Automatic Image Captioning

Automatically generating this textual description from an artificial system is the task of image captioning.

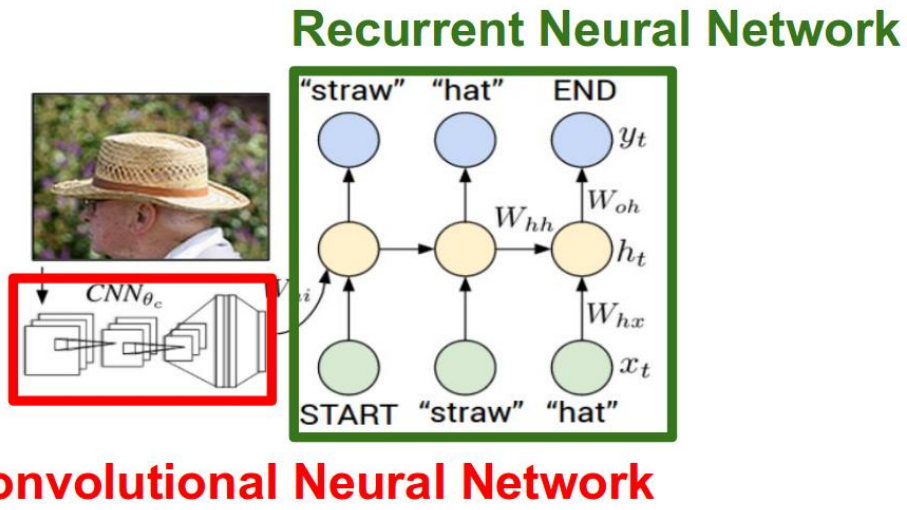
The task is straightforward – the generated output is expected to describe in a single sentence what is shown in the image – the objects present, their properties, the actions being performed and the interaction between the objects, etc. But to replicate this behaviour in an artificial system is a huge task, as with any other image processing problem and hence the use of complex and advanced techniques such as Deep Learning to solve the task.

Methodology to Solve the Task

The task of image captioning can be divided into two modules logically – one is an **image based model** – which extracts the features and nuances out of our image, and the other is a **language based model** – which translates the features and objects given by our image based model to a natural sentence.

For our image based model (viz encoder) – we usually rely on a Convolutional Neural Network model. And for our language based model (viz decoder) – we rely on a Recurrent Neural Network. The image below summarizes the approach given above.

Describing images



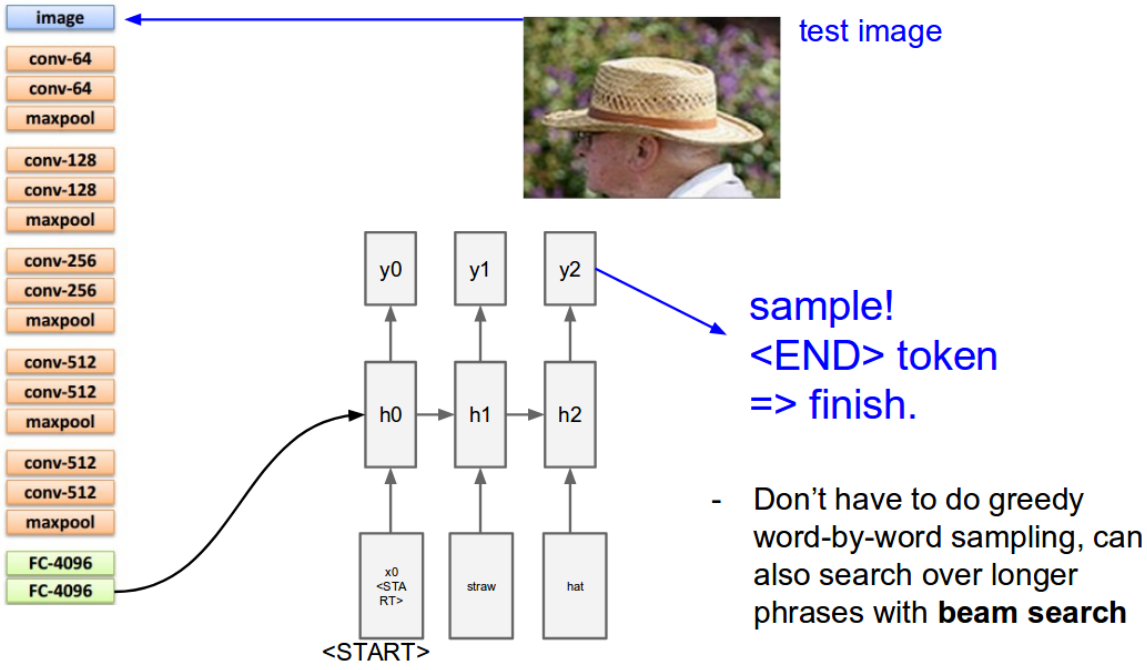
Usually, a pretrained CNN extracts the features from our input image. The feature vector is linearly transformed to have the same dimension as the input dimension of the RNN/LSTM network. This network is trained as a language model on our feature vector.

For training our LSTM model, we predefine our label and target text. For example, if the caption is "A man and a girl sit on the ground and eat.", our label and target would be as follows –

Label – [<start>, A, man, and, a, girl, sit, on, the, ground, and, eat, .]

Target – [A, man, and, a, girl, sit, on, the, ground, and, eat, ., <end>]

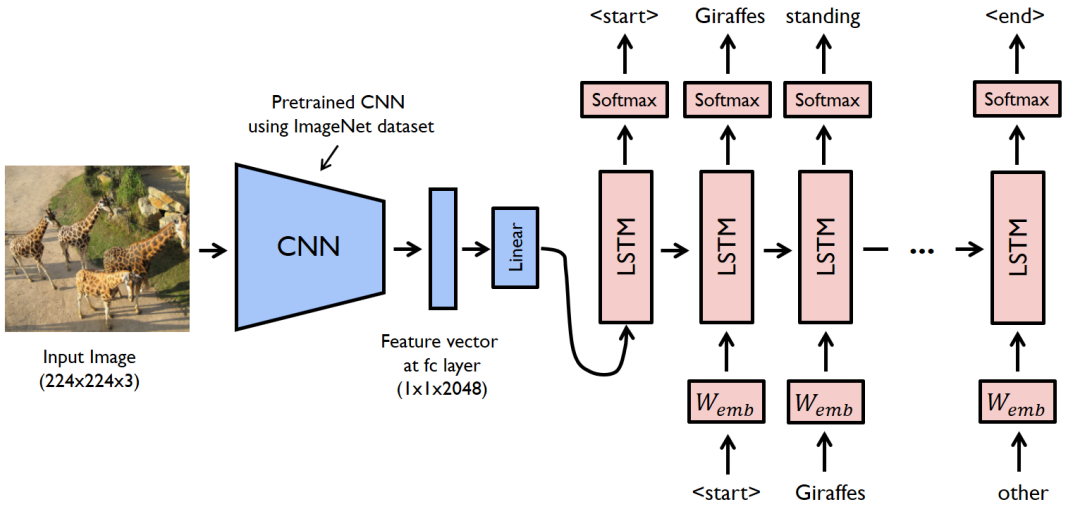
This is done so that our model understands the start and end of our labelled sequence.



Walkthrough of Implementation

Let's look at a simple implementation of image captioning in Pytorch. We will take an image as input, and predict its description using a Deep Learning model.

In this walkthrough, a pre-trained resnet-152 model is used as an encoder, and the decoder is an LSTM network.


<start>
Giraffes
standing
<end>
Softmax
Softmax
Softmax
Softmax
Pretrained CNN
using ImageNet dataset
CNN
Linear
LSTM
LSTM
LSTM
LSTM
W_{emb}
W_{emb}
W_{emb}
Input Image
(224x224x3)
Feature vector
at fc layer
(1x1x2048)
<start>
Giraffes
other