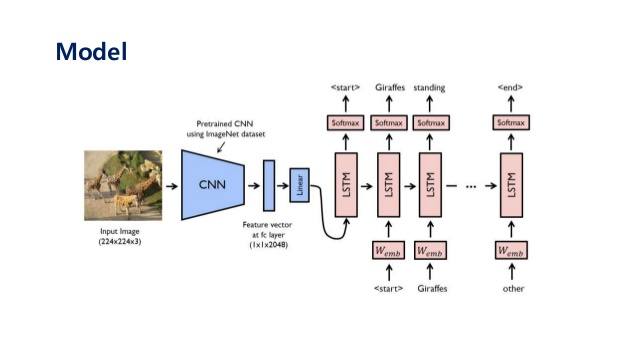
Image Caption Generator

using CNN and LSTM



|  |  |
| --- | --- |
| Name | PRN Number |
| Gopal Gupta | 22030242019 |
| Shankhini Chaudhury | 22030242047 |
| Vedant Shrivastava | 22030242049 |
| Saloni Jain | 22030242074 |
| Sameer Roongta | 22030242075 |

Table of contents

|  |  |
| --- | --- |
| No. | Particulars |
|  | Objectives |
|  | Introduction |
|  | Importing libraries |
|  | Understanding of code |
|  | Usage of CNN and LSTM |
|  | Conclusion |

Objectives

The objective of the image generator is to predict the captions for the input image. The dataset consists of 8k images and 5 captions for each image. The features are extracted from both the image and the text captions for input. The features will be concatenated to predict the next word of the caption. CNN is used for image and LSTM is used for text. BLEU Score is used as a metric to evaluate the performance of the trained model.

Introduction

Training computers to be able to automatically generate descriptive captions for images is currently a very hot topic in Computer Vision and Machine Learning. This task is a combination of image scene understanding, feature extraction, and translation of visual representations into natural languages.

It involves both computer vision technologies to comprehend the image’s content and a language models from the field of natural language processing to converts the image’s comprehension into words in the correct order

Importing libraries

* **NumPy:**

is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python. It is open-source software.

* **TQDM-**

tqdm is a library in Python which is **used for creating Progress Meters or Progress Bars**. tqdm got its name from the Arabic name taqaddum which means 'progress'.

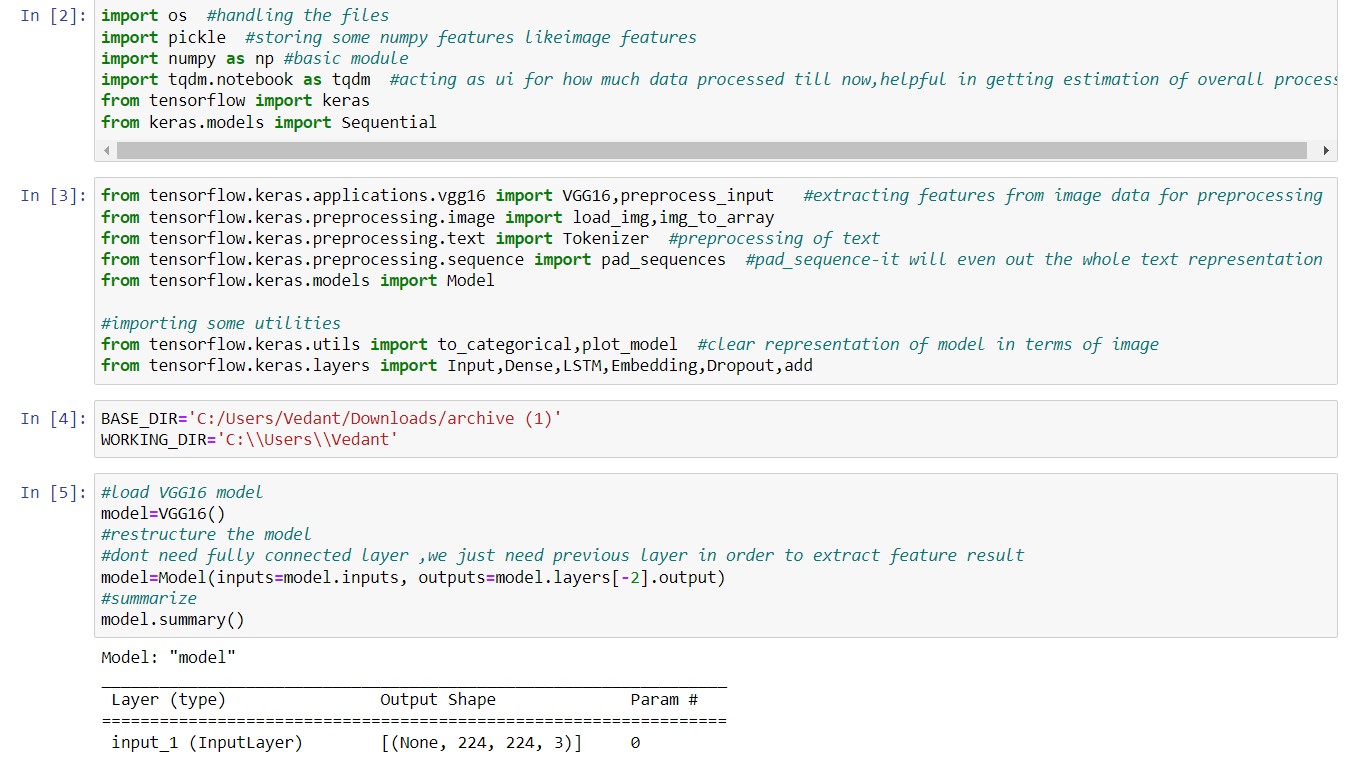
* **Tensorflow:**

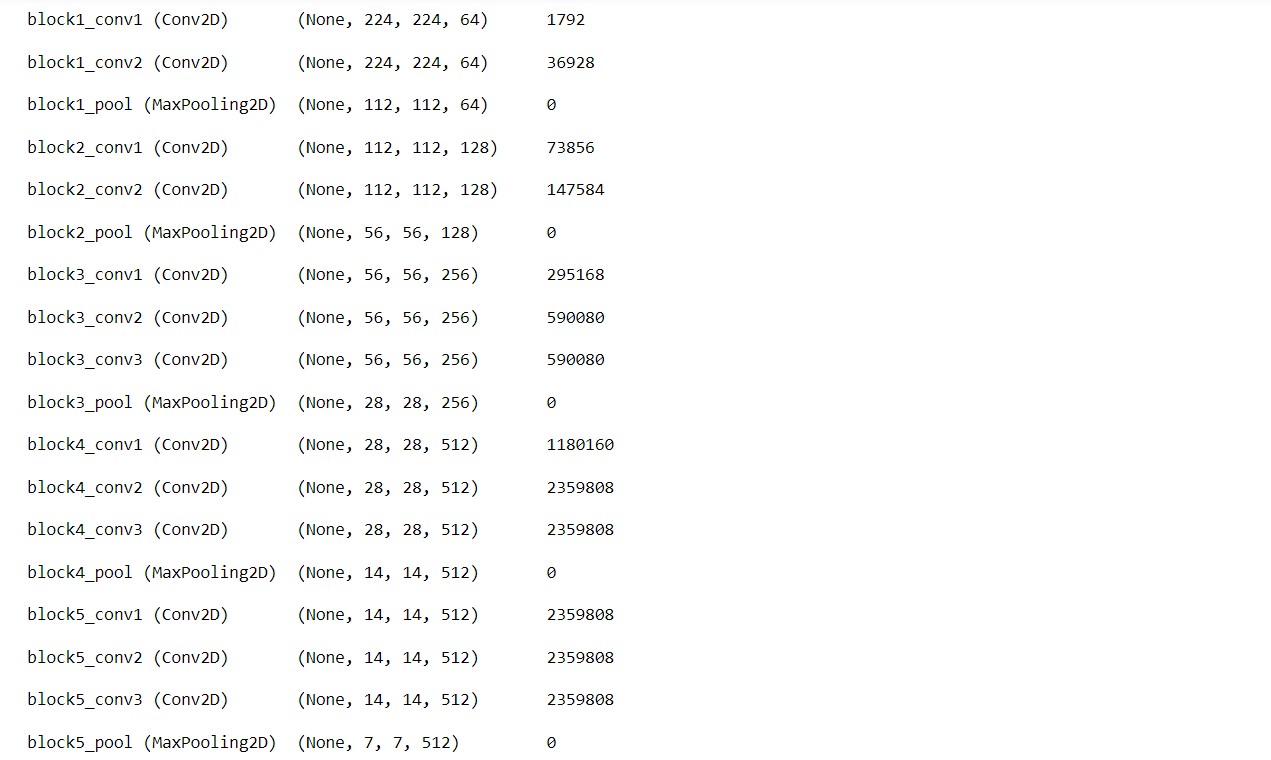
The TensorFlow platform **helps you implement best practices for data automation, model tracking, performance monitoring, and model retraining**. Using production-level tools to automate and track model training over the lifetime of a product, service, or business process is critical to success.

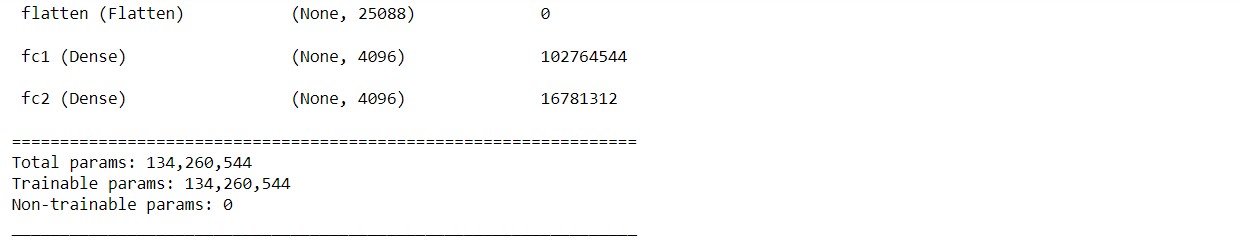
* **Keras:**

Keras is **a high-level, deep learning API developed by Google for implementing neural networks**. It is written in Python and is used to make the implementation of neural networks easy. It also supports multiple backend neural network computation.

Understanding of code

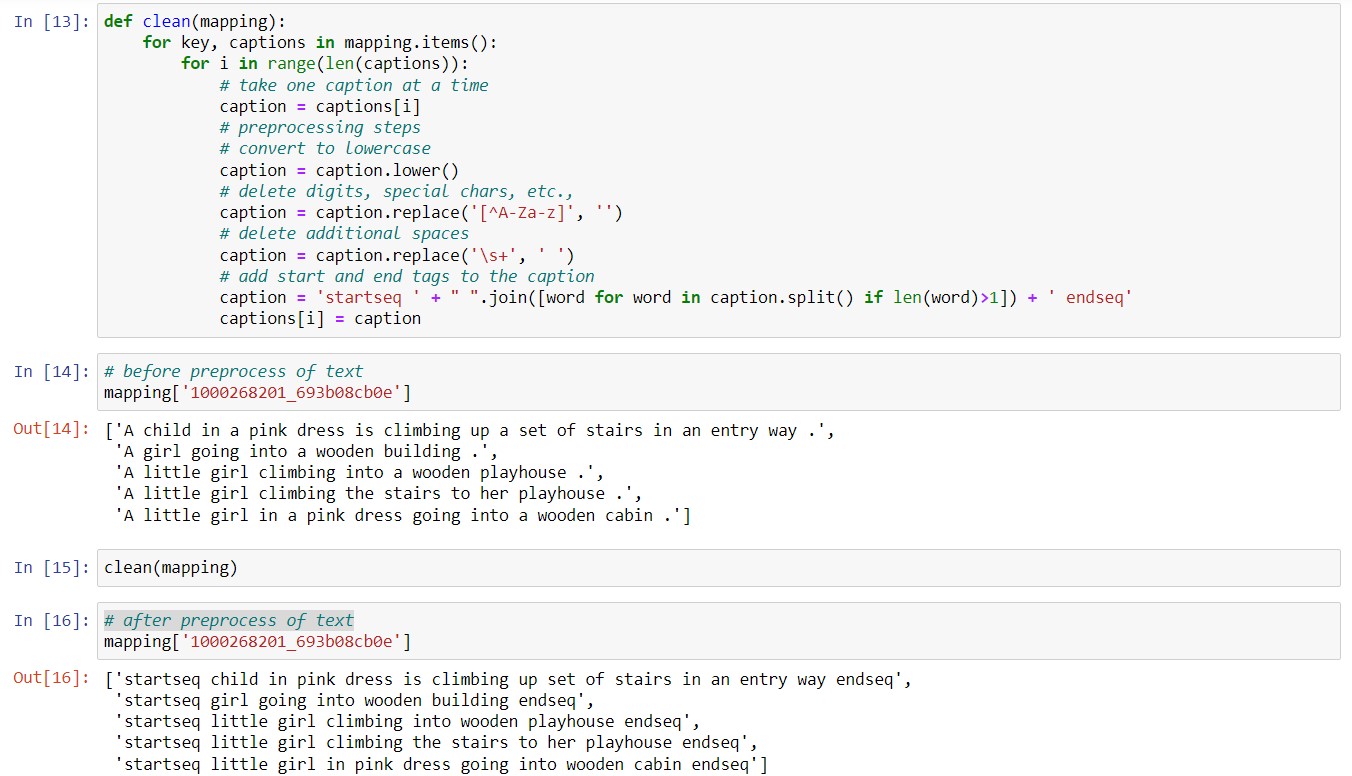


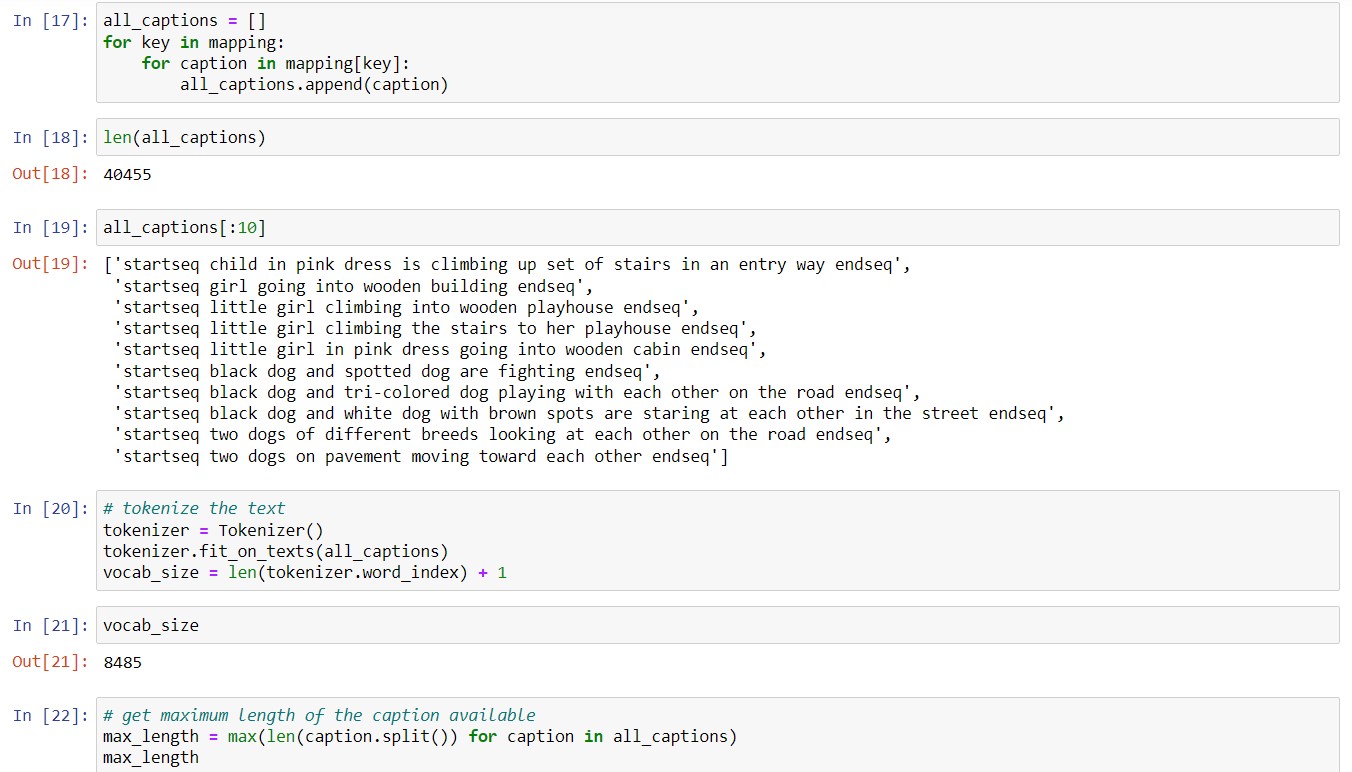


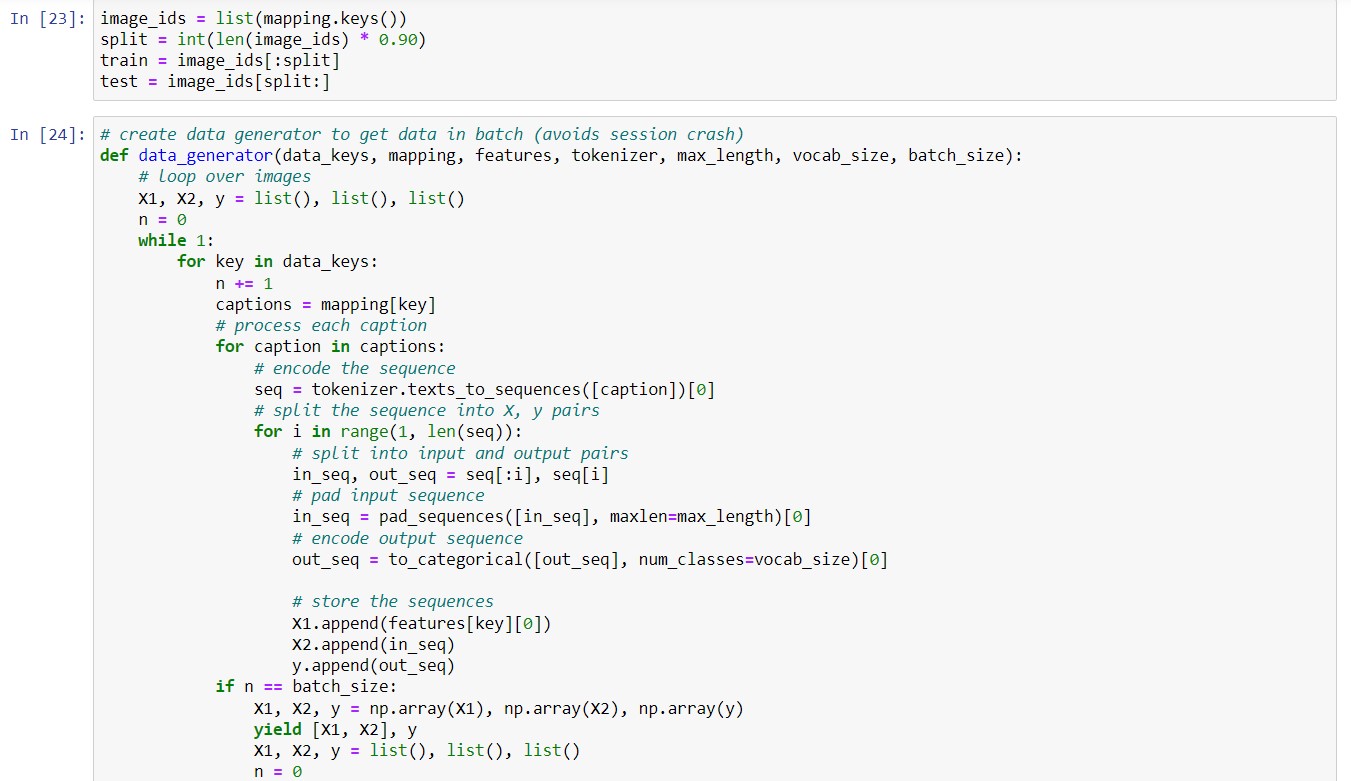




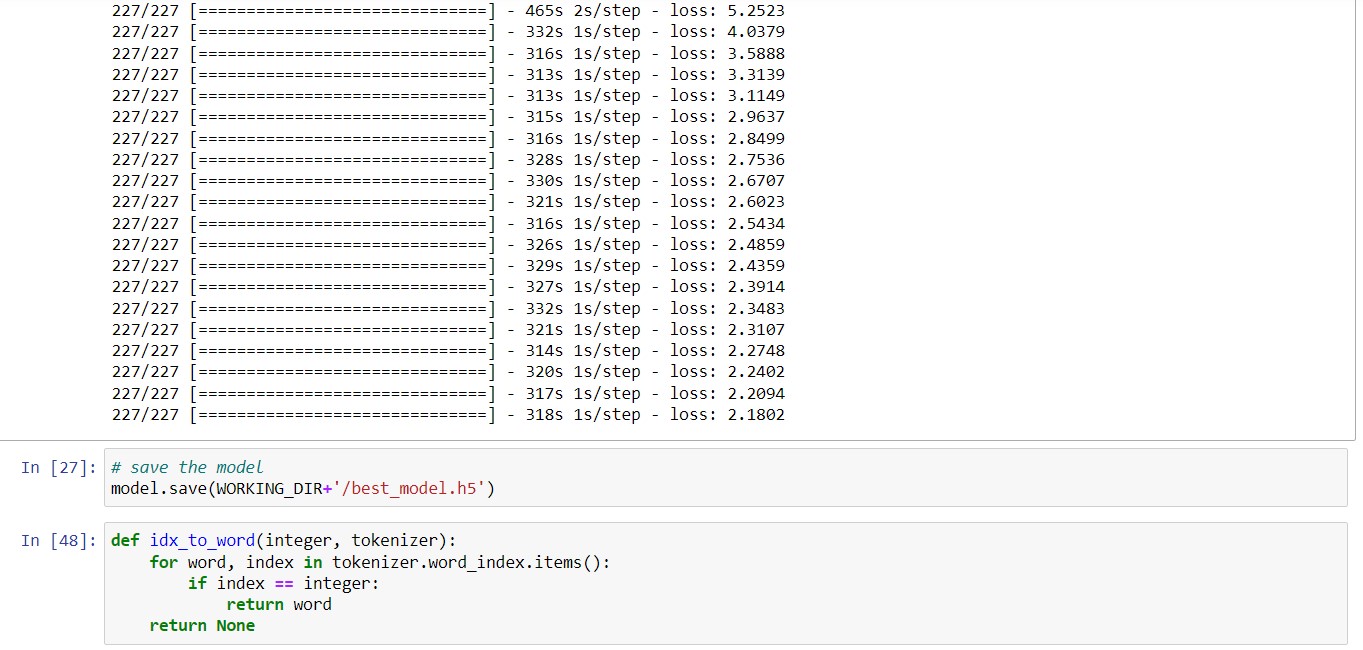


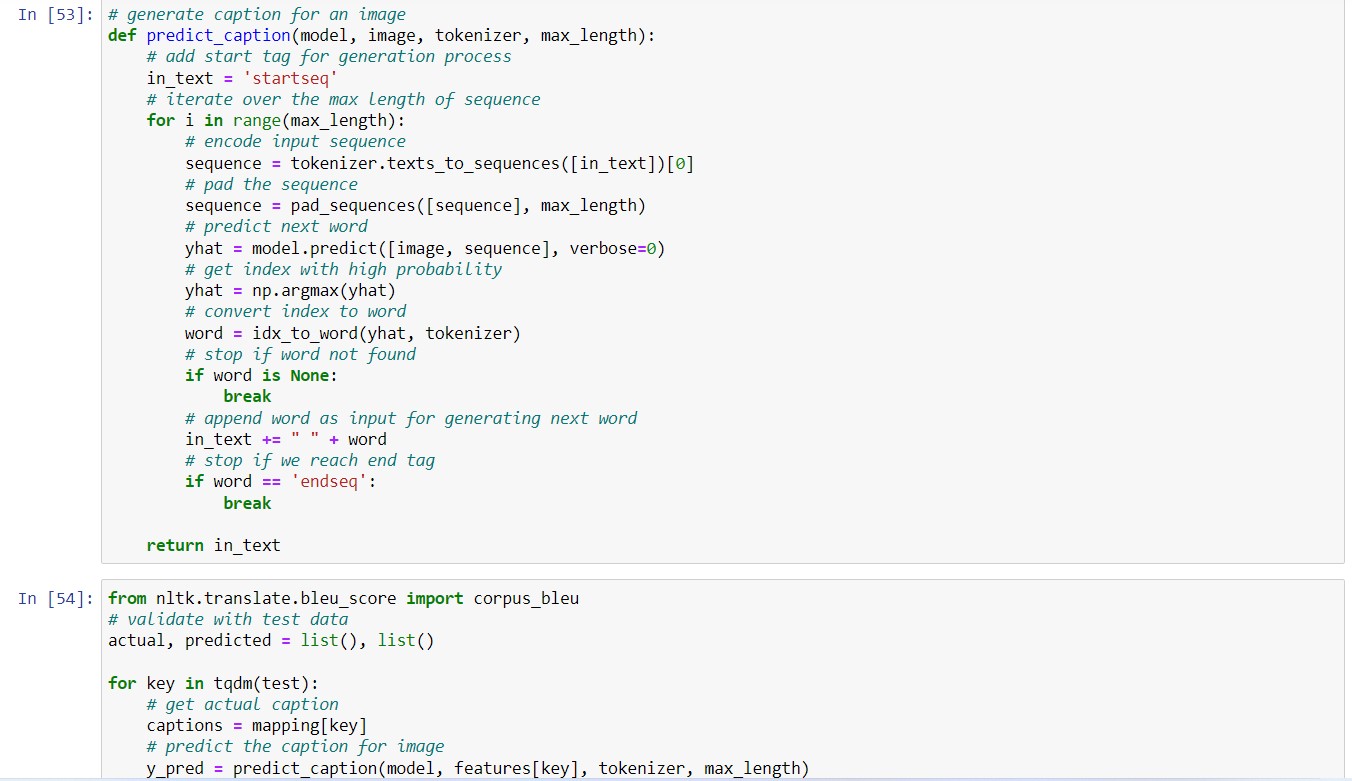


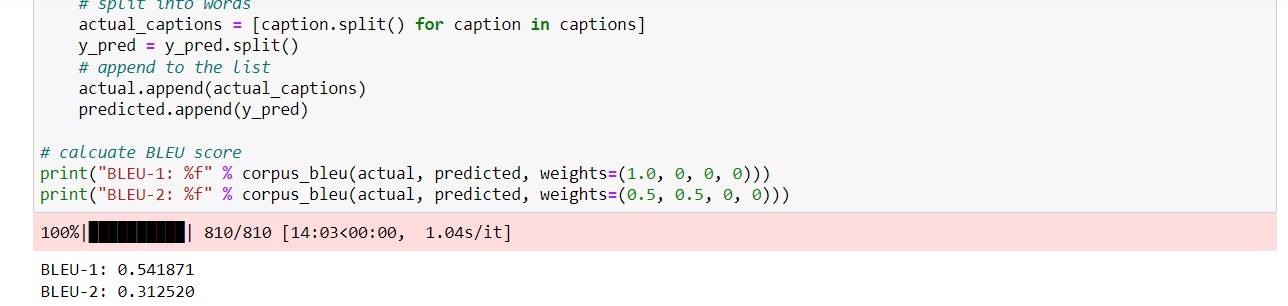




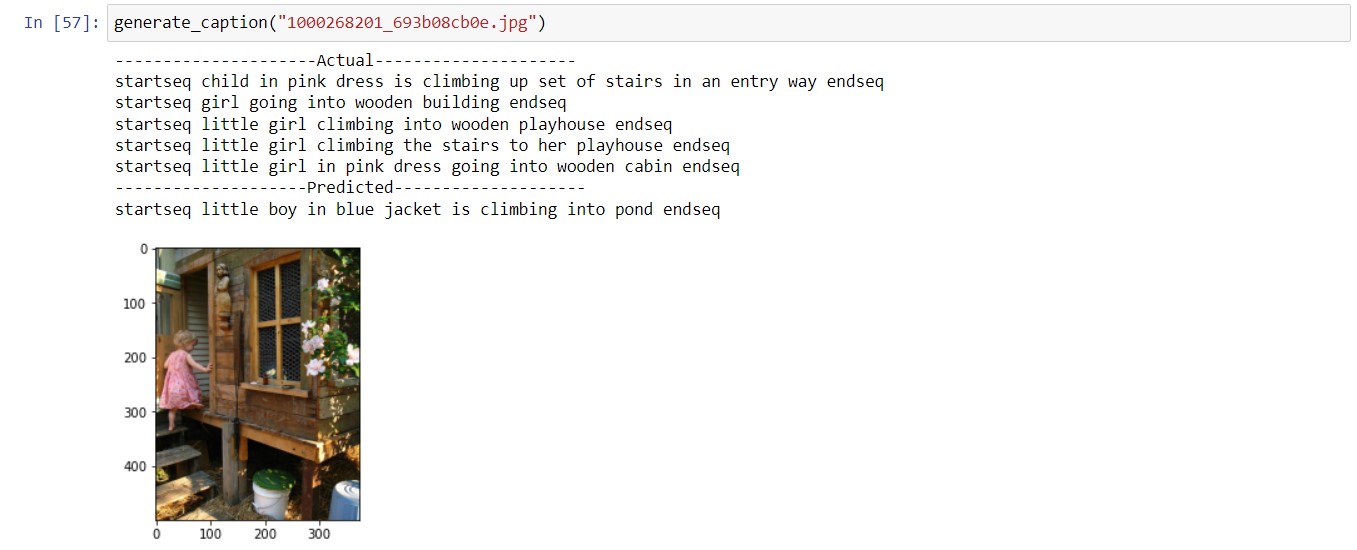


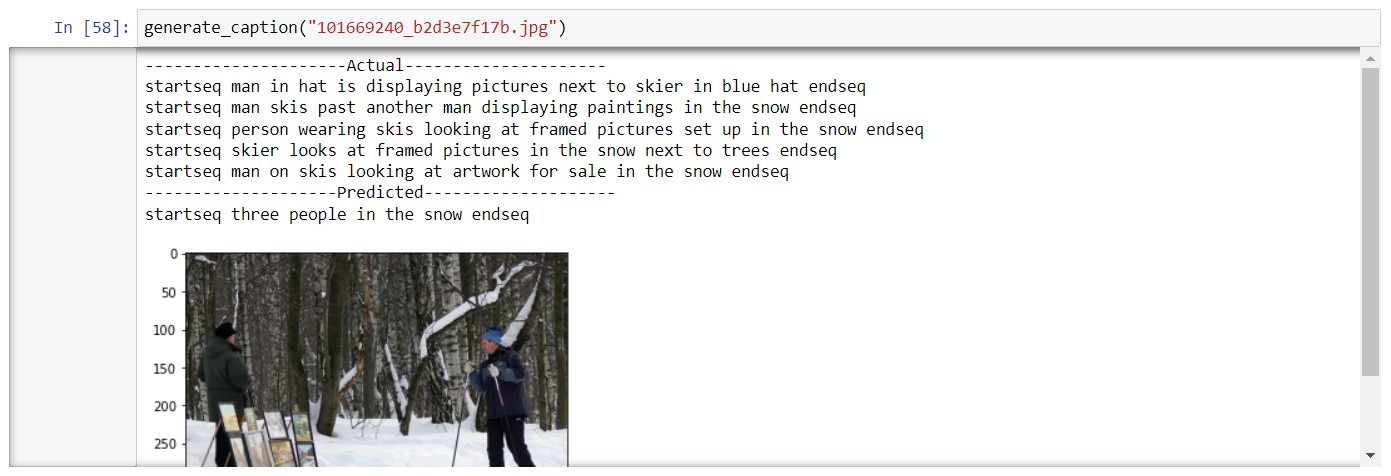


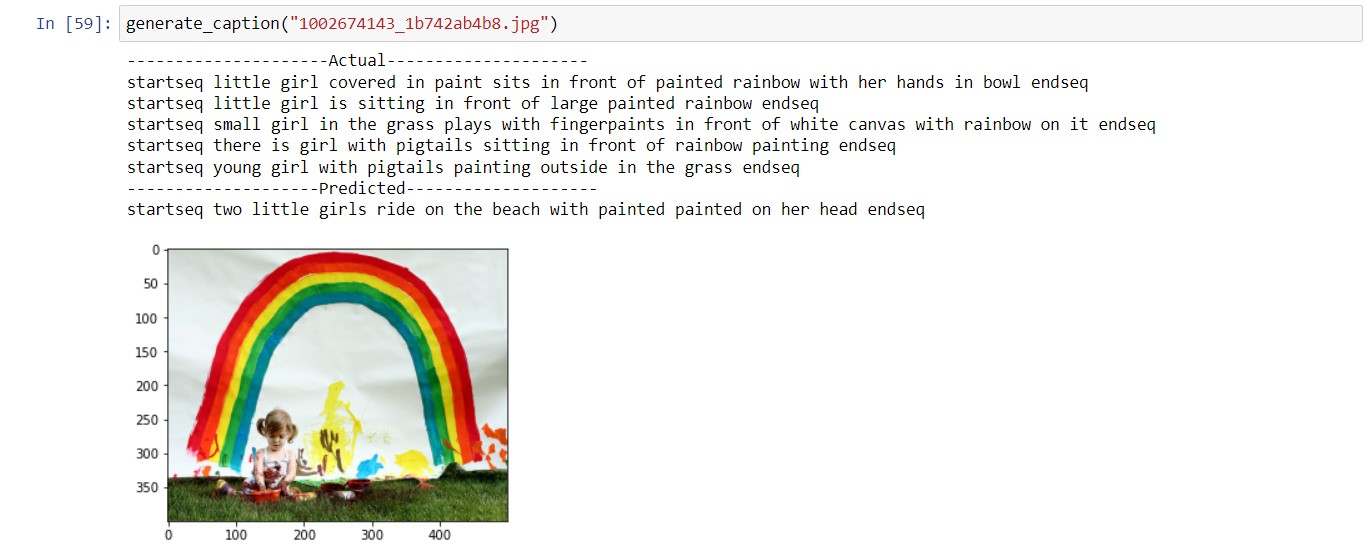












Usage of CNN and LSTM

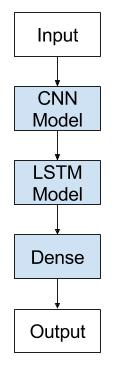
One of the most interesting and practically useful neural models come from the mixing of the different types of networks together into hybrid models.

EXAMPLE-Consider the task of generating captions for images. In this case, we have an input image and an output sequence that is the caption for the input image.

**Can we model this as a one-to-many sequence prediction task?**-Yes, but how would the LSTM or any other sequence prediction model understand the input image. We cannot directly input the RGB image tensor as they are ill-equipped to work with such inputs. Input with spatial structure, like images, cannot be modeled easily with the standard Vanilla LSTM.

**Can we extract some features from the input image?**-Yes, this is precisely what we need to do in order to use the LSTM architecture for our purpose. We can use the deep CNN architecture to extract features from the image which are then fed into the LSTM architecture to output the caption.

This is called the CNN LSTM model**,**specifically designed for sequence prediction problems with spatial inputs, like images or videos. This architecture involves using Convolutional Neural Network (CNN) layers for feature extraction on input data combined with LSTMs to perform sequence prediction on the feature vectors. In short, CNN LSTMs are a class of models that are both spatially and temporally deep and sit at the boundary of Computer Vision and Natural Language Processing. These models have enormous potential and are being increasingly used for many sophisticated tasks such as text classification, video conversion, and so on. Here is a generic architecture of a CNN LSTM Model.



Conclusion

A CNN-LSTM architecture has wide-ranging applications as it stands at the helm of Computer Vision and Natural Language Processing. It allows us to use state of the art neural models for NLP tasks such as the transformer for sequential image and video data. At the same time, extremely powerful CNN networks can be used for sequential data such as the natural language. Hence, it allows us to leverage the useful aspects of powerful models in tasks they have never been used for before. This post was just to introduce the concept of hybrid neural models and encourage the people to increasingly use different architectures of the CNN-LSTM models.