2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT) | 978-1-7281-8595-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICCCNT51525.2021.9579967

IMAGE CAPTION GENERATOR USING ATTENTION MECHANISM

Vaishnavi Agrawal
Electronics and Telecommunication Engineering
College of Engineering, Pune
Pune, India
agrawalvj17.extc@coep.ac.in

Shariva Dhekane
Electronics and Telecommunication Engineering
College of Engineering, Pune
Pune, India
dhekanesv17.extc@coep.ac.in

Neha Tuniya
Electronics and Telecommunication Engineering
College of Engineering, Pune
Pune, India
nehapt17.extc@coep.ac.in

Vibha Vyas
Electronics and Telecommunication Engineering
College of Engineering, Pune
Pune, India
vsv.extc@coep.ac.in

Abstract—Image captioning is used to generate sentences describing the scene captured in the form of images. It identifies objects in the image, performs a few operations, and tries to find the salient features of the image. Once the system identifies this information, it should further generate the most relevant and brief description for the image, which should be both syntactically and semantically correct. With the advancements of Learning techniques, algorithms can generate text in the form of natural sentences that will be able to describe an image in its best form. The natural ability of humans to understand image content and generate descriptive text is a challenging task for a machine to imitate. The applications of image caption generation are extensive and significant. The task involves generating brief captions using various techniques like Natural language processing (NLP), Computer vision (CV), and Deep Learning (DL) techniques. This paper introduces a system that uses an attention mechanism alongside an encoder and a decoder to generate the captions. It uses a pre-trained Convolutional Neural Network (CNN) viz. Inception V3 to extract features of the image and then a Recurrent Neural Network (RNN) viz. GRU to generate a relevant caption. To generate captions, the proposed model uses an attention mechanism that is Bahdanau attention. MS-COCO dataset is used to train the model. The results validate the model's reasonable ability to understand images and generate text.

Keywords—Convolutional Neural Networks (CNN), Recurrent Neural Network (RNN), Attention mechanism, Deep Learning (DL), and Computer Vision (CV).

I. INTRODUCTION

Technology being an integral part of our daily lives; all devices collect data to enhance user experience. Technologies like Artificial Intelligence and Deep Learning are proliferating in many areas. The absolute fortune for the research community working on Machine learning and Artificial Intelligence is the ready availability of scalable volumes of open-source data. In this era of Artificial Intelligence, generating captions for images is crucial for many conglomerates like Google to enhance the search experience by allowing users to search by images. Social media platforms like Facebook, Twitter, Instagram, and Snapchat also employ this kind of image captioning algorithm to predict what the user might be interested in and alter that user's feed accordingly. The task of finding direct objects in the image is not very difficult. However, the task is arduous for machines to identify the salient features of the image, like children playing at a playground. The captions generated should be syntactically and semantically correct. Free accessibility to massive datasets like ImageNet, Flickr 8k, Flickr30k, and the Microsoft COCO: Common Objects in Context (MS COCO) has made research in this area more robust. Moreover, the encoder-decoder framework inspired by neural networks, i.e., Convolutional Neural Networks and Recurrent Neural Networks, has added value to the research in this area.

The critical difference between the methods involving deep learning and machine learning techniques is the approach they follow. Deep learning requires incredible computation power, high processing hardware, and a tremendous amount of memory space compared to machine learning algorithms. Neural networks are used to implement Deep Learning [9] algorithms. On the other hand, machine learning algorithms learn from data provided and apply it further. If a machine learning algorithm makes a wrong prediction, the developer must manually adjust the hyperparameters and try to make a good prediction.

Mimicking the human brain using a computer system is nowadays an ongoing trend. In accordance with this trend, computer systems should possess the ability to determine what they see and translate it into words, just like a human brain does [3]. This involves Computer Vision, i.e., the machine processing what it can see, and then involves Natural Language Processing, i.e., translating the visible scene into natural language like English. This whole conversion can be dubbed as "Image Captioning using Computer Vision and Natural Language Processing". A significant application of this can be realized on social media platforms, where posted images can be processed to generate captions. And, these captions can help monitor which user likes what kind of posts. This system can also help scene-to-text conversion for people with visual disabilities if this generated text (caption) is converted to speech.

In this paper, Section 2 describes the previous works related to this project. This section describes the shortcomings of the existing methods and the novelty of the method presented in this paper. Section 3 includes the information of various datasets and their pre-processing. Section 4 introduces the model used to generate image captions. Section 5 further elaborates on the model used in detail. Finally, the results obtained are shown in section 6 followed by the conclusion and limitations of this paper in section 7.

II. Related works

Image caption generators can be constructed using either the conventional machine learning approach or the deep learning approach.

The machine learning approach includes retrieval-based image captioning and template-based image captioning. In retrieval-based, caption for a query image is retrieved from an existing caption pool by identifying an image that is visually to the query image from the training dataset. The caption corresponding to the image closest to the query image is considered the output caption. This method generates grammatically correct captions but fails to generate an appropriate caption for images that are similar to more than one query image. In template-based image captioning, caption templates are pre-defined. It detects objects, attributes, and actions to fill the obtained data in the caption template. This method can generate grammatically correct captions but fails to generate variable length captions.

The deep learning approach includes multimodal learning-based image captioning and encoder-decoder based captioning. In multimodal learning-based image captioning a deep convolutional network is utilized for extracting features of an image and RNN along with a multimodal part is used to model word distribution of the caption. In the encoder-decoder based image captioning, visual data is encoded by deep Convolutional Neural Network (CNN) and the coded visual content is then fed into an embedding space extended by Recurrent Neural Network (RNN) hidden states which encodes textual data and generates captions. The captions obtained by this method are close to the natural language and improvement in this method is capable of generating more accurate results.

To generate more focused captions which cover all the highlights of the image, an attention mechanism is implemented. Attention mechanism emphasizes the most relevant information in given query image. This paper discusses the implementation of encoder-decoder based image caption generator which also implements attention mechanism for accurate results.

Finding the correctly suited encoder (CNN) and a correspondingly compatible and efficient decoder (RNN) from a myriad of different combinations available is a difficult task. Previously existing similar projects struggle to find the right CNN-RNN combination, or sometimes inefficiently implement attention mechanisms. This paper discusses the perfect combination that perfectly balances the trade-off between accuracy and complexity of the training model.

III. Data

A. Dataset Used

MS-COCO(Microsoft Common Objects in Context) dataset [5] was used, which contains 82,000 unique images, each of which has five captions annotated to it. Out of these, 30,000 captions that correspond to 6,000 images were used to develop this model. These images are preprocessed using a TensorFlow [7] preprocessing library.

B. Pre-processing of Data

The dataset consists of images and five captions corresponding to each image. Hence, there need to be two kinds of preprocessing: one for images and another for text(captions). In the first part, i.e., image preprocessing, the images need to be re-shaped into an input format compatible with the input of the CNN (encoder) [1]. Furthermore, the captions (text) need to be mapped to their corresponding image names. Later, the captions are broken down word by word to form a dictionary of unique words. The words in the dictionary are processed to form vectors. This process is called tokenization. Post tokenizing words from the vocabulary, the resultant vectors are padded to maintain uniformity of size amongst all the tokens(word vectors). While training, caption vectors and image features vectors are mapped to each other and trained accordingly.

IV. Methodology

To implement this, the model contains two systems. The first converts the image information into features, and the second converts the features into a meaningful English sentence (caption). The first part can be called the encoder, as it encodes image information into a feature vector. The second part will be referred to as a decoder that converts the features into a caption [2]. Fig. 1. represents this classic image caption generator flowchart.

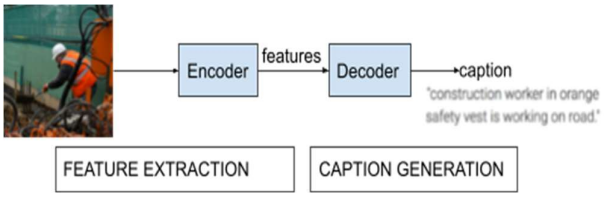

Fig. 1. Flowchart for classic image caption generator.

This method, however, is referred to as classic Image Captioning. The shortcoming of this classic image captioning method is that it does not consider the spatial features of the image and generates captions considering the whole image as a scene. Since this project attempts to mimic the human brain and its capabilities to identify and caption a scene, it was imperative to add an element that will enhance the machine's ability to focus on more important parts of the scene. For example, if an image from a scene represents a woman crossing the road, the woman and the road are more important than the blue sky or the buildings in the background. The machine needs to hence, understand which part needs more "attention" than others [4]. Hence, Fig. 1. can be modified as shown in Fig. 2.

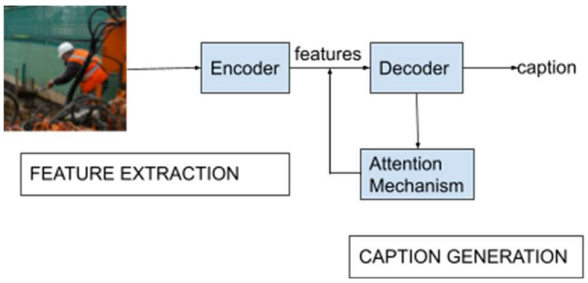

Fig. 2. Modified image caption generator

V. Model Overview

For the encoder, Convolutional Neural Networks(CNN) need to be used, as features are to be extracted from images. Pre-trained InceptionV3 [6] is used as the CNN for the encoding part. Bahdanau attention is used as the attention mechanism, and Recurrent Neural Network (RNN) is used for the decoder part. A Gated Recurrent Unit (GRU) [11] is used as the RNN for decoding.

A. Convolutional Neural Network

Inception V3 was built for object detection. Upon consuming an input image of (299x299x3), it outputs object-wise probabilities. Since InceptionV3 is an object-detection deep network, it needs to be tweaked a little to work as an encoder. To get a feature vector from this deep network, the final layer that classifies the images into objects is removed. This layer is a fully connected layer that classifies and predicts the objects. Once the fully connected layer on top is removed, a feature vector of size (8x8x2048) is obtained from InceptionV3. The internal architecture of this encoder (InceptionV3) is summarized in the table below.

In this paper, the Inception V3 model was used due to its several advantages over other models. Inception V3 is computationally less expensive than most other models. Compared to other models, this model is memory efficient, and comparatively fewer parameters are generated. Other factors also such as depth and accuracy, are better than other models. In the Inception V3 model, constraints are loose for more accessible model adaptation, optimizing overall network performance. Parallelized computations, Regularization, factorized convolutions, and dimension reduction are the key techniques used in the InceptionV3 architecture. The architecture of the proposed model is built step by step and progressively. The details of which are explained below:

- *Factorized Convolutions* – This parameter controls the network efficiency by keeping a check on it. It also reduces the number of parameters involved in the network. Thus, it helps in reducing computational efficiency.
- *Smaller Convolutions* – To increase the training speed, bigger convolutions are replaced with smaller convolutions. For example, a 5x5 filter has 25 parameters; when replaced with two 3x3 filters will have only 18 (2*3*3) parameters instead.
- *Asymmetric convolutions* – Replacing bigger convolutions with smaller convolutions every time may not reduce the parameters significantly. Thus, bigger convolutions are replaced with asymmetric convolutions. For example, replacing a 3x3 filter by two 2x2 filters will generate a slightly higher number of parameters. This shows that applying smaller convolutions techniques does not work in this case. Hence, the better solution would be to replace 3x3 convolution with a 1x3 convolution followed by a 3x1 convolution.
- *Auxiliary classifier* – This classifier is a type of CNN that is small in size. It acts as a regularizer in the network. It is between layers during training, and the loss incurred is added to the main network loss.
- *Grid Size reduction* – Pooling operations are generally used to reduce the Grid size.

After consolidating all the above mentioned concepts, the final architecture adopted is as represented by Fig. 3.

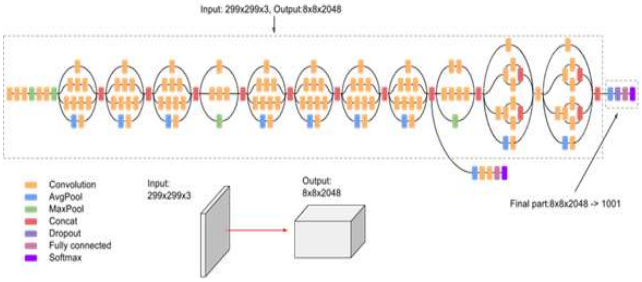

Fig. 3. Convolutional Neural Network architecture

The feature vector extracted from this CNN will be fed to a Recurrent Neural Network (RNN) in case of classic image captioning. The words will be predicted according to the features that the RNN receives. However, here, a visual attention mechanism is being proposed.

B. Attention Mechanism

The attention mechanism takes as input the features from CNN and a the output of RNN from the previous time step. Here, the Bahdanau attention [8] mechanism is used. Its architecture is represented by Fig. 4.

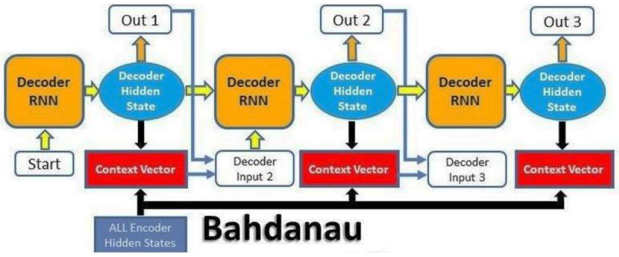

Fig. 4. Attention model

In the attention unit, the first step is to calculate alignment scores. The alignment scores are numerical measures of the amount of attention that the decoder pays to the features when the next output is being produced. The output of the decoder, i.e., the hidden state in the previous time step and the features are passed through a linear layer to obtain $W_1 h_1$ and $W_2 h_2$, respectively. A tan hyperbolic activation is then applied to the summation of the terms obtained above. As shown in (1), the score is finally calculated.

$$\text{score} = \tanh(W_1 h_1 + W_2 h_2) \quad (1)$$

Using these scores calculated in (1), the attention weights (α) are obtained using (2).

$$\alpha = \text{softmax}(\text{score}) \quad (2)$$

Next, using (3), the context vector (c) is calculated using the attention weights (α) from (2) and features (h_2) that were obtained from the encoder.

$$c = \alpha h_2 \quad (3)$$

Finally, the context vector (c) and the decoder output from previous time step (h_1) are fused together and fed to the RNN cell to produce the decoder output for the current time step.

C. Recurrent Neural Network

The decoder mechanism is an RNN that predicts words in natural language. The (Recurrent Neural Networks) RNNs typically used are Vanilla RNN, Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU) [11]. Vanilla RNN is outright rejected as it exhibits the vanishing gradient problem, i.e., the words that were predicted first are forgotten as and when the next words are predicted. Hence the first words finally get very low weightage in the generated output. To deal with this issue, LSTM (Long Short Term Memory) [10] was introduced. GRU is similar to LSTM but is less complex in cell architecture. Hence, GRU was used to implement this caption generator.

Fig. 5. Shows the working of a Gated Recurrent Unit (GRU). Here, x[t] represents input vector, z[t] is the update gate vector, h[t-1] represents previous output, h[t] represents current output, r[t] is the reset gate vector and $\hat{h}[t]$ represents activation vector. The σ operator represents the sigmoid function, and tanh represents the tan hyperbolic operation. The final output of GRU is h[t] which is given by (4).

$$h[t] = (1 - z[t])(h[t-1])(\hat{h}[t]) \quad (4)$$

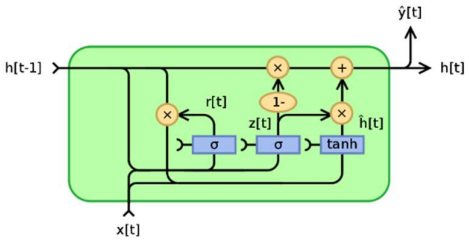

Fig. 5. Gated Recurrent Unit architecture

D. Overall architecture:

In Fig. 6., y_n represent encoder outputs (features), C represents the context vector, z_k represents attention weights, and h represents the previous decoder hidden states.

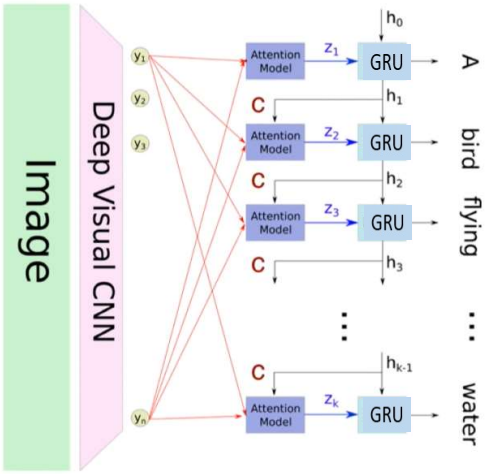

Fig. 6. Overall model architecture

VI. Results

A. Image 1:

Fig. 7. An image under scrutiny

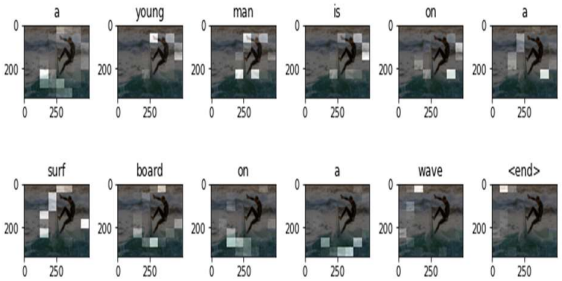

Fig. 8. Attention map generated by attention layer for test image 1 (Fig. 7.)

Predicted caption– a young man is on a surf board on a wave <end>

B. Image 2:

Fig. 9. An image under scrutiny

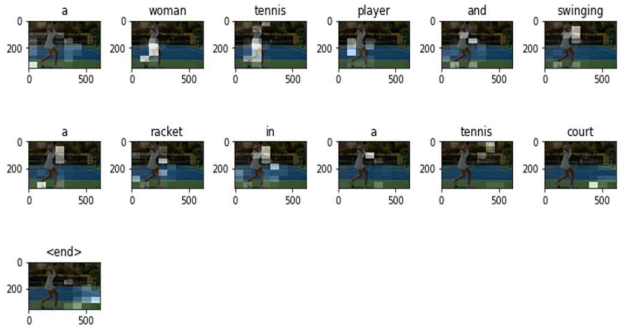

Fig. 10. Attention map generated by attention layer for test image 2 (Fig. 9.)

Predicted caption– a woman tennis player and swinging a racket in a tennis court <end>

C. Image 3:

Fig. 11. An image under scrutiny

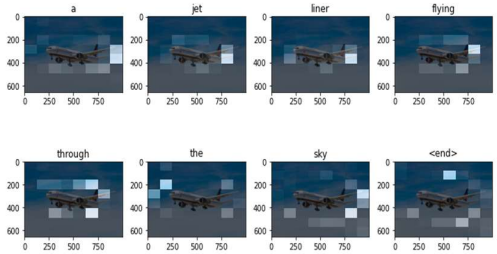

Fig. 12. Attention map generated by attention layer for test image 3 (Fig. 11.)

Predicted caption– a jetliner flying through the sky <end>

D. Image 4:

Fig. 13. An image under scrutiny

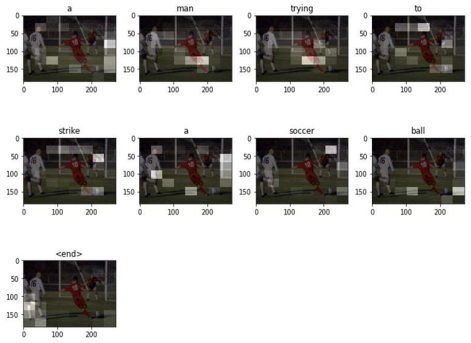

Fig. 14. Attention map generated by attention layer for test image 4 (Fig. 13.)

Predicted caption– a man trying to strike a soccer ball <end>

E. Loss versus Epochs:

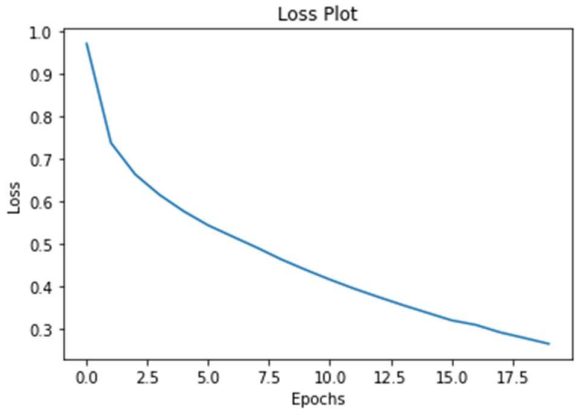

Fig. 15. Loss versus Epoch plot to justify the number of epochs

VII. Conclusion

An image caption generator based on the encoder and attention-based decoder is proposed in this paper. The attention mechanism introduced after the convolutional neural network encoder makes the model pay attention to the most relevant information in the input scene image so that the decoder only uses specific parts of the image to generate the caption. This improves the caption as compared to a conventional encoder-decoder-based model. The obtained results are promising, and the captions generated by the model is intelligible.

However, since the training data and the corresponding vocabulary was limited, the model might fall short of identifying objects from input scenes that do not feature in its vocabulary. Such words are replaced with an unknown tag <unk>. The model might not perform well if the input query image consists of more than one such unknown objects. A caption hence generated may or may not make sense. Example of such a caption: A <unk> at a <unk>.

Acknowledgment

Microsoft Common Objects in Context, thanks for making this colossal dataset public. Extending gratitude to the Center of Excellence in Signal and Image Processing, Electronics and Telecommunication Engineering Department at the College of Engineering, Pune.

REFERENCES

[1] Ansar Hani, Najiba Tagougui, Monji Kherallah, “Image caption generation using a deep architecture”, International Arab Conference on Information Technology ACIT, 2019.

[2] Himanshu Sharma, Manmohan Agrahari, Sujeet Kumar Singh, Mohd Firoj, Ravi Kumar Mishra, “Image captioning: a aomprehensive survey”, International Conference on Power Electronics and IoT Applications in Renewable Energy and its Control PARC, 2020.

[3] Adela Puscasiu, Alexandra Fanca, Dan-Ioan Gota, Honoriu Valean, “Automated image captioning”, IEEE International Conference on Automation, Quality and Testing, Robotics AQTR, 2020.

[4] Phyu Phyu Khaing, May The` Yu, “Attention-based deep learning model for image captioning: a comparative study”, International Journal of Image, Graphics and Signal Processing, June 2019.

[5] O. Vinyals, A. Toshev, S. Bengio et al., "Show and tell: Lessons learned from the 2015 MSCOCO image captioning challenge", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 4, 2017.

[6] Asifullah Khan, Anabia Sohail, Umme Zahoora, Aqsa Saeed Qureshi, “A survey of the recent architectures of deep convolutional neural networks”, Artificial Intelligence Review, DOI, 2020.

[7] Jojo John Moolayil, Learn Keras for Deep Neural Networks: A Fast-Track Approach to Modern Deep Learning with Python, 1st edn, Apress, New York, 2019.

[8] Shailja Gupta, Manpreet Kaur, Sachin Lakra, Mayank Khattar, “Application-based attention mechanisms in deep learning”, International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) ICRITO, September 2020.

[9] Murk Chohan , Adil Khan , Muhammad Saleem Mahar, Saif Hassan , Abdul Ghafoor, Mehmood Khan, “Image captioning using deep learning: a systematic literature review”, International Journal of Advanced Computer Science and Applications, Vol. 11, No. 5, 2020.

[10] Yong Yu , Xiaosheng Si , Changhua Hu , Jianxun Zhang, “A review of recurrent neural networks: LSTM cells and network architectures”, Neural Computation, Volume 31, Issue 7, July 2019.

[11] Kanchan M. Tarwani, Swathi Edem, “Survey on recurrent neural network in natural language processing”, International Journal of Engineering Trends and Technologies, IJETT, Volume 48, Number 6, June 2017.