Automated Image Caption Generator Using Deep Learning

Abhinav Sharma[1], Hansin Malhotra[2], Jaskaran Singh[3], Mr. Prem Prakash Agarwal [4]

[1,2,3] Student, Information Technology, Galgotias College of Engineering and Technology, Gr. Noida

[4]Asstiant Professor, Information Technology Engineering, Galgotias College of Engineering and Technology, Gr. Noida

Abstract

This paper aims to generate automatic captions by reading image content. At present images are defined by human intervention and become an impossible task in big business details. The image website is provided as an integral part of the Convolutional Neural Network (CNN) to create a "vector thought" that removes the features and nuances from our image and the RNN (Recurrent Neural Network) decoder is used to translate the features and features provided by our image to find the sequence, a logical description of the image. In this paper, we systematically analyze the various methods of image depth with network-based networks and fictional models to conclude with a highly efficient model with good optimization. The analyzed models contain both and without the 'attention' concept of enhancing the caption capabilities that produce the model. All models are trained in the same concrete comparison database.

***Keywords**—Deep Neural Network, Image, automated Caption, Description, Long Short Term Memory (LSTM), Deep Learning, CNN, RNN, feature extraction, attention, Visual to text, image and video captioning, content 29 understanding, text description, summarization.*

Introduction

Caption production is an interesting artificial problem when finding a descriptive sentence for a given image.

Making a caption for an image is called a caption for an image. Captioning of a photo requires identifying the main features, their attributes and their relationship to the image. It also needs to produce sentences that are accurate and concise. Deep learning modes are able to handle the difficulties and challenges of image captioning.

It involves two techniques from a computer perspective to understand the content of an image and a language model from the field of natural language processing to convert image comprehension into words in the right way. Image captioning has a variety of applications such as recommendations for editing programs, use of visual assistants, image insertion, visually impaired people, social media, and many other language processing applications. More recently, in-depth study methods have yielded superior results in examples of this problem. It has been shown that in-depth learning models are able to achieve good results in the field of caption problems. Instead of requiring complex data fixes or a pipeline of specially designed models, a single-end model can be defined to predict captions, given a picture.

Aim

The aim of this project is to develop software that can generate descriptive captions for images using neural network language models. This project can serve as a visual aid for visually impaired people, as it can pinpoint nearby objects with a camera and provide output in the form of sound. The application can provide a highly interactive platform for people with special disabilities..

Project Scope

The goal is to design a web application that integrates all image definition functions and provides a user interface.

Through in-depth learning strategies, the project makes:

1. Photography: Recognizing the various types of objects in a picture and creating a logical sentence that describes the image to people with poor eyesight.

2. Text to be a transformation of speech: By giving an effect in the form of sound.

Feasibility Study

Product

Image caption generator is a function that captures computer imagery and natural language processing concepts to visualize the image and interpret it in the native language such as English.

Technical Feasibility

For this Python project, we will use a caption generator using CNN (Convolutional Neural Networks) and LSTM (Long-Term Memory). Image features will be released on Xception which is a CNN model trained in the image database and feeds the features on the LSTM model which will be responsible for captioning the image. So we can say that our project is possible..

Socially Feasible

Image Captioning Generator will work with Windows OS, Mac OS and even Android. We will also use this on iOS..

Economically Feasible

Image Image Captioning Project is more expensive than other similar projects on the market and is easy to use.

Requirements

Dataset:

The database used here is FLICKR 8K containing around 8091 images and 5 captions for each image. If you have a powerful system with more than 16 GB RAM and an image card with more than 4 GB memory, you can try taking the FLICKR 30K with around 30,000 images with captions.

Dependencies:

- Keras
- Tensorflow GPU
- Pre-trained ResNet 50 weights
- NLTK
- Matplotlib

Methodology and Architecture

We use a pre-trained Convolutional Neural Network format as an encoder to extract and insert image elements into a vector space of large size. CNN is a type of Artificial Neural Network feed unlike unlike fully connected layers, it has only a subset of notes in the previous layer connected to describe the image.

To take advantage of this opportunity in native language, the most common way is to extract sequential information and translate it into language. With this function, the LSTM-based Recurrent Neural Network is used as a decoder to convert embedded features into natural language definitions. In recent photo captioning functions, they extract a feature map from the top layers of CNN, transfer them to a specific RNN format and use softwaremax to get word points in every step.

With the caption function of image captions, duplicate neural networks help to collect semantics of a sentence. Sent strings are separated by words, each with a GloVe vector presentation found in the view table.

These vocal cords are integrated into a recurring neural network in sequence, incorporating the concept of semantic meaning over the sequence of words in their hidden form. We are dealing with a hidden situation after a recurring net sees the last word in a sentence as a punctuation.

The core of the LSTM model is a memory cell that constantly informs the step of the input that has been detected up to this point (see Figure 2). The behavior of the cell is controlled by "gates" - the layers that are used in the plural are less and thus can keep the value from the gate with a gate if the gate is 1 or zero this number if the gate is 0. Specifically, three gates are used to control that you forget the current value of the cell (forget gate f), if you must h σ σ c σ input LSTM memory block block predict the name of the softmax input gate output f forget gate f update term ct-1 ct mt x read its input (input gate i) and that you must extract the value of the new cell (output gate o).

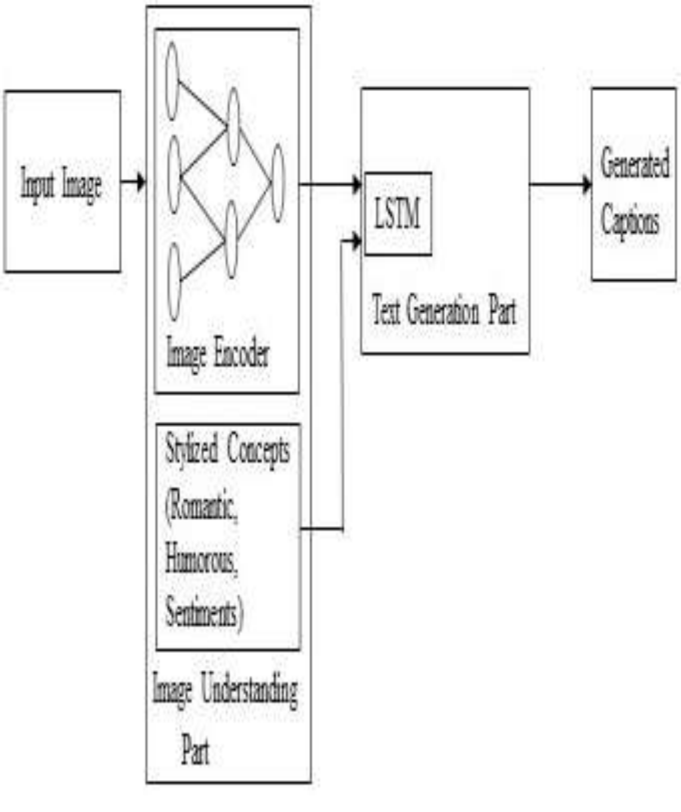


Figure 1. Architecture overview of the Image Caption Generator

System Architecture

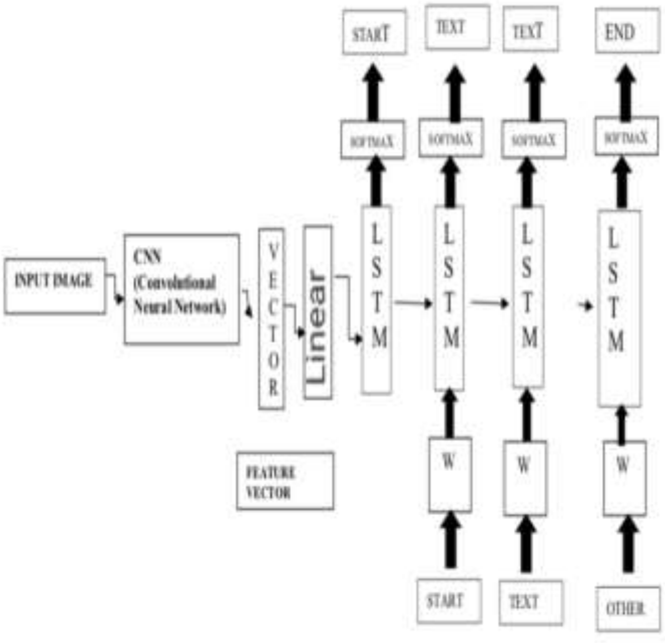


Figure2. Proposed Model for Image Caption Generator

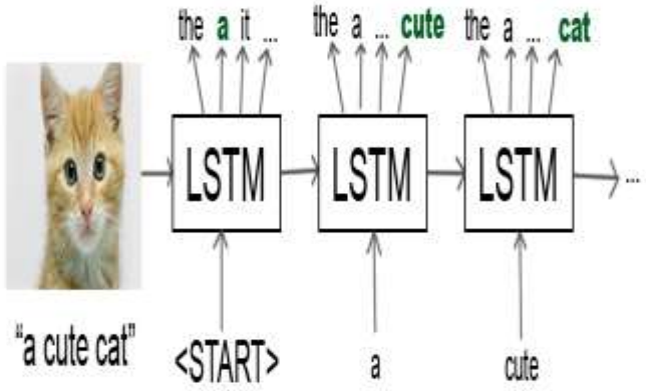
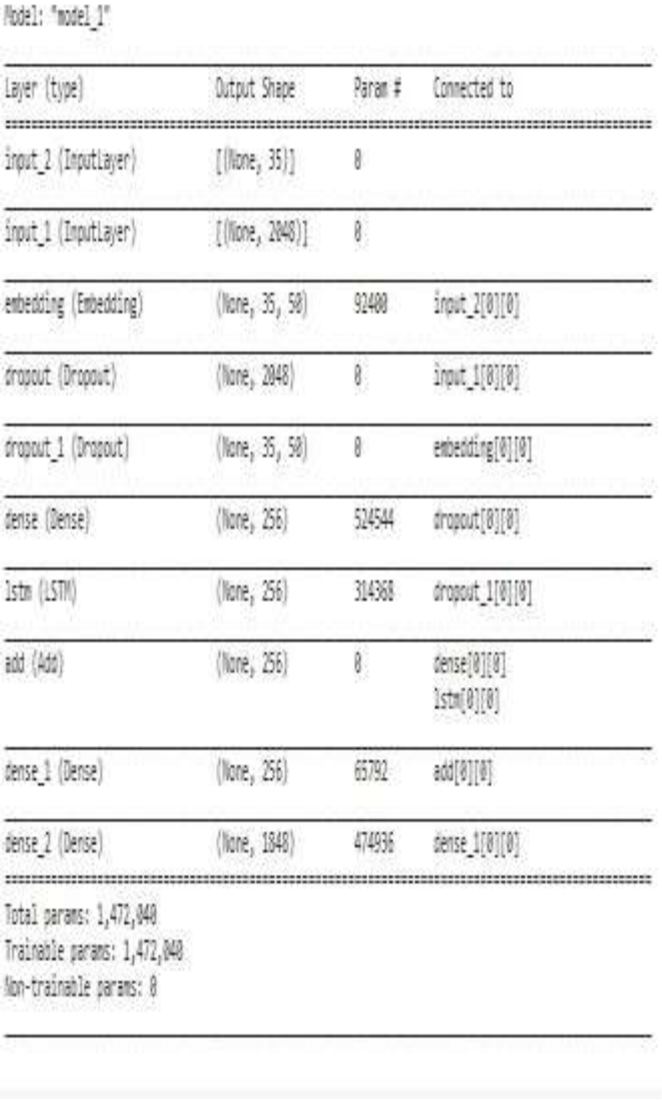


Figure3. RNN model for describing pictures

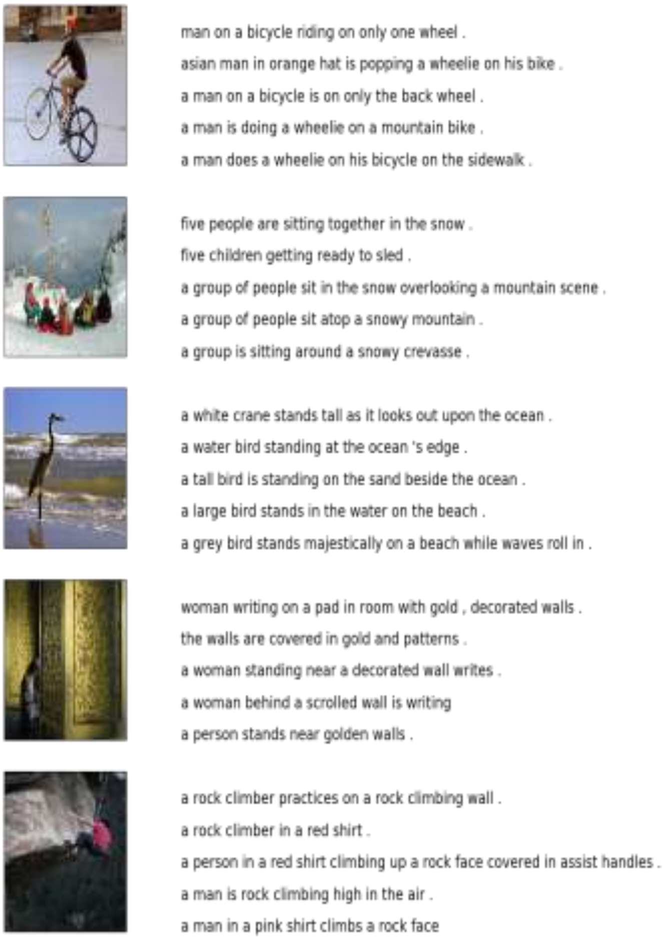
Training

The LSTM model is trained to predict each word of a sentence after seeing the image and all previous words as defined by p (St | I, S0, .., St - 1). For this purpose, it is instructive to think about LSTM in an open way - a copy of the LSTM memory is designed for the LSTM LSTM LSTM WeS1 WeSN-1 p1 p2 pN log p1 (S1) log p2 (S2) log pN (SN)) ... image of -LSTM WeS0 S0 S1 SN-1 Figure 3. The LSTM model is integrated with CNN image embedding (as described in [12]) and word embedding. Uncontrolled links between LSTM memories are blue and corrects spells for recurring connections in Figure 2. All LSTMs share the same parameters. the image with each word in the sentence as all LSTMs share the same parameters and the mt-1 output of LSTM during t - 1 is given to LSTM during t (see Figure 3). All recurring connections are converted into forwarding links in an unregistered version. In detail, if we define the input image and in S = (S0, ..., SN) the actual state of the image description, the registration process reads as follows: x - 1 = CNN (I) (10) xt = WeSt, t ∈ { 0. . . N - 1} (11) pt + 1 = LSTM (xt), t 0 {0. . . N - 1} (12) where we represent each word as one hot vector St of dimension equal to the size of the dictionary. Note that we define in S0 a basic starting word and SN a special stop word that describes the beginning and end of a sentence. Mainly by releasing the stop name LSTM signs to produce a complete sentence. Both the image and the words are drawn in the same place, the image through the CNN concept, the words through the embedding of the word We. The image I insert only once, in t = -1, informs LSTM about the image content. We have conceptually verified that each image feed as an additional extension produces negative results, as the network can explicitly use the image in the image and override it. Our loss is the sum of the negative details of each word log in each step as follows: L (I, S) = - X N t = 1 log pt (St). (13) The above losses are reduced w.r.t. all LSTM parameters, the top layer of CNN embedded image and We embed text. Input There are many methods that can be used to make a sentence given a picture, you have a NIC. The first is Sampling where we simply sample the first word ac by going to p1, and then provide the corresponding embedding as input with the sample p2, it continues like this until we measure a special end-of-sentence token or longer length. The second is BeamSearch: think of a set of better sentences k until t as selected to produce sentences of size t + 1, and save only the best results for them. This is closer to S = arg maxS0 p (S 0 | I). We used the BeamSearch method for the following experience sessions, with a size 20 beam. Using 1 ray size (e.g., greedy search) reduced our results by 2 BLEU points on average.



```
Model: "model_1"
____________________________________________________________________________
Layer (type)              Output Shape        Param #    Connected to
============================================================================
input_2 (InputLayer)      [(None, 35)]        0
____________________________________________________________________________
input_1 (InputLayer)      [(None, 2048)]      0
____________________________________________________________________________
embedding (Embedding)     (None, 35, 50)      92400      input_2[0][0]
____________________________________________________________________________
dropout (Dropout)         (None, 2048)        0          input_1[0][0]
____________________________________________________________________________
dropout_1 (Dropout)       (None, 35, 50)      0          embedding[0][0]
____________________________________________________________________________
dense (Dense)             (None, 256)         524544     dropout[0][0]
____________________________________________________________________________
lstm (LSTM)               (None, 256)         314368     dropout_1[0][0]
____________________________________________________________________________
add (Add)                 (None, 256)         0          dense[0][0]
                                                         lstm[0][0]
____________________________________________________________________________
dense_1 (Dense)           (None, 256)         65792      add[0][0]
____________________________________________________________________________
dense_2 (Dense)           (None, 1848)        474936     dense_1[0][0]
============================================================================
Total params: 1,472,040
Trainable params: 1,472,040
Non-trainable params: 0
____________________________________________________________________________
```

Figure4. Model Summary



man on a bicycle riding on only one wheel .
asian man in orange hat is popping a wheelie on his bike .
a man on a bicycle is on only the back wheel .
a man is doing a wheelie on a mountain bike .
a man does a wheelie on his bicycle on the sidewalk .

five people are sitting together in the snow .
five children getting ready to sled .
a group of people sit in the snow overlooking a mountain scene .
a group of people sit atop a snowy mountain .
a group is sitting around a snowy crevasse .

a white crane stands tall as it looks out upon the ocean .
a water bird standing at the ocean 's edge .
a tall bird is standing on the sand beside the ocean .
a large bird stands in the water on the beach .
a grey bird stands majestically on a beach while waves roll in .

woman writing on a pad in room with gold , decorated walls .
the walls are covered in gold and patterns .
a woman standing near a decorated wall writes .
a woman behind a scrolled wall is writing
a person stands near golden walls .

a rock climber practices on a rock climbing wall .
a rock climber in a red shirt .
a person in a red shirt climbing up a rock face covered in assist handles .
a man is rock climbing high in the air .
a man in a pink shirt climbs a rock face

Figure5. Training Dataset

Planning of work

1. Initially, we studied image classification and natural language production algorithms for example CNN and RNN.
2. After that, we will link the techniques to create a hybrid approach to building the Image Caption Generation model.
3. Next, we will test the hybrid method in different test cases and measure its accuracy.
4. The hybrid algorithm produced will be able to produce multiple captions in a single frame.
5. Captions are produced in the form of text and are converted into audio by helping the visually impaired to understand the content of the image.
6. Finally, we will get the Real-Time Simulation Results.

Procedure for Creating Model Image Caption Generator
So, to make our own photo caption model, we will incorporate this art. Also called the CNN-RNN model.

1. First, we import all the required packages.
2. Obtaining and performing data purification.
3.The vector feature is removed from all images
4. Model Training database loaded.
5. Marking.
6. Create a Data generator.
7. Explain the CNN-RNN model.
8. Model training
9. Model testing

Results

Photo captioning has greatly improved in recent years. Recent work based on in-depth reading techniques has led to the emergence of an accurate caption of the image. Image text interpretation can enhance the effectiveness of image acquisition with content, the wide range of application for increased visual perception in the medical, security, military and other fields, with a wide range of application capabilities. At the same time, theoretical framework and research methods of captioning can promote the development of theory and the use of image annotation and visual questionnaire (VQA), media retrieval, video writing and video chat, with educational and practical application value.

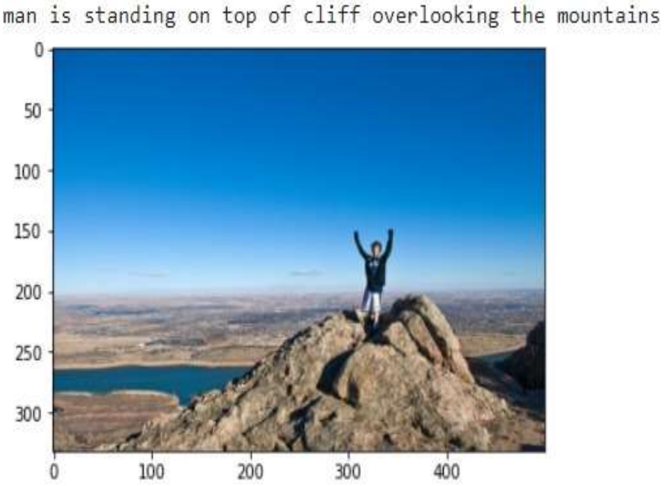


Figure6. Results

We have introduced the AICG, an end-to-end network system that can automatically view an image and make a meaningful interpretation in plain English. AICG is based on a convolution neural network that captures image in a composite representation, followed by a recurring neural network that produces the same sentence. The model is trained to increase the chances of a sentence being given a picture. Multiple database tests show the strength of the AICG in terms of quality results (the sentences that are most logical) and quantitative tests, using standard metrics or BLEU, metrics used for machine translation to test the quality of the sentences produced. It is clear from this study that, as the size of the available data stocks for image definition increases, so does the effectiveness of methods such as the AICG. In addition, it will be interesting to see how one can use unattended data, both from images alone and text alone, to improve image definition methods.

The input in our model is [x1, x2] and the result will be y, where x1 is the vector of the 2048 element of that image, x2 input text sequence and y output sequence output model should predict.

Future Scope

Next time we will use this model in the video to produce a whole story line for short clips. This will help visually impaired people to understand the visual clips as this model will help to create the text and then the text can be converted to audio format.

We can make a blind product for them to lead them on the streets without the support of anyone else. We can do this by first turning the scene into a text and then the text into a word. Both are now popular applications for in-depth learning.

It will also help with CCTV cameras Security surveillance cameras are everywhere today, but also with worldviews, if we can also produce relevant captions, then we can raise alarms as soon as something malicious happens somewhere else. This can help to reduce some crime or accidents.

Automatic driving is one of the biggest challenges and if we can't properly caption the area around the car, it can give power to the self-driving system.

Human Evaluation Figure

The result of a personal assessment of the definitions provided by the NIC, as well as the reference system and global reality in various databases. We see that the NIC is better than the reference system, but it is clearer than the real, as expected. This shows that BLEU is not a complete metric, because it does not take well with the difference between NIC and human definitions tested by the characters. Examples of scaled images can be seen in Figure 5. It is interesting to see, for example in the second picture of the first column, how the model is able to see the frisbee given its size.

Analysis of Embeddings

In order to represent the previous word St−1 as input to the decoding LSTM producing St, we use word embedding vectors [22], which have the advantage of being indepen dent of the size of the dictionary (contrary to a simpler one hot-encoding approach). Furthermore, these word embed dings can be jointly trained with the rest of the model. It is remarkable to see how the learned representations have captured some semantic from the statistics of the language. Table 4 shows, for a few example words, the nearest other words found in the learned embedding space. Note how some of the relationships learned by the model will help the vision component. Indeed, having “horse”, “pony”, and “donkey” close to each other will encourage the CNN to extract features that are relevant to horse-looking animals. We hypothesize that, in the extreme case where we see very few examples of a class (e.g., “unicorn”), its proximity to other word embeddings (e.g., “horse”) should provide a lot more information that would be completely lost with more traditional bag-of-words based approaches.

Conclusion

The model has been successfully trained to generate the captions as expected for the images. The caption generation has constantly been improved by fine tuning the model with different hyper parameter. Higher BLEU score indicates that the generated captions are very similar to those of the actual caption present on the images. Below you will find a table displaying different BLEU scores obtained by tuning the parameters:

With the help of Tensorboard, we were able to see how different training process had an impact on the model.

The validation loss falls upto 5th epoch and then increases afterwards, while the training loss still continues falling

The following were the major outcomes and aboservations of the training process and testing the model on the test data:

The validation loss increases after 5th epoch in most cases even though the training loss decreases over time. This indicates that the model is over fitting and the training needs to stop.

Higher BLEU score doesn't aways translate to better generated captions. If the model overfits on your training data, it will lead the model to go through details in the image and generate out captions which don't make sense. It can be seen in the good and the bad captions generated above.

References

[1] S. Yan, F. wu, J. Smith and W. Lu, "Image Captioning via a Hierarchical Attention Mechanism and Policy Gradient Optimization," LATEX CLASS FILES, vol. 14, 11 January 2019.

[2] R. Staniute and D. Sesok, "A Systematic Literature Review on Image Captioning," Applied Sciences, vol. 9, no. 10, 16 March 2019.

[3] G. Ding, M. Chen, S. Zhao, H. Chen, J. Han and Q. Liu, "Neural Image Caption Generation with Weighted Training and Reference," Cognitive Computation, 08 August 2018.

[4] J. Chen, W. Dong and M. Li, "Image Caption Generator Based On Deep Neural Networks," March 2018.

[5] S. Bai and S. An, "A Survey on Automatic Image Caption Generation," Neurocomputing, 13 April 2018.

[6] D. S. Whitehead, L. Huang, H. and S.-F. Chang, "Entity-aware Image Caption Generation," in Empirical Methods in Natural Language Processing, Brussels, 2018.

[7] Sun Chengjian, Songhao Zhu, Zhe Shi, “Image Annotation Via Deep Neural Network”, Published in: 2015 14th IAPR International Conference on Machine Vision Applications (MVA).

[8] Venkatesh N. Murthy, Subhransu Maji, R Manmatha, “Automatic Image Annotation using Deep learning representations”, ICMR '15 Proceedings of the 5th ACM on International Conference on Multimedia Retrieva.

[9] Oriol Vinyals, Alexander Toshev, Samy Bengio, Dumitru Erhan, “Show and Tell: A Neural Image Caption Generator”, published 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)

[10] Kelvin Xu, Jimmy Lei Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S. Zemel, Yoshua Benjio, “Show, attend and tell: neural image caption generation with visual attention”, ICML’15 Proceedings of the 32nd International Conference on Machine Learning – Volume 37, Pages 2048-2057.

[11] Kishore Papineni, Salim Roukos, Todd Ward, Wei-Jing Zhu, “BLEU: a Method for Automatic Evaluation of Machine Translation”, Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), Philadelphia, July 2002, pp. 311-318.

[12] R. Socher, A. Karpathy, Q. V. Le, C. Manning, and A. Y. Ng. Grounded compositional semantics for finding and describ ing images with sentences. In ACL, 2014.

[13] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describ ing images using 1 million captioned photographs. In NIPS, 2011.

[14] Simonyan, Karen, and Andrew Zisserman. ”Very deep convolutional networks for large-scale image recognition.” arXiv preprint arXiv:1409.1556 (2014).

[15] Yao, Benjamin Z., et al. ”I2t: Image parsing to text description.” Proceedings of the IEEE 98.8 (2010): 1485-1508.

[16] Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ”ImageNet Large Scale Visual Recognition Challenge.” International Journal of Computer Vision Int J Comput Vis 115.3 (2015): 211-52. Web. 19 Apr. 2016

[17] A. Aker and R. Gaizauskas. Generating image descriptions using dependency relational patterns. In ACL, 2010.

[18] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. arXiv:1409.0473, 2014.

[19] J. Mao, W. Xu, Y. Yang, J. Wang, and A. Yuille. Explain images with multimodal recurrent neural networks. In arXiv:1410.1090, 2014.

[20] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. In ICLR, 2013.

[21] M. Mitchell, X. Han, J. Dodge, A. Mensch, A. Goyal, A. C. Berg, K. Yamaguchi, T. L. Berg, K. Stratos, and H. D. III. Midge: Generatingimagedescriptionsfromcomputervision detections. In EACL, 2012.

[22] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describing images using 1 million captioned photographs. In NIPS, 2011.

[23] K. Papineni, S. Roukos, T. Ward, and W. J. Zhu. BLEU: A method for automatic evaluation of machine translation. In ACL, 2002.

[24] C. Rashtchian, P. Young, M. Hodosh, and J. Hockenmaier. Collecting image annotations using amazon’s mechanical turk. In NAACL HLT Workshop on Creating Speech and Language Datawith Amazon’s Mechanical Turk, pages139– 147, 2010.

[25] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge, 2014.

[26] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, andY.LeCun. Overfeat: Integratedrecognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229, 2013.

[27] J. Mao, W. Xu, Y. Yang, J. Wang, and A. Yuille. Ex plain images with multimodal recurrent neural networks. In arXiv:1410.1090, 2014.

[28] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. In ICLR, 2013.

[29] M. Mitchell, X. Han, J. Dodge, A. Mensch, A. Goyal, A. C. Berg, K. Yamaguchi, T. L. Berg, K. Stratos, and H. D. III. Midge: Generating image descriptions from computer vision detections. In EACL, 2012.

[30] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describ ing images using 1 million captioned photographs. In NIPS, 2011.

[31] K. Papineni, S. Roukos, T. Ward, and W. J. Zhu. BLEU: A method for automatic evaluation of machine translation. In ACL, 2002.

[32] C. Rashtchian, P. Young, M. Hodosh, and J. Hockenmaier. Collecting image annotations using amazon’s mechanical turk. In NAACL HLT Workshop on Creating Speech and Language Data with Amazon’s Mechanical Turk, pages 139– 147, 2010.

[33] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge, 2014.

[34] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229, 2013.

[35] R. Socher, A. Karpathy, Q. V. Le, C. Manning, and A. Y. Ng. Grounded compositional semantics for finding and describ ing images with sentences. In ACL, 2014.

[36] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to seq6uence learning with neural networks. In NIPS, 2014.

[37] R. Vedantam, C. L. Zitnick, and D. Parikh. CIDEr: Consensus-based image description evaluation. In arXiv:1411.5726, 2015.

[38] B. Z. Yao, X. Yang, L. Lin, M. W. Lee, and S.-C. Zhu. I2t: Image parsing to text description. Proceedings of the IEEE, 98(8), 2010.

[39] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier. From im age descriptions to visual denotations: New similarity met rics for semantic inference over event descriptions. In ACL, 2014.

[40] W. Zaremba, I. Sutskever, and O. Vinyals. Recurrent neural network regularization. In arXiv:1409.2329, 2014

[41] Fang, Hao, Saurabh Gupta, Forrest Iandola, Rupesh K. Srivastava, Li Deng, Piotr Dollar, Jianfeng Gao, Xiaodong He, Margaret Mitchell, John C. Platt, C. Lawrence Zitnick, and Geoffrey Zweig. ”From Captions to Visual Concepts and Back.” 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015). Web. 27 Apr. 2016

[42]” Visual Impairment and Blindness.” World Health Organization. (2014). Web. 10 Apr. 2016

[43] Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ”ImageNet Large Scale Visual Recognition Challenge.” International Journal of Computer Vision Int J Comput Vis 115.3 (2015): 211-52. Web. 19 Apr. 2016

[44] Everingham, Mark, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. ”The Pascal Visual Object Classes (VOC) Challenge.” International Journal of Computer Vision Int J Comput Vis 88.2 (2009): 303-38. Web. 22 May 2016

[45] Farhadi, Ali, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hocken□ maier, and David Forsyth. ”Every Picture Tells a Story: Generating Sentences from Images.” Computer Vision ECCV 2010 Lecture Notes in Computer Science (2010): 15-29. Web. 5 Apr. 2016

[46] Kulkarni, Girish, Visruth Premraj, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C. Berg, and Tamara L. Berg. ”Baby Talk: Understanding and Generating Simple Image Descriptions.” Cvpr 2011 (2011). Web. 27 May 2016

[47] Li, Li-Jia, R. Socher, and Li Fei-Fei. ”Towards Total Scene Understanding: Classification, Annotation and Segmentation in an Automatic Framework.” 2009 IEEE Conference on Computer Vision and Pattern Recogni tion (2009). Web. 21 Apr. 2016

[48] Gould, Stephen, Richard Fulton, and Daphne Koller. ”Decomposing a Scene into Geometric and Semanti□ cally Consistent Regions.” 2009 IEEE 12th International Conference on Computer Vision (2009). Web. 6 May 2016

[49] Fidler, Sanja, Abhishek Sharma, and Raquel Urtasun. ”A Sentence Is Worth a Thousand Pixels.” 2013 IEEE Conference on Computer Vision and Pattern Recognition (2013). Web. 18 May 2016

[50] Li, Li-Jia, and Li Fei-Fei. ”What, Where and Who? Classifying Events by Scene and Object Recognition.” 2007 IEEE 11th International Conference on Computer Vision (2007). Web. 10 Apr. 2016.

[51] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In Advances in neural information processing systems. 2672–2680.

[52] Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra. 2015. DRAW: A recurrent neural network for image generation. In Proceedings of Machine Learning Research. 1462–1471.