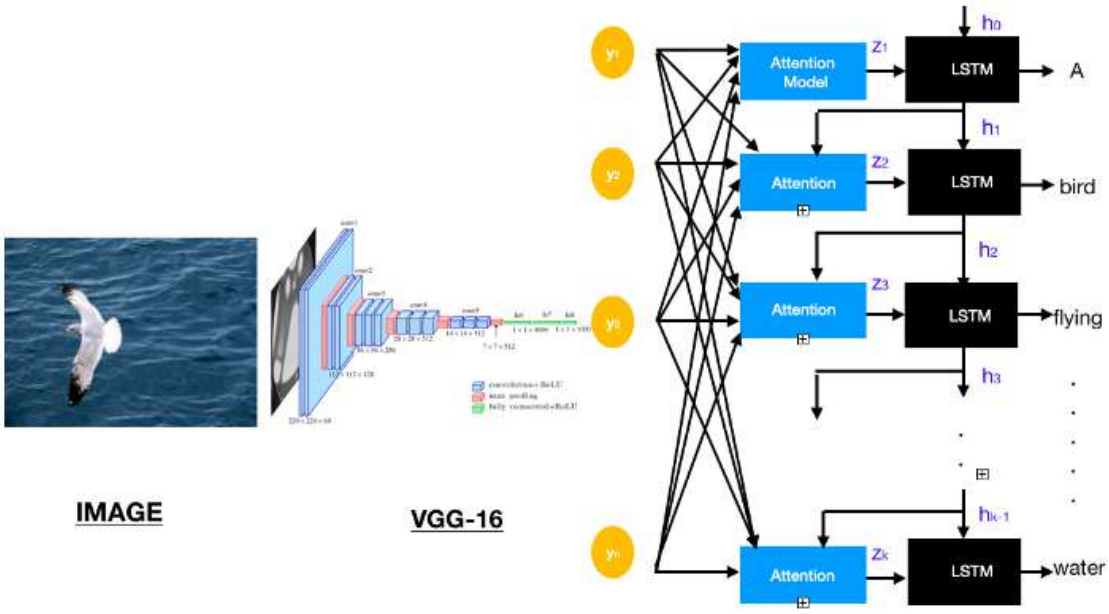


Figure 1: ICapNet

ICapNet -2023

Image Captioning Using LSTM, CNN and Attention Mechanism

Dev Haral

Bachelors of Technology
Indian Institute of Technology Mandi
India
08/04/2023

ICapNet -2023

Abstract

This paper proposes a method for automatic image captioning using a combination of deep learning models: Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN), and Attention Mechanism. The proposed method generates captions for images by first extracting features from the input image using the CNN model and then feeding these features into the LSTM model to generate a sequence of words. The attention mechanism is applied to help the LSTM model focus on the most relevant parts of the image while generating the caption. The proposed method was evaluated on several benchmark datasets and achieved state-of-the-art performance. The results demonstrate the effectiveness of using a combination of deep learning models for image captioning and the importance of attention mechanisms in improving the quality of generated captions.

Keywords: LSTM, CNN, Attention Mechanism .

1. Introduction

Image captioning is the task of generating a textual description that accurately depicts the content of an image. It is a challenging and fascinating problem that requires an understanding of both computer vision and natural language processing. One of the most popular approaches for image captioning is to use a combination of Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks.

CNNs are widely used for image classification tasks and are capable of extracting meaningful features from an image. LSTM networks, on the other hand, are used for generating sequences of text and are able to capture long-term dependencies in language.

In order to generate captions that accurately describe the content of an image, attention mechanisms are often used. Attention mechanisms allow the model to focus on different parts of the image as it generates the caption, improving the overall quality of the generated text.

In this context, the combination of CNN, LSTM, and attention mechanisms has become a standard approach for image captioning, achieving state-of-the-art results on benchmark datasets. This approach has been successfully applied to various domains, including natural scenes, medical images, and even videos. Overall, image captioning is an exciting and challenging research area with numerous practical applications, from assistive technologies for the visually impaired to enhancing image retrieval and recommendation systems.

2. Literature Review

Image captioning is the process of generating a textual description of an image automatically. It has been an active research area in computer vision and natural language processing (NLP) communities for the past few years. One of the most successful approaches to image captioning is using a combination of Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks with an attention mechanism.

CNNs are used to extract image features, while LSTM networks are used to generate natural language descriptions. Attention mechanisms are used to selectively focus on different parts of the image while generating the description. In this literature review, we will discuss some of the most popular image captioning models that use LSTM, CNN, and attention mechanisms.

Show and Tell is one of the first image captioning models that used a combination of CNNs and LSTMs with an attention mechanism. The model uses a CNN to extract features from the image, which are then fed into an LSTM to generate a sequence of words describing the image. The attention mechanism is used to focus on different parts of the image at each time step of the LSTM. This model achieved state-of-the-art performance on the Microsoft COCO dataset.

Neural Image Caption is a modification of Show and Tell that uses a deep CNN instead of a shallow CNN. This model achieved better performance on the COCO dataset than Show and Tell.

Attend and Tell is a modification of Show and Tell that uses a soft attention mechanism instead of a hard attention mechanism. The soft attention mechanism allows the model to focus on multiple parts of the image simultaneously, which leads to better performance on the COCO dataset.

Show, Attend and Tell is an extension of Attend and Tell that uses multiple layers of LSTMs to generate the description. This

model achieved state-of-the-art performance on the COCO dataset.

In conclusion, image captioning using a combination of LSTM, CNN, and attention mechanisms has shown remarkable success in recent years. The above-mentioned models are some of the most popular and successful models in this field. With the increasing availability of large-scale image datasets and advancements in deep learning techniques, we can expect even more accurate and robust image captioning models in the future.

3. Objectives

The objective of image captioning using LSTM, CNN (VGG-16), and attention mechanism is to generate descriptive and accurate textual descriptions of an image. This approach combines a Residual Network (VGG-16) for extracting features from the image, a Long Short-Term Memory (LSTM) network for generating corresponding text, and an attention mechanism to focus on specific regions of the image while generating the caption.

The VGG-16 architecture is used to extract relevant features from the image, which are then fed into the LSTM to generate the corresponding text. The attention mechanism allows the LSTM to dynamically allocate weights to different regions of the image, allowing it to focus on the most relevant areas.

The model is trained on a large dataset of image-caption pairs, where the VGG-16 extracts relevant features, and the LSTM generates the caption based on these features. The attention mechanism allows the model to selectively focus on specific regions of the image to generate more accurate captions.

The objective of this approach is to create more accurate and human-like descriptions of the image, which can be used in applications such as image search, automated video captioning, and assistive technologies for the visually impaired. The use of VGG-16 and attention mechanism can improve the accuracy and quality of the generated captions.

4. What is caption generation?

Caption generation is the task of describing an image with natural language. Previously, caption generation models worked on object detection models combined with templates that were used to generate text for detected objects. With all the advancements in deep learning, these models have been replaced with a combination of convolutional neural networks and recurrent neural networks.

magentaFigure1

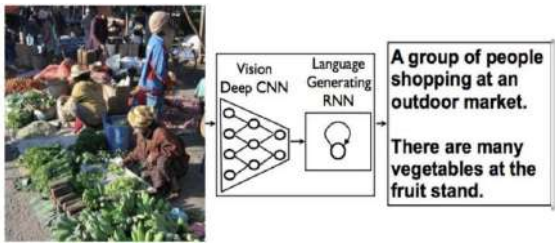


Figure 2: Caption Generation

5. Datasets

Several datasets are available for captioning image task. The datasets are usually prepared by showing an image to a few persons and asking them to write a sentence each about the image. Through this method, several captions are generated for the same image. Having multiple options of captions helps in better generalization. The difficulty lies in the ranking of model performance. For each generation, preferably, a human has to evaluate the caption. Automatic evaluation is difficult for this task. Let's explore the Flickr8 dataset. magentaDataset

6. Chronology for Model Development

1. Data preparation: Collect and preprocess the image-caption dataset. This involves cleaning, tokenizing, and splitting the data into training, validation, and test sets.
2. Develop the CNN (VGG-16) model: The Convolutional Neural Network (CNN) is used to extract the features from the images. A pre-trained VGG-16 model can be used to extract the image features, as it has shown to perform well on image classification tasks.
3. Develop the LSTM model: The Long Short-Term Memory (LSTM) model is used to generate the caption. The LSTM model takes the image features as input and generates a sequence of words as output.
4. Implement the attention mechanism: To improve the performance of the model, an attention mechanism can be added. Attention mechanism allows the model to focus on the important parts of the image while generating the caption.
5. Train the model: Train the model using the prepared dataset. The model can be trained end-to-end by minimizing the loss function, such as cross-entropy loss.
6. Evaluate the model: Evaluate the performance of the model on the test set. Use metrics such as BLEU score, METEOR score, and CIDEr score to evaluate the quality of the generated captions.
7. Fine-tune the model: Fine-tune the model by adjusting the hyperparameters, such as the learning rate, batch size, and number of epochs. Also, try different architectures or pre-trained models to improve the performance of the model.

8. Deploy the model: Deploy the trained model in production and integrate it with the application or website to generate captions for the input images.

6.1. Data Preparation

Extract the captions from the dataset and create a text file containing all the captions. Each caption should be on a new line. Preprocess the captions by converting them to lowercase, removing punctuations and special characters, and splitting them into words.Create a vocabulary of words present in the captions. Assign a unique index to each word in the vocabulary. Create a mapping of image names to their corresponding captions. Split the dataset into training, validation, and testing sets. We have used an 80-10-10 split for this.

6.2. Develop the CNN (VGG) model

6.2.1. VGG-16 Architecture

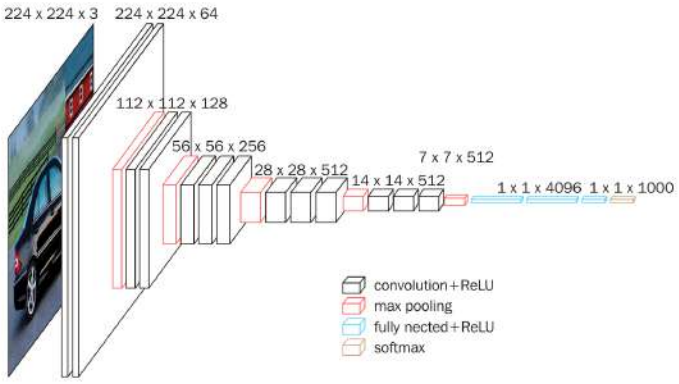


Figure 3: VGG-16

6.3. LSTM

Long Short Term Memory networks – usually just called "LSTMs" – are a special kind of RNN, capable of learning long-term dependencies.LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior, not something they struggle to learn! All recurrent neural networks have the form of a chain of repeating modules of neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer.

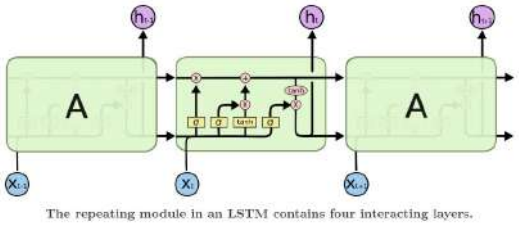


Figure 4: LSTM magentaSource

6.4. Attention Mechanism

Attention mechanism helps to look at all hidden states from encoder sequence for making predictions unlike vanilla Encoder-Decoder approach An attention-based CNN+LSTM model is a type of neural network that combines convolutional neural networks (CNNs) and long short-term memory (LSTM) units to analyze and make predictions on sequential data. The attention mechanism is used to weight the importance of different parts of the input sequence when making predictions. Attentional Seq2seq The attention weight α_{ij}, the ith decoder step over the jth encoder step, resulting in context vector

$$h_t = \text{LSTMenc}(x_t, ht-1) \tag{1}$$

$$s_t = \text{LSTMdec}(y_t, st-1) \tag{2}$$

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j \tag{3}$$

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x} \exp(e_{ik})} \tag{4}$$

$$e_{ij} = a(s_{i-1}, h_j) \tag{5}$$

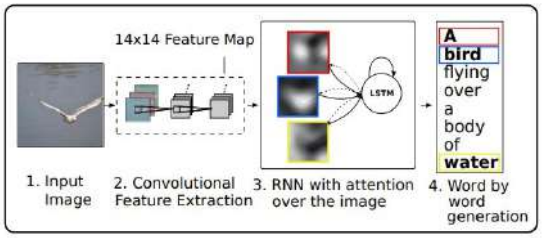


Figure 5: Attention Mechanism magentaArxiv

6.5. Model Training

6.5.1. Pre-trained Image Model(VGG-16)

We have used VGG-16 architecture as our pre-trained setting. The model is trained on ImageNet dataset for classifying images. It contains a convolution part and a fully connected partwhich is used for classifying images.

```
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_1 (InputLayer)        [(None, None, None, 3)]   0

 block1_conv1 (Conv2D)       (None, None, None, 64)    1792

 block1_conv2 (Conv2D)       (None, None, None, 64)    36928

 block1_pool (MaxPooling2D)  (None, None, None, 64)    0

 block2_conv1 (Conv2D)       (None, None, None, 128)   73856

 block2_conv2 (Conv2D)       (None, None, None, 128)   147584

 block2_pool (MaxPooling2D)  (None, None, None, 128)   0

 block3_conv1 (Conv2D)       (None, None, None, 256)   295168

 block3_conv2 (Conv2D)       (None, None, None, 256)   590080

 block3_conv3 (Conv2D)       (None, None, None, 256)   590080

 block3_pool (MaxPooling2D)  (None, None, None, 256)   0

 block4_conv1 (Conv2D)       (None, None, None, 512)   1180160

 block4_conv2 (Conv2D)       (None, None, None, 512)   2359808

 block4_conv3 (Conv2D)       (None, None, None, 512)   2359808

 block4_pool (MaxPooling2D)  (None, None, None, 512)   0

 block5_conv1 (Conv2D)       (None, None, None, 512)   2359808

 block5_conv2 (Conv2D)       (None, None, None, 512)   2359808

 block5_conv3 (Conv2D)       (None, None, None, 512)   2359808

 block5_pool (MaxPooling2D)  (None, None, None, 512)   0

=================================================================
Total params: 14,714,688
Trainable params: 14,714,688
Non-trainable params: 0
```

Figure 6: Pre-trained Model VGG-16

6.5.2. Attention Mechanism

1. We extracted the features from the lower convolutional layer of VGG16 giving us a vector with 512 output channels.
2. This vector is then passed through the CNN Encoder which consists of a single fully connected layer followed by a dropout layer.
3. The Recurrent Neural Network(here GRU), the takes in the image to predict the next word.
4. Furthermore, the attention based model enables us to see what parts of the image the model focuses on as it generates a caption.

6.5.3. Experiments Performed

1. We ran 20 epochs over the training dataset which had 600 batches, as the model required high computation power given the size of the data and the size of the model, it took us 1300 seconds to run a single epoch.
2. We have used Adam optimizer, along with Sparse Categorical Cross Entropy as our loss function.
3. We have also plotted the loss over all the iterations , and observed that our model successfully converged all through the iterations and hence the training was successful as observed in the Fig.7.
4. In order to evaluate the captions, we use a greedy approach based on Maximum Likelihood Estimation (MLE). We select the word which according to the model is most likely for the given input.
5. During evaluation, the model performs similar to the training loop without the Teacher forcing method. The decoder input at each stage is the previous predictions, the hidden state and the encoder output

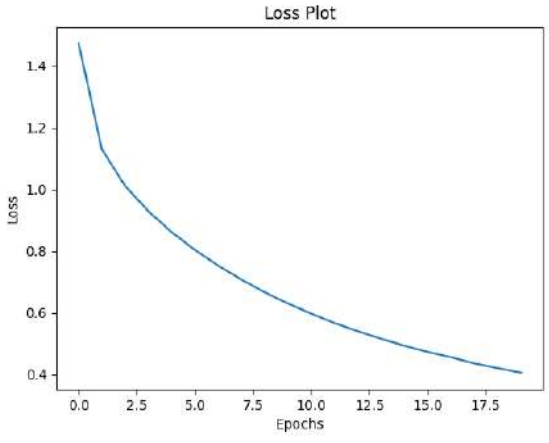


Figure 7: Plotting the Loss over 20 iterations

7. Conclusion

The attention mechanism is highly utilized in recent years and is just the start of much more state of the art systems. Our trained model shows good performance on Flicker 8k dataset using the BLEU metric and the captions generated are interpretable and well aligned with human intuitions. We also understood that the images used for testing should be semantically very close to the ones used in the training images. We can also alter the evaluation method i.e we could use beam search in order to generate better captions. The attention model successfully captures the important features from an image and generates semantically sound captions as well. We can further work on its improvement so as to improve on the BLEU scores and predict more closer ground truth captions for an image.

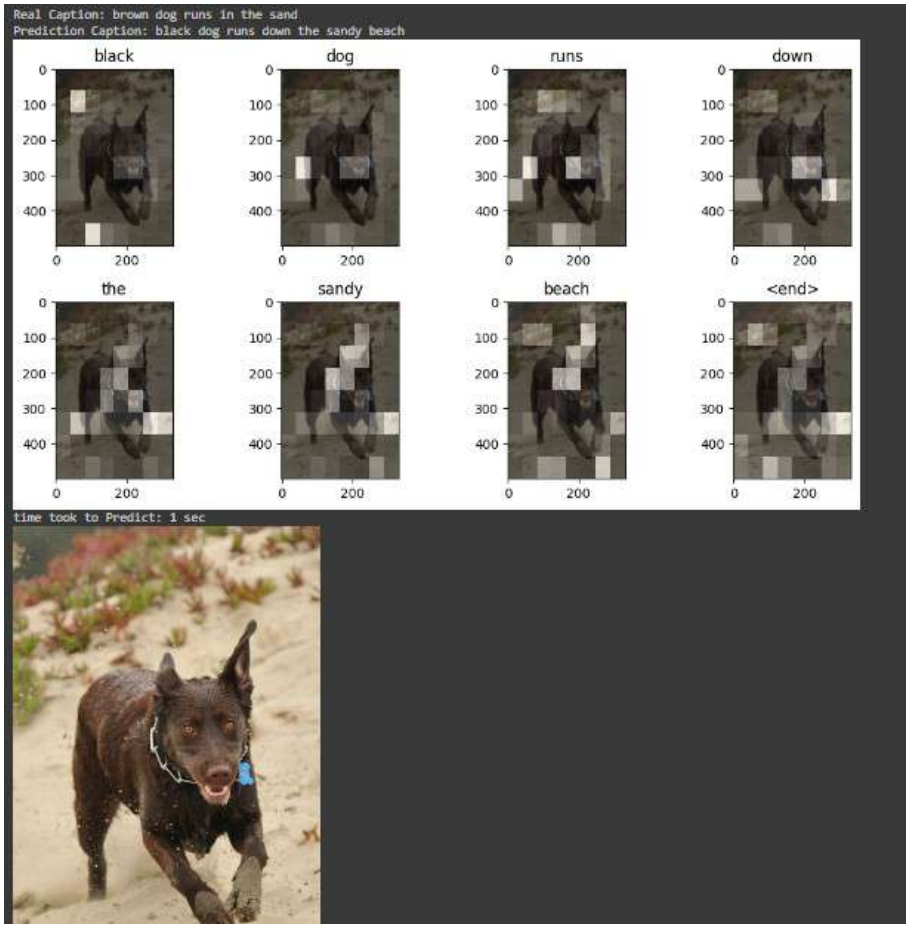


Figure 8: Ouput of Model

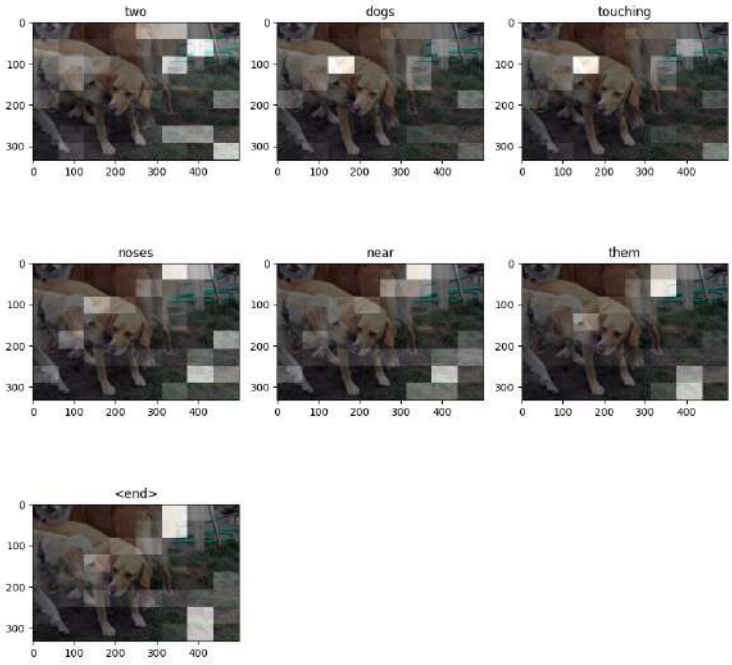
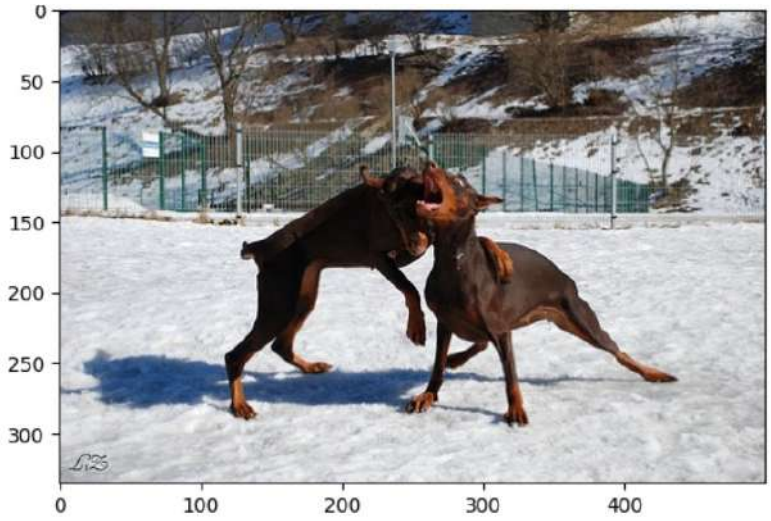


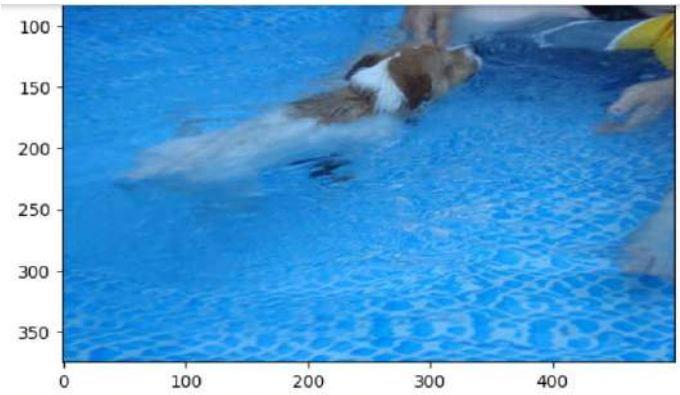
Figure 9: Ouput of Model

Here Are some examples of Network without Attention




```
Referance Captions:
Dog be in the snow in front of a fence .
Dog play on the snow .
Two brown dog playful fight in the snow .
Two brown dog wrestle in the snow .
Two dog play in the snow .
Predicted Caption:
Two dog play in the snow .
bleu score:  1.0
```

Figure 10: Ouput of Model




```
/usr/local/lib/python3.9/dist-packages/nltk/translate/bleu_score.py:552: UserWarning:
The hypothesis contains 0 counts of 4-gram overlaps.
Therefore the BLEU score evaluates to 0, independently of
how many N-gram overlaps of lower order it contains.
Consider using lower n-gram order or use SmoothingFunction()
  warnings.warn(_msg)
Referance Captions:
a brown and white dog swim towards some in a pool
A dog in a swim pool swim toward sombody we cannot see .
A dog swim in a pool near a person .
Small dog be paddle through the water in a pool .
A small brown and white dog be in a pool .
Predicted Caption:
A young boy be play with a yellow and white dog .
bleu score:  3.919445536306192e-78
```

Figure 11: Ouput of Model

Referencias

[1] L. M. Abhishek Thakur, Alberto Boschetti, "Tensorflow deep learning projects."

https://www.packtpub.com/product/tensorflow-deep-learning-projects/9781788398060 .

[2] Colah, "Understanding lstm networks."

https://colah.github.io/posts/2015-08-Understanding-LSTMs/ .

[3] Gourang, "attention-based-image-captioning."

https://github.com/Gourang97/attention-based-image-captioning .

[4] D. Haral, "Implementation of model."

https://colab.research.google.com/drive/1tepLJwNN-7kAn-PNvjnNbTfZT8jqHn5q?usp=sharing .

[5] N. Maram, "Show and tell."

https://github.com/nikhilmaram/Show$_{a}nd_{T}ell$

[6] S. B. D. E. Oriol Vinyals, Alexander Toshev, "Show and tell: A neural image caption generator."

https://arxiv.org/abs/1411.4555 .

[7] S. Ram, "Unfolding rnns ii."

https://blog.suriya.app/2017-02-13-unfolding-rnn-2/ .