGNet.

2. **Norm I:** These embeddings are from the last hidden layer of VGGNet and are L2 normalized activations.

ii) **Question Channel:** the authors here experiment with 3 embeddings for the questions:

a) **Bag of words Question (BoW Q):** Here top 1000 words are used to create a bag of words representation with 30 words that usually repeat in both the question and answer. These 1030 features are used to get a 1030 dimension embedding.

b) **LSTM Q:** The LSTM with one hidden layer outputs a 1024 dimension embedding for a question. Each word given as an input to the LSTM is encoded as a 300-dimension embedding by a fully connected layer + tanh non-linearity.

c) **Deeper LSTM Q:** Here the LSTM has two hidden layers and aims to output a 2048 dimension embedding for the question in addition added by a fully connected layer + tanh non-linearity which helps it to transform to 1024 dimension embedding.

These image and question embedding are combined to obtain a single embedding with either BoW Q + I. For LSTM Q + I and deeper LSTM Q + norm I the image embedding is transformed to 1024 dimension by a fully connected layer with tanh non-linearity to match the LSTM embedding of the question. The transformed image and the LSTM embeddings are merged via element wise multiplication. The combined features are later passed on to a multilayer perceptron which has 2 hidden layers and 1000 hidden units with a dropout of 0.5 in each layer with tanh non-linearity with the last layer being a SoftMax over the k classes or answers as mentioned in [6]. Cross Entropy Loss is applied and learned from end to end. The whole architecture is represented in the below diagram.

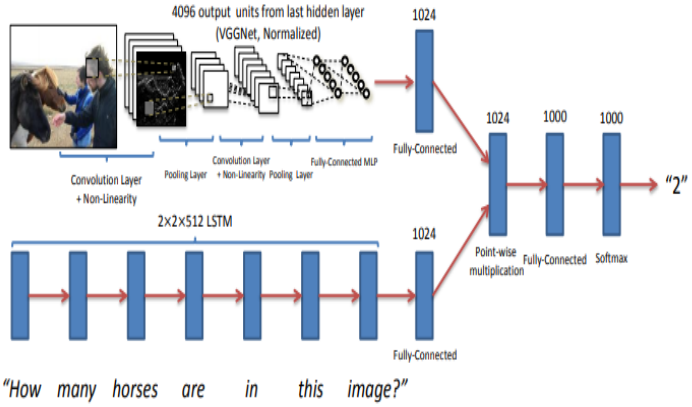


Figure Reference: The best performing model deeper LSTM Q + norm I as mentioned in [6].

VI. MODEL EVALUATION

We could see that Stacked Attention Network (SAN) is consistently performing well across all the datasets.

Model	Accuracy %	WUPS at (0.9%)	WUPS at 0 (%)
SWQA	9.69	14.73	48.57
MWQA	12.73	18.10	51.47
Vis+LSTM	34.41	46.05	82.23
AYN	34.68	40.76	79.54
2Vis + BiLSTM	35.78	46.83	82.15
Full CNN	42.76	47.58	82.60
DPPNet	44.48	49.56	83.95
ATP	45.17	49.74	85.13
SAN	45.5	50.2	83.60

Figure Reference: Results of models on the DAQUAR dataset. [5].

Model	Accuracy %	WUPS at (0.9%)	WUPS at 0 (%)
Vis+LSTM	53.31	63.91	88.25
AYN			
2Vis+ BiLSTM	55.09	65.34	88.64
Full CNN	54.95	65.36	88.58
DPPNet	61.19	70.84	90.61

ATP	63.18	73.14	91.32
SAN	61.60	71.60	90.9
CoAtt	65.4	75.10	92.00
AMA	69.73	72.14	92.50

Figure Reference: Results of models on the COCO-QA dataset. [5].

Model		Open Ended				MC Q
	Y/N	Number	Other	ALL		All
IBowimg	76.8	35	42.6	55.9		
DPPNet	80.3	36.9	42.2	57.4		
WTL						62.4
AYN	78.2	36.3	46.3	58.4		
SAN				58.9		
ATP	80.3	37.8	47.6	60.1		
NMN	81.2	37.7	44	58.7		
CoAtt				62.1		66.1
AMA	81.07	37.12	45.83	59.44		

Figure Reference: Results of various models on VQA dataset. [5]

In the above table we could see that Hierarchical co-attention models and NMN (Neural Model Networks) performing well across the VQA dataset. Neural Model Networks design the neural architecture based on the sub modules chosen according to the given question and image composition. The module composition is chosen by parsing the question using the dependency tree and the submodules from Attention, Classification, Reattention, Measurement and Combination have to be chosen. Then accordingly the whole system is trained through backpropagation.

We can see that deep learning models perform better than non-deep learning models, but they are comparable in performance to them. As discussed in [5], it is important to select the image parts that have relevance rather than just using deep neural models as attention type prediction (ATP) is seen to perform better than models using attention. The use of attention has drastically improved performance comparatively.

Above are the table of comparison of various models built over the VQA dataset as referenced in [6].

	Open-Ended				Multiple-Choice			
	All	Yes/No	Number	Other	All	Yes/No	Number	Other
prior ("yes")	29.66	70.81	00.39	01.15	29.66	70.81	00.39	01.15
per Q-type prior	37.54	71.03	35.77	09.38	39.45	71.02	35.86	13.34
nearest neighbor	42.70	71.89	24.36	21.94	48.49	71.94	26.00	33.56
BoW Q	48.09	75.66	36.70	27.14	53.68	75.71	37.05	38.64
I	28.13	64.01	00.42	03.77	30.53	69.87	00.45	03.76
BoW Q + I	52.64	75.55	33.67	37.37	58.97	75.59	34.35	50.33
LSTM Q	48.76	78.20	35.68	26.59	54.75	78.22	36.82	38.78
LSTM Q + I	53.74	78.94	35.24	36.42	57.17	78.95	35.80	43.41
deeper LSTM Q	50.39	78.41	34.68	30.03	55.88	78.45	35.91	41.13
deeper LSTM Q + norm I	**57.75**	**80.50**	**36.77**	**43.08**	**62.70**	**80.52**	**38.22**	**53.01**
Caption	26.70	65.50	02.03	03.86	28.29	69.79	02.06	03.82
BoW Q + C	54.70	75.82	40.12	42.56	59.85	75.89	41.16	52.53

Figure Reference: Table of various accuracy comparison of models for open and multiple-choice question where Q stands for question, C for caption and I for image. [6]

From the above table we can see that deeper LSTM Q with norm I embeddings give a better performance.

VII. PROPOSED MODEL.

A VQA system takes input image and a natural language question corresponding the image and generates a natural language answer as the output. We propose two different solutions to work upon to build our model. We have planned to take VQA dataset from Microsoft (http://www.visualqa.org/).

Our first model, based on [28], is aimed as a classification problem as questions with multiple choice are taken into consideration only. The top frequent answers from those multiple choices are taken as classes to which any input would be mapped into. The proposed architecture is shown below:

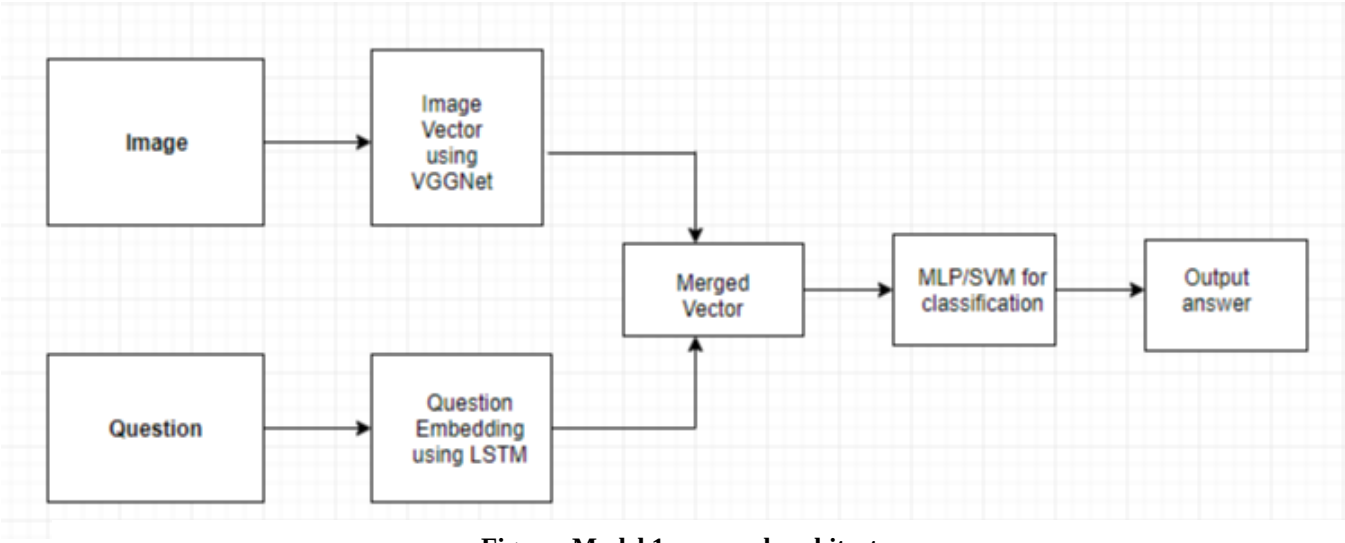


Figure: Model 1 proposal architecture

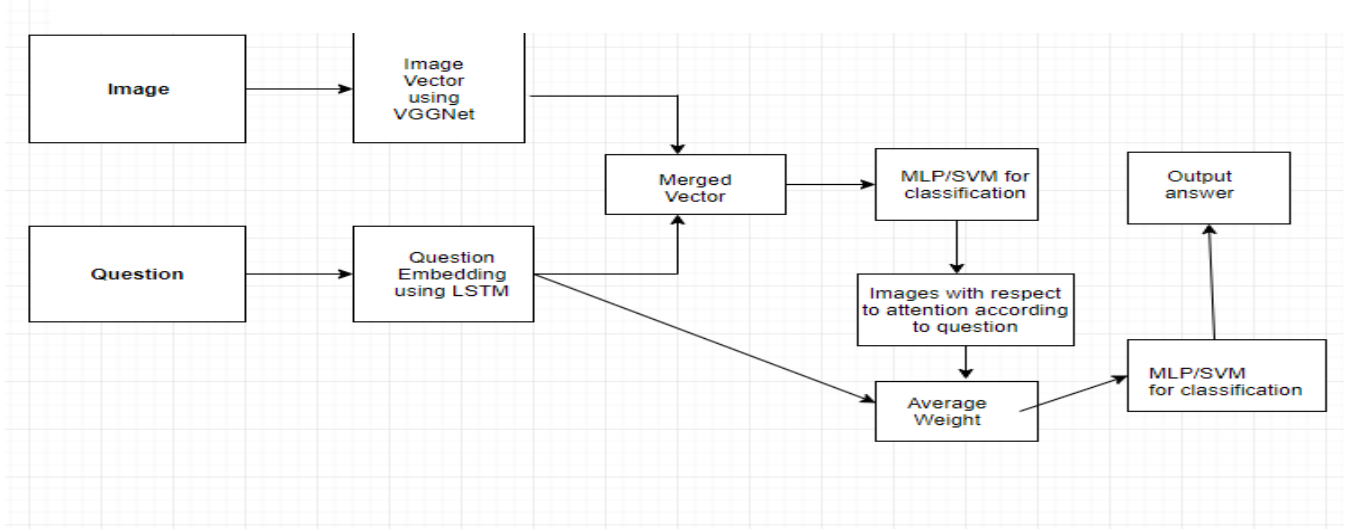


Figure: Model 2 proposal architecture

Data: The data used here is Visual Question Answering v2 contains three folders Input questions (training of 443,757 questions, validation of 214,354 questions and testing of 447, 793 questions), images folder (training of 82,783 , testing of 81,434 and validation of 40,504), Annotations folder (4,437,570 and validation of 2, 143,540).

So, a dataset is created with questions and their correct answers matching from one of these columns and the rest questions are discarded. For getting the image features, we planned to use pretrained VGGNet model which has convolution filter size of (3*3). Each image after rescaling to 224 x 224 size is encoded into a vector size of 1000 using the VGG-19 embeddings. The feature extraction for the questions could be done using sentence embeddings or averaging the word vector for every word in the question. But using skip thought vectors provides a better representation for sentence representation. So, we could use InferSent model provided by Facebook to get an equal vector representation size of 4096 dimensions for each sentence. We could provide a combination of these two features as an input to models like SVM or neural networks like Multilayer Perceptron with SoftMax activations as the last layer to classify the merged vector representation into one of these classes. We plan to implement this using Keras library, with a VGG-19 model.

The second proposed model according to [49], we transfer the same concept by taking most frequent answers as the classes, we take the image features using pretrained model VGG19 and encode the question using LSTM. Finally, using Stacked attention network, relevant images parts are given more

	answer	image_id	mc_answers	question	question_id
0	no	262145	[green, blue, 1, on chest, no, behind head, wh...	Is this a fancy supermarket?	2621452
1	yes	524291	[alive, hang gliding, yes, paper cup, 2, 3, so...	Is the dog waiting?	5242910
2	no	524297	[no, enix, 3, london, corduroy, sunset, blue, ...	Was this photo taken recently?	5242970
3	yes	524297	[green, rabbits, united states, genius, red, g...	Judging from the dress, was this taken in a La...	5242971
4	yes	262159	[ilford, blue, 4, space, against wall, 3, yes,...	Is there a shower curtain?	2621592

Figure Reference: Dataset for building models.

weightage with respect to the question. The proposed architectures are shown below.

Following is the expected model output demo:

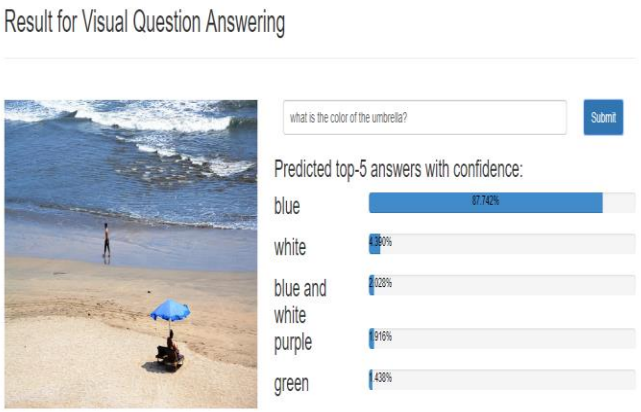


Figure Reference: The expected output for VQA. [71]

In the above proposed models, various experimentations can be done with different pretrained models like ResNet GoogLeNet and different sentence embeddings to see improvement in model performance. Finally, we can also compare performance of models with and without attention. All these are in conformation with the baselines. We had cleaned and built our dataset as shown below:

We had the image embeddings of the images using the already available VGGNet features. We were stuck in inability to compute the question embeddings through our LSTM model which required a high-performance computing machine to train such a big dataset. Due to this major halting point and facing environment issues in the NYU HPC cluster while resolving library dependencies, we plan to take this project further in computer vision course offered in the fall 2019 as this problem lies in the intersection of computer vision and NLP.

VIII. DISCUSSION AND FUTURE WORK

We can see that there are gradual mechanisms being bought into picture to improve performance from involving deep neural networks, attention etc. There are still various challenges to address to improve the performance and evaluation. Knowing the upper bound performance of the VQA systems can help in knowing the areas where to improve the model. The vision and language model are crucially interdependent for the features to be considered for isolation.

It has been found that language content affects the visual question answering system more than the image which hinders deployment. [6] [57] have shown that question only models perform better than image only models and this is partly since most answers depend on how the questions are formed. Moreover, analysis in [6] show that the hints or language type used was critical in affecting the performance of the VQA system. So, for a new system to perform better, the datasets should consider this bias.

[57][69][70] have used basic pretrained neural networks without attention models and they have found them to perform better than attention models that are complex. Moreover, it has been seen through many studies that it is not necessary that generally attention models alone improve performance. It is seen that when good models use attention mechanism with them, they improve the performance. Moreover, there can be biases to attention mechanism due to

the questions, where the model can attend to a various part of the image than the actual part of the image that it should be attending to. Future datasets should be incorporative of such biases in image and models should be able to handle it.

As per [7], it has been found that it is easier to answer questions with a 'is' and 'are' rather than a 'why' and 'where' which start to effect the performance of the models as they belong to harder set of questions and need generative answers rather than a single type of answer. Moreover, it should also be capable of answering more than a yes/no or multiple-choice question to make the system more robust and be more applicable as a real-world utility.

With the growing capabilities in the field of computer vision and NLP, current benchmarks for VQA system are not enough. The datasets should incorporate a lot of variety to reduce bias, of which one of the ways would be to increase the size of the dataset. Another thing to consider would be to giving weights to the questions in the dataset as some questions seem easier to answer than others. By giving each question a type, we can measure performance on the type of question which could be useful for benchmarking. As a part of future work, we have planned to implement the above proposed models and do the comparative analysis practically.

IX. CONCLUSION

Visual Question answering system is a vast research topic where deep learning techniques have been producing state of art performance. It involves common tasks from computer vision and NLP, and it should be a part of visual Turing test. We have done an analysis of the major datasets which are used for these tasks. Dataset selection is an important criterion for formulating the model and getting accuracy because the system is implicitly biased to the data beneath. This paper was successful in analyzing models or algorithms for VQA and analyzing the evaluation criteria. Significant improvements can be seen on the model with many upcoming techniques and there is a lot of scope for innovation. A more successful VQA system would be the one that can answer any random question about the image with precision. Further work would be based on creating varied and bias free datasets, evaluating question types. The VQA algorithms need to be developed further so that they can reason about image content which could lead to improvement in VQA models.

X. ACKNOWLEDGMENT

We would like to express our gratitude to Dr. Ralph Grishman for guiding us through this study. We also would acknowledge our individual contributions as follows. Sree Gowri Addepalli has been responsible for drafting and researching sections V, VI, VII, VIII, IX which required understanding the baseline and state of art models for this task proposed by various researchers. On the other hand, Sree Lakshmi Addepalli has worked on the sections I, II, III, IV, IX which involved working on the datasets and its evaluation along with the deep learning jargon.

XI. REFERENCES

[1] Mateusz Malinowski, Mario Fritz, "A Multi-World Approach to Question Answering about Real-World Scenes based on Uncertain Input", NIPS 2014.
[2] Mateusz Malinowski, Marcus Rohrbach, Mario Fritz "Ask Your Neurons: A Neural-based Approach to Answering Questions about Images".
[3] Mateusz Malinowski, Mario Fritz, "Towards a visual Turing challenge."
[4] Mateusz Malinowski, Mario Fritz, "Hard to cheat: A Turing Test based on Answering Questions about Images."
[5] Akshay Kumar Gupta, "Survey of Visual Question Answering: Datasets and Techniques."
[6] Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Dhruv Batra, Devi Parikh "VQA: Visual Question Answering"
[7] Kushal Kafle and Christopher Kanan, "Visual Question Answering: Datasets, Algorithms, and Future Challenges."
[8] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, Piotr Dollar, "Microsoft COCO: Common Objects in Context."
[9] http://tamaraberg.com/visualmadlibs/
[10] Licheng Yu, Eunbyung Park, Alexander C. Berg, and Tamara L. Berg, "Visual Madlibs: Fill in the blank Image Generation and Question Answering"
[11] Haoyuan Gao, Junhua Mao, Jie Zhou, Zhiheng Huang, Lei Wang, and Wei Xu. 2015. Are you talking to a machine? dataset and methods for multilingual image question.
[12] https://datasets.d2.mpi-inf.mpg.de/mateusz14visual-turing/challenges.pdf
[13] https://github.com/anujshah1003/VQA-Demo-GUI
[14] https://iamaaditya.github.io/research/vqa/
[15]https://iamaaditya.github.io/2016/04/visual_question_answering_demo_notebook
[16] http://avisingh599.github.io/deeplearning/visual-qa/
[17] https://colah.github.io/
[18] https://karpathy.github.io/
[19] https://github.com/facebookresearch/InferSent
[20] https://github.com/GT-Vision-Lab/VQA
[21] https://cs.stanford.edu/people/jcjohns/clevr/
[22]https://towardsdatascience.com/deep-learning-and-visual-question-answering-c8c8093941bc
[23] https://datasets.maluuba.com/FigureQA
[24] Samira Ebrahimi Kahou1, Vincent Michalski, Adam Atkinson, Ákos Kádár, Adam Trischler ,Yoshua Bengio,

‘FIGUREQA: AN ANNOTATED FIGURE DATASET FOR VISUAL REASONING’
[25] https://visualgenome.org/
[26] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, Li Fei-Fei “Visual Genome Connecting Language and Vision Using Crowdsourced Dense Image Annotations.”
[27] https://cs224d.stanford.edu/reports/shuhui.pdf
[28] https://scholarworks.sjsu.edu/cgi/viewcontent.cgi?referer=https://www.google.com/&httpsredir=1&article=1635&context=etd_projects
[29] https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6
[30]https://medium.com/the-theory-of-everything/understanding-activation-functions-in-neural-networks-9491262884e0
[31] https://machinelearningmastery.com/transfer-learning-for-deep-learning/
[32] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson, “How transferable are features in deep neural networks?”
[33]https://towardsdatascience.com/3-silver-bullets-of-word-embedding-in-nlp-10fa8f50cc5a
[34] Yoshua Bengio, Ducharme Rejean &Vincent Pascal. A Neural Probabilistic Language Model. 2001. https://papers.nips.cc/paper/1839-a-neural-probabilistic-language-model.pdf
[35] Yoshua Bengio, Ducharme Rejean, Vincent Pascal & Janvin Christian. A Neural Probabilistic Language Model. March 2003. http://www.jmlr.org/papers/volume3/bengio03a/bengio03a.pdf
[36] Collobert Ronan, & Weston Jason. A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. 2008. https://ronan.collobert.com/pub/matos/2008_nlp_icml.pdf
[37] Tomas Mikolov, Greg Corrado, Kai Chen & Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. September 2013. https://arxiv.org/pdf/1301.3781.pdf
[38] https://skymind.ai/wiki/attention-mechanism-memory-network
[39] https://en.wikipedia.org/wiki/Multilayer_perceptron
[40]https://stackoverflow.com/questions/43035827/whats-the-difference-between-a-bidirectional-lstm-and-an-lstm
[41] ÖzalYildirim, A novel wavelet sequence based on deep bidirectional LSTM network model for ECG signal classification.
[42]https://machinelearningmastery.com/develop-bidirectional-lstm-sequence-classification-python-keras/
[43] Karen Simonyan and Andrew Zisserman, Very Deep Convolutional Networks for Large-Scale Visual Recognition
[44]https://cv-tricks.com/cnn/understand-resnet-alexnet-vgg-inception/
[45] https://www.cs.toronto.edu/~frossard/post/vgg16/
[46]https://www.quora.com/What-is-the-VGG-neural-network
[47] Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, Sanja Fidler, Skip-Thought Vectors
[48] https://medium.com/@sanyamagarwal/my-thoughts-on-skip-thoughts-a3e773605efa
[49]https://scholarworks.sjsu.edu/cgi/viewcontent.cgi?referer=https://www.google.com/&httpsredir=1&article=1617&context=etd_projects
[50] https://en.wikipedia.org/wiki/Residual_neural_network
[51] https://towardsdatascience.com/an-overview-of-resnet-and-its-variants-5281e2f56035
[52] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep Residual Learning for Image Recognition
[53] https://medium.com/@14prakash/understanding-and-implementing-architectures-of-resnet-and-resnext-for-state-of-the-art-image-cf51669e1624
[54] https://www.cs.unc.edu/~wliu/papers/GoogLeNet.pdf
[55] https://medium.com/@sidereal/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5
[56] https://iamaaditya.github.io/research/literature/
[57] Kushal Kafle and Christopher Kanan, Answer-Type prediction for visual question answering system.
[58] Bolei Zhou, Yuandong Tian, Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. 2015. Simple baseline for visual question answering.
[59] Lin Ma, Zhengdong Lu, and Hang Li. 2015. Learning to answer questions from image using convolutional neural network.
[60] Mengye Ren, Ryan Kiros, Richard S. Zeme, Exploring Models and Data for Image Question Answering
[61] Hyeonwoo Noh, Paul Hongsuck Seo, and Bohyung Han. 2016. Image question answering using convolutional neural network with dynamic parameter prediction.
[62] A. Fukui, D. H. Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach, “Multimodal compact bilinear pooling for visual question answering and visual grounding,” in Conference on Empirical Methods on Natural Language Processing (EMNLP), 2016.
[63] J.-H. Kim, K.-W. On, J. Kim, J.-W. Ha, and B.-T. Zhang, “Hadamard product for low-rank bilinear pooling,”
[64] Kevin J. Shih, Saurabh Singh, Derek Hoiem, Where To Look: Focus Regions for Visual Question Answering
[65] Yuke Zhu, Oliver Groth, Michael Bernstein, and Li FeiFei. 2016. Visual7w: Grounded question answering in images.
[66] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. 2016. Stacked attention networks for image question answering.
[68] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical question-image coattention for visual question answering.

[69] A. Jabri, A. Joulin, and L. van der Maaten, “Revisiting visual question answering baselines,” in European Conference on Computer Vision (ECCV), 2016
[70] J.-H. Kim, S.-W. Lee, D.-H. Kwak, M.-O. Heo, J. Kim, J.-W. Ha, and B.-T. Zhang, “Multimodal residual learning for visual qa,” in Advances in Neural Information Processing Systems (NIPS), 2016.
[71] https://vqa.cloudcv.org/