Image captioning using Deep Neural Networks

Rahul Singh[a,1] **and Aayush Sharma**[a,1]

[a]Iowa State University

This manuscript was compiled on May 22, 2018

In this project, we develop a framework leveraging the capabilities of artificial neural networks to "caption an image based on its significant features". Recurrent Neural Network (RNN) are increasingly being used as encoding-decoding frameworks for machine translation. Our objective is to replace the encoder part of the RNN with a Convolutional Neural Network (CNN). Thus, transforming the images into relevant input data to feed into the decoder of the RNN. The image will be converted into a multi-feature dataset, characterizing its distinctive features. The analysis will be carried out on the popular *Flickr8K* dataset.

Computer Vision | Language Processing| Machine Learning |

Machine translation is advancing at an alarming rate due to the technical developments in the field of machine learning. (1–3) The rapid developments in the field of machine learning is affecting and improving many branches of the technical industry. The application of Artificial Intelligence and Neural Networks to complicated natural language processing challenges like speech recognition and machine translation is leading to remarkably rapid advancements.(4–7) Among of the many advancements, one such example is the success in the field of "Describing Images". The task to provide a description of an image is challenging. It requires to first understand the visual knowledge and translate the knowledge into sentences using natural language processing tools. This requires to develop a model that can capture the correlation present in the visual and natural language for the associated image. The problem is multimodal that creates the necessity to create a hybrid model that can utilize the multidimensionality of the problem. Traditionally, methods (8) like template based and retrieval based methods have been used to solve the problem. However, the major drawback of these methods is the results don't translate to new images and hence, these methods fail in generalization. These methods focus on labeling images with a fixed set of visual categories and hence, fail in describing new images. However, a human can provide various descriptions for the same image and this just describes the restrictive nature of these methods. Therefore the requirement to create a method that can be generalized to create description of new images.
With the advent of machine learning and especially in the field of deep neural networks (DNN), the computer vision and language processing has advanced exponentially in the last few years. The power of deep neural networks has also been tested successfully to create captions of images(9, 10), and their generalization capability is much better than the traditional methods. These models often employ different deep neural networks like convolutional neural network(CNN)(11), long short term memory(LSTM) networks(12), recurrent neural network(RNN)(13) to implicitly learn the common embedding by encoding and decoding the direct modalities. These methods gives an improved results compared to earlier methods on all common datasets of caption generation. In the last few years, a numbers of models have been developed. The common theme in these methods is a modified framework of "merge model" developed in (10) by combining CNN and LSTM.
The most popular and easily available datasets used to test new methods are:

- Flickr8k(14) - It has 8000 images and 5 captions for each image.
- Flickr30k(14) - This dataset has 31783 images with 5 full sentence level caption for each image.
- MSCOCO (15) - This dataset has 82783 images. Each image has 5 captions.

In the following sections, we give brief description of the theory behind the model and dataset used in this project. Finally we summarize the results and give conclusion.

Method

For this project, we utilize one of the first models that was proposed to provide an end to end training capability. This model uses a neural and probabilistic framework to generate descriptions from images. It generates the descriptions by maximizing the probability of the correct translation in an "end-to-end" fashion. With a sequence given:

$$\theta^* = argmax_\theta \sum_{(I,S)} logp(S|I;\theta)$$

where θ are the parameters of the model, I is an image, and S its correct transcription. Now, we can represent the sentence as a joint probability and suppose, $S_0, S_1....S_N$ is the sequence, and N is the length of:

$$logp(S|I) = \sum_{t=0}^{N} logp(S_t|I, S_0, ..., S_{t-1})$$

This can be easily modeled using Recurrent neural networks (RNN), using RNN to calculate the probability from the input image I and the $t-1$ words expressed as a sequence. This can be represented as

$$h_{t+1} = f_{LSTM}(h_t, x_t)$$

, where h_t is the memory and x_t is the image.
The functions f_{LSTM} is defined according to the following:

$$i_t = \sigma_o(W_i x_t + U_i h_{t-1})$$

$$f_t = \sigma_o(W_f x_t + U_f h_{t-1})$$

$$o_t = \sigma_o(W_o x_t + U_o h_{t-1})$$

$$c_t = \tanh_o(W_c x_t + U_c h_{t-1})$$

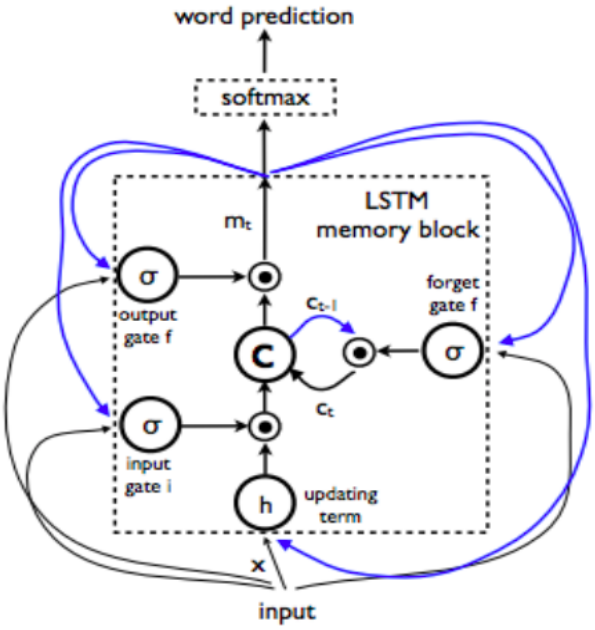

Fig. 1. Description of an LSTM network showing the details of different blocks.(10)

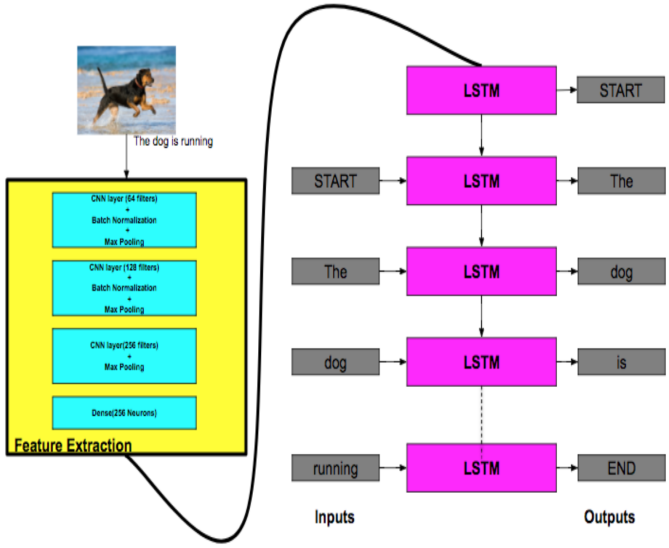

Fig. 2. Schematic diagram of the flow of the model.

$$m_t t = f_t \odot c_{t-1} + i_t c_t$$

$$h_t = o_t \odot tanh_o(m_t t) = f_{LSTM}(x_t, h_{t-1})$$

where W_i, W_f, W_o, $W_c \in R^{D_h x D_x}$, U_i, U_f, U_o, $U_c \in R^{D_h x D_h}$, $\odot$ represent element-wise multiplication.

In this model, we use Convolution neural networks (CNN) to extract the features of the images, represented as x_0 in the above equations, and RNN's are designed using the most common model known as LSTM (Long short-term memory). So, the model calculates the conditional probability of the next word in the caption given the image and the previous words. The algorithm for the model can be written as :

1.) The first step in the model is to extract features using CNN. This step can be done using the normal CNN steps such as Convolutions, MaxPooling and Batch Normalization. The image features is the first input in the model.

$$x_0 = I_{CNN} = \sigma(W_{CNN} I + b)$$

where W_{CNN} are weights of the convolutional network, I is the image (represented as 3-dimensional matrix) and b is the bias in the model.

2.) The subsequent inputs are the sequence of words generated from the sentence.

$$x_t = W_e s_t$$

where W_e are the embedding weights and s_t is the word vector.

3.) Calculate the probabilities of the words using the function defined above using the function, f_{LSTM} defined above,

$$h_t = f_{LSTM}(x_t, h_{t-1})$$

4.) Guess the final word in the sequence.

$$p_{t+1} = \sigma_o(W h_t + b)$$

where σ_o is the softmax function.

Evaluation Metrics. BLUE-N: Bilingual Evaluation Understudy

$$logB_N = min(0, 1 - r/c) + \sum_{n=1}^{N} w_N logp_n$$

where, B_N is BLEU-N metric, r is the effective corpus length, c is the length of the candidate translation and w_N are the weights.

This is an algorithm developed for evaluating the quality of text generated from machine translation. The output of this value ranges from 0 to 1. Higher values indicates the similarity between the generated text and reference text.

Results

This section contains the experimental results obtained from applying this algorithm on the Flickr8k dataset.

A. Dataset. We have used the *Flickr8K* dataset for both training, validation and evaluation. The dataset contains a total of 8000 images, 6000 for training, 1000 for validation and 1000 for testing. Figure 3 shows an example of the image with their captions. Each image has five captions. We pre-process the captions in the training set by removing all non-alphanumeric characters, making all words lower case, prepending each caption with a special "START" token and appending with an "END" token.

B. Models. A schematic representation of Model-2 is shown in Figure 2 and the summary is provided in Figure 7. The model is divided in the following categories. A link to the code Caption-generator

Encoder. We have tested two models for the dataset. The models differ in their image extraction capabilities.

Model-1 used features from a pre-trained model on ImageNet dataset (VGG16)(16). The VGG neural network is an image classification CNN and given an image, the VGG network will output the probabilities of the different classes that an image could potentially belong to.

Fig. 3. A sample image from the dataset.

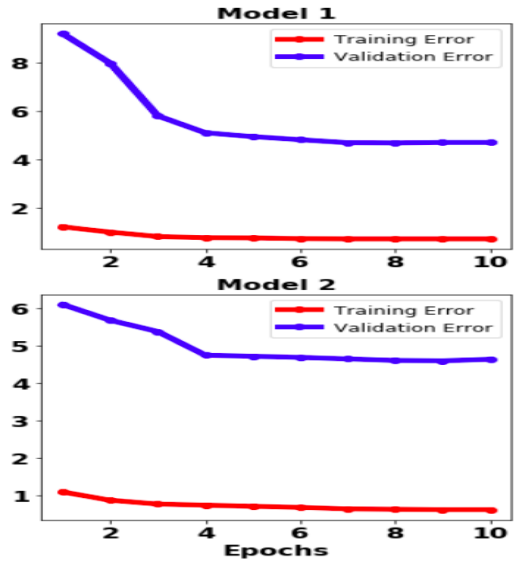

Fig. 4. Variation of training and validation error with epochs.

Table 1. BLEU metric evaluation of the models and the benchmarks set by earlier models

	Model1	Model2	Benchmarks
BLEU - 1	0.54	0.51	0.401 - 0.578
BLEU - 2	0.28	0.25	0.176 - 0.390
BLEU - 3	0.19	0.17	0.099 - 0.260
BLEU - 4	0.082	0.07	0.059 - 0.170

Model-2 is a 4 layer CNN network that extracts features from the images present in the dataset. So, it is specifically trained on the *Flickr8K* image dataset.

Decoder. The decoder consists of an embedding layer and a LSTM as the recurrent unit.

Embedding Before feeding the word vector, the model does embeddings on the target and source words. The vocabulary is calculated from the annotations provided with the data that contains all the words from the image captions. The most frequent words are treated as unique. The embedding weights are learned during training. The decoder consists of one LSTM layer and is the recurrent in the RNN.

C. Training. In this section, we present results from the two different models (Model-1 and Model-2) investigated for this project. The models differ in the method used to extract features from the images. Figure 4 shows the results from the variation of training error, and also the validation error with increase in number of epochs. It is visible from the figure that the simulation becomes stagnant after 3-4 epochs in both the models. Model-1 performs much better in validation accuracy with values reaching below ≈ 4.0 compared to Model-2 in which values stay above ≈ 4.0. In table 1, we present the BLEU-N metric for the two models and comparison with the benchmarks set by other models.

Table 1 shows the results from two machine learning models employed in the present work. The difference between the two models is in the way they extract features from the images.

Fig. 5. An image with five initial captions and the final predicted caption.

D. Caption generation. The captions generated from Model-1 and Model-2 are approximately similar and does not provide any qualitative difference. So, in this section, we focus on results generated from Model-2. Figure 5 shows the captions predicted by the model along with the five initial captions. In this case, the model accurately predicts the major features in the image such as the "dog", and "water". The relationship between these features also depicts the image appropriately. However, if we look at Figure 6, the primary feature of the image is captured accurately. It captures the major feature which is "the girl" but fails to depict minor features and incorrectly predicts features like the "blue shirt" and "grass". In addition, to the limitations in capturing these minor details, the model also fails to describe the exact relationship between the image and the caption. This shows the limitations of our model and highlights the necessity for future improvements in the model. Finally, we also predict caption for a completely random image taken from Iowa state web page and is shown in Figure 6. This image was not present in the *Flickr8K* dataset and would serve as the test for generalization, which is a serious limitation for traditional approaches like "template matching" or "ranking based retrieval". In this scenario, our model (both model-1 and

Fig. 6. Captions generated for an image in the data set and a random image from internet.

model-2) is able to predict the caption describing the person in the red shirt and the street in the image (Fig. 6). However, it fails to detect the building and people walking on the street. The image dataset had ≈ 500 instances of "building", ≈ 3000 instances of "dog", ≈ 6000 instances of "water" and ≈ 1000 instances of "street". We think this difference in numbers is somewhat responsible for the inaccurate description of the images and to accurately predict "dog", "water" and "street" but failing to recognize the "building" and the group of people. We are confident that a much larger and unbiased dataset would resolve these issues, and our model would accurately describe the relationship between images and its caption even for random image sets.

Conclusion

We combine "Image Labeling" and "Automatic Machine Translation" into an end-to-end hybrid neural network system. The developed model is capable to autonomously view an image and generate a reasonable description in natural language with reasonable accuracy and naturalness. Further extension of the present model can be in regard to increasing additional CNN layers or increasing/implementing pre-training, which could improve the accuracy of the predictions.

References

1. Wu Y, et al. (2016) Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR* abs/1609.08144.
2. Jordan M, Mitchell TM (2015) Machine learning: Trends, perspectives, and prospects. *Science* 349(6245):255–260.

Layer (type)	Output Shape	Param #	Connected to
input_2 (InputLayer)	(None, 64, 64, 3)	0	
conv2d_5 (Conv2D)	(None, 64, 64, 32)	2432	input_2[0][0]
batch_normalization_3 (BatchNor	(None, 64, 64, 32)	128	conv2d_5[0][0]
max_pooling2d_5 (MaxPooling2D)	(None, 32, 32, 32)	0	batch_normalization_3[0][0]
conv2d_6 (Conv2D)	(None, 32, 32, 64)	18496	max_pooling2d_5[0][0]
batch_normalization_4 (BatchNor	(None, 32, 32, 64)	256	conv2d_6[0][0]
max_pooling2d_6 (MaxPooling2D)	(None, 16, 16, 64)	0	batch_normalization_4[0][0]
conv2d_7 (Conv2D)	(None, 16, 16, 64)	36928	max_pooling2d_6[0][0]
max_pooling2d_7 (MaxPooling2D)	(None, 8, 8, 64)	0	conv2d_7[0][0]
dropout_2 (Dropout)	(None, 8, 8, 64)	0	max_pooling2d_7[0][0]
conv2d_8 (Conv2D)	(None, 8, 8, 64)	36928	dropout_2[0][0]
max_pooling2d_8 (MaxPooling2D)	(None, 4, 4, 64)	0	conv2d_8[0][0]
input_3 (InputLayer)	(None, 37)	0	
dropout_3 (Dropout)	(None, 4, 4, 64)	0	max_pooling2d_8[0][0]
embedding_1 (Embedding)	(None, 37, 256)	1428480	input_3[0][0]
flatten_1 (Flatten)	(None, 1024)	0	dropout_3[0][0]
dropout_4 (Dropout)	(None, 37, 256)	0	embedding_1[0][0]
dense_1 (Dense)	(None, 256)	262400	flatten_1[0][0]
lstm_1 (LSTM)	(None, 256)	525312	dropout_4[0][0]
add_1 (Add)	(None, 256)	0	dense_1[0][0] lstm_1[0][0]
dense_2 (Dense)	(None, 256)	65792	add_1[0][0]
dense_3 (Dense)	(None, 5580)	1434060	dense_2[0][0]

Fig. 7. Model summary of Model-2. The image passes through a 4-layer CNN network before going through a LSTM network

3. Amershi S, Cakmak M, Knox WB, Kulesza T (2014) Power to the people: The role of humans in interactive machine learning. *AI Magazine*.
4. Meltzoff A, Kuhl P, Movellan J, Sejnowski T (2009) Foundations for a new science of learning. 325:284–8.
5. Hinton G, et al. (2012) Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine* 29(6):82–97.
6. Misra J, Saha I (2010) Artificial neural networks in hardware: A survey of two decades of progress. *Neurocomputing* 74(1):239 – 255. Artificial Brains.
7. Maier HR, Dandy GC (2000) Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. *Environmental Modelling and Software* 15(1):101 – 124.
8. Bhute AN, Meshram BB (2014) Text based approach for indexing and retrieval of image and video: A review. *CoRR* abs/1404.1514.
9. Karpathy A, Fei-Fei L (2017) Deep visual-semantic alignments for generating image descriptions. *IEEE Trans. Pattern Anal. Mach. Intell.* 39(4):664–676.
10. Vinyals O, Toshev A, Bengio S, Erhan D (2014) Show and tell: A neural image caption generator. *CoRR* abs/1411.4555.
11. O'Shea K, Nash R (2015) An introduction to convolutional neural networks. *CoRR* abs/1511.08458.
12. Lipton ZC, Kale DC, Elkan C, Wetzel RC (2015) Learning to diagnose with LSTM recurrent neural networks. *CoRR* abs/1511.03677.
13. Schmidhuber J (2015) Deep learning in neural networks: An overview. *Neural Networks* 61:85 – 117.
14. Hodosh M, Young P, Hockenmaier J (2013) Framing image description as a ranking task: Data, models and evaluation metrics. *J. Artif. Int. Res.* 47(1):853–899.
15. Lin T, et al. (2014) Microsoft COCO: common objects in context. *CoRR* abs/1405.0312.
16. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. *CoRR* abs/1409.1556.