Image Caption Generation Methodologies

Omkar Shinde[1], Rishikesh Gawde[2], Anurag Paradkar[3]

[1-3]*Student, Department of Information Technology, Vidyalankar Institute of Technology, Mumbai, India*

Abstract - *Scene understanding has always been an important task in computer vision, and image captioning is one of the major areas of Artificial intelligence research since it aims to mimic the human ability to compress an enormous amount of visual information in a few sentences. Image caption generation aims to generate a sentence description for an image. The task aims to provide short but detailed caption of the image and requires the use of techniques from computer vision and natural language processing. Recent developments in deep learning and the availability of image caption datasets such as Flickr and COCO have enabled significant research in the area. In this paper, we propose methodologies used such as multilayer Convolutional Neural Network (CNN), Recurrent Neural Network (RNN) and a Long Short Term Memory (LSTM) to accurately identify and construct meaningful caption for a given image.*

***Key Words*: Image Captioning, Computer Vision, Convolutional Neural Network, Recurrent Neural Network, Long Short term memory**

1. INTRODUCTION

Problem Statement:

Most of the people spend about hours deciding about what to write as a caption. A picture is incomplete without a good caption to go with it. The problem introduces a captioning task, which requires a computer vision system to both localize and describe salient regions in images in natural language. The image captioning task generalizes object detection when the descriptions consist of a single word. Given a set of images and prior knowledge about the content find the correct semantic label for the entire image(s).

Artificial Intelligence (AI) is now at the heart of innovation economy and thus the base for this project is also the same. In the recent past a field of AI namely Deep Learning has turned a lot of heads due to its impressive results in terms of accuracy when compared to the already existing Machine learning algorithms. The task of being able to generate a meaningful sentence from an image is a difficult task but can have great impact, for instance helping the visually impaired to have a better understanding of images. With the great advancement in computing power and with the availability of huge datasets, building models that can generate captions for an image has become possible.

On the other hand, humans are able to easily describe the environments they are in. Given a picture, it's natural for a person to explain an immense amount of details about this image with a fast glance. Although great development has been made in computer vision, tasks such as recognizing an object, action classification, image classification, attribute classification and scene recognition are possible but it is a relatively new task to let a computer describe an image that is forwarded to it in the form of a human-like sentence.

2. LITERATURE REVIEW

One of the influential papers by Andrej Karpathy et al. in image captioning divides the task into two steps: mapping sentence snippets to visual regions in the image and then using these correspondences to generate new descriptions (Karpathy and Fei-Fei 2015). The authors use a Region Convolutional Neural Network (RCNN) to represent images as a set of h dimensional vectors each representing an object in the image, detected based on 200 ImageNet classes. The authors represent sentences with the help of a Bidirectional Recurrent Neural Network (BRNN) in the same h dimensional space. Each sentence is a set of h dimensional vectors, representing snippets or words. The use of the BRNN enriches this representation as it learns knowledge about the context of each word in a sentence. The authors find that with such a representation, the final representation of words aligns strongly with the representation of visual regions related to the same concept. They define an alignment score on this representation of words and visual regions and align various words to the same region generating text snippets, with the help of a Markov Random Field. With the help of these correspondences between image regions and text snippets, the authors train another model that generates text descriptions for new unseen images (Karpathy and Fei-Fei 2015).

The authors train an RNN that takes text snippets and visual regions as inputs and tries to predict the next word in the text based on the words it has seen so far. The image region information is passed to the network as the initial hidden state at the initial time step, and the network learns to predict the log probability of the next most likely word using a softmax classifier. The authors use unique START and END tokens that represent the beginning and end of the sentence, which allows the network to make variable length predictions. The RNN has 512 nodes in the hidden layer (Karpathy and Fei-Fei 2015).

The network for learning correspondences between visual regions and text words was trained using stochastic gradient descent in batches of 100 image-sentence pairs. The authors used dropouts on every layer except the recurrent layers and clipped the element-wise gradients at 5 to prevent gradient explosion. The RNN to generate descriptions for unseen images was trained using RMSprop which dynamically adjusts the learning rate (Karpathy and Fei-Fei 2015).

Kelvin Xu et.al (Xu et al. 2015) use the concept of attention to better describe images. The authors propose models that focus on which area of the image, and what objects in the image are being given attention and evaluate these models on different image captioning datasets. The idea be- hind the approach is that much like the human visual system, some parts of the image may be ignored for the task of image description, and only the salient foreground features are considered. The authors use a CNN to learn important features of the image and an LSTM (Long short-term memory network) to generate description text based on a context vector.

Jyoti Aneja et al. in (Aneja, Deshpande, and Schwing 2017) use a convolutional approach to generate description text instead of a simple RNN, and show that their model works at par with RNN and LSTM based approaches.

Andrew Shin et al. (Shin, Ushiku, and Harada 2016) use a second neural network, finely tuned on text-based sentiment analysis to generate image descriptions which capture the sentiments in the image. The authors use multi-label learning to learn sentiments associated with each of the images, and then use these sentiments, along with the input from the CNN itself as inputs to an LSTM to generate sentences which include the sentiment. The LSTM is restricted so that each description contains at least one term from the sentiment vocabulary.

Alexander Mathews et al. (Mathews, Xie, and He 2016) emphasize how only a few image descriptions in most datasets contain words describing sentiments, and most descriptions are factual. The authors propose a model that consists of two CNN + RNN models each with a specific task. While one model learns to describe factual content in the image, the other learns to describe the sentimental associated, thus providing a framework that learns to generate sentiment based descriptions even with lesser image sentiment data.

Quanzeng You et.al in (You, Jin, and Luo 2018) propose approaches to inject sentiment into the descriptions generated by image captioning methods.

Tsung Yi Lin et.al in (Lin et al. 2014) describes the Microsoft Common Objects in Context dataset that is widely used for benchmarking image captioning models.

3. METHODOLOGIES

3.1 Model Overview:

The model proposed takes an image I as input and is trained to maximize the probability of p(S|I) where S is the sequence of words generated from the model and each word St Is generated from a dictionary built from the training dataset. The input image I is fed into a deep vision Convolutional Neural Network (CNN) which helps in detecting the objects present in the image. The image encodings are passed on to the Language Generating Recurrent Neural Network (RNN) which helps in generating a meaningful sentence for the image as shown in the fig. 13. An analogy to the model can be given with a language translation RNN model where we try to maximize the p (T|S) where T is the translation to the sentence S. However, in our model the encoder RNN which helps in transforming an input sentence to a fixed length vector is replaced by a CNN encoder. Recent research has shown that the CNN can easily transform an input image to a vector.

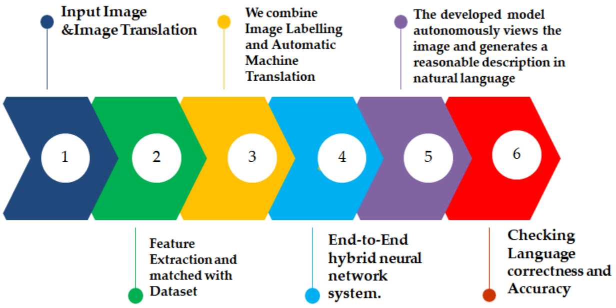
Input Image &Image Translation
We combine Image Labelling and Automatic Machine Translation
The developed model autonomously views the image and generates a reasonable description in natural language
1
2
3
4
5
6
Feature Extraction and matched with Dataset
End-to-End hybrid neural network system.
Checking Language correctness and Accuracy

Fig-1: Flow of Model

For the task of image classification, we use a pretrained model VGG16. The details of the models are discussed in the following section. A Long Short-Term Memory (LSTM) network follows the pretrained VGG16. The LSTM network is used for language generation. LSTM differs from traditional Neural Networks as a current token is dependent on the previous tokens for a sentence to be meaningful and LSTM networks take this factor into account.

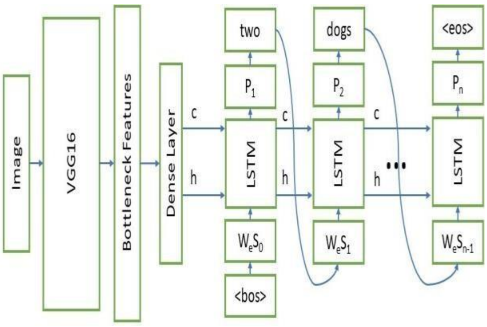

Fig- 2: Training the model using VGG16 image features

The model that was can before the project consists of two different input streams, one for the image features, and the other for the preprocessed input captions. The image features are passed through a fully connected (dense) layer to get a representation in a different dimension. The input captions are passed through an embedding layer. These two input streams are then merges and passed as inputs to an LSTM layer. The image is passed as the initial state to the LSTM while the caption embedding's passed as the input to the LSTM. The architecture is shown in figure

3.2 Recurrent Neural Network:

Recurrent Neural Network is a generalization of feed forward neural network that has an internal memory. RNN is recurrent in nature as it performs the same function for every input of data while the output of the current input depends on the past one computation. After producing the output, it is copied and sent back into the recurrent network. For making a decision, it considers the current input and the output that it has learned from the previous input.

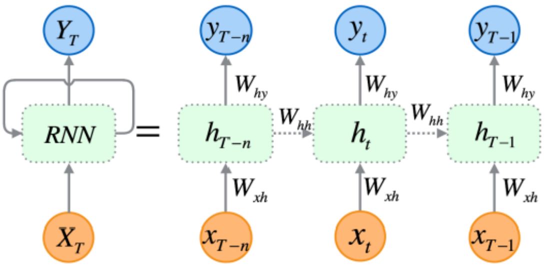

Fig-3: RNN

A recurrent neural network (RNN) is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. This allows it to exhibit temporal dynamic behavior. Derived from feed forward neural networks, RNNs can use their internal state (memory) to process variable length sequences of inputs. This makes them applicable to tasks such as unsegmented, connected handwriting recognition or speech recognition.

The term "recurrent neural network" is used indiscriminately to refer to two broad classes of networks with a similar general structure, where one is finite impulse and the other is infinite impulse. Both classes of networks exhibit temporal dynamic behavior. A finite impulse recurrent network is a directed acyclic graph that can be unrolled and replaced with a strictly feed forward neural network, while an infinite impulse recurrent network is a directed cyclic graph that cannot be unrolled.

Both finite impulse and infinite impulse recurrent networks can have additional stored states, and the storage can be under direct control by the neural network. The storage can also be replaced by another network or graph, if that incorporates time delays or has feedback loops. Such controlled states are referred to as gated state or gated memory, and are part of long short-term memory networks (LSTMs) and gated recurrent units. This is also called Feedback Neural Network (FNN).

RNN have a "memory" which remembers all information about what has been calculated. It uses the same parameters for each input as it performs the same task on all the inputs or hidden layers to produce the output. This reduces the complexity of parameters, unlike other neural networks.

The input to the hidden layer in an RNN (with a single hidden layer) is the input vector, along with the output of the hidden layer for the previous time step. The RNNs are trained to learn to predict the next word given the current example. This is done for multiple times in each iteration, which represents the length of the sequence that the RNN can learn and predict later. The Network is trained using back propagation through time, which adjusts the weights between the hidden layer for a given time step and the next time step. Once trained for various iterations, the RNN can learn to model the sequence (contributors 2018c).

There are problems with RNNs, when learning long sequences, in situations such as the language translation of large documents, where it may be necessary to remember only the context over a small time period. For this purpose, Long Short Term Memory (LSTM) networks are used, where each cell has three gates - input, forget and output and can learn when to forget the previous context, along with other parameters. The LSTM is trained such that each LSTM cell updates its weights at each time step and the all the weights are updated after each iteration. This helps the network learn long sequences and decide which parts of the sequences are related with some context (Trask 2015) (contributors 2018b).

3.3 Convolutional Neural Network:

In deep learning, a convolutional neural network (CNN, or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery. They are also known as shift invariant or space invariant artificial neural networks (SIANN), based on the shared-weight architecture of the convolution kernels that scan the hidden layers and translation invariance characteristics. They have applications in image and video recognition, recommender systems, image classification, Image segmentation, medical image analysis, natural language processing, brain-computer interfaces, and financial time series.

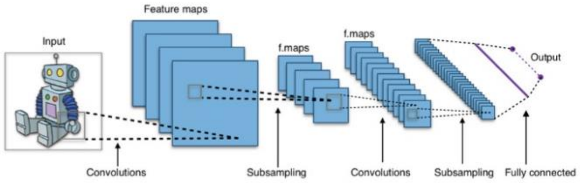

Fig-4: CNN

CNNs are regularized versions of multilayer perceptrons. Multilayer perceptrons usually mean fully connected networks, that is, each neuron in one layer is connected to all neurons in the next layer. The "fully-connectedness" of these networks makes them prone to overfitting data. Typical ways of regularization include varying the weights as the loss function gets minimized while randomly trimming connectivity. CNNs take a different approach towards regularization: they take advantage of the hierarchical pattern in data and assemble patterns of increasing complexity using smaller and simpler patterns embossed in the filters. Therefore, on the scale of connectedness and complexity, CNNs are on the lower extreme. Convolutional networks were inspired by biological processes in that the connectivity pattern between neurons resembles the organization of the animal visual cortex. Individual cortical neurons respond to stimuli only in a restricted region of the visual field known as the receptive field. The receptive fields of different neurons partially overlap such that they cover the entire visual field.

CNNs use relatively little pre-processing compared to other image classification algorithms. This means that the network learns to optimize the filters or convolution kernels that in traditional algorithms are hand-engineered. This independence from prior knowledge and human intervention in feature extraction is a major advantage. Convolutional neural networks are composed of multiple layers of artificial neurons. Artificial neurons, a rough imitation of their biological counterparts, are mathematical functions that calculate the weighted sum of multiple inputs and outputs an activation value. The behavior of each neuron is defined by its weights. When fed with the pixel values, the artificial neurons of a CNN pick out various visual features.

When you input an image into a ConvNet, each of its layers generates several activation maps. Activation maps highlight the relevant features of the image. Each of the neurons takes a patch of pixels as input, multiplies their color values by its weights, sums them up, and runs them through the activation function. The first (or bottom) layer of the CNN usually detects basic features such as horizontal, vertical, and diagonal edges. The output of the first layer is fed as input of the next layer, which extracts more complex features, such as corners and combinations of edges. As you move deeper into the convolutional neural network, the layers start detecting higher-level features such as objects, faces, and more.

3.4 Long Short-Term Memory:

Long Short-Term Memory (LSTM) networks are a type of recurrent neural network capable of learning order dependence in sequence prediction problems. This is a behavior required in complex problem domains like machine translation, speech recognition, and more.

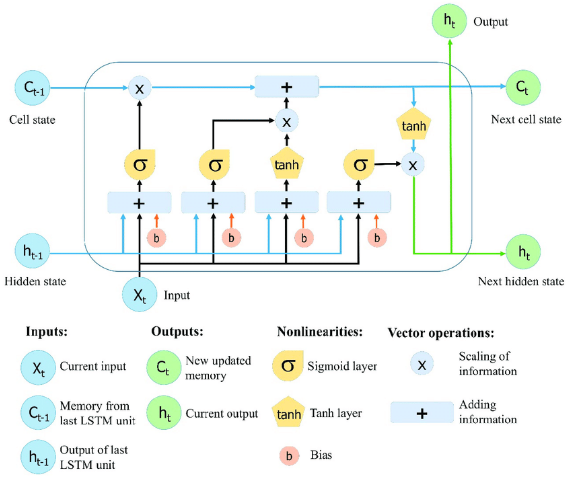

Fig-5: LSTM

LSTMs are a complex area of deep learning. It can be hard to get your hands around what LSTMs are, and how terms like bidirectional and sequence-to-sequence relate to the field. In this post, you will get insight into LSTMs using the words of research scientists that developed the methods and applied them to new and important problems. There are few that are better at clearly and precisely articulating both the promise of LSTMs and how they work than the experts that developed them. We will explore key questions in the field of LSTMs using quotes from the experts, and if you're interested, you will be able to dive into the original papers from which the quotes were taken. Unlike standard feed forward neural networks, LSTM has feedback connections. It can not only process single data points (such as images), but also entire sequences of data (such as speech or video). For example, LSTM is applicable to tasks such as unsegmented, connected handwriting recognition, speech recognition and anomaly detection in network traffic or IDSs (intrusion detection systems).

A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell. LSTM networks are well-suited to classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important events in a time series. LSTMs were developed to deal with the vanishing gradient problem that can be encountered when training traditional RNNs. Relative insensitivity to gap length is an advantage of LSTM over RNNs, hidden Markov models and other sequence learning methods in numerous applications.

3.5 VGG16:

It is considered to be one of the excellent vision model architecture till date. Most unique thing about VGG16 is that instead of having a large number of hyper parameters they focused on having convolution layers of 3x3 filters with a stride 1 and always used same padding and maxpool layer of 2x2 filter of stride 2. It follows this arrangement of convolution and max pool layers consistently throughout the whole architecture. In the end it has 2 FC (fully connected layers) followed by a softmax for output. The 16 in VGG16 refers to it has 16 layers that have weights. This network is a pretty large network and it has about 138 million (approx.) parameters.

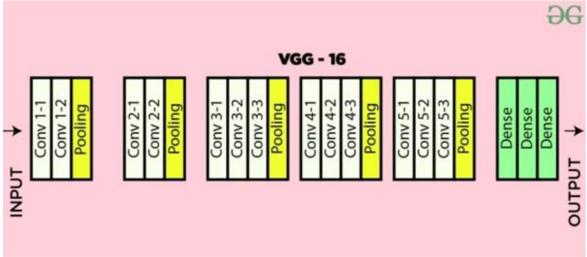

Fig-6: VGG16

VGG16 is a convolutional neural network model proposed by K. Simonyan and A. Zisserman from the University of Oxford in the paper "Very Deep Convolutional Networks for Large-Scale Image Recognition". The model achieves 92.7% top-5 test accuracy in ImageNet, which is a dataset of over 14 million images belonging to 1000 classes. It was one of the famous model submitted to ILSVRC-2014. It makes the improvement over AlexNet by replacing large kernel-sized filters (11 and 5 in the first and second convolutional layer, respectively) with multiple 3×3 kernel-sized filters one after another. VGG16 was trained for weeks and was using NVIDIA Titan Black GPU's.

VGG-16 is a convolutional neural network that is 16 layers deep. You can load a pretrained version of the network trained on more than a million images from the ImageNet database [1]. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images. The network has an image input size of 224-by-224. This network is characterized by its simplicity, using only 3×3 convolutional layers stacked on top of each other in increasing depth. Reducing volume size is handled by max pooling. Two fully-connected layers, each with 4,096 nodes are then followed by a softmax classifier.

4. CONCLUSIONS

In this paper we presented the deep learning techniques used for image captioning problem. We have presented methodologies such as Convolutional Neural Network, Convolutional Neural Network, VGG16, Long short-term memory models.

The image caption generator has the capabilities to generate captions for the images, provided during the Training purpose & also for the new images as well. The model takes an image as an input and by analyzing the image it detects objects present in an image and can create a suitable caption for it.

ACKNOWLEDGEMENT

We thank Professor Yash Shah for giving us the opportunity to work on the project and for providing the necessary resources.

REFERENCES

[1] Aarthi, S., and Chitrakala, S. 2017. Scene understandinga survey. In Computer, Com- munication and Signal Processing (ICCCSP), 2017 Interna- tional Conference on, 1–4. IEEE.
[2] Amazon Web Services. 2018. Amazon ec2 p2 instances.
[3] Aneja, J.; Desh- pande, A.; and Schwing, A. 2017. Convolutional image captioning. arXiv preprint arXiv:1711.09151.
[4] contributors, W. 2018a. Convolutional neural network — wikipedia, the free encyclopedia. [On- line; accessed 30-March-2018].
[5] contributors, W. 2018b. Long short- term memory — wikipedia, the free encyclopedia. [Online; accessed 30-March-2018].
[6] contributors, W. 2018c. Recurrent neu- ral network — wikipedia, the free encyclopedia. [Online; accessed 30-March-2018].
[7] He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In Proceed- ings of the IEEE conference on computer vision and pattern recognition, 770–778.
[8] Jason Brownlee. 2017. A gentle in- troduction to calculating the bleu score for text in python.
[9] Karpathy, A., and Fei-Fei, L. 2015. Deep visual-semantic alignments for generating image descriptions. In Proceedings of the IEEE conference on computer vision and pattern recognition, 3128–3137.
[10] Karpathy, A. 2015. The unreasonable ef- fectiveness of recurrent neural networks. Andrej Karpathy's Blog.
[11] Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dolla´r, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In European conference on computer vision, 740–755. Springer.
[12] Mathews, A. P.; Xie, L.; and He, X. 2016. Senticap: Generating image descriptions with sentiments. In AAAI, 3574–3580.
[13] Rashtchian, C.; Young, P.; Hodosh, M.; and Hockenmaier, J. 2010. Collecting image annotations using amazon's mechanical turk. In Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, 139–147. Association for Computational Linguistics.
[14] Rosebrock, A. 2017. Imagenet: Vggnet, resnet, inception, and xception with keras. pyimagesearch website.
[15] Shin, A.; Ushiku, Y.; and Harada, T. 2016. Image captioning with sentiment terms via weakly-supervised sentiment dataset. In BMVC.
[16] Simonyan, K., and Zisser- man, A. 2014. Very deep convolutional networks for large- scale image recognition. arXiv preprint arXiv:1409.1556.
[17] Dr. Vinayak D. Shinde, Mahiman P. Dave, Anuj M. Singh, Amit C. Dubey; Image Caption Generator using Big Data and Machine Learning.
[18] Trask, A. 2015. Anyone can learn to code an lstm-rnn in python (part 1: Rnn). iamtrask github.io blog.
[19] Vinyals, O.; Toshev, A.; Bengio, S.;and Erhan, D. 2015. Show and tell: A neural image caption gen- erator. In Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on, 3156–3164. IEEE.
[20] Wikipedia contributors. 2018a. Bleu — Wikipedia, the free encyclopedia. [Online; accessed 30-April-2018].
[21] Wikipedia contributors. 2018b. Cross entropy — Wikipedia, the free encyclopedia. [Online; accessed 30-April-2018].
[22] Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; and Bengio, Y. 2015. Show, attend and tell: Neural image caption generation with visual attention. In International Conference on Machine Learning, 2048–2057.