Video Captioning using Policy Gradient Optimization

Vamsi Krishna Reddy Satti
160050064

Vighnesh Reddy Konda
160050090

Yaswanth Kumar Orru
160050066

Department of Computer Science and Engineering
Indian Institute of Technology Bombay
{vamsikrishna, vighnesh, yaswanth}@cse.iitb.ac.in

Abstract

Video captioning is a challenging task due to complexity in interpreting the video and various ways of describing it in natural language. Recent advances in deep learning have increasingly improved the performance of this task. Most state-of-the-art approaches follow an encoder-decoder framework with attention mechanism. However in this paper, we describe a novel decision-making framework for video captioning by using Policy Gradient optimization. Prior work of utilizing policy gradients on captioning have shown promising results specifically in the face of Image Captioning [8]. We propose to extend this idea by making suitable changes to existing Image Captioning works to support Video Captioning. We release the code to public here[1]. To the best of our knowledge, this attempt has had a limited exploration. Extensive experiments on MSR-VTT [18] dataset show that policy gradient optimization helps in improving various metrics.

1. Introduction

Video captioning, the task of automatically describing the content of a video with natural language, has caught increasing interest in computer vision. Despite being challenging, it has huge importance, for instance by helping visually impaired people better understand the content of videos on web and media. When compared with image captioning, video captioning is more difficult due to the diverse sets of objects, scenes, actions, attributes and salient contents with actions being performed across time.

Recent state-of-the-art approaches follow an encoder-decoder framework with attention mechanism to generate captions for videos. They usually utilize convolutional neural networks to encode the visual information and utilize recurrent neural networks to decode that information into naturally meaningful sentences. The architectures are usually optimized over a differentiable metric using back-propagation [7]. During training and inference, they try to maximize the probability of the next token given the current hidden state.

In this paper, we utilize a very similar architecture but optimized using Policy Gradient methods [5] as shown in Figure 1. To achieve this, we suggest to interpret the each possible output at each time step as an `action`. In this view, the timestep itself denotes a `state`. We try to deduce an optimal policy which is equivalent to finding the best prediction word for that timestep. The `reward` is given by the metric we try to optimize over, which is BLEU [9] score in our case. The output probability distribution at each time step from the model is used to determine an estimated `Q value` using Monte Carlo rollouts. Due to increased bias originating from rollouts [13], we employ a baseline linear network on top of the architecture to get estimates of prediction and hence reduce bias.

To learn the networks, we use policy gradient method to get the gradients of various parameters of the model using back-propagation. We begin by pretraining the whole architecture using standard traditional supervised learning with cross entropy loss before optimizing over BLEU [9] scores. This has been observed to be required for stable training.

To accommodate videos as inputs to our network, we propose to add an additional LSTM [6] layer to encode the video into a feature space, which is the most common way present in the existing literature. These features are then treated in a very similar way compared to image captioning based-methods.

We conduct detailed analyses on our framework to understand it's performance and improvements over the current baseline. Experiments on the MSR-VTT dataset [18] show that the proposed method is variedly better than the baselines when compared particularly over BLEU [9] scores. The goals of this work are summarized as follows:

[1] Code available at: `www.cse.iitb.ac.in/~vighnesh/160050064_66_90.zip`

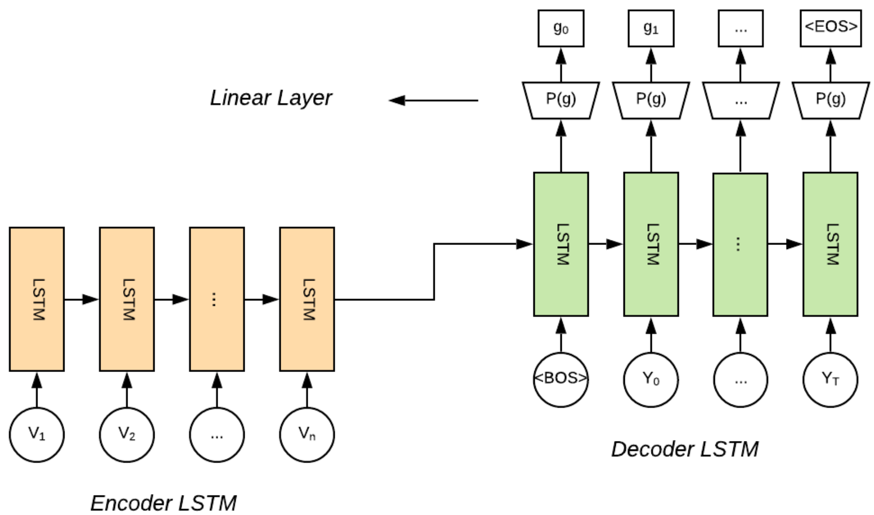

Figure 1. Architecture of our model

- We extend the idea of utilizing policy gradients in optimization for non-differentiable scores (BLEU in our case) for the task of video captioning. To the best of our knowledge, utilizing policy-based methods from [5] to video captioning has been limitedly explored.
- To realize the extension, we add an additional LSTM layer to extract features from the video sequence. This is needed to realize the actions being done on the video across time.

2. Related Work

In recent years, maximum likelihood estimation algorithm has been widely used in video captioning which maximizes the probability of the next work to be predicted given the previous ground truth words. But this method has two major limitations on its own.

First is the exposure bias which is existent between training and inference. During training, the prediction of decoder is based upon ground truth instead of model predictions. While during inference, the decoder has to depend and make its predictions based on its previous predictions. Bengio et al. [2] suggest scheduled sampling to mitigate this gap between training and inference by essentially selecting ground truth more often initially and gradually sampling more model predictions towards the end.

The other limitation is mismatch of objective between training and inference. During training, it optimizes the classification loss at word level. While in inference, discrete metrics such as BLEU [9] score are used for evaluation. This problem has been attempted to mitigate by using reinforcement learning. In image captioning, Ren et al. [12] introduce an actor-critic method in addition to a look ahead inference algorithm. Liu et al. [8] employ a policy gradient method to optimize the SPIDEr score. Dai et al. [3] use a conditional generative adversarial network with policy gradient to produce natural sentences. However, relatively less works using reinforcement learning for video captioning are present in literature.

3. Deep Reinforcement Learning-based Video Captioning

In this section, we first define our formulation for deep reinforcement learning-based video captioning using policy gradient optimization and BLEU scores. Specifically, we describe the training procedure as well as the inference mechanism. We later describe the architecture, which is a standard RNN-RNN network *i.e.* has one recurrent network to encode videos and other to prediction the word sequence. Our work on video captioning using policy gradient is particularly inspired from Liu et al. [8] work on image captioning.

3.1. Training using policy gradient

At every time step we pick a action which corresponds to a word g_t with a stochastic policy $\pi_\theta\left(g_t|s_t, x\right)$ where $s_t = g_{1:\,t-1}$, x is the image and θ are the model parameters. Instead of getting reward at the end, we are estimating the intermediate reward *via* Monte-Carlo rollouts. The value function of the partial sequence is given by

$$V_\theta\left(g_{1:\,t}|x^n, y^n\right) = E_{g_{t+1:\,T}}\left[R\left(g_{1:\,t}; g_{t+1:\,T}|x^n, y^n\right)\right] \tag{1}$$

where the expectation is *wrt.* $g_{t+1:\,T} \sim \pi_\theta\left(.|g_{1:\,t,x^n}\right)$

Goal is to maximise the average expected reward over the complete time period starting from state s_0.

$$J(\theta) = \frac{1}{N} \sum_{n=1}^{N} V_\theta(s_0|x^n, y^n) \tag{2}$$

Where N is the number training sample size and x^n, y^n is the n^{th} training sample. The gradient corresponding to $J(\theta)$ is calculated using policy gradient theorem from [5] as follows

$$\nabla_\theta V_\theta(s_0) = E_{g_{1:T}} \left[\sum_{t=1}^{T} \sum_{g_t \in V} \nabla_\theta \pi_\theta(g_t|g_{1:\,t-1}) \times Q_\theta(g_{1:\,t-1}, g_t) \right] \tag{3}$$

where Q is the Expected future of a state $g_{1:\,t-1}$ and action g_t pair.

We used Monte-Carlo rollouts to estimate the Q by averaging the rewards of k continuation samples of sequence s_t, g_t to give $g^k_{t+1:\,T}$ which gives

$$Q_\theta(g_{1:\,t-1}, g_t) = E_{g_{t+1:\,T}}\left[R(g_{1,t-1}; g_t; g_{t+1:\,T})\right] \tag{4}$$

$$= \frac{1}{K} \sum_{k=1}^{K} R(g_{1:\,t-1}; g_t; g_{t+1:\,T}) \tag{5}$$

Though the Monte-Carlo is an unbiased estimate, it has a high variance. So, in order to reduce the variance we introduce an expected baseline reward here $E_{g_t}[Q(g_{1:\,t-1}, g_t)]$ which will be estimated using a baseline model $B_\phi(g_{1:\,t-1})$ with ϕ as parameters by simply using MLE estimate. We then subtract the baseline reward from $Q_\theta(g_{1:\,t-1}, g_t)$ to estimate the gradient of value function as

$$\nabla_\theta V_\theta(s_0) \approx \sum_{t=1}^{T} \sum_{g_t} \left[\pi_\theta(g_t|s_t) \nabla_\theta \log \pi_\theta(g_t|s_t)\right] \times (Q_\theta(s_t, g_t - B_\phi(s_t))) \tag{6}$$

where $s_t = g_{1:\,t-1}$. We train baseline estimator model by minimising the squared-error between $Q_\theta(s_t, g_t)$ and $B_\phi(s_t)$ as

$$L_\phi = \sum_t E_{s_t} E_{g_t} \left(Q_\theta(s_t, g_t) - B_\phi(s_t)\right)^2 \tag{7}$$

The Policy Gradient algorithm as mentioned in [8], is written with proper detail and pseudo code in Algorithm 1. We note that this algorithm is similar to REINFORCE as described here [17].

Algorithm 1: Policy Gradient algorithm

```
Input : D = {(x^n, y^n) : n = 1: N};
Train φ_θ(g_{1:T}|x) using MLE on D;
Train B_φ using MC estimates of Q_θ on small subset
  of D;
for each epoch do
    for example(x^n, y^n) do
        Generate Sequence g_{1:T} ~ π_θ(.|x^n);
        for t = 1: T do
            Compute Q(g_{1:t-1}, g_t) for g_t with K
              Monte-Carlo rollouts;
            Compute estimated baseline
              B_φ(g_{1:t-1});
        Compute G_θ = ∇_θ V_θ(s_0);
        Compute G_φ = ∇_φ L_φ;
        SGD update on θ using G_θ;
        SGD update on φ using G_φ;
```

4. Implementation Details

4.1. Dataset

We report results on the MSR-VTT [18] dataset. This has 6513 training , 497 validation and 2990 testing videos, each with 20 ground truth captions.

We preprocess the text data by lower casing, and replacing words which occur less than 4 times in the 23666 training set with `<UNK>`, which represents the unknown token. This results in a vocabulary size of 7816. At training time, we keep all captions to their maximum lengths by appropriate padding wherever needed. At test time, the generated sequences are truncated to 30 symbols.

The video frames are sampled at 6 fps (frames per second) and the features are extracted from a ResNet backbone [11] which was pretrained on ImageNet dataset [4].

4.2. Model

The baseline model used to reduce bias while calculating gradients is a Linear layer over the LSTM hidden layers like in Figure 1 (not shown). It takes the output of recurrent network at a time step as input and estimates the prediction by a simple MLE estimate. During training, the parameters of LSTM network are fixed and SGD updates the linear layers' parameters.

Our model trained using Policy gradient (PG) follows the architecture shown in Figure 1. We sample 3 sentences to estimate the Q value using Monte Carlo rollouts. We note that all our poliwcy gradient optimization was only over BLEU scores during training. This simplification was done due to high computational load to calculate the other metrics while training. After training, the model was evaluated over various other scores presented in Section 5.

Models	BLEU-4	METEOR	ROUGE-L	CIDEr
v2t navigator [15]	40.8	28.2	60.9	44.8
videoLAB [10]	39.1	27.7	60.6	44.1
Aalto [14]	39.8	26.9	59.8	45.7
our model	30.6	24.3	54.2	32.8
our model with PG	33.8	25.2	56.1	34.9

(We refer to Policy Gradient as PG)

Table 1. Comparison with the top models on MSR-VTT [18] test dataset. We are quite far from the state of the art results on all the evaluation metrics which rely on completely different methods. However, our model when optimized with policy gradients is better compared to our model with just optimized on MLE thereby using policy gradients improves the score on all evaluation metrics.

Case	Video ID	Ground Truth (GT)	MLE	Policy Gradient (PG)
Both are good	7428	a child is in a bounce house	a girl is sitting on a trampoline	a girl is doing gymnastics
	8440	a man is punching another man in a wrestling match	two wrestlers are having a match	two men are wrestling
PG better than MLE	7960	a band performing in a small club	a man is playing guitar	a band is playing a concert
	8563	a person is on a motorcycle on the streets	a man is riding a motorcycle	a man is riding a motorcycle on a road
MLE better than PG	7732	a young man was being filmed while he was singing	a person is playing a guitar and singing	a man is singing a song in a stage
	8113	a congressman from maryland is own the news to discuss president obama	a man is talking about politics	a man is talking about a news story
Both are bad	8555	tv chef gordon shows the front door sign of a restaurant and walks in the front door	a man is talking about a new school and its history	a <eos>
	8501	a man playing video games	a <eos>	a man is talking about a map

Table 2. Example captions comparing our model with PG and without PG

5. Results

Table 1 shows the results of some models, along with our model on the MSR-VTT dataset. Clearly using policy gradients improve the score on all the evaluation metrics. We believe that other models have better scores because their video encoders are sophisticated compared to ours because they use various attention schemes combined with multi-modal features such as C3D [16] and audio MFCC features.

Table 2 shows generated sentences for some videos in the test set. As we can see there are cases where policy gradient generates better sentences than MLE and in some cases MLE generates better sentences than policy gradient.

6. Future Work

On closer inspection of results, we can observe that policy gradient optimization has the potential to improve a baseline algorithm like MLE estimate. One way our architecture can be improved is by adding attention [1] so that the decoder can look at specific frames of the video it has to pay attention to while generating a word at a time step. Another possible direction from pre-processing perspective is to utilize multi-modal features such as C3D[16] and MFCC features to boost the performance of the model.

7. Conclusion

Though the results are meaningful, our metrics are quite far from the state of the art results which rely on completely different methods over all the evaluation metrics. However, our model when optimized with policy gradients is better compared to our model with just optimized on MLE thereby using policy gradients improves the score on all evaluation metrics.

Acknowledgement We would like to thank Prof. Shivaram Kalyanakrishnan for this opportunity to work on the project as a part of the requirements for the Reinforcement Learning course at Indian Institute of Technology, Bombay.

References

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate, 2014.

[2] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. *CoRR*, abs/1506.03099, 2015.

[3] Bo Dai, Dahua Lin, Raquel Urtasun, and Sanja Fidler. Towards diverse and natural image descriptions via a conditional GAN. *CoRR*, abs/1703.06029, 2017.

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[5] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[6] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[7] Yann LeCun, D Touresky, G Hinton, and T Sejnowski. A theoretical framework for back-propagation. In *Proceedings of the 1988 connectionist models summer school*, volume 1, pages 21–28. CMU, Pittsburgh, Pa: Morgan Kaufmann, 1988.

[8] Siqi Liu, Zhenhai Zhu, Ning Ye, Sergio Guadarrama, and Kevin Murphy. Improved image captioning via policy gradient optimization of spider. In *Proceedings of the IEEE international conference on computer vision*, pages 873–881, 2017.

[9] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

[10] Vasili Ramanishka, Abir Das, Dong Huk Park, Subhashini Venugopalan, Lisa Anne Hendricks, Marcus Rohrbach, and Kate Saenko. Multimodal video description. In *Proceedings of the 24th ACM international conference on Multimedia*, pages 1092–1096. ACM, 2016.

[11] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[12] Zhou Ren, Xiaoyu Wang, Ning Zhang, Xutao Lv, and Li-Jia Li. Deep reinforcement learning-based image captioning with embedding reward. *CoRR*, abs/1704.03899, 2017.

[13] Zhou Ren, Xiaoyu Wang, Ning Zhang, Xutao Lv, and Li-Jia Li. Deep reinforcement learning-based image captioning with embedding reward, 2017.

[14] Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Christopher Pal, Hugo Larochelle, Aaron Courville, and Bernt Schiele. Movie description. *International Journal of Computer Vision*, 123(1):94–120, 2017.

[15] Zhiqiang Shen, Jianguo Li, Zhou Su, Minjun Li, Yurong Chen, Yu-Gang Jiang, and Xiangyang Xue. Weakly supervised dense video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1916–1924, 2017.

[16] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pages 20–36. Springer, 2016.

[17] Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3):229–256, May 1992.

[18] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.