GENERATING IMAGE DESCRIPTIONS WITH A DEEP CONVOLUTIONAL NEURAL NETWORK

Vedang Matey

Lalkrishna Joshi

ABSTRACT:

Computer Vision and Natural Language Processing are used for automatically describing the content of an image. Humans can relatively quickly describe the environments they are in. Given an image, it is natural for a human to describe an immense amount of details about this image with a glance. It is one of the humans' basic abilities. Making computers imitate humans' ability to interpret the visual world has been a long-standing goal of researchers in the field of artificial intelligence. It has a tremendous potential impact. We executed an image caption generating system that uses convolutional neural network (CNN) for extracting the feature embedding of an image and feed that as an input to long short-term memory cells that generates a caption. We have used pre-trained CNN models on ResNet-101.These models were tested and compared on Flickr8K dataset.

Keywords: Neural Networks, Image Caption, RNN, LSTM, RESNET-50, Deep Learning.

INTRODUCTION:

Humans can relatively quickly describe the environments they are in. Given an image, it is natural for a human to describe an immense amount of details about this image with a glance. It is one of humans' basic abilities. Making computers imitate humans' ability to interpret the visual world has been a long-standing goal of researchers in artificial intelligence. It has a tremendous potential impact. For example, it could help visually impaired people better understand the content of images on the web. It could also provide more accurate and compact pictures/videos in scenarios such as image sharing in social networks or video surveillance systems. This project accomplishes this task using deep learning. The job of presenting a summary of the picture is complicated. First, it requires understanding the visual information and using natural language processing software to translate it into sentences. This includes creating a model capable of capturing the association present on the related image in the visual and natural language. The problem is multimodal, which generates the need to construct a hybrid model that can leverage the problem's multidimensionality. Deep neural networks have been successfully tested to build and generalize image captions. Such models often use specific deep neural networks such as a convolutional neural network (CNN), long-term memory (LSTM) networks, recurrent neural network (RNN) to learn the common embedding implicitly by encoding and decoding the direct modalities.

EXISTING WORK:

Many image captioning projects are based on vgg16 and the inception 3 model, but these models suffer a vanishing gradient effect. Through the backpropagation stage, the error is calculated, and gradient values are determined. Finally, the gradients are sent back to hidden layers, and the weights are changed accordingly. The gradient becomes lesser and lesser as it reaches the bottom of the network. Hence, the weights of the initial layers will either change very slowly or remains the same. Hence, deep network training will not converge, and accuracy will either starts to reduce or saturate at a particular value.

PROPOSED WORK:

Our image caption generator has a state-of-the-art architectural approach concatenating a high-level image feature extractor with a natural language generator. The image is fed to an image feature generator which undergoes a series of complex mathematical operations. Then it learns every aspect of the input image where it extracts the present prominent entities, contrasting objects, sharp features in the foreground, and background and jointly learned relations between all the image features. Finally, this is represented and stored in the form of a highly crucial data structure, i.e. standard sized vector, that serves as one of the main components of the architecture.

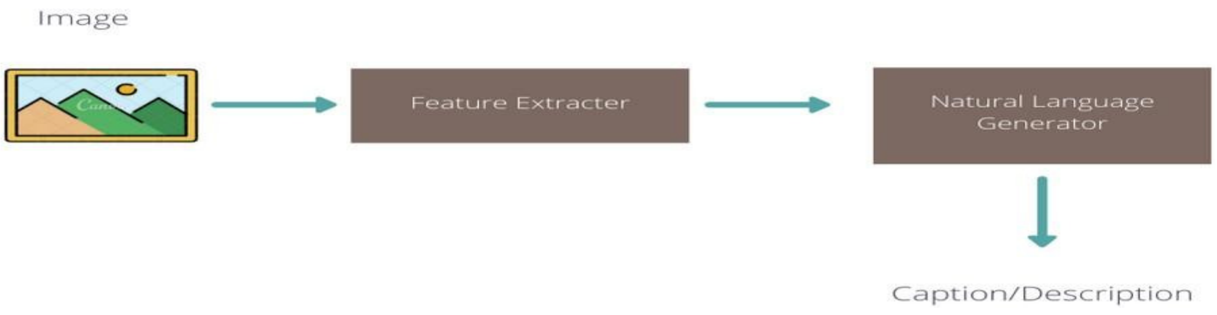


Figure 1: Image Caption Generator

LITERATURE SURVEY:

Generation of Image Captions Using VGG and ResNet CNN Models Cascaded with RNN Approach [1] In this paper, the developed system uses a combination of ResNet and VGG16 CNN architecture. VGG16 is a shallow network with less no of layers, as you go deeper it suffers vanishing gradient problem. ResNet overcomes this problem using skip connections. They have also done a comparative study between the two and find out that ResNet model has more time complexity compared to VGG16, both produce similar quality of result based of BLEU score.

IMAGE CAPTION GENERATOR USING DEEP LEARNING [2] The paper is intended to identify objects and inform people through audio and text messages. It recognizes image and converts to audio using GTTS and converts to text using LSTM. Initially, the input image is converted to a grayscale image that is processed through the Convolution Neural Network (CNN) to correctly identify the objects. Objects in the image are correctly identified using OpenCV, which is then converted to audio and text messages. The proposed method for blind people is designed to expand to people with vision loss in order to achieve their full potential.

A survey on automatic image caption generation [3] Image captioning computer vision and natural language processing, high level understanding of the semantic contents of an image, but also needs to express the information in a human-like sentence. This paper talks about to generate human readable sentence in correct English grammar. The authors do a comparative research on different methods of image caption generation like Multimodal learning, Encoder-decoder framework, Attention guided and compositional architectures.

Deep visual-semantic alignments for generating image descriptions[9] This paper uses an alignment model based on combination of Convolutional Neural Networks over image areas, bidirectional Recurrent Neural Networks over sentences and a structured objective that aligns the two modalities through a multimodal embedding. They present a model that positions sentences to visual regions that is described through a multimodal embedding. They divide the image into different areas and then predict dense descriptions.

Show and tell: A neural image caption generator [22] This paper presents a generative model based on a deep recurrent architecture that combines recent developments in computer vision and machine translations and can be used for image description predictions. This paper had used LSTM for sentence generation and has produced a good BLEU 4 of 27.7.

IMAGE FEATURE EXTRACTION MODEL

We have used pretrained model resnet-101 for image feature generation. We provide image as an input to the model which give output as feature vector of 2048 values, we have removed the last layer and used second last layer as an output layer which takes image of 224x224 size and gives output layer of each neuron having 2048 values.

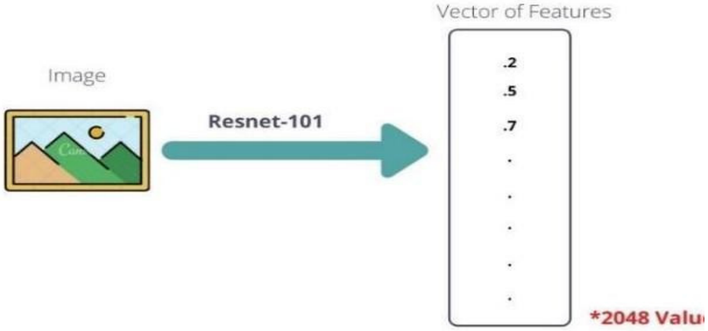


Figure 2: Feature Extraction model

ResNet (Residual Networks)-101 provides a solution to the vanishing gradient problem by adding the identity mapping of the previously occurring intermediate layers to the present layer hence performing better feature extraction than its shallower counterparts. The operation by which the resultant layer is generated is called as the residual function. This layer is represented as Y,where x is the layer taken from the shallower network and F(x) is the output learned from the current layer. Y is given by equation:

$$Y=F(x)+X$$

RESNET-101

vgg16 is a shallow network with less number of layers, as you go deeper the feature extraction gets improved but it introduces vanishing gradient problem. In order to address this problem we used ResNet (Residual Networks)-101, it provides a solution by adding the identity mapping of the previously occurring intermediate layers to the present layer hence performing better feature extraction than its shallower counterparts. The operation by which the resultant layer is generated is called as the residual function. This layer is represented as H(x), where x is the layer taken from the shallower network and F(x) is the output learned from the current layer.

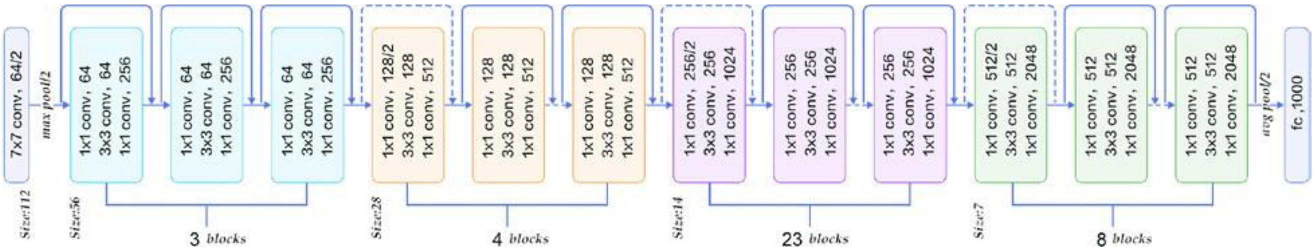


Figure 3: Resnet-101 architecture [5]

ARCHITECTURE

Our image caption generator has a state-of-the-art architectural approach concatenating a high level image feature extractor with a natural language generator. Image feature extractor takes an entire image as input and undergoes a series of complex mathematical operations and then it learns every aspect of the input image where it extracts the present prominent entities, contrasting objects, sharp features in the foreground as well as background and jointly learned relations between all the image features. This is represented and stored in the form of a extremely crucial data structure i.e. standard sized vector, that serves as one of the main components of the architecture.

NATURAL LANGUAGE GENERATION MODEL

We have used LSTM [8] model for generation of natural language that describes the image. As shown in Figure 1 we pass the fully connected layer to the hidden LSTM layer of our model. Consider a sentence Si = W0W1 W2...Wj-1 where j<=n and n is the length of the longest caption in our dataset. LSTM uses probabilistic language model where a word Wk at a position k (k<=n) is predicted by Wk = max(p(Wj | W0W1...Wk-1)) where Wj belongs to the vocabulary of the corpus and p(x) is the conditional probability of the word Wj.

LONG SHORT-TERM MEMORY BASED SENTENCE GENERATOR(LSTM)

In order to forecast / predict the next word in a sentence LSTM layer takes all the tokens predicted until current point in time as input. These tokens are converted into word vectors and are passed to the LSTM cells of this layer. LSTM cell as shown The output in time is called the cell state Ct passed to the next LSTM cell. The cell state plays a crucial role in storing the context of the subject of the sentence being generated over long sequences that are generally lost in an RNN produces an output in time and an output in depth as a result obtained from the functions in the LSTM cell. The output in depth is used to compute the next word's probability by using a categorical cross-entropy function, which gives a uniformly distributed representation of predicted probabilities of all the words in the dataset. The words with the top-m probabilities are chosen to be the most probable words where m is the beam size. he probabilities of top-m partial caption predicted till the last layer are evidential probabilities, considered to compute the conditional probabilities of top-m words from the current cell output and concatenate the words corresponding to these conditional probabilities with the partial captions corresponding to their evidential probabilities, in order to generate the new partial caption until the special '<END>' token is sampled.

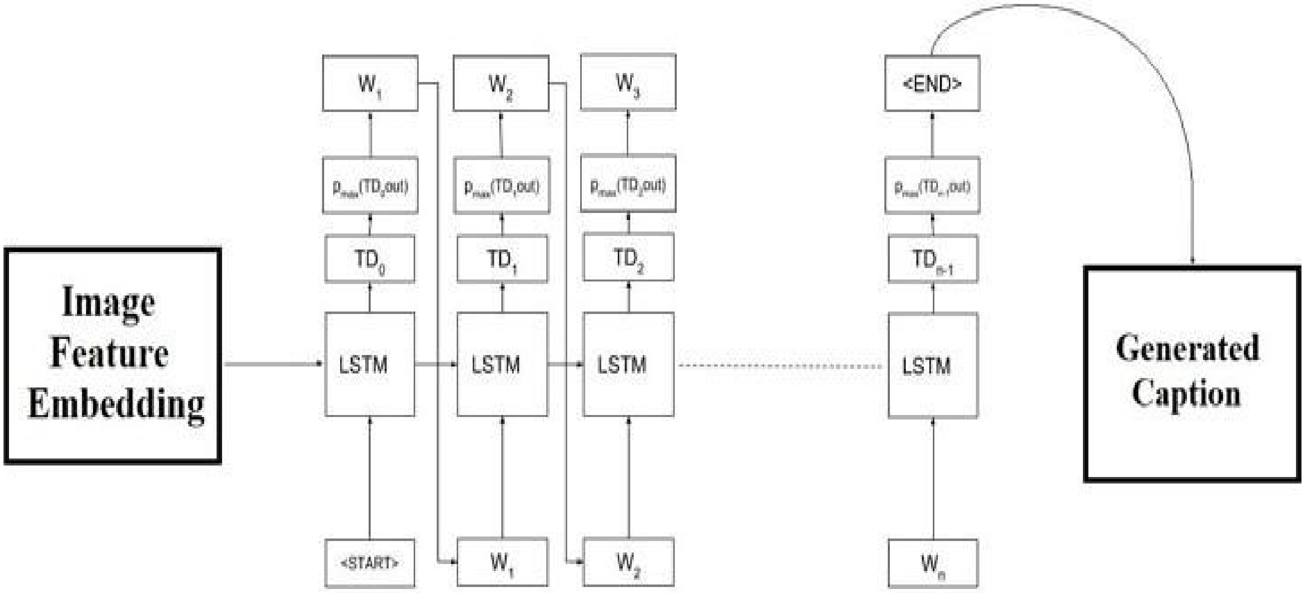


Figure 4: LSTM (Long Short Term Model)[1]

LSTM cell

The key to LSTMs is the cell state Ct, it runs straight down the entire chain, with only some minor linear interactions. The LSTM removes or adds information to the cell state, filtered down by sigmoid activation function followed by an elementwise multiplication operation called as gates. The cell state is like a vector whose values refer to context from the previous cells. Due to the fact that the new information as well as new context has to be added by the newly predicted words the forget gate ft has to make the cell state forget few of the holding contexts to various degrees defined by the sigmoid function. The tanh activation function is used to add new context from the current input word vector. The sigmoid used along with the tanh function (i ⊙ g) actually is supposed to ignore certain values in the vector obtained from tanh for reducing the noise in the new information. This selectively ignored information is excluded and the included candidate context from the new word vector is updated to the cell state (ft ⊙ ct-1 + i ⊙ g). When it comes to generating ht (ot ⊙ tanh(ct)). It has to contain only selective information in the cell state which is much useful for the next hidden cell or short term and avoid the cell state information that is relevant in the long term. The LSTM cell functions are defined as follows:

$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1})$$

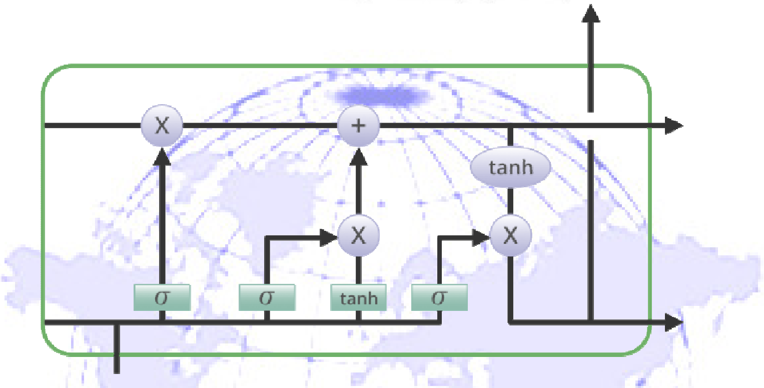
$$I = \sigma(W_{ix}x_t + W_{ih}h_{t-1})$$

$$g = \tanh(W_{gx}x_t + W_{gh}h_{t-1})$$

$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1})$$

$$C_t = f_t \odot C_{t-1} + g \odot i$$

$$h_t = \tanh(C_t) \odot o_t$$



sFigure 5: LSTM Cell Architecture[23]

MATHEMATICAL MODEL

Image is provided to the first layer which calculated the F(x), this is then added with the input from the previous layer result of which is feeded to the next layer. F(x) + X can be done only when the image size is same, if it is same the layer is called identity block but if it is not the input from the previous layer result is first processed through a convolutional layer to match the size and then added with the result before feeding to the next layer this layer is called convolutional layer.

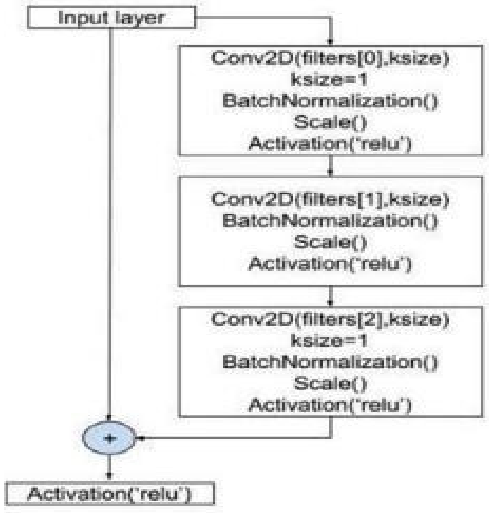


Figure 6: Identity block[1]

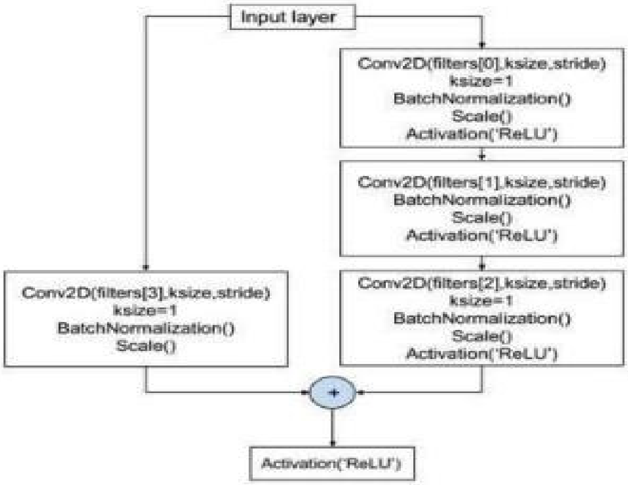


Figure 7: Convolutional block[1]

The Resnet model performs the following operation with the input provided to it:

$$\frac{n+2P-f}{s}+1*\frac{n+2P-f}{s}+1$$

Where,

n= image size, P= padding, f= filter size, s= stride

EVALUATING THE MODEL

We trained the model for over 60 epochs. After processing each epoch we got loss and accuracy of the model. To analyse the model, we plotted graphs of loss vs epochs (fig. 10) and accuracy vs epochs (fig. 11).

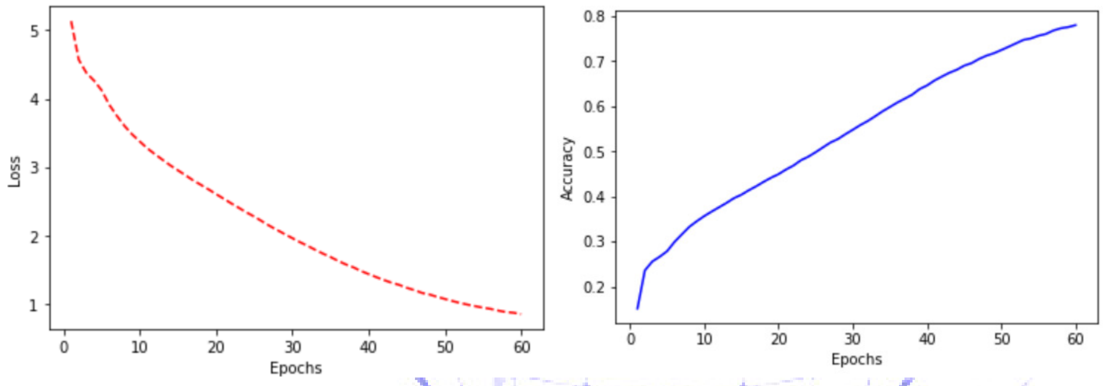
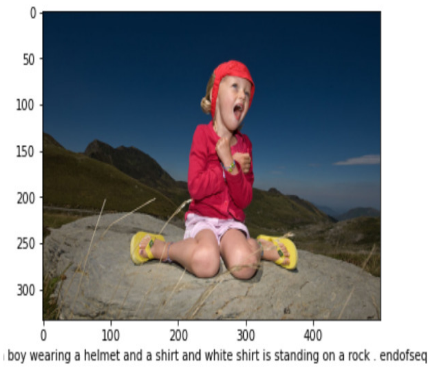


Figure 8: Loss vs. Epochs

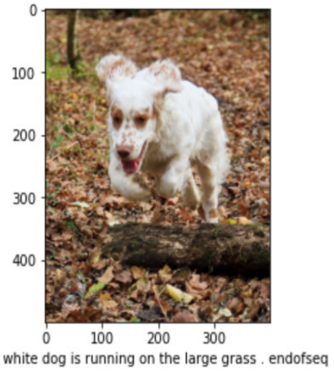
After analysing the graphs in fig. 10 and fig. 11, it is observed that as the number of epochs increases the loss drastically from 0-50 Epochs and then starts to converge at 50-60 epochs. In the same ways, accuracy increases exponentially from 0-50 epochs and then starts to converge at 50-60 epochs.

Screenshots



boy wearing a helmet and a shirt and white shirt is standing on a rock . endofseq

Figure 10: Sample Prediction 1



white dog is running on the large grass . endofseq

Figure 11: Sample Prediction 2

CONCLUSION

We have presented an Image Captioning System that uses convolutional neural network to generate image embeddings followed by recurrent neural network that finally produces description of the image in simple language understood by humans. We have executed a CNN-RNN model by building an image caption generator. Key points to be noted are that our model depends on the data. So, it cannot forecast / predict the words that are out of its vocabulary. We trained model on 8091 image and achieved accuracy of 78.82%.

REFERENCES

1. Bhalekar M., Sureka S., Joshi S., Bedekar M. (2020) Generation of Image Captions Using VGG and ResNet CNN Models Cascaded with RNN Approach. In: Agarwal S., Verma S., Agrawal D. (eds) Machine Intelligence and Signal Processing. MISP 2019. Advances in Intelligent Systems and Computing, vol 1085. Springer, Singapore. https://doi.org/10.1007/978-981-15-1366-4_3
2. D.Kaviyarasu, B. , K. S. R. (2020). IMAGE CAPTION GENERATOR USING DEEP LEARNING. International Journal of Advanced Science and Technology, 29(3s), 975 - 980. Retrieved from http://sersc.org/journals/index.php/IJAST/article/view/5927
3. Bai, S. and An, S., 2018. A survey on automatic image caption generation. Neurocomputing, 311, pp.291-304. ISSN 0925-2312, https://doi.org/10.1016/j.neucom.2018.05.080
4. J. Dai, K. He, and J. Sun. BoxSup:Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. IEEE International conference on computer vision ICCV, pp 1635-1643, In arXiv:1503.01640v2, 2015.
5. S. Das. CNN architectures: Lenet, alexnet, vgg, googlenet, resnet and more. https://medium.com/@sidereal/cnns-architectures-lenetalexnet-vgg-googlenet-resnet-and-more666091488df5, 2017.
6. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. FeiFei. Imagenet: A large-scale hierarchical image database. IEEE Computer Vision and Pattern Recognition (CVPR), pages 248–255, 2009.
7. H. Fang, S. Gupta, F. Iandola, R. Srivastava, L. Deng, P. Dollar, J. Gao, X. He, M. Mitchell, J. C. Platt, C. L. Zitnick, and G. Zweig. From captions to visual concepts and back. IEEE Conference on Computer Vision and Pattern Recognition(CVPR), pp 1473-1482, 2015. In arXiv:1411.4952v3.
8. M. Guillaumin, J. Verbeek, and C. Schmid. Multimodal semi- supervised learning for image classification. In IEEE Computer Society Conference on Computer Vision and Pattern

Recognition, 2010.

9. Karpathy, L. Fei-Fei. Deep visual-semantic alignments for generating imag descriptions. IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 14, No. 8, August 2015.
10. R. Kiros, R. Salakhutdinov, and R. Zemel. Multimodal neural language models. In International Conference on Machine Learning, pp 595-603, 2014.
11. R. Kiros, R. Salakhutdinov, and R. S. Zemel. Unifying visual- semantic embeddings with multimodal neural language models. Computing Research Repository (CoRR) In Machine Learning, arXiv: 1411.2539v1, 2014
12. T.-Y. Lin, M. Maire, S. Belongie, L.Bourdev, R.Girshick , J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick,. Microsoft COCO: Common Objects in Context. Springer European Conference on Computer Vision ECCV, Springer, pp 740-755, 2014.
13. C.Olah. Understanding LSTM networks. http://colah.github.io/posts/2015-08-Understanding-LSTMs/, 2015.
14. Joseph Redmon, Ali Farhadi. YOLO 9000: Better, Faster, Stronger. IEEE conference on Computer Vision and Pattern Recognition (CVPR), In arXiv:1612.08242v1, July 2017.
15. K. Simonyan and A. Zisserman. Very deep convolutional networks for large scale image recognition. International Conference on Learning Representations (ICLR), arXiv1409.1556v6, 2015.
16. Moses Soh. Learning CNN-LSTM Architectures for Image Caption Generation. Stanford University, Proceeding, 2016.
17. C. Szegedy, A. Toshev and D. Erhan. Deep neural network for object detection. Proceeding of Neural Information Processing Systems NIPS, 2013.
18. Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond Mooney, Kate Saenko. Translating Videos to Natural Language Using Deep Recurrent Neural Networks. Annual Conference of the North American Chapter of the Association for Computational Linguistics (ACL). In arXiv:1412.4729v3, 2015.
19. Oriol Vinyals, Alexander Toshev, Samy Bengio, Dumitru Erhan. Show and Tell: A Neural Image Caption Generator. In arXiv:1411.4555v2, 2015.
20. Xinggang Wang, Zhuotun Zhu, Cong Yao, Xiang Bai. Relaxed Multiple-Instance SVM with Application to Object Discovery. In IEEE International Conference on Computer Vision, 2015.
21. Kelvin Xu, Jimmy Lei Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan

Salakhutdinov, Richard S. Zemel, Yoshua Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In arXiv:1502.03044v3, 2016.

22. O. Vinyals, A. Toshev, S. Bengio and D. Erhan, "Show and tell: A neural image caption generator," 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3156-3164, doi: 10.1109/CVPR.2015.7298935.

23. https://www.geeksforgeeks.org/understanding-of-lstm-networks/