Text aware semi-supervised image captioning

Report submitted in fulfillment of the requirements
for the B.Tech Project

Third Year B.Tech.

by

Chetanya Kr Bansal(18075073), Bharat Kumar(18075016)

Under the guidance of

Dr. Tanima Dutta

Department of Computer Science and Engineering
INDIAN INSTITUTE OF TECHNOLOGY (BHU) VARANASI
Varanasi 221005, India
May 2021

Abstract

In this project, we are dealing with Finding image captions from given images. But there are some problems associated with image captioning such as: Having a large paired dataset requires intense human resources and it is a very time-consuming task. While on the contrary having a large number of images and captions separately seems an easier task than above. Our Model first assigns Pseudo labels to unpaired samples with the use of Generative Adversarial Networks to facilitate the training of unpaired dataset.

Contents

Chapter 1

Introduction

Having a large image caption paired dataset is a very time-consuming and intense task. It needs a huge effort in terms of resources. While on the contrary having a dataset with many images and many sentences separately is a much easier task than the former. In this project, we devise a semi-supervised method [1] for training an image-captioning model. In this project, our model firstly assigns pseudo/Generated labels to all unpaired images with the help of GANs(Generative adversarial Networks) [2] for using them in the Image caption training Model. Image captioning is a task in which a description(in natural language) is generated from a given image using the given description of the given image. Most of the previously trained model uses labeled datasets such as MS COCO caption dataset for better results. In this dataset there are 1.2M images with 5 captions each image. In our scenario, due to unpaired images and captions, Conventional Supervision loss can't be calculated in a standard supervised way as we have a semi-supervised dataset, hence, our model purposes pseudo labeling to unpaired images using adversarial training. In this discriminator learns to find if the given image and captions are relevant to each other or not. Our project deals with training an image captioner with unpaired data with the help of GAN by creating a scarcely paired dataset created using flicker8k dataset in which a certain percent(less than 4/10) are used as paired and the rest of the pairs are created after adversarial training.

Chapter 2

Related Literature

2.1 CNN Encoder

CNN(Convolutional neural network) is a class of deep learning neural network which is mostly applied to images, videos, visuals objects. In this project **Resnet-101** [3] is acting as a CNN encoder for given images. We use Resnet101 CNN encoder for a given input image and it is initialized on the pre-trained weights ImageNet, where it is trained for over 1000 data classes. Here, Each region is represented as 2048-Dimensional output.

2.2 LSTM

LSTM stands for Long short term memory. It is an RNN architecture. It has feedback connections also. It can process an entire sequence of data along with single-point data. In our project Channel size of hidden layers for LSTM is set to 1024, the channel size for the attention layer is set to 512, and the channel size for word embeddings is set to 1024.

2.3 Generative adversarial network

GAN has shown good effectiveness in unpaired datasets. They are mostly used in Image to image translation. Here, We have used them for the image to caption and caption to image translation. In GAN model we have two generators.G1, G2. And a common discriminator. G1 takes the image feature vector as input and output caption feature vector.G2 does vice-versa. While Discriminator takes both an image feature vector and caption feature vector as input and output a sigmoid to detect fake or real pair.

2.4 Related Research Papers

2.4.1 Image Captioning with Very Scarce Supervised Data: Adversarial Semi-Supervised Learning Approach

In this paper, we got the basic idea of our GAN Model. Our baseline model was the same as this paper's model.

2.4.2 Deep TextSpotter: An End-to-End Trainable Scene Text Localization and Recognition Framework

From this paper, we get an idea about Object Detection and Text Localization.

2.4.3 MSCap: Multi-Style Image Captioning with Unpaired Stylized Text

From this paper, we derived training styles and Working with an unpaired dataset. Further, we used evaluation matrices of this paper (BLEU scores) in our project.

Chapter 3

Proposed Methodology

Firstly here, we would like to describe normal image captioning with a paired dataset then we would describe image captioning with an unpaired dataset by using Pseudo label Assignment using GAN.

3.1 Proposed Method

Generative adversarial networks proved to be effective in the case of an unpaired dataset. As GAN doesn't need paired data for learning other than discriminator model which can be trained from a small paired dataset. Generators can be trained from unpaired datasets. As For unpaired datasets GAN has been used in image-to-image translation. For that purpose, we have used CycleGAN. Here, In our project, we can't use cycleGAN as the area and diversity of the two domains(images and captions) are much different. Hence cycle adversarial loss(forward and backward cycle loss) is difficult to calculate. Resnet-101 has been proved useful for the extraction of visual features and it is pre-trained also on the ImageNet dataset for around 1000 data classes. For purpose of our project, we have removed its last layer, which gives its extracted feature vectors. For converting our caption embeddings to feature vectors, we have used a single-layer LSTM layer followed by a dropout and dense layer. Generators are implemented with multi-layer-perceptron with four FC layers with ReLU nonlinearity. Discriminators are used for retrieving pseudo labels and for training generators.

3.2 Model Details from our code

Here is Our Model snippet.

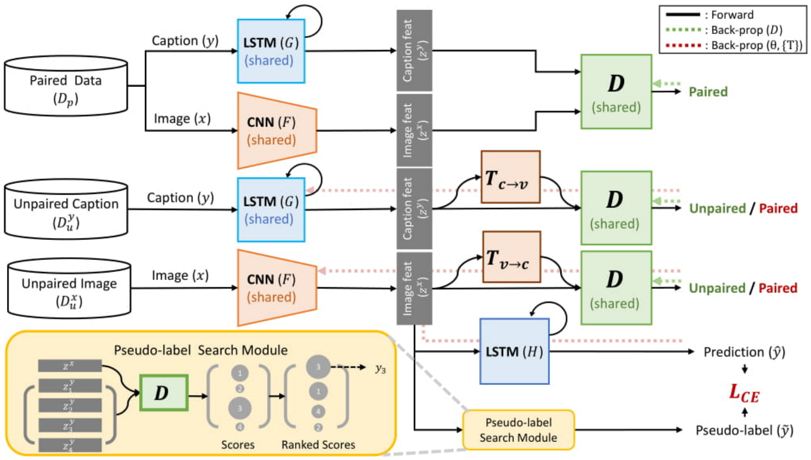


Figure 3.1: Description of the proposed method. LSTM and CNN models give feature vectors of respective inputs. A discriminator is trained to discriminate if a feature vector pair is real or fake. Then with the help of back-propagation, both generators are trained to fool the discriminator.

We have a paired dataset D_p with total N_p instances of image-caption samples as $D_p = \{(x_i, y_i)\}_{i=0}^{N_p}$. We first train the the discriminator(D) so as to identify if a given image-caption pair is semantically meaningful matching pair. We train the discriminator using the paired dataset D_p. We use a caption encoder G, which embed a caption y into its feature vector z^y. We perform this with a single layer LSTM, We take the last time step output as z^y. Similarly we use an image encoder F. Given an image x, F will give a feature vector z^x. We implemented F with the help of an CNN object detection model Resnet-101, removing its last layer. Now for discriminator training we have two comparable feature vectors z^x and z^y. We train the discriminator to identify if a given image-caption pair (z^x, z^y) is such that $(x, y) \in D_p$.

Now that our discriminator is trained we can utilize unpaired data to generate pseudo labels for that. We have a large caption dataset as $D_u^y = \{(y_i)\}_{i=0}^{M_y}$ where M_y is total no of captions in D_u^y. Similarly we have only image dataset as $D_u^x = \{(x_j)\}_{j=0}^{M_x}$, where M_x is the total no of images in D_u^x.

Now we have caption data, with the help of single layer LSTM again we will find feature vector of captions z^y. We use a generator $T_{c \to v}$ which when given a caption as input it outputs a semantically matching image to that caption such that discriminator D can be fooled. We take a noise vector then concatenate it with the caption feature vector ad give input to $T_{c \to v}$ which gives an output image z^x_{fake}.

$$z^x_{fake} = T_{c \to v}(z^y)$$

Now both z^y and z^x_{fake} are given input to the discriminator D. As the last layer of the D is sigmoid() layer which gives a float value between [0,1]. This value, with a target value '1' are used to calculate the loss of the generator $T_{c \to v}$.

Similarly we have image data, with the help of CNN network we find image feature vector z^x. We use a generator $T_{v \to c}$ which when given a image as input it outputs a semantically matching caption to that image such that discriminator D can be fooled.

$$z^y_{fake} = T_{v \to c}(z^x)$$

Both generators are implemented using Multi-layer-perceptron.

For pseudo label search module suppose we have an image x $x \in D_u^x$ and we dog the caption $y_{pseudo} \in D_u^y$ such that the discriminator score is maximum for this caption.

This caption is most likely to be paired with the given image.

$$y_{pseudo} = y_{pseudo}(x) = argmax(D(F(x), G(y))\forall y \in D_u^y$$

Similarly vice versa for unpaired captions also-

$$x_{pseudo} = x_{pseudo}(y) = argmax(D(F(x), G(y))\forall x \in D_u^x$$

We assign a confidence score [4] for each of the assigned labels from discriminator output which is output of a sigmoid layer.

$$\alpha^x = D(x, y_{pseudo})$$

$$\alpha^y = D(x_{pseudo}, y)$$

Here, $\alpha \in [0, 1]$. Therefore, we utilize the confidence scores to assign weights to the unpaired samples.

Chapter 4

Experimental Details

4.1 Dataset

In our experiment of Semi-Supervised Image Captioning, we have mainly used Flickr 8k Image-Caption Dataset which contains Total of 8k images having 5 caption labels for each image i.e. total of 40k image-caption pairs. In our experiments, to implement the Supervised and Unsupervised learning simultaneously where both paired and unpaired data exist, we use two diverse Data arrangements:

1. Rarely Paired Flickr data for supervised learning and

2. Unpaired Data for Unsupervised learning.

To prepare the first arrangement, we take around 40% of the total Flickr data which consists of 30k paired image-caption samples. We depict this small paired data as D_p. For Unpaired data D_u we have further two parts, First one is D_u^x which contains the 100% image data of Flickr dataset. D_u^x contains only 8k image samples to train the Tvc. Second part is D_u^y containing 40k sentences of image descriptions to train the generator Tcv which converts descriptions to visual information.

4.2 Training Details

In order to implement models and to train them we have used keras library. Basically We have three models to train in our work.

4.2.1 Discriminator

We have used a sequential multi-layered architecture model in Keras which takes a Feature vector of an image and a feature vector of a corresponding caption. We have concatenated both feature vectors resulting in an input feature vector of shape (1,512) for the discriminator. Another input associated with this model is a vector of labels for each image-caption sample. If an image has correct descriptions in its corresponding caption, its label is ’1’ else ’0’. When passing data to the underlying training loops of the model, we have utilized NumPy arrays. We specify the training configuration (optimizer, loss, metrics, etc). Then We call the ”fit()” method, which will train the model by slicing the data into ”batches” of size ”batch$_s$*ize”andrepeatedlyiteratingoverthewholedatasetforagivennumberof”epochs”.Theobjectsreturne*

4.2.2 Text-to-image Generator

Training of this model is a part of unsupervised learning. For this part, we have a large dataset of captions only. With Text-to-image Generator we will generate pseudo images after training it with the help of Discriminator. This part of the training is similar to GANs only difference being the discriminator which is already trained. The Generator takes an input vector of size(1,512) which is a concatenation of the feature vector of caption(1,256) and a noise vector(1,256). Since the purpose of this training is to train the model which finally generates an image for a given caption, we here use the trained discriminator as a part of the loss function. We generate a batch of images using the generator and pass the same batch of captions and generated batch of images in the already learned discriminator. Then we calculate the loss by taking target labels to '1'. We do this because the generator's objective is to "fool" the discriminator. Then with that loss, we perform Gradient Descent i.e. we change the weights of the Generator so that it keeps getting better to generate an actual visual image of a caption/sentence. We have used "Adam Optimizer" [5] with some custom parameters that are known for working fine in the case of GAN.

4.2.3 Image-to-text Generator

After training this model we finally can generate a text description of an input image. Training of this model is done via unsupervised method as in this case also we have unlabelled data of only images. Training of this model also involves the help of the already trained discriminator. The input of this model is also a vector of size (1,512) which is the concatenation of an input image feature vector and a noise vector of size(1,256) each. Similarly in this scenario also we use the discriminator as a part of the loss function so that the generator can learn to fool the discriminator to generate fake captions of an image that looks exactly like the actual caption of the image. The rest of the training process is similar to that of Text-to-image generator training.

Chapter 5

Further Ablation Study

As in our project performance is directly connected to the input to the GAN models. How precisely we can extract information without any loss from dataset is a direct measure to the performance of the model. For feature extraction from the images we can use a CNN encoder including ROI(Region Of Interest) Proposal and HOI(Human Object interaction) Graph. With this change, we can add some extra neural network layers so as to avoid that information loss in pre-processing of input images. This change can affect the overall results drastically.

Chapter 6

Qualitative Results

We have worked on an image-captioning model with less paired or labeled data to make it a semi-supervised implementation. We have achieved satisfying results as compared to the supervised methods despite of having unlabelled data in bulk which is easy to collect as we can collect any number of images and sentences separately from web. We could save a lot of time to manually pair the images to their captions.

For Evaluation purposes we have used Bleu scores(i.e. BLEU-1,BLEU-2,BLEU-3,BLEU-4) and ROUGE-L scores. Here are the following results produced from our model. After Pseudo label assignments we got these results:

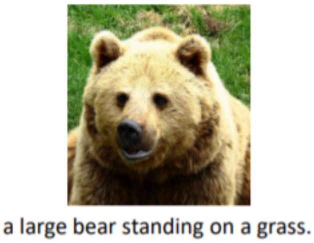
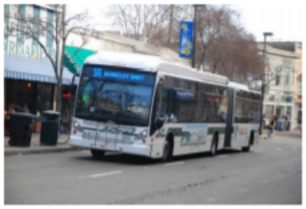


Figure 6.1: Examples of the pseudo-labels (captions) assigned to the unpaired images



a bus is parked on a street.

Figure 6.2: Examples of the pseudo-labels (captions) assigned to the unpaired images

Here are the Final results:

Model Name	BLEU-1	BLEU-2	BLEU-4
Mao et. al. 2015	0.725	0.551	0.281
Ours(with 40 % paired)	0.681	0.425	0.210
ours(with 50 % paired)	0.695	0.502	0.220

Bibliography

[1] D.-J. Kim, J. Choi, T.-H. Oh, and S.-O. Kweon, "Image captioning with very scarce supervised data: Adversarial semi-supervised learning approach," 01 2019, pp. 2012–2023.

[2] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Advances in Neural Information Processing Systems*, vol. 3, 06 2014.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," vol. 7, 12 2015.

[4] K.-H. Lee, L. Zhang, and L. Yang, "Cleannet: Transfer learning for scalable image classifier training with label noise," 11 2017.

[5] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, 12 2014.