Image Caption Learning

Shenghui Zhou
Center for Data Science
New York University
sz2396@nyu.edu

Yiyan Chen
Center for Data Science
New York University
yc2462@nyu.edu

Yulin Shen
Center for Data Science
New York University
ys2542@nyu.edu

Abstract

In this project, we aimed to implement models that convert an image into a natural language description. The proposed method construct input into encoder-decoder framework, which conduct image encoder with Deep Residual Convolutional Neural Network on ILSVRC-2012-CLS image classification dataset and text decoder with Elman RNN, long short-term memory (LSTM) network and Gated Recurrent Unit (GRU).

1 Introduction

Image caption, which contains a short description about explaining or elaborating on pictures, is common used on publish photographs or exhibitions. However, it can be challenging when delivering detailed image captions for large amount of pictures from different photographers or painters: firstly it might cause mistakes on images captions by misleading the paints with different authors, secondly the format of image captions would be various and messy by different writers, thirdly the large amount of image captions will cause huge time consuming and expensive on labor hiring. Moreover, the application of provide a accurate, syntactically reasonable text description for the a picture and find all the important features is also significant for many uses, for instance, we can generate descriptions for movies or videos, transit pictures to words for further use.

In this project, we want to implement encoder-decoder framework which gives accurate and well structured image caption automatically. In the encoder part, we plan to implement three CNN models (ResNet-34 & ResNet-101 & ResNet-152[6]) respectively, compare and performances of each models and eventually gives best image caption. In the decode part, we use Elman RNN, long short-term memory (LSTM) network and Gated Recurrent Unit (GRU), we compare the 9 combination results and evaluate each model with Bleu score, then select a most significant model for this project. With accomplishment of this project, we can save time and labor, provide good recommendations for each image on captioning and precise descriptions for images. .

2 Literature Review

It has been studied for a long time in computer vision for the language description of the videos, which has a complex system composed structured formal language on sports shows or traffic scenes[8,9]. With the development of CNN models, there are lots of applications focusing on image recognition, which is able to find the specific subjects in the pictures with high accuracy. On top of that, Li et al.[19], who started to conduct the correlation between items in the pictures and the correlation of the structured sentences, provided a more accurate description of image caption.

In the most of the recent researches of generating the description of the images are mainly involve the combination of recurrent network (RNN) and convolutional network. In the structure, CNN aim to deliver the image feature extraction as a encoder, the RNN model plans to treat the language model as decoder. In [6], the image caption is delivered by a simple recurrent neural network with LSTM, we

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

also find several models combinations to implement image captioning, however, mots of the research choose to use LSTM as the RNN model but lose the consideration of GRU, also, there is no structured comparison or evaluation about different CNN models with Elain RNN, LSTM or GRU.

3 Methodology and Models

3.1 Dataset

We use Microsoft COCO dataset to train our models. COCO is a large-scale object detection, segmentation, and captioning dataset. It has several features to help us fit models with accurate prediction. At first, it has more than 200,000 images with labeled information. The large train set could definitely increase the prediction accuracy. At second, it has 1.5 million object instances with 80 object categories and 91 stuff categories. At third, each image has object segmentation, and superpixel stuff segmentation. Moreover, each image has 5 captions, and recognition in context. The features above make COCO become a very clean and abundant source for training a image caption model.

3.2 Encoder Framework

The encoder is a neural network takes in input and transform the input into feature map/vectors that can easily be recognized by the decoder. In our model, we used a pre-trained CNN due to the limitation of time. The pre-trained model trains over five hundred thousand images from ImageNet[6], and it contains all the weights and biases output that can be used on other images to recognize the necessary objects in a picture.

1. Deep Residual Convolutional Neural Network models (ResNet-34 & ResNet-101 & ResNet-152[6]) Given the difficulty of training deeper Neural Network, Deep Residual Convolutional Neural Network, which reformulate the layers by learning residual functions with reference to the layer inputs gives quicker and easier implements. The residual network are easily to optimize and gives better accuracy when increase the depth. We respectively use depths of 34 layers 101 layers and 152 layers and expected to give a better classification for image recognition.

   For example, the 34-layer ResNet starts with a 7×7 CONV layer, and maxpool layer with tride as 2. And then 32 3×3 CONV layers, then push the output to an average pool layer. And last connect all the outputs in a fully connected layer.

3.3 Decoder Framework

The decoder takes feature vectors from the encoder and gives out the best performed match to the actual input.

1. Base Line Recurrent Neural Network

   Recurrent Neural Network are popular in NLP tasks such as dealing with languages. RNN can use their memory to process sequential information, and that is why it can be trained on language model and output a sequence of words. Usually the number of output words equal the number of neural network layer. The sentence that input into the model is converted to one-hot vector in numeric values. RNN unlike the traditional neural network that uses different parameters at each layer, RNN shares the same weight and bias across all layers. The formula at each step is: $S_t = f(Ux_t + Ws_{t-1})$, x_t is the current input, and f function is a nonlinear function such as tanh and ReLU. In our case, it is a tanh function. $O_t = softmax(Vs_t)$ is the output. RNN suffers from the vanishing gradient problem that means it cannot take account of the information a long steps back. And some aproaches to address this problem are proper inilization of weight matrices, or improved model such as Long Short-Term Memory and Gated Recurrent Unit.

2. Long Short-Term Memory Model (LSTM)

   RNN can remember the previous computations in order to generate outputs, and this architecture is great for speech recognition, translation and image captioning. However, RNN becomes incapable of capturing previous information if the gap between current

state and previous state is too large. Long Short Term Memory Networks (LSTM) model tackle this problem by adding gates in the states to evaluate the removal or addition of information. It is done by adding sigmoid neural network layer, and the convert output interval between 0 and 1, with 0 as no information should go through and 1 as all the information should go to next stage. LSTM have 4 gates: Forget gate, Input gate, Gate gate and the Output gate. The Forget gate decides what information need removal. $f_t = \sigma(W_f[h_{t-1}, x_t] + b_f)$. The Input gate decides what part of information needs to be added. $i_t = \sigma(W_i[h_{t-1}, x_t] + b_i), \tilde{C}_t = tanh(W_c[h_{t-1}, x_t] + b_c)$. In the third stage, the gate put the two previous steps into action. $C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$. At last, the Output gate decides what the output is, which is a filtered vision of cell state that combines a sigmoid layer and tanh layer. $O_t = \sigma(W_o[h_{t-1}, x_t] + b_o), h_t = O_t * tanh(C_t)$.

3. Gated Recurrent Unit Model (GRU)
   The GRU is similar to LSTM in a sense that it also solves the vanishing gradient problem. It is a variant of LSTM. The GRU contains two gates: the update gate and reset gate. These two gates decide which information should be passes to the output. Compared to LSTM, GRU only has two gates. It is simpler than standard LSTM models. Also, GRU is relatively new and computationally more fast than LSTM.

   GRU has two gates: update gate and reset gate. The update gate is to determine how much of the past information needs to be passed along. $z_t = \sigma(W^{(z)}x_t + U^{(z)}h_{t-1})$. h_{t-1} is the hidden layer that is the output from previous layers. σ function pushes value between 0 and 1 to determine how much information to pass along. The reset gate is used to decide how much information to forget. $r_t = \sigma(W^{(r)}x_t + U^{(r)}h_{t-1})$. The current memory content uses the reset gate to store the relevant information from the past. $h'_t = tanh(Wx_t + r_t \bigodot Uh_{t-1})$. The final memory at the current time step using update gate. $h_t = z_t \bigodot h_{t-1} + (1 - z_t) \bigodot h'_t$. [7]

4 Results

From the result, we can see that GRU 34 usually describe things in a quite simple and straightforward way compared to GRU 101 and GRU 134 which can catch more details. For example: GRU 34 describes: a group of people standing around a snow covered field while GRU 134 describes this group of people are skiing. This trend happens with the group of LSTMs and RNNs. We can see that with more layers, the captions are prone to be more accurate and entail details.

The RNN compared to other two neural networks of same layer size has higher possibility to have non-complete sentences and have captions that make no sense. For example in the RNN 101: a man in a red shirt and a black shirt and a black and white dog on a sk. And LSTM: a man is jumping in the air with a skateboard, while GRU: a man is jumping in the air with a frisbee. In conclusion, GRU and LSTM with more layers tend to perform better.

5 Evaluation

In the evaluation part, we aims to use the BLEU (bilingual evaluation understudy) score to evaluate the performance of each model we used according to this image caption. From the central idea of BLUE: which consider the correspondence between the sentence from human and the machine generated text with 4 scores regarding to the performance of N-grams, we can find the metrics on both accuracy and inexpensive computation. The results are summarized in the following table:

	Elaine RNN & ResNet34	GElaine RNN & ResNet101	Elaine RNN& ResNet152
BLEU Score 1	0.645	0.639	0.643
BLEU Score 2	0.459	0.455	0.460
BLEU Score 3	0.316	0.315	0.318
BLEU Score 4	0.218	0.217	0.219

Table 1: Evaluation results for Elaine RNN with CNN models

	GRU & ResNet34	GRU & ResNet101	GRU& ResNet152
BLEU Score 1	0.656	0.674	0.673
BLEU Score 2	0.475	0.492	0.492
BLEU Score 3	0.334	0.348	0.347
BLEU Score 4	0.234	0.246	0.244

Table 2: Evaluation results for GRU with CNN models

	LSTM & ResNet34	LSTM& ResNet101	LSTM & ResNet152
BLEU Score 1	0.658	0.673	0.668
BLEU Score 2	0.479	0.495	0.489
BLEU Score 3	0.338	0.352	0.346
BLEU Score 4	0.239	0.250	0.244

Table 3: Evaluation results for LSTM with CNN models

From the evaluation we can find that ResNet101 with LSTM gives the best result with BLEU Score 0.250, which is quit close to the result from Google (0.27). Moreover, all CNN models with Elaine RNN give worse results and ResNet101 provides better result with all RNN models.

6 code repository

Our code with description about our project is posting at the Github address: $https$: $//github.com/ys2542/Image_Caption_DS1008$ containing the models, results,and evaluation of all the 9 models that we implement in tis project.

References

[1] Lei Zhang, Yangyang Feng, Jiqing Han, Xiantong Zhen. (2015). *REALISTIC HUMAN ACTION RECOGNITION: WHEN DEEP LEARNING MEETS VLAD*

[2] F.Perronnin & C.Dance. (2007) *Fisher kernels on visual vocabularies for image categorization. In CVPR, 2007.*

[3] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. (2014) *Imagenet large scale visual recognition challenge* **15**(7) :arXiv:1409.0575.

[4] Zhou, Z., Zhao, G., Hong, X., & Pietikainen, M. (2014). A review of recent advances in visual speech decoding. Image and vision computing, 32(9), 590-605.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun . (2015). *Deep Residual Learning for Image Recognition.* arXiv:1512.03385

[6] The image-net URL: http://www.image-net.org

[7] Simeon Kostadinov. (2017). *Understanding GRU networks.*. URL:https://towardsdatascience.com/understanding-gru-networks-2ef37df6c9be

[8] R. Gerber and H.-H. Nagel. *Knowledge representation for the generation of quantified natural language descriptions of vehicle traffic in image sequences.* In ICIP. IEEE, 1996.

[9] R.KirosandR.Z.R.Salakhutdinov. *Multimodalneurallan- guage models*. In NIPS Deep Learning Workshop, 2013.

[10]S.Li,G.Kulkarni,T.L.Berg,A.C.Berg,andY.Choi.*Com- posing simple image descriptions using web-scale n-grams.* In Conference on Computational Natural Language Learn- ing, 2011.