

Auto-Generation of Captions for Ancient Indian Historical Images using Deep Learning Networks

P Rama Devi¹ and Shylaja S S²

¹⁻²Department of CSE, PES University, Bangalore, India
Email: pramadevi@pes.edu, shylaja.sharath@pes.edu

Abstract—Automatic image annotation is the process through which a machine can automatically describe an input digital image in the form of keywords or captions. In this paper we focus on generating captions for various sculptures carved in the ancient past by the Indian kingdoms in various heritage sites such as temples, caves and other places. Towards this direction, a new dataset along with annotations is created. This dataset is used on various deep learning models such as VGG16-LSTM, ResNet50-LSTM, VGG19-LSTM, Xception-LSTM, InceptionResNetV2-LSTM for the auto generation of captions and the performance measure for each of the models is evaluated.

Index Terms— Image Captioning, Deep Learning, Computer Vision, Natural Language Processing, CNN, LSTM.

I. INTRODUCTION

In our daily life a lot of images from sources like Internet, News articles, Documents etc. are seen. We humans can understand the images even though the description is not given. But, for the machines it is a complex task. Hence, several attempts have been carried out to automatically caption the images which is also known as Image tagging. Applications of Image Captioning include Content-Based Image Indexing, Assistance for Visually Impaired, Visual Positioning and Tracking, Video Surveillance, etc.

Image captioning can be broadly categorized into two tasks namely 1) Understanding an image and getting context-based information 2) Generating descriptions for the image. To achieve these two tasks, we need a good connect between two important fields namely Computer Vision and Natural Language Processing. Inspired by the successful application of the encoder-decoder paradigm in neural machine translation, a large number of models have been developed in the context of image captioning. Before the exploration of Deep Learning Models, Traditional Machine Learning Models were used to caption images. Handcrafted features such as Scale-Invariant Feature Transform [1], Local Binary Patterns [2], Histogram of Oriented Gradients [3] are extensively used in Traditional Machine Learning. Using the above-mentioned techniques important attributes are extracted from the input data. These are passed to a classifier such as Support Vector Machines (SVM) [4] for classification of objects. Obtaining features from a huge and diverse collection of data is not easy since these hand-crafted features are task oriented. Real world data such as images and videos are very complex and have various semantic representations. But Deep Learning Models can automatically learn features from training data without manual intervention. We in this paper, focus on Deep Learning Models since real world data is complex and have different semantic interpretations.

II. RELATED WORK

There have been Traditional and Deep Learning Methods in place for auto-tagging of images. Traditional methods for auto-tagging of images fall into two categories namely generative and retrieval models. The figure 1 below shows generative based method for retrieving the captions. Here the method requires extraction of some objects, attributes or semantic information from the image, and then uses a specified grammar rule to combine the information or fills the obtained data into the pre-defined blanks of the sentence template to form the image description.

[Li et al.,] [5] uses an image recognizer to obtain visual information from the image, including objects, attributes of objects and spatial relationships between different objects. Object detector to detect objects, attributes, propositions in the image has been used by [Kulkarni et al.,] [6]. Furthermore, a Conditional Random Field (CRF) is constructed to deduce the relevant information from the extracted features.

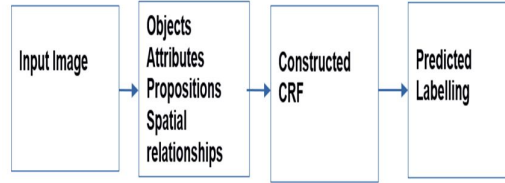


Figure 1: Generative Model

The figure 2 shows the retrieval model. Here for a given image, the method aims to retrieve the matching sentence(s) from a set of given image-description pairs. [Ordonez et al.,] [7] retrieves a series of related images from the image-description pairs by Gist and Tiny image descriptors. Later they classify the query image by specific objects and scenes. After that they reorder the retrieved images and the image with top rank is considered.

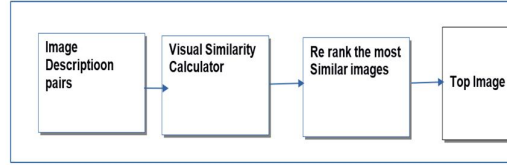


Figure 2: Retrieval Model

The other way of image captioning is using Deep Learning Models. Encoder-Decoder has been the basic structure for most of the state-of-the-art techniques. In these models Convolutional Neural Networks (CNNs) act as encoder to fetch the image features. Recurrent Neural Networks (RNNs) act as decoder to represent semantic interpretations to generate sentences. The Vanilla RNN can only remember the contents of previous few time steps. This is mainly due to the vanishing gradient issue. Therefore, Long-Short-Term-Memory Networks (LSTMs) [9] have been introduced. These LSTMs are able to learn long series of time steps, and can continue to have huge memory. Vinyal's [8] worked on the above model in his paper. In this network, Encoder will extract the hidden activations of CNN global image features. These features are sent to the decoder network i.e., LSTM to generate a sequence of words. Wu's [11] use attribute generation model based on CNN. They use attribute-based-captioning prediction model to generate the captions.

The encoder and decoder mechanism used by Aneja et al. [10] is VGGNet, Language CNN. In this architecture MSCOCO dataset T. Lin et al., [15] has been used and evaluated on various evaluation metrics. Karpathy et al. [12] developed a Recurrent Neural Network (RNN) model with VGGNET which obtains the probability distribution of the next word in the caption based on previously generated words. The architecture used, generates whole scene images by taking Multispace model. Zhang et al. [13] have used Inception-V3 and LSTM model using MS-COCO dataset. Image and text features are mapped in to the space using visual space technique. In our paper we propose to generate a new dataset which can give description to Ancient Indian historical images and have evaluated the results against various architectures.

III. DATASET GENERATION

In this section we look at the dataset generation for Ancient Indian historical images. In India alone, there are more than 2 million temples (census 2001) while the number has been substantially increasing even

more during the past two decades. But until now as far as the authors know there has been no proper image captioning model that can caption these types of architectures, sculptures, idols, artworks, etc for Indian scenario. Although there has been significant progress in the recent years in the field of image captioning, there has not been many models that can properly classify and caption the images of Ancient Indian Artworks or Temples for India. By properly classifying and captioning such images, not only would it help improve and simplify the work of researchers in this field, it could even help travellers and the common people. To carry out such a task it is imperative that a good dataset with proper annotations is made available to the learning models.

TABLE I: DATASET CREATION

	Vishnu Sculpture	A stone sculpture has a smooth carving while adorned with traditional Hindu ornaments with a conch on the right and a disc on the left of the sculpture	<u>Halebidu, Hassan, Karnataka</u>
---	------------------	---	------------------------------------

As shown in the table I above as an example, the sculpture is of Lord Vishnu which is located in Halebidu of Hassan District in the State of Karnataka. The dataset is created in such a way that images are considered if and only if they are images of Ancient Indian Artworks, Architectures, or Idols. The dataset mainly contains, the filename which consists of the name of the file and the image format. The name of most sculptures follows official given names but, in the event where the name of the sculpture or artifact is not obtained, the name is either left blank, or the name of the temple is used instead. The description for the images acts as its captions.

The Image file consists of 831 images and index is given for each image. Caption file consists of four captions for each of the 831 images totalling to 3,324 captions creation. Sample data entry is shown in below table II.

TABLE II: SAMPLE DATA ENTRY

Image index for image file	Captions given to each image for captions file
 1.jpg	1.jpg#0 This is a sculpture of Lord Shiva performing his cosmic dance. 1.jpg#1 This can be found in the <u>Nellaippar Temple</u> located in <u>Tirunelveli, Tamil Nadu</u>. 1.jpg#2 Lord Shiva is worshipped as <u>Nellaippar</u> or <u>Venuvananathar</u> and Goddess Parvati is worshipped as <u>Kantimathi Amman</u>. 1.jpg#3 This temple signifies one of the five places where Lord Shiva is said to have performed his Cosmic dance (<u>Thandav</u>).

IV. METHODOLOGY

A. Pre-processing

In the pre-processing stage first step is to prepare data for images and text. VGG model is loaded excluding the last layer of the model. 1- dimensional 4,096 feature vectors are generated from VGG output that are recorded in a pickle file. After extracting the features from the images in the dataset, the captions from the caption files for each image are processed. Processing includes cleaning the captions. Here, the punctuations, stop letter words, as well as words containing numbers, are removed. This reduced the complexity of the captions making it easier for the computer to understand and learn. After doing so, the captions for each individual image are stored in a dictionary. The contents of this dictionary are then saved. Now both files with and without cleaning can be seen in Table 3 below.

TABLE III: CAPTIONS WITHOUT AND WITH PRE-PROCESSING

1.jpg#0 This is a sculpture of Lord Shiva performing his cosmic dance.	1 this is sculpture of lord shiva performing his cosmic dance
1.jpg#1 This can be found in the Nelliappur Temple located in Tirunelbeli, Tamil Nadu.	1 this can be found in the nelliappur temple located in tirunelbeli tamil nadu
1.jpg#2 Lord Shiva is worshipped as Nelliappur or Venuvananathar and Goddess Parvati is worshipped as Kantiathi Amman.	1 lord shiva is worshipped as nelliappur or venuvananathar and goddess parvati is worshipped as kantiathi amman
1.jpg#3 This temple signifies one of the five places where Lord Shiva is said to have performed his Cosmic dance (Thandav).	1 this temple signifies one of the five places where lord shiva is said to have performed his cosmic dance thandav
2.jpg#0 This is located in the Brihadisvara Temple.	2 this is located in the brihadisvara temple
2.jpg#1 This shows Shiva seated with Parvathi, with Chandesha below him.	2 this shows shiva seated with parvathi with chandesha below him
2.jpg#2 Shiva is in the process of winding a garland of flowers around Chandesha's head.	2 shiva is in the process of winding garland of flowers around chandesas head
2.jpg#3 It is believed Chandesha represents King Rajendra Chola.	2 it is belived chandesha represents king rajendra chola

B. Deep Learning Models

This is one of the most important stages in the process. Here, our Deep Learning Models are trained to fit our dataset which helps to identify the features in the image by linking it to our RNN. Different combinations of CNN and RNN were used and evaluated with our Deep Learning Models. Some of these were VGG16-LSTM, ResNet50-LSTM, VGG19-LSTM, InceptionResNetV2-LSTM and Xception-LSTM. As shown in the figure 3 below, given an image to the model it should give caption to the image. Here is a brief description about various architectures used.

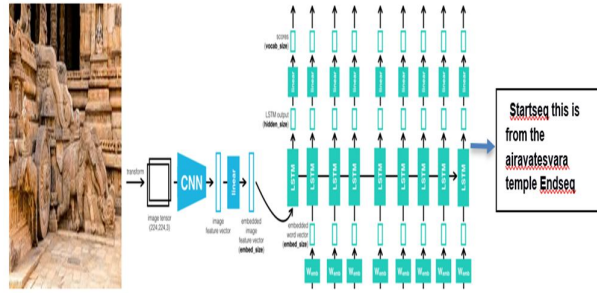


Figure 3: End to End Architecture of CNN and LSTM

VGG Architecture

The input to the network is image of dimensions $224 \times 224 \times 3$. As shown in the figure 4 below we have sixteen layers excluding max pool layer as max pool layers does not have parameters. Each of the convolution layers have varying number of filters like 64, 256, 512 with filter size as 3×3 . The output of last max pool layer, as shown in the figure 4 is flattened and connected to three fully connected layers. Connections of VGG Model has been shown in figure 4.

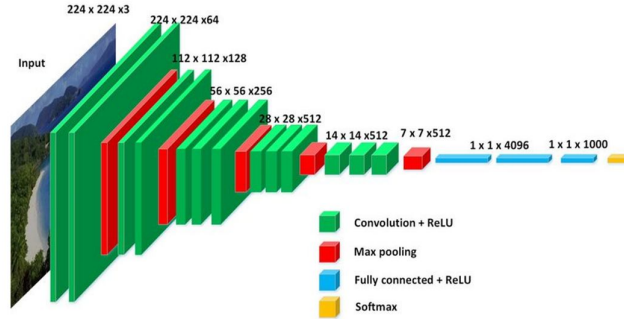


Figure 4: VGG Model

LSTM

Recurrent Neural Networks face vanishing and exploding gradient problem. To overcome this, we use LSTMs as they deal with both Long-Term Memory and Short-Term Memory. To make the calculations easy LSTM uses three gates called input gate, output gate and forget gate. The cell recollects values over arbitrary time periods. The information is flown into and out of the 3 gates. The architecture of LSTM is shown below in Figure 5. *Input Gate tries to retain and see what data can be kept. x : current input; h : previous hidden state are the inputs to the LSTM.*

$$i_t = \sigma(w_i[h_{t-1}, x_t] + b_i) \quad (1)$$

Forget Gate is used to check what information is to be kept from the previous time step. f_t is the forget gate given below.

$$f_t = \sigma(w_f[h_{t-1}, x_t] + b_f) \quad (2)$$

Output Gate is used to know what information is outputted and sent to the next layer. O_t is the output gate.

$$o_t = \sigma(w_o[h_{t-1}, x_t] + b_o) \quad (3)$$

Where each term is indicated as

σ : sigmoid function

w_x : weights for the neuron gates

h_{t-1} : preceding output at timestamp t-1

x_t : present time step information

b_x : Input, Forget, Output gate biases

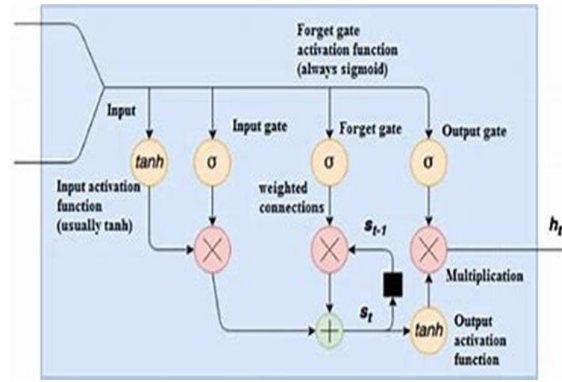


Figure 5: LSTM Architecture

Inception model

In Inception model filters of size 1x1, 3x3, 5x5 are applied at the same time and then concatenated to the feature map instead of applying filters of different sizes at different layers. Also, the number of computations has been reduced by using 1x1 filters. The below figure 6 shows an Inception module in the Inception Architecture. An extension of the Inception Architecture is Xception. In this Architecture depth wise Separable Convolutions are substituted for Inception modules. Xception model is 71 layers deep.

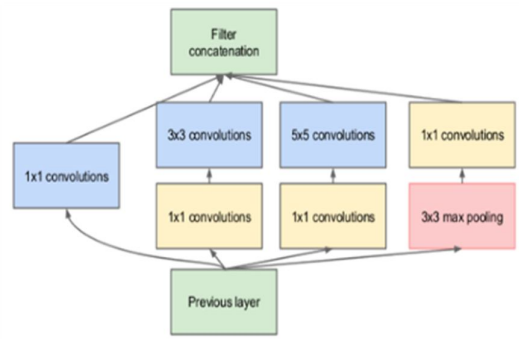


Figure 6: Inception Module

Residual Networks

The idea of Resnet [Kaiming He et al.,14] is that if shallow network works well then the deep network should also work well by simply learning to compute identity function in the new layers. The shallow network solution space is a subset of the deep network solution space. This identity function from the input allows a ResNet to retain a copy of input as shown in figure 7.

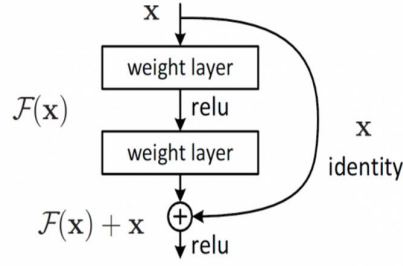


Figure 7: Residual Networks

C. Model Definition

The Photo Feature Extractor model and the Sequence Processor model produced a 256-element vector. Drop out of 50% is used for both the models. With this overfitting of the training dataset is reduced. The Decoder model is merged with both input models. This output is then passed to a 256 dense neuron layer. The last layer in the architecture is the dense output layer. Finally, a softmax activation function is used on the entire output vocabulary. This will predict the next word in the sequence. Figure 8 shows the architecture of the model defined.

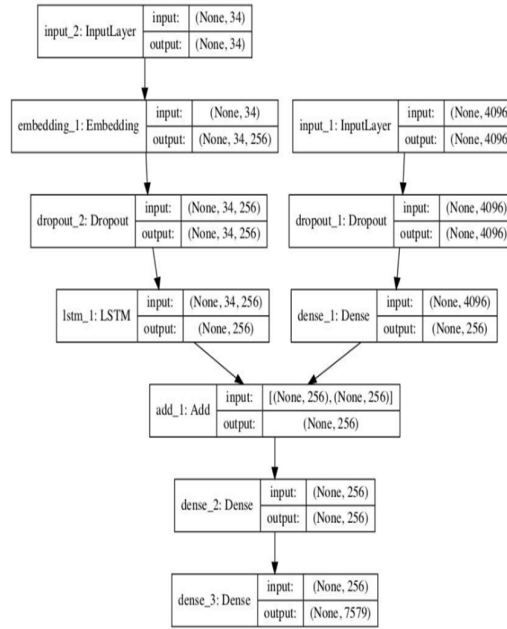


Figure 8: Architecture of the Model

D. Model Evaluation

After the model is fit, evaluation of the model is done. The generated descriptions for all the input images in the test dataset are evaluated with the cost function. Using the corpus BLEU score [K. Papineni et al., 17] the actual and predicted captions are evaluated. This will summarize how near the text generated is to the text expected.

V. RESULTS

The results on the various Deep Learning Models such as VGG16 LSTM, ResNet50-LSTM, VGG19-LSTM, InceptionResNetV2-LSTM, Xception-LSTM are shown in the table 4 below. VGG16-LSTM, ResNet50-LSTM, VGG19-LSTM models worked well for the set of images given with BLEU scores above between 0.5 to 1.0 as we can see in the table 5. Performance of the InceptionResNetV2-LSTM model is poor compared to all the models used even though BLEU scores are good. This is probably due to the model learning the task too quickly and basically stops improving after. And thus, it might be probable that InceptionResNetV2 doesn't do better over other models.

TABLE IV: EVALUATION SCORES ON THE VARIOUS DEEP LEARNING MODELS USED

Model Name	VGG16 - LSTM	ResNet50 - LSTM	VGG19 - LSTM	Xception - LSTM	InceptionResNetV2 - LSTM
BLEU-1	0.816195	0.981031	0.684759	0.816195	0.956732
BLEU-2	0.767668	0.973135	0.557215	0.767668	0.944815
BLEU-3	0.767898	0.971876	0.534872	0.767898	0.943907
BLEU-4	0.734439	0.966028	0.465427	0.734439	0.933493

The table V above shows the BLEU Scores on VGG16-LSTM, ResNet50-LSTM, VGG19-LSTM, InceptionResNetV2-LSTM, Xception-LSTM Deep Learning Models we have used.

Model	Image1	Image2	Image3
VGG16 -LSTM			
Caption generated	startseq this is located in the brihadisvara temple endseq	startseq this is the airavatesvara temple in darasuram tamil nadu endseq	startseq it is rockcut buddhist cave monuments from century bce to century ce endseq
ResNet50 -LSTM			
Caption generated	startseq this is located in the brihadisvara temple endseq	startseq this is temple dedicated to lord vishnu endseq	startseq buddha depiction carved into the wall in seated position with frame around it endseq
VGG19 -LSTM			
Caption generated	startseq this is located in the brihadisvara temple endseq	startseq this is the airavatesvara temple in darasuram tamil nadu endseq	startseq it is rockcut buddhist cave monuments from century bce to century ce endseq

InceptionResNetV2-LSTM			
Caption generated	startseq it	startseq it	startseq it
Xception-LSTM			
Caption generated	startseq this is d etailed sculpture of an elephant end seq	startseq this is det ailed sculpture of a n elephant endseq	startseq this is fro m the durga temple i n aihole built durin g the century by the chaluhyas endseq

Table V: Generated Captions on the various Deep Learning Models

VI. CONCLUSIONS

We have implemented Image Captioning on the images of Ancient Indian historical Artworks, Architectures using various Deep Learning Models. In this process a new dataset containing images and captions of Indian Ancient Artworks, Temples, Architectures, and Statues in India has been created. Various Deep Learning Models such as VGG16 LSTM, ResNet50-LSTM, VGG19-LSTM, InceptionResNetV2-LSTM and Xception-LSTM, have been used on the dataset created and evaluation scores for the same is done. VGG16 LSTM, ResNet50-LSTM, VGG19-LSTM models performed well compared to InceptionResNetV2-LSTM. Through the generation of captions for Indian Ancient Artworks it becomes easier to understand the artwork in the image, this helps in easier identification of the artwork along with the type of artwork. This can help researchers and avid travelers who want to know more about the artwork through the click of a single image. Further enhancements can be done to increase the dataset so that large variety of images are available for the models.

REFERENCES

- [1] David G Lowe. Distinctive image features from scale-invariant key points. *International journal of computer vision* 60, 2 (2004), 91–110.
- [2] Timo Ojala, Matti Pietik inen, and Topi M n nen. 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. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition. CVPR 2005. IEEE Computer Society Conference on*, Vol.1. IEEE, 886–893, 2005.
- [4] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*. ACM, 144–152.
- [5] S. Li, G. Kulkarni, T. L. Berg, A. C. Berg, and Y. Choi. Composing simple image descriptions using web-scale n-grams. In *CoNLL*, pages 220–228, 2011.
- [6] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Baby talk: Understanding and generating simple image descriptions. In *CVPR*, pages 1601–1608, 2011.
- [7] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describing images using 1 million captioned photographs. In *NIPS*, pages 1143–1151, 2011.
- [8] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3156–3164, 2015.

- [9] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation* 9.8 pp. 1735–1780, 1997.
- [10] Jyoti Aneja, Aditya Deshpande, and Alexander G Schwing. Convolutional image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5561–5570, 2018.
- [11] Qi Wu, Chunhua Shen, Peng Wang, Anthony Dick, and Anton van den Hengel. Image captioning and visual question answering based on attributes and external knowledge. *IEEE transactions on pattern analysis and machine intelligence* 40, 6, 1367– 1381, 2018.
- [12] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3128–3137, 2015.
- [13] Li Zhang, Flood Sung, Feng Liu, Tao Xiang, Shaogang Gong, Yongxin Yang, and Timothy M Hospedales. Actor-critic sequence training for image captioning. *arXiv preprint arXiv:1706.09601*, 2017.
- [14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778, 2016.
- [15] T. Lin, M. Maire, S. J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: common objects in context. In *ECCV*, pages 740–755, 2014.
- [16] K. Papineni, S. Roukos, T. Ward, and W. Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002.