IMAGE CAPTIONING USING IMAGENET

[1]S.INIYAN, [2]PRASHANTH MAHESWARI, [3]RAHUL AJITH

ABSTRACT*—There are many cases wherein an image has to be described to people, or a caption is needed for multiple reasons. Giving pre-defined captions for each specific image can be a long and dreary job for a human being when there is an excessive number of images involved. This is where the image captioning system comes into play. In this paper, we explore the mapping between images and their descriptions in a sentence form. It can be useful in creating something that generates natural language which can describe the image in a manner that is understandable by human beings. Making for more human like responses can greatly benefit the human race as many things can be computerized in the near future which takes the tedious work of captioning given images in a large scale off our hands. What is the use for computer generated image captioning? People may need to find out what the object in front of them is, in case it is something that they aren't acquainted with, or they may want a description of what's happening in the given image. If the system has a reference that can be used to detect the image, it can be beneficial to the end user. On a large scale, this can be used as a tool that can work as an assistant, potentially connected to a camera or a storage device which contains images for it to work on.*

Keywords—*Captioning, Imagenet*

I. INTRODUCTION

There is an elephantine amount of images from a multitude of sources like the internet, Ads, articles as well as documents. These consist of images that humans would have to understand and describe on their own. We can understand them without the captions. On the other hand, machines have to interpret an image caption, if us humans require one generated by it..

Image captioning is relevant because of a multitude of reasons. For example, they can be utilized for autonomous image indexing, which is important for Content-Based Image Retrieval and thus, it can be utilized in many areas, such as medicine, military, education, commercial sector, digital libraries, and web surfing. We can observe that some social media platforms can generate descriptions for the images. The descriptions may include our location (e.g., beach, mountains), our clothes and importantly what we are doing there. For example, the sentence "X is playing with Y on the beach" can be a possible description for an image.

[1] Assistant Professor, Department of Computer Science &Engineering ,SRM Institute of Science & Technology SRM Nagar, Kattankulathur, Kancheepuram, Tamil Nadu, India,603203

[2] UG Student, Department of Computer Science & Engineering ,SRM Institute of Science & Technology SRM Nagar, Kattankulathur, Kancheepuram, Tamil Nadu, India,603203

[3] UG Student, Department of Computer Science & Engineering ,SRM Institute of Science & Technology SRM Nagar, Kattankulathur, Kancheepuram, Tamil Nadu, India,603203

Received: 22 Feb 2020 | Revised: 13 Mar 2020 | Accepted: 05 Apr 2020

Image captioning is currently a booming part of Artificial Intelligence, where the system has to understand an image and then describe it in a human language.

Image understanding relates to the detection and recognition of objects in the given images. It detects the type of place or location, characteristics of certain objects and interactions between multiple objects or entities.

Image understanding depends on gathering the features of an image. There are two categories of techniques used for this purpose.

(i) Traditional ML based

(ii) Deep ML based

Some of the widely used features in traditional machine learning based techniques are (SIFT)[1], (LBP)[2], and Histogram of Oriented Gradients[3].

Characteristics are taken from the input data using these techniques. They are passed to the Support Vector Machine, to classify something.

The main drawbacks of such techniques are that these are task specific, and real life data like images and video may be complicated, and can have multiple semantic elucidations.

Extraction of features from huge sets of data is not viable with task-specific techniques like these.

On the other hand, in techniques based on deep learning, characteristics are learned from the training data and large sets of media are handled. For instance, 'Convolutional Neural Networks' (CNN)[4] are ubiquitously made use of for feature learning, and the Softmax classifier is used for classifying them. 'Recurrent Neural Networks'(RNN) usually follows CNN to create the relevant captions.

II. State of the art (LITERATURE SURVEY)

The objective of image captioning is to generate correct, natural looking descriptions for a given image.

There are a multitude of reasons why Image Captioning is relevant. For instance, automatic image indexing is one of the uses. Image indexing can be made use of in many fields, including biomedical, military, commerce purposes, digital libraries, surfing the web and education. Descriptions are generated from images by some social media platforms to guess the location, items in the image which look similar to something, and the like.

In deep ML based methods, training data is used from where the system learns the features, and they manage a large-scale varying set of pictures. For instance, CNNs are quite commonly used for feature based learning.

The table shown below depicts the datasets that were used by people, with their evaluation methods. After analyzing the works of multiple people, we can conclude that Flickr8k, Flickr30k and MSCOCO are the most prevalent.

There are two main methods of generating captions from images, one being a form of supervised learning where the output is usually one of the sentences from the training set. The other is an unsupervised learning method called multimodal space based, which contains an encoder, a vision part, a multimodal space part, and a language decoder part. The vision part uses a deep CNN to extract the characteristics of the image. The multimodal space maps the word features and the image features together.

Table 2.1 : Datatsets and Evaluation Metrics used

Reference	Dataset used	Evaluation
Jia et al. (2015)	Flickr8k/30k, MSCOCO	BLEU, Meteor, CIDEr
Mao et al. (2014)	Flickr8k/30k, IAPR TC-12, MSCOCO	BLEU, R@ K, mrank
Chen et al. (2017)	Flickr8k/30k, MSCOCO, PASCAL	CIDEr, BLEU, ROUGE, METEOR
Wu et al. (2018)	Flickr8k/30k, MSCOCO	BLEU, CIDEr, METEOR
Wang et al. (2016)	Flickr8k	BLEU, PPL, METEOR

Supervised Learning compared to Other Deep Learning

Networks have been utilized in the classification of photos[5,6,7], object detection[8,9,10], and attribute learning[11] with good success rates for many years, using supervised learning. Researchers have expressed good interest in using them for automatic image captioning[12,13,14] for this reason.

A large number of methods have been identified for this type of learning.

Some different categories are:

(a) Dense image captioning,

(b) Semantic concept-based,

(c) Stylized Captions,

(d) Compositional Architecture,

(e) Attention Based,

(f) Encoder-Decoder Architecture, and

(g) Novel object-based.

Others

Different image encoders are used by existing captioning methods that use deep learning to extract image characteristics. Characteristics are then added to language decoders to create the captions that are required. There are two major drawbacks of these methods:

(I) Their training is done using approaches of maximum likelihood estimation as well as back propagation[15]. In this method, given the image and all the previously generated ground-truth words, the next word is predicted. hence, the captions that are created appear similar to the ground-truth captions. This is known as the exposure bias[16] problem.

(ii) Test time evaluation metrics aren't differentiable. Ideally, the sequence models for image captioning must be trained to elude the exposure bias and directly optimize the test time metrics. The critic is utilized to estimate the expected reward to train the actor in an actor-critic-based reinforcement learning algorithm (captioning policy network). Image captioning methods based on reinforcement learning take the next token from the model depending on the incentives they obtain in each state. In this type of learning, policy gradient methods can maximize the gradient to predict the cumulative reward in the long term. Hence, it solves the non-differentiable evaluation metrics problem.

Zhang et al.[17] proposed the image captioning method based on the actor-critic reinforcement learning. Currently existing evaluation metrics have problems that can be directly optimized using this method. This architecture has something called a policy network (which is the actor) and a value network (which is the critic). The actor predicts the following token of sequence by handling the job like a sequential decision problem. In each state, the network will obtain some reward depending on the task(in this case, it is the evaluation metrics score). The critic has to predict the previously mentioned reward. If it predicts the required reward, the actor then continues to show the results, which is related to the probability distribution.

The methods in this category follow the steps below:

(1) captions are generated by combined network based on CNN and RNN

(2) A different network based on CNN-RNN is used to evaluate the captions and send the feedback to the previous network to generate captions with improvement in quality.

Compositional Architecture based Image Captioning

This method consists of numerous independent constituents that are functional: First, a Convolutional Neural Network is used to get the semantic concepts from a picture. Then a language model is applied for creating some possible descriptions. On generating the final description, they are ranked again by making use of a deep multimodal similarity model.

Usually, the method of this category is shown in the steps below:

(a) Features of the image are taken using a Convolutional Neural Network.

(b) Visual concepts such as attributes, are then taken from the visual features.

(c) Many captions are generated by a language model using the information of the previous steps.

(d) The generated descriptions are then ranked again by utilizing a deep multi-modal similarity model to select the satisfactory captions.

Existing methods have shortcomings in generating a group of different, diverse descriptions as they have to foresee the upcoming word on a previously defined word by word format.

However, an amalgamation of attributes, subjects as well as their relationship in sentences, regardless of the location, can give rise to a wide range of descriptions.

Wang et al.[18] gave a method that finds the objects and their interactions as the initial step, and then recognizes and takes the required attributes to create captions. The main objective of this is to break the ground truth captions down into two parts, which are “skeleton sentence” and “attribute phrases”.

RNNs have been ubiquitously utilized in many sequence learning tasks. On the other hand, traditional RNNs seem to have issues like vanishing and exploding gradients and they cannot handle the temporal dependencies in the long term satisfactorily.

LSTM[19] networks are examples of Recurrent Neural Network that has some special units as well, along with the standard units. LSTM units utilize a memory cell that stores all the information in the memory for quite a while. sequence to sequence learning tasks use models based on LSTM recently. A different network called the Gated Recurrent Unit (GRU)[20] has a somewhat similar structure to LSTM, but it doesn’t utilize separate memory cells and it happens to make use of a lower number of gates to control the information flow. LSTMs happen to not detect the inherent hierarchical structure of sentences. They need large amounts of storage because of the dependencies through the memory cell. On the other hand, CNNs can learn the structure of the sentences and hence they are faster in processing when compared to LSTMs. Hence, recently, other tasks such as conditional image generation[21] make use of convolutional architectures.

Multiple datasets are used ubiquitously for Image Captioning, examples being MS-COCO, Flickr8K, and Flickr30K. For instance, we have used the Flickr8K Dataset, from which the features have been extracted from the images. We can compute the features of the image using the previously trained model and save them to file beforehand. Next, we can load these features and input them into the model as the interpretation of a given picture in the dataset. It is similar to running the photo through the full InceptionV3 model, but we will have done it once beforehand. The dataset has been trained in advance and we are using the said information for generating captions. This method is called “Transfer Learning”.

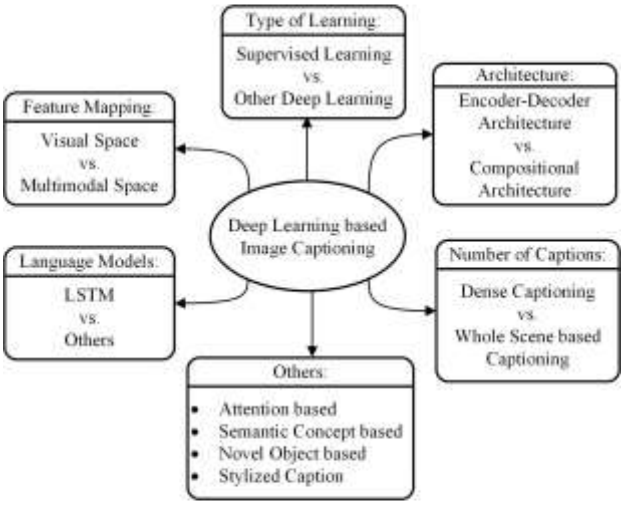


Figure 2.1: Components of Deep Learning based Image Captioning

III. PROPOSED WORK

Training with a lot of images takes a very long time and is a tedious process. This is why we use transfer learning, which is a research topic in machine learning (ML) that focuses on storing the knowledge obtained while solving a certain problem and applying it to a different problem which may be somewhat related. Here, InceptionV3 is used. This model was originally trained on some very powerful machines. The dataset used for InceptionV3 is ImageNet, which consists of more than a million images. This model has attained more than 78.1% accuracy on this dataset.

Imagenet is a database of pictures which is organized based on the WordNet hierarchy, in which each node consists of more than thousands of images.

We also use GloVe(which is the abbreviation of Global Vectors for Word Representation), an unsupervised learning algorithm whch is made use of to get representations of the words in a vector form.

This model is trained on the filled entries of a global word-word co-occurrence matrix, which makes a table showing the frequency of the co-occurrence of the words between themselves, in a corpus. Filling this matrix needs one pass through the complete corpus to get the data. For bigger corpora, the pass may be extremely expensive computationally, but it is a single time up-front cost. The following iterations are considerably quicker as the count of filled entries is usually a lot smaller than the actual amount of words in the given corpus. "Word embeddings" are a family of NLP (Natural Language Processing) techniques which aim at mapping semantic meanings into geometric spaces. This is done by linking a numerical vector to every word in a dictionary, in a way that the cosine distance between any two vectors would capture a portion of the semantic relationship between the words. The geometric space made by the said vectors is called “embedding space”.

For example, "coconut" and "polar bear" are two words that are very different semantically, so a good embedding space would represent them as vectors that would be quite far apart. But "kitchen" and "dinner" are words that have a relation, so they must be embedded close by.

Ideally, in a satisfactory embedding space, the "path" to go from the words "kitchen" and "dinner" would capture correctly the semantic relationship between these concepts. Here, the relationship is "where x occurs", so you would expect the path/vector “kitchen - dinner” (difference between the two embedding vectors, that is, path to go from dinner to kitchen) to capture the "where x occurs" relationship. Basically, we should be having the vectorial identity: “dinner + (where x occurs)” = “kitchen” (approximately). If that’s what happens, we can make use of a relationship vector like this to answer some questions. For instance, a new vector, "work", applied to this relationship vector, should give us something meaningful, like “work + (where x occurs)” = “office”, answering the question "where does work occur?".

Word embeddings are computed by implementing techniques like dimensionality reduction to the datasets of co-occurence statistics between words. This can be done using neural networks (the "word2vec" method), or by the matrix factorization technique.

IV. IMPLEMENTATION

Dataset

Received: 22 Feb 2020 | Revised: 13 Mar 2020 | Accepted: 05 Apr 2020

The dataset used here is Flickr8k that contains 8092 images with multiple captions for every image, that will cover enough features in the image.

Given below is a sample image with its five captions.

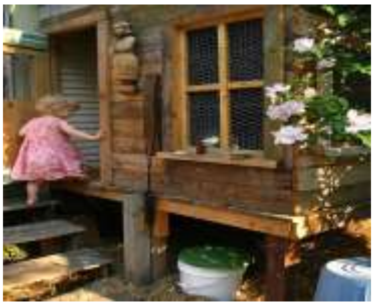


figure 4.1: A sample image with its five captions.

A child is climbing up a flight of stairs in an entryway.

A girl entering a brown building.

A little girl climbing into a wooden house.

A little girl climbing up the stairs to her wooden house.

A child in pink dress getting into a wooden cabin.

Features of objects in the image like girl, dress, cabin are there in the given captions, as well as the color like pink. All the captions are stored in a list. The dataset contains other words too that do not add much to the meaning such as ‘a’, ‘an’, that are eliminated.

The dataset is passed through the image-language encoder that will encode the image and its captions so that they are understandable to the LSTM neural network that is used.

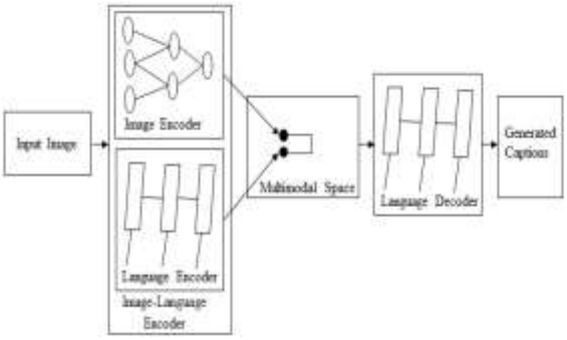


Figure 4.2: Representation of Captioning based on Multimodal Space

Received: 22 Feb 2020 | Revised: 13 Mar 2020 | Accepted: 05 Apr 2020

Image Encoder

For image encoding, InceptionV3 neural network is used, that is trained in ImageNet dataset that carries upwards of 14 million images and is extensively used in software research for visual recognition of objects.

The model is made up of many building blocks, including convolutions, average pooling, max pooling, dropouts, and fully connected layers. Batch normalization is used extensively throughout and applied to activation inputs. Loss is calculated by Softmax.

We remove the last layer from the InceptionV3 network and add an output layer of 2048, that means the network will classify image into 2048 objects. Each image is converted into a 2048 x 1 vector, that contains probabilities of the image belonging to one of the 2048 classes

We resize the image, which is actually in pixel format and provide it to the changed IncepionV3 network. This technique of using already built network is called transfer learning, which is a very popular area in deep learning and natural language processing.

InceptionV3 neural network extracts all the features within the images, for example cat, dog, grass, field, beach, road, shirt as well as colors such as black, red etc.

Language encoder

Glove word embeddings are used for language encoding purpose. It is a predefined vectors for each words that we have in our dictionary. Word embeddings are essential in the field of NLP, as it is necessary to make our deep learning model to understand the meaning of the sentences that we have.

The model must understand the correlation between different word so that it will be able to predict sensible and realistic captions as output. For example,

"A man swims in a ?????"

If our model has generated the above sentence, and is looking for the last word, it is clear that the last word cannot be ground, road or field. It will have something to with a water body such as lake, river, sea, or a pool. This a human can understand using his intelligence, but the deep learning model l can understand by using word embeddings, which are nothing but numerical vectors, that are correlated in the same manner the meanings of the words are correlated to each other.

The embedding matrix is created that contains embeddings for each and every word that are used in the vocabulary.

LSTM Model

The LSTM neural network contains the 2048 x 1 sized vector of image and embedding matrix.

The image and one caption is being provided for computation at a time. It generates captions according to that, similarly for another caption and so on till all five captions are used.

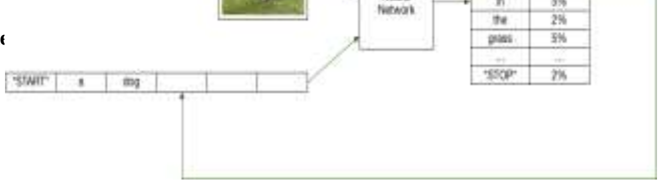


Figure 4.3 : Working of LSTM model

First, image and STARTSEQ keyword is in given as input to the LSTM model. It then predicts a keyword, say ‘a’. Then next time, ‘STARTSEQ a ‘ is given ass input along with image. And say the output is ‘dog’.

The third time the input is ‘STARTSEQ a dog’ and image.

This process keeps on going as long as STOPSEQ is not generated. Once it is generated the model stops generating anymore captions.

Data Generator

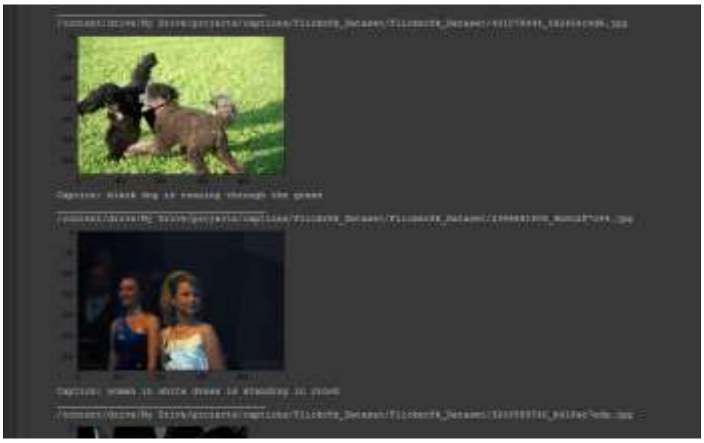
Data generator iterates through each image and for each image it takes captions and provides it to the LSTM model after creating a padding sequence for each. After providing caption to the model it yields or waits for some time until LSTM finishes training and then data generator provides another caption or iterates through next image.

Caption Generation

The image is provided to the LSTM model after training.

The model will then give the caption as output. That caption will contain STARTSEQ and STOPSEQ that needs to be removed manually.

V. RESULTS DISCUSSION

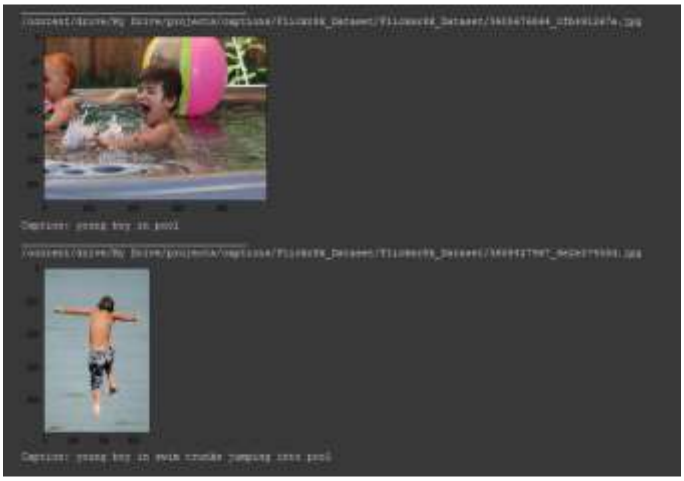


```
/content/drive/My Drive/[illegible]/Flickr8k_Dataset/Flicker8k_Dataset/[illegible].jpg
Caption: black dog is running through the grass
/content/drive/My Drive/[illegible]/Flickr8k_Dataset/Flicker8k_Dataset/[illegible].jpg
Caption: woman in white dress is standing in crowd
/content/drive/My Drive/[illegible]/Flickr8k_Dataset/Flicker8k_Dataset/[illegible].jpg
```

Figure 5.1: A screenshot of the results (1)

As we can see in the first image, the dog and the grass were detected and a sentence has been generated. Although varied results have been obtained for different inputs, they were mostly satisfactory.

In the second image, a woman, her white dress, and the presence of people in the background has been detected. Therefore, the caption, although not entirely true, has some elements shown correctly.



```
/content/drive/My Drive/projects/captions/Flickr8k_Dataset/Flickr8k_Dataset/[illegible].jpg
Caption: young boy in pool
/content/drive/My Drive/projects/captions/Flickr8k_Dataset/Flickr8k_Dataset/[illegible].jpg
Caption: young boy in swim trunks jumping into pool
```

Figure 5.2 : A screenshot of the results(2)

Here, the program has detected a boy and a body of water in his vicinity. Hence, the caption "young boy in a pool" has been generated.

Similarly, in the second image, the action of jumping into a pool has been detected, along with the boy wearing swim trunks. We can observe that the generated caption is apt.

VI. CONCLUSION

In this paper, we have discussed methods to create captions for images based on deep learning. We have given a classification of techniques used for captioning the images. We have also discussed different metrics for evaluation and data sets with the pros and cons. A short synopsis of the outcome has also been given. We have shown possible research prospects in this particular field as well. Even though captioning methods based on deep learning have seen a laudable progress in the past few years, a completely reliable image captioning method that is known to give good quality descriptions for almost every image has not yet been achieved. Now that there is a rise of newer deep learning architectures, automated captioning of images may remain a fairly active research topic for a while.

Methods that are based from Encoder-Decoder architecture use a basic Convolutional Neural Network and a text generator to create image captions. Attention based image caption methods observe different major sections of the image and obtain more satisfactory results when compared to methods that are based on encoder-decoder architecture. Dense image caption methods can create region related captions for pictures. Stylized image captions can show the different emotions involved.

REFERENCES

1. David G Lowe. 2004. Distinctive image features from scale-invariant keypoints. International journal of computer vision 60, 2 (2004), 91–110.
2. Timo Ojala, Matti PietikAďinen, and Topi MAďenpAďAď. 2000. Gray scale and rotation invariant texture classification with local binary patterns. In European Conference on Computer Vision. Springer, 404–420.
3. Navneet Dalal and Bill Triggs. 2005. Histograms of oriented gradients for human detection. In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, Vol. 1. IEEE, 886–893.
4. Yann LeCun, LAľon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. Proc. IEEE 86, 11 (1998), 2278–2324.
5. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition. 770–778.
6. Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In International Conference on Learning Representations (ICLR).
7. Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems. 1097-1105.
8. Ross Girshick. 2015. Fast r-cnn. In Proceedings of the IEEE international conference on computer vision. 1440–1448.
9. Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition. 580–587.
10. Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems. 91–99.
11. Chuang Gan, Tianbao Yang, and Boqing Gong. 2016. Learning attributes equals multi-source domain generalization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 87–97.
12. Xinlei Chen and C Lawrence Zitnick. 2015. Mind’s eye: A recurrent visual representation for image caption generation. In Proceedings of the IEEE conference on computer vision and pattern recognition. 2422–2431.
13. Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 3128–3137.
14. Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In Proceedings of the IEEE conference on computer vision and pattern recognition. 3156–3164.
15. Marc’Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2016. Sequence level training with recurrent neural networks. In International Conference on learning Representations (ICLR).
16. Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction

17. with recurrent neural networks. In Advances in Neural Information Processing Systems. 1171–1179.
18. Li Zhang, Flood Sung, Feng Liu, Tao Xiang, Shaogang Gong, Yongxin Yang, and Timothy M Hospedales. 2017. Actor-critic sequence training for image captioning. arXiv preprint arXiv:1706.09601.
19. Yufei Wang, Zhe Lin, Xiaohui Shen, Scott Cohen, and Garrison W Cottrell. 2017. Skeleton Key: Image Captioning by Skeleton-Attribute Decomposition. In Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). 7378–7387.
20. Sepp Hochreiter and JAijrgen Schmidhuber. 1997. Long short-term memory. Neural computation 9, 8 (1997), 1735–1780.
21. Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empiricalevaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555.
22. Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. 2016. Conditional image generation with pixelcnn decoders. In Advances in Neural Information Processing Systems. 4790–4798.
23. Chris Callison-Burch, Miles Osborne, and Philipp Koehn. 2006. Re-evaluation the Role of Bleu in Machine Translation Research.. In EACL, Vol. 6. 249–256.
24. Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In Text summarization branches out: Proceedings of the ACL-04 workshop, Vol. 8. Barcelona, Spain.
25. Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization, Vol. 29. 65–72.