CNN-LSTM based Image Captioning

Mohith Damarapati
md4289@nyu.edu

Introduction

- Describing an image is an easy and obvious task for humans
- But, is it so for machines?
- No! Because they lack commonsense and intelligence
- They cannot understand the meaning of context and content in the image
- The problem of making machines to automatically describe an image in a natural language is referred as “Image Captioning”
- It is an interesting problem because making machines understand the visual scene is an important sub goal of True AI

Artificial Intelligence

Computer Vision

Aims to make machines understand the visual scenes
Like an ‘eye’ to AI machine

Natural Language Processing

Aims to make machines understand and generate natural language
To converse with humans

Image Captioning Problem

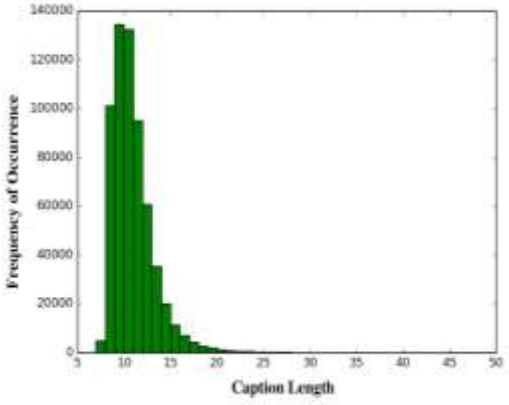
Connects CV and NLP, two sub fields of AI

A baseball player swinging a bat at a ball

Generated by my model

Dataset (MSCOCO)

- MSCOCO dataset has a total of 82783 train images and 40592 test images
- Each image has 5 captions on an average
- Caption lengths vary from 7 to 57



- **REFERENCE CAPTIONS**
- Candles, pumpkins and skull decorations on table
- A table topped with Halloween decorations and food
- A table decorated with skulls, apples and oranges

CNN – LSTM Framework

- CNNs provide a rich representation of the features which can be used for various vision tasks like classification, localization and detection
- We also call the process of extraction of rich image features as learning disentangled representations
- These disentangled representations are given as input to sequence models like RNN
- In this project, I used the ResNet-101, which is pre-trained on the ImageNet data to capture image features
- Features in the last layer of ResNet-101 after removing the fully connected layer are given as inputs to an LSTM, which is a modified version of an RNN
- LSTM architecture is similar to the architecture used in NIC

Experiments

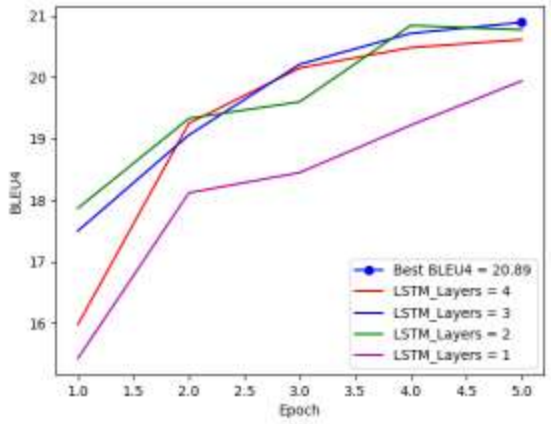
- LSTM architecture is similar to NIC. Dimension of LSTM hidden states is 512 and word embedding is 256
- Trained the model for LSTM layers ranging from 1 to 4
- LSTM layers above 2 gave almost similar results, but there is a comparable difference in the results when compared to 1 layer
- Batch size and learning rate are set as 64 and 0.001 respectively
- Also, investigated the effect of minimum word frequencies of vocabulary set on the model
- Minimum word frequency of 4 in the vocabulary set gave better results than 2 and 8

Results

MODEL	BLEU1	BLEU2	BLEU3	BLEU4
LSTM - 4	74.22	57.28	36.18	20.61
LSTM - 3	74.3	57.32	36.42	**20.98**
LSTM - 2	74.14	57.24	36.26	20.7
LSTM - 1	72.82	55.43	34.12	19.99

Baseline Results

Model	BLEU4
Random Forest	4.6
Nearest Neighbor	9.9
Human	21.7
NIC	27.7



BEST BLEU4 = 20.98

Results – Sample Captions generated by my model

A kitchen with a sink and a microwave
BLEU4: 0.397
Successful example

A couple of cows standing next to each other
BLEU4: 7.8Xe^-15
Failed example

Problems in the current captioning models

- **Lack distinctiveness**: Images with similar kind of objects but different interactions between them are not captured accurately
- **Cannot do basic arithmetic**: If model is trained with 2 cats playing, and if we test an image with 4 cats playing, current captioning models still caption it as 2 cats as they cannot understand arithmetic
- **Cannot capture deep meaning**: Humans can infer deep meanings, but current captioning models cannot. For instance, if an image consists of a man petting a cat, humans can infer the love shown towards the cat, but machine can only describe that a man is rubbing cat's fur