DEEP LEARNING BASED IMAGE CAPTION GENERATOR

Charan Reddy[1], Rohith Reddy[2], Sampath[3], Niteesh[4], Anmol Bhayana[5], Jaisakthi S M [6], Archit[7]

[1-7]*Department of Computer Science, Vellore Institute of Technology, Vellore, Tamil Nadu, India*

Abstract - *Understanding visual scenes is one of the primary goals of computer vision. Scene understanding involves several tasks, including recognizing what objects are present, localizing the objects in 2D and 3D, characterizing relationships between objects, determining the objects and scene's attributes, and to supply a meaningful description of the scene. Image captioning simply means describing a picture which is fed to the model. The auto generation of accurate syntaxial and semantical image captions is a crucial problem in AI. It essentially combines methods from computer vision to know the content of the image and a language model from natural language processing field to visualize the understanding of the image into words within the correct order.*

In this paper, we will use the ideas of the Convolutional Neural Network (CNN) as well as Long Short-Term Memory (LSTM) model to create the proposed working model of Image Caption Generator.

Key Words: Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), Captions, Transfer Learning and Generator.

1. INTRODUCTION

1.1 Objective

Image caption generator is basically a task that involves both computer vision and natural language processing concepts like CNN, LSTM to recognize the context of an image and describe them in a natural language like English. It essentially combines the methods from computer vision to acknowledge the content of the image and language model from natural language processing field to visualize the image into words within the correct order.

The objective of our project is to learn the concepts of a CNN and LSTM model and build a working model of Image caption generator by implementing CNN with LSTM. we will use the ideas of the Convolutional Neural Network (CNN) as well as Long Short-Term Memory (LSTM) model to create the proposed working model of Image Caption Generator
Basically, in our model the CNN will work as encoder to extract features from given images and LSTM works as decoder to get words that might describe the given image.

Basically, in our model the CNN will work as encoder to extract features from given images and LSTM works as decoder to get words that might describe the given image.

1.2 Motivation

The importance of this problem to world scenarios are often understood using few applications where an answer to the present problem is often very useful. Few applications where the answer of this problem could also be useful are:

1. Automatic driving is one among the most important challenges and if we can properly caption the scene round the car, it can enhance to the self-driving system.

2. We can develop a product for the blind that will allow them to navigate the roadways without the assistance of others. We could accomplish this by turning the scene to text and then the text to voice.

3. Automatic captioning can help Google Image Search, which will become nearly as excellent as Google Search, as each image is first converted into a caption, and then searches are conducted based on the caption.

4. CCTV cameras are now ubiquitous, but if we combine viewing the globe with the generation of appropriate captions, we will be able to raise alarms as soon as criminal conduct is detected. This will undoubtedly aid in the reduction of crime and accidents.

1.3 Background Study

One of our primary goals of computer vision is to know the visual scenes. Scene understanding involves several tasks, including recognizing what objects are present, localizing the objects in 2D and 3D, characterizing relationships between objects, determining the objects and scene's attributes, and to supply a meaningful description of the scene. The several image caption models have been developed in the past.

The model based on LSTM and CNN uses deep convolutional neural network (CNN) is to create a dense feature vector from the images. The dense vectors are also called embedding. For the image caption model, this embedding acts as a dense representation of the image which can be used as the initial state of the LSTM. The CNN network can be trained directly on the images in our dataset. An LSTM is recurrent neural network architecture that is mainly used for problems having temporal dependences. It useful for capturing information about previous states to better inform the current prediction through its memory cell state. There are three main components of LSTM: input gate, output gate

and a forget gate. Updates are altered to cell's memory state using each of these three gates.

2. LITERATURE SURVEY

It concentrates on survey and analysis of different Type of image captioning models.

A set of K Nearest Neighbor images are found for the given image using cosine similarity with the feature spaces GIST, fc7 and fc7-fine. Image features are calculated for every image present in the training dataset. For the k nearest training images for a given test image, union of their captions is taken and set is created with n candidate captions and among them best is selected using similarity score between the captions using BLEU and CIDEr similarity functions. Similarity function needed to be improved since the captions generated here is far from the caption that would have been generated by the human for the given image. Future work involves finding more robust approaches for selecting the best caption from the set of candidates generated [1].

Both top-down and bottom-up features from an input image are extracted and intermediate filer responses from a classification CNN are used to build a global visual description. All visual features are fed into RNN for caption for caption generation. Input attention model. Input and output attention model uses bilinear function to evaluate the score for the relevance of predicted word accordingly the caption gets generated. Attention model need to get modified because when compared with Google NIC it was found that better captions would have been generated. Since the performance is affected by applied visual attributes generation method, better methods are needed to be used [2].

The CNN and RNN architectures to generate the captions are additionally incorporated with the detected high-level attributes. To explore the semantic correlations between the attributes Multiple Instance Learning (MIL)-based model with Inter-Attributes Correlations (IAC) is devised. The method is able to predict the attributes probability distribution over the numerous massive attributes available. Model then integrates the high-level attributes into the LSTM. Since the attributes have huge role here, large-scale benchmark like YFCC-100M dataset can be used to learn attributes to get better captions. Generation of open-vocabulary and free-form sentences with the learnt attributes can also be expected [3].

It uses two networks namely policy network and value network both containing CNN and a RNN. Same CNN and RNN architectures are used for both. Policy network provides the probability for the agent to take actions in each state whereas value network is used as predictor of the total reward. Training is done using deep reinforcement learning and visual-semantic embedding is used as reward here. The system fails to understand important visual contents which only are present in small portions of the images which is due to policy network architecture and this implies policy network improvement. Network architecture needs to be improved and reward design needs to get investigated by considering other embedding [4].

Encoder and decoder are separated to different LSTMs for constructing deep hierarchical encoder-decoder to fuse the visual and textual semantics before decoding. Deep CNN has been used to encode the hierarchical image features, Sentence-LSTM to encode sentence inputs, Vision-Sentence Embedding LSTM for fusing and transforming CNN visual feature and SLSTM encoded sentence feature to join semantic space, semantic fusion LSTM decoder to decode the composition of the image feature, vision-sentence embedded vector and to target sentence, sentence encoded feature. The method is better suited to large datasets and in case of small datasets there is the need to update the model for more meaningful caption. To figure out how to stack different layers and set the parameters so as the vertical depth of the encoder decoder model can be increased. To figure out how to stack different layers and set the parameters so as the vertical depth of the encoder-decoder model can be increased [5].

Encoder-decoder architecture has been adopted which incorporates visual attention mechanism to generate image captioning. The encoder part being the CNN and the decoder part using visual. attention module. ADAM has been used as optimizer during training. Stochastic Gradient Descent (SGD) has been used to minimize the loss function. There is need of more robust thinking while working with the natural language processing models. Deep CNN with the use of BLEU metric can be used to enhance the performance of model [6].

The proposed model can generate more unique and novel captions, when compared to the baseline in all three datasets used in this experiment. When compared with the methods that only use the CNN as encoder, the model performed better. Improvements are observed in all the sub-metrics when image caption is decoded in the phrase based hierarchical manner. The average length of the captions generated by phi LSTM model is shorter as compared to baseline models. In Flickr8k dataset, baseline models managed to infer correctly more words than the model proposed in this paper • The phrase-based LSTM (phi-LSTM) has less chance to predict particle word 'up and conjunction 'from', with influence from longer sequence of previous words [7].

A deep neural network architecture Word2VisualVec, is proposed, which is capable of transforming a natural language sentence into a meaningful visual feature representation Multimodal query composition, by adding and/or subtracting the predicted visual features of specific words to a given query image is also supported by Word2VisualVec and it can also easily generalize to predict a visual-audio representation from text for video caption retrieval. A sentence is given as input in the form of word

sequence into a recurrent neural network (RNN). The RNN output is projected into a latent subspace. The encoding is transformed into a higher dimensional visual feature space via a multi-layer perceptron. As the visual features are predicted from text, Word2VisualVec is called. Word2VisualVec applies a mapping from the textual to the visual modality. Therefore, at run time, Word2VisualVec allows the caption retrieval to be performed in the visual space. The drawback with the caption retrieval task is that it works with the assumption that for a query image or video, there must be least one sentence relevant with respect to the query [8].

Aneja, J., Deshpande, A., & Schwing, A. G. discussed a convolutional approach for image captioning and also demonstrated that it performs at par with existing LSTM techniques. The differences between RNN based learning and the proposed convolutional method is analyzed, and discovered gradients of lower magnitude as well as overly confident predictions to the existing LSTM network concerns. The model used in this paper is based on the convolutional machine translation model. A simple feed forward deepnet for modeling is used. Prediction of a word relies on past words or their representations. To prevent convolution operations from using information of future word tokens, a masked convolutional layer that operates only on "past" data is used. After sampling, the word is fed back into the feed-forward network to generate subsequent words. Inference continues until the end token is predicted, or until reach a fixed upper bound of N steps is reached. Though comparable CIDEr scores and better SPICE scores than LSTM on test set with the proposed CNN+Attn method, the BLEU, METEOR, ROUGE scores are less than the LSTM which is an area of improvement [9].

Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., & Zhang, L proposed a novel combined visual attention mechanism, bottom-up and top-down. The approach allows images' attention to be calculated more naturally at the level of objects and other salient regions. Applying the abovementioned approach to image captioning and visual question answering, state-other-art results in both tasks is achieved, while it improves the interpretability of the resulting attention weights. The bottom-up mechanism proposes a set of salient image regions, with each region representing a pooled convolutional feature vector. Bottom-up attention is implemented using Faster R-CNN, which represents a natural expression of a bottom-up attention mechanism. The top-down mechanism, on the other hand, uses task-specific context to predict an attention distribution over the image regions. Finally, the attended feature vector is then computed as a weighted average of image features over all regions. Relative to the Self-critical Sequence Training (SCST) models, the ResNet baseline used in this paper and slightly worse performance when optimized for CIDEr score [10].

3. PROPOSED SYSTEM

3.1 Analysis and Design

Our proposed paper "Image Caption Generator" uses the concepts of Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) models. Basically, the CNN will work as encoder to extract features from given images and LSTM works as decoder to generate words that would describe the given image. The LSTM model cannot understand the directly fed images which are basically The RGB image tensor is needed because the LSTM isn't designed to handle such inputs. For using the LSTM model for prediction, we need to first extract some features from the images which can be then fed to the LSTM architecture. In our proposed model, deep convolutional neural network (CNN) is used to create a dense feature vector from the images. The dense vectors are also called embedding. For our model, this embedding acts as a dense representation of the image which can be used as the initial state o LSTM. The CNN network can be trained directly on the images in our dataset

An LSTM is recurrent neural network architecture that is mainly used for problems having temporal dependences. It useful for capturing information about previous states to better inform the current prediction through its memory cell state. There are three main components of LSTM: input gate, output gate and a forget gate. Updates are altered to cell's memory state using each of these three gates. In our model, LSTM based model is used to predict the next sequence of words in the caption. The previously created image embedding using CNN model is fed as initial state into an LSTM. The image embedding acts as the first previous state to the LSTM based language model and influences the next sequence of words in the caption. At each iteration, the LSTM considers the previous cell state and outputs a prediction for the most probable next words in the sequence. This process is repeated until the end token is sampled. The end token is used for signaling the end of the caption has been reached.

Caption generator used for caption generation uses beam search to improve the quality of sentences that are generated. At each iteration, the previous state of the LSTM (initial state is the image embedding) and previous sequence is passed by the caption generator in order to generate the next SoftMax vector. It keeps the top N most probable candidates and uses it in the next inference step. This process continues until either the max sentence length is reached, or end-of-sentence token is generated by all the tokens.

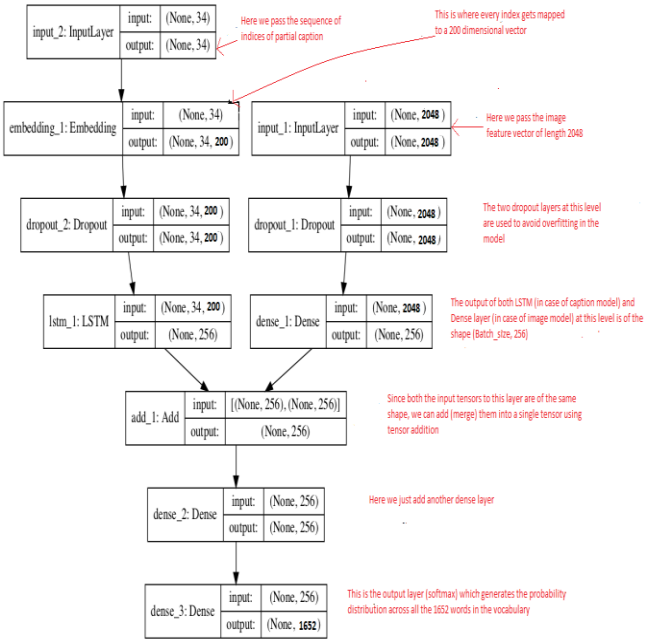

Fig.1 Flowchart of architecture

3.2 Module Description

3.2.1 Data Preprocessing

The following steps are involved in data pre-processing:

1) DATA COLLECTION: We have used the Flickr 8k dataset in our project which contains 8000 images in total. Each image with 5 captions, since a single image can have multiple caption and all of them being relevant simultaneously. Among the 8000 images, the images are divided into 3 categories. They are
    1. Training Set = 6000 images
    2. Development Set =1000 images
    3. Testing Set= 1000 images

2) UNDERSTANDING THE DATA: The dataset contains some text files related to the images. One among them is "Flickr8k.token.txt" which holds the name of each image of the dataset along with its 5 captions. Hence, the every line contains #i , where 0≤i≤4 the name of the image, caption number (0 to 4) and actual caption.

3) DATA CLEANING: The data cleaning steps involves some of the basic steps like lower-casing all words for eg. to avoid "hello" and "Hello" to be regarded as separate words), removing the special tokens (such as '%', '$', '#', etc.). we are also eliminating words which contain numbers. We are creating a vocabulary of all unique words which are present across the 8000*5 (i.e. 40000) image captions present in our data set. In total, we have 8763 unique words present across the 40000 image captions.

4) LOADING THE TRAIN AND TEST DATA: The names of the images belonging to the training set is contained in file "Flickr_8k.trainImages.txt". Hence, we load those names in a list "train" and hence we separated 6000 training images in the list named "train".

3.2.2 Transfer Learning

Transfer learning is applied on: (a) images and (b) captions.

1) IMAGES: The input (X) to our model are the images. As we know the input to the model is given in the form of vector so we need to convert every image into a fixed sized vector and then that vector can be fed as input to neural network. In our project, we have used the transfer learning by using Resnet50 model (Convolutional Neural Network). The model was already trained to classify 1000 different classes of images present in ImageNet dataset. Since we do not need to classify images but just need the information vector of the image, we have removed last SoftMax layer present in the model and extracted the 2048 length vector and this process is known as automatic feature engineering.

2) CAPTIONS: We have mapped every word (index) to the 200 long vector by using a pretrained GLOVE model. And then for all the unique 1652 words in the vocabulary, we created the embedding matrix which will be loaded to the model before training.

3.2.3 Data Preparation Using Generator Function

For the purpose of feeding the data to the model we have to process it and make it convenient to be given as the input to the deep learning model. We created a generator function which passes the previous state of LSTM, the previous sequence and image vector in order to generate the next SoftMax vector. The most probable top N candidates are kept and are utilized in the next upcoming inference. The process is continued till either the sentence has generated the end sentence token or the max length is reached.

3.2.4 Building the Model, Training and Prediction

Our LSTM model is being fed the partial caption and then we are using this LSTM and image vector generated by CNN in order to generate the full captions. We can't utilize the Sequential API supplied by the Keras library because our input consists of two parts: an image vector and a partial caption. As a result, we use the Functional API, which allows us to combine the two models we generated, LSTM and image vector, to predict the next word in the sequence. For training, we will be using batch size of 3 and we train our

model for 20 epochs. Since, we have 6000 training images; we will call the generator function two thousand times so that all the training images will pass through it. After training the model we will pass the image to the model to generate the required caption

4. CONCLUSION & FUTURE ENHANCEMENT

We successfully created the image captioning model using the CNN and LSTM which was able to predict the caption for the given image. Since our input consisted of two parts, an image vector and a partial caption, we used the Functional API provided by the keras which allowed us to merge the two models which we created namely LSTM and image vector which then predicts the next word of the sequence.

Our future works involves training our model on large dataset like flick30k and improving the network architectures to improve the accuracy and also exploring the natural language processing technique more deeply for better caption formulation.

REFERENCES

[1] S. M. Metev and V. P. Veiko, Laser Assisted Microtechnology, 2nd ed., R. M. Osgood, Jr., Ed. Berlin, Germany: Springer-Verlag, 1998.

[2] Devlin, J., Gupta, S., Girshick, R., Mitchell, M., & Zitnick, C. L. (2015). Exploring nearest neighbor approaches for image captioning. arXiv preprint arXiv:1505.04467.

[3] You, Q., Jin, H., Wang, Z., Fang, C., & Luo, J. (2016). Image captioning with semantic attention. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4651-4659).

[4] Yao, T., Pan, Y., Li, Y., Qiu, Z., & Mei, T. (2017). Boosting image captioning with attributes. In Proceedings of the IEEE International Conference on Computer Vision (pp. 4894-4902).

[5] Ren, Z., Wang, X., Zhang, N., Lv, X., & Li, L. J. (2017). Deep reinforcement learning based image captioning with embedding reward. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 290-298).

[6] Xiao, X., Wang, L., Ding, K., Xiang, S., & Pan, C. (2019). Deep hierarchical encoder– decoder network for image captioning. IEEE Transactions on Multimedia, 21(11), 2942-2956.

[7] Hani, A., Tagougui, N., & Kherallah, M. (2019, December). Image Caption Generation Using A Deep Architecture. In 2019 International Arab Conference on Information Technology (ACIT) (pp. 246-251). IEEE.

[8] Tan, Y. H., & Chan, C. S. (2019). Phrase-based image caption generator with hierarchical LSTM network. Neurocomputing, 333, 86-100.

[9] Dong, J., Li, X., & Snoek, C. G. (2018). Predicting visual features from text for image and video caption retrieval. IEEE Transactions on Multimedia, 20(12), 3377-3388.

[10] Aneja, J., Deshpande, A., & Schwing, A. G. (2018). Convolutional image captioning. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 5561-5570).