REVNET: BRING REVIEWING INTO VIDEO CAPTIONING FOR A BETTER DESCRIPTION

Huidong Li, Dandan Song, Lejian Liao, Cuimei Peng*

Beijing Engineering Research Center of High Volume Language Information Processing & Cloud Computing Applications, Beijing Key Laboratory of Intelligent Information Technology, School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China
{2120171029, sdd, liaolj, 3120181022}@bit.edu.cn

ABSTRACT

Recently, the task of automatically generating a textual description of a video is attracting increasing interest. The attention-based encoder-decoder framework has been extensively applied in this domain. However, compared with other captioning tasks, such as image captioning, video captioning is more challenging because semantic information among frames is hard to be extracted. In this paper, we propose a reviewing network (REVnet) to reconstruct the previous hidden state, which is combined with the conventional encoder-decoder framework. REVnet brings backward flow into the caption generation process, which encourages the hidden state embedding more information and enables the semantics of the generated sentence more coherent. Furthermore, REVnet can regularize the attention mechanism within the framework, which encourages the model better utilizing the semantic information extracted from multiple different frames. Our experimental results on benchmark datasets demonstrate that our proposed REVnet has a significant improvement over the baseline method. Furthermore, we use a reinforcement learning method to finetune the model, and get better results than the state-of-the-art methods.

Index Terms— Video Caption, Backward Flow, Attention Mechanism, Reinforcement Learning

1. INTRODUCTION

Generating a textual description of a video is a task to link video with language, which has received increasing attention in computer vision communities. Different from image captioning which generates caption from one still picture, the input of video captioning is a set of frames. As the interrelationship between frames is hard to be learned, video captioning is a more challenging task.

Recent work is primarily based on an encoder-decoder architecture to build the mapping from visual contents to words [1, 2]. Specifically, an encoder (CNN or RNN) takes video frames as input and extracts a compact video representation. Then the video representation is fed into decoder RNN to generate sentence word by word. Additionally, [3] suggested to dynamically select multiple visual representations based on temporal attention mechanism which is driven by previous hidden states from the decoder.

Based on the previous work and our observations, we summarize the problems that need to be addressed in this task. Firstly, rich semantic information needs to be extracted, and the contribution of each dynamic semantic fragment within a video should be assigned appropriately. Secondly, the caption generation process needs to ensure syntactic relevance of the generated sentence, and requires modeling the relationship between adjacent words. However, the conventional frameworks only consider forward flow (previous word to the current word) in the caption generation process. As words within a sentence interact with each other from the beginning to the end, current information should also restrict the previous state in turn.

In this paper, we bring information from current to the past by rebuilding the previous hidden state. Firstly, as the adjacent hidden states within the decoder have a close relationship, we use the current hidden state to reconstruct the previous one, which can not only encourage the hidden state to embed more information, but also enable the semantic meaning of generated sentence more coherent. Secondly, the current input of the decoder is an attention-based visual representation vector, which is generated according to the previous state of the decoder. We propose to reconstruct the previous hidden state with the current input, which can regularize the attention mechanism and encourage a better assignment on the contribution of multiple different visual representations.

In this paper, we propose several different implementations to reconstruct the previous hidden state in the decoder. Our main idea is to bring backward flow to the decoder and encourage the attention mechanism getting a more focused context vector. Furthermore, the reinforcement learning method is utilized for a better performance.

The main contributions of this paper can be summarized as:

1) We propose to bring backward flow in the decoder to

978-1-5386-9552-4/19/$31.00 ©2019 IEEE
DOI 10.1109/ICME.2019.00228

strengthen the relationship between adjacent words and get a more coherent semantic representation.

2) We encourage the model to find a better contribution assignment on visual representations which lead to a more accurate result.

3) We utilize the reinforcement learning method to further optimize our model and get a better performance than the state-of-the-art methods.

2. RELATED WORK

In this section, we introduce two types of methods that used in visual captioning and some reinforcement learning works.

2.1. Template-based Approaches

Template-based methods predefine some sentence generation templates and specific grammar rules. Words are detected from the visual content to produce the final description with predefined templates. For example, [4] proposed to describe human activities based on concept hierarchies of actions. [5] formulated video captioning as machine translation problem and learned a CRF to predict the semantic representation. A semantic hierarchy was defined in [6] to learn the semantic relationship between different sentence fragments. [7] exploited both statistics gleaned from parsing large quantities of text data and recognition algorithms from computer vision.

2.2. Sequence Learning Approaches

As templates limit the ability of the model to express, encouraged by the development of deep neural networks, sequence learning approaches start to be noticed. Typically, these methods used CNN or RNNs as an encoder to extract visual feature, and then other RNNs were used to generate captions. Attention mechanism [3, 8] and bi-RNN [9] were also applied. Recently, several reconstruction methods were proposed for visual captioning. [10] reproduced the video features based on the hidden state sequence generated by the decoder. [11] proposed to regularize the transition dynamics of RNNs by hidden state reconstruction for image captioning. Different from their works, in this video captioning task, we use both current hidden state and attention-based context vector to reconstruct previous hidden state, which can make full use of the information contained in multiple video frames.

2.3. Reinforcement Learning

While the encoder-decoder architecture maximizes the probability of current ground-truth given the previous ground-truth output during training step, that previous ground-truth is unknown during testing. This inconsistency problem has been known as exposure bias [12], and REINFORCE algorithm [13] is exploited to solve this problem. There were several improvements proposed, [14] added a baseline to improve the stability while training. Actor-critic methods were proposed to train an additional critic to estimate the value of generated words automatically [15, 16].

In this work, reinforcement optimization similar to [14] are applied to further improve the performance of our model.

3. FRAMEWORK

The whole architecture of our model is illustrated in Fig. 1. Our proposed REVnet cooperates with an attention-based encoder-decoder architecture for video captioning.

3.1. Encoder-Decoder

We apply an attention-based encoder-decoder architecture as our basic framework, where we encode input frame level video features V = $\{v_1, v_2, ..., v_m\}$ via a bi-directional LSTM and then generate the caption $\{w_1, w_2, ..., w_n\}$ using another LSTM with an attention mechanism. We use θ denote the parameters of whole architecture and $\{w_1^*, w_2^*, ..., w_n^*\}$ denote the ground-truth caption, and then the cross entropy loss can be formulated as:

$$L_{xe} = -\sum_{i=1}^{n} logP(w_i|w_{<i}^*, V; \theta) \quad (1)$$

where $P(w_i|w_{<i}^*, V; \theta) = softmax(W^T h_i^d)$. h_i^d is the decoder's hidden state at time step i, which is generated by standard RNN recursion with inp_i^d as input at time step i.

$$h_i^d = LSTM(inp_i^d, h_{t-1}^d) \quad (2)$$

where inp_i^d denotes the concatenation of attention-based context vector c_t and the embedding of previous ground-truth word x_t. Context vector c_t is computed as a weighted sum over all the encoder's hidden states:

$$c_t = \sum_{i=1}^{m} \alpha_{t,i} h_i^e \quad (3)$$

the weight α_t denote the measure of correlation between last hidden state and all the encoder's hidden states. α_t is defined as:

$$\alpha_{t,i} = \frac{exp(e_{t,i})}{\sum_{k=1}^{n} exp(e_{t,k})} \quad (4)$$

And

$$e_{t,i} = \omega^T tanh(W_a h_i^e + U_a h_{t-1}^d + b_a) \quad (5)$$

where ω,W_a,U_a and b_a are trainable paremeters, and h_i^e denote hidden state in the encoder, while h_{t-1}^d is the previous hidden state of the decoder.

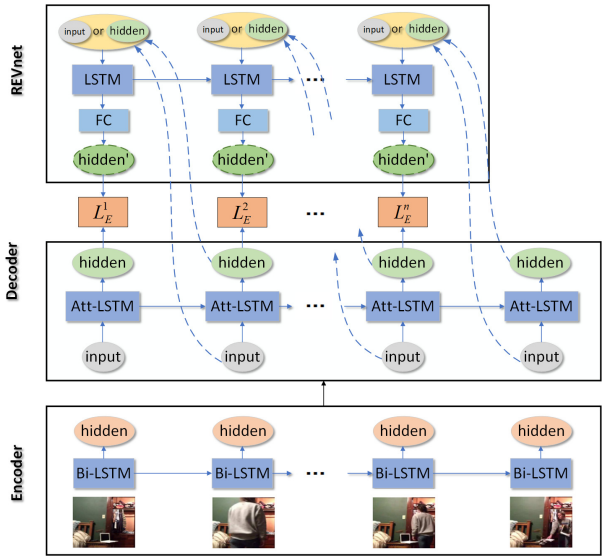

Fig. 1. The overview of the attention-based encoder-decoder architecture along with our proposed REVnet. As can be seen from the picture, the target of REVnet is to reconstruct the previous hidden state of the decoder with current hidden state or current input of decoder.

3.2. REVnet

There are two methods in our proposed REVnet to reconstruct the previous hidden state, and they are realized by two kinds of LSTMs. The first LSTM takes the current hidden state of the decoder as input, while the second LSTM fed with the current input of decoder to perform a reconstruction operation.

We use inp_i^r to represent the input of LSTMs. h_i^d is the current hidden state of decoder which is the dynamic transition result from the previous hidden state. inp_i^d is the current input of decoder, which is an attention-based visual representation driven by the previous hidden state. inp_i^r is an alternative in h_i^d or inp_i^d.

The LSTM unit of REVnet can be formulated as:

$$\begin{aligned} \begin{pmatrix} i_t^r \\ f_t^r \\ o_t^r \\ g_t^r \end{pmatrix} &= \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} \mathbf{T} \begin{pmatrix} inp_t^r \\ h_{t-1}^r \end{pmatrix}, \\ c_t^r &= f_t^r \odot c_{t-1}^r + i_t^r \odot g_t^r, \\ h_t^r &= o_t^r \odot \tanh(c_t^r), \end{aligned} \tag{6}$$

where i_t^r,f_t^r,o_t^r and h_t^r are the input gate, forget gate, output gate and hidden state of the LSTM unit.

Furthermore, in order to facilitate the comparison of the similarities between the previous hidden state and reconstructed hidden state, a fully-connected layer is employed to map h_t^r into the common space with h_{t-1}^d :

$$\widehat{h}_{t-1} = w_{fc} h_t^r + b_{fc} \tag{7}$$

where w_{fc} and b_{fc} are the weights of the fully-connected layer, and $\widehat{h}_{t-1}$ is the final reconstructed previous hidden state.

The reconstruction loss function is defined as Euclidean distance between h_{t-1} and $\widehat{h}_{t-1}$:

$$L_E = \|h_{t-1}^d - \widehat{h}_{t-1}\|_2^2 \tag{8}$$

As mentioned above, two kinds of LSTMs are proposed to reconstruct the previous hidden state with different input vectors. And we can use L_{E-i} and L_{E-h} to represent the reconstruction loss of two different kinds of LSTMs. However, there are three forms of our proposed REVnet, and we use $\mathrm{REVnet_{v1}}$, $\mathrm{REVnet_{v2}}$, and $\mathrm{REVnet_{v3}}$ to distinguish them. $\mathrm{REVnet_{v1}}$ is realized by the LSTM which takes the current hidden state as input, while LSTM with in $\mathrm{REVnet_{v2}}$ is fed with current input of decoder. $\mathrm{REVnet_{v3}}$ combines two kinds of LSTMs.

Then the reconstruction loss of three forms of REVnet can be formulated as:

$$\begin{aligned} L_{rev-v1} &= L_{E-h} \\ L_{rev-v2} &= L_{E-i} \\ L_{rev-v3} &= \alpha L_{E-h} + (1-\alpha) L_{E-i} \end{aligned} \tag{9}$$

where α is a trade-off parameter to balance the contributions of two reconstruction methods.

Through minimizing these reconstruction losses, we enhance the ability of the attention-based encoder-decoder architecture from different aspects. $\mathrm{REVnet_{v1}}$ brings backward flow to the architecture and encourages the current hidden state to embed more information from the previous one. And $\mathrm{REVnet_{v2}}$ provides a constraint to attention mechanism, which encourages the model to find a better contribution assignment on visual representations. As $\mathrm{REVnet_{v3}}$ uses a combination of the two LSTMs to reconstruct the previous hidden state, it has the two advantages mentioned above. The correlations between h_i^d and h_{i-1}^d are further exploited and enhanced.

3.3. Training Procedure

In our experiments, it is hard to make a suitable initialization of parameters in REVnet. So we use a training strategy to optimize our models. Firstly, we only train attention-based encoder-decoder framework without REVnet, the target is to minimize the negative log-likelihood loss function proposed in Equation 1. Then, in order to suitably initialize the parameters of REVnet, parameters pre-trained in encoder-decoder architecture are frozen and parameters in REVnet will be

trained for one epoch. Finally, the whole network will be trained together with the following objective function.

$$L_1 = L_{xe} + \beta L_{rev} \quad (10)$$

where L_{rev} is one of the reconstruction loss functions proposed in Equation 9. And the coefficient β determines the relative weight of cross entropy loss versus REVnet loss.

In order to overcome the inconsistency problem known as exposure bias and optimize the sentence-level test metrics directly, we introduce the self-critical REINFORCE algorithm [13, 14] to obtain a better performance. Here, we use CIDEr score as the reward of the REINFORCE algorithm. The objective function of the REINFORCE algorithm is:

$$L_r = -E(R(S)) \quad (11)$$

where $R(S)$ is the reward of generated sentence sampled from the network. In order to reduce variance, a baseline b which is the reward of the generated sentence applying greedy search method is introduced. The general updates of parameters can be written as:

$$\nabla_\theta L_r(\theta) = -(R(S) - b)\nabla_\theta log\, p_\theta(S|V;\theta) \quad (12)$$

and the whole loss function used in REINFORCE optimization can be formulated as:

$$L_2 = \gamma L_r + (1-\gamma)L_1 \quad (13)$$

where γ represents the tradeoff between the two losses.

4. EXPERIMENTAL RESULTS

To prove the effectiveness of our proposed REVnet, we test its performance with extensive experiments and compare our results with many existing works. We evaluate our model on the benchmark dataset – Microsoft Research video to text (MSR-VTT). And in this section, we introduce the experimental setting and our competitive results in details.

4.1. Dataset

MSR-VTT [17] is the largest dataset for video captioning, which is derived from a wide variety of video categories. There are 10000 video clips in MSR-VTT, and each clip is annotated with 20 sentences. We use the public splits for training and testing.

4.2. Implementation Details

In particular, we use pre-trained ResNet-152 [18] to extract frame level features to obtain static semantic information, and apply pre-trained ResNeXt-101 [19] to extract motion features for dynamical semantic information. They are concatenated and 4096-dimensional vectors are attained to be input. The hidden size of LSTMs in our model is 1024. Word embedding size is 512. And the coefficients α, β, γ in loss functions are 0.666, 0.005, and 0.985, respectively. Learning rate trained with cross-entropy loss is set to 0.0001, while that within REINFORCE algorithm is 0.00001. We use four diverse automatic evaluation metrics: BLEU, METEOR, ROUGE-L, and CIDEr. We use the standard evaluation code from MSCOCO server [20] to obtain the results.

Table 1. Comparison with the state-of-the-art methods on MSR-VTT dataset.

Models	BLEU-4	METEOR	ROUGE-L	CIDEr
v2t_navigator	40.8	28.2	60.9	44.8
Aalto	39.8	26.9	59.8	45.7
VideoLAB	39.1	27.7	60.6	44.1
PickNet	41.3	27.7	59.8	44.1
CIDEnt-RL	42.2	28.2	**62.3**	53.0
baseline-XE	40.8	28.1	60.7	48.2
$REVnet_{v1}$-XE	41.0	28.3	61.1	49.2
$REVnet_{v2}$-XE	40.9	28.2	61.0	49.0
$REVnet_{v3}$-XE	41.3	**28.4**	61.3	49.2
baseline-RL	41.8	28.0	62.2	52.5
$REVnet_{v3}$-RL	**42.4**	28.1	**62.3**	**53.2**

4.3. Comparative Methods

In this paper, We propose three forms of REVnet, which coupled with the attention-based encoder-decoder architecture. Firstly, we train the three models with cross-entropy loss, and get $\mathrm{REVnet_{v1}}$-XE, $\mathrm{REVnet_{v1}}$-XE and $\mathrm{REVnet_{v1}}$-XE respectively. Then the best performing model ($\mathrm{REVnet_{v3}}$-XE) is selected to be further optimized with REINFORCE algorithm ($\mathrm{REVnet_{v3}}$-RL). We compare our results with the state-of-the-art methods on MSR-VTT dataset: v2t navigator [21], Aalto [22], VideoLAB [23], PickNet [24], CIDEnt-RL [25].

Baselines. We use an attention-based encoder-decoder architecture as a baseline. Firstly, the baseline is trained with a cross entropy loss (**baseline-XE**). Then further finetune the model with REINFORCE algorithm with CIDEr score as a reward (**baseline-RL**).

v2t_navigator. [21] ranks top one on the leaderboard of the MSR-VTT challenge. In this work, fused features are used to represent the video contents.

Aalto. [22] is the second best method in the MSR-VTT challenge. It trains an evaluator network to drive the captioning model towards semantically interesting sentences.

VideoLAB. [23] ranks third in the MSR-VTT challenge. This model fuses multiple sources of information judiciously and use different modalities separately.

PickNet. [24] proposes a plug-and-play PickNet to perform informative frame picking, and develops a

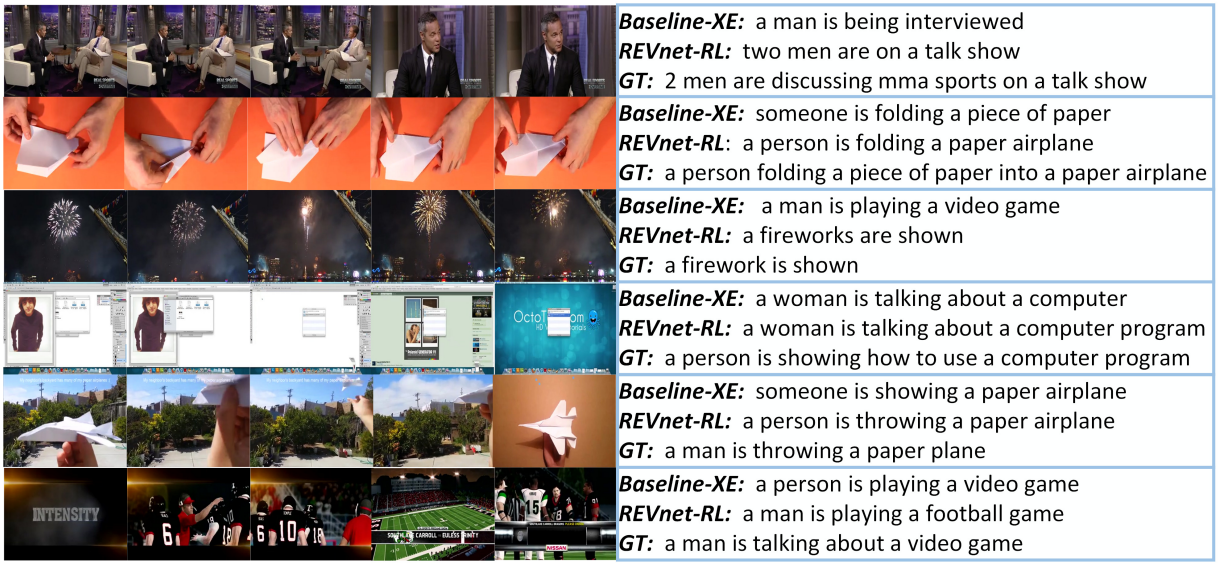

Fig. 2. Qualitative comparison with the baseline methods. The given examples are from the test set of MSR-VTT.

reinforcement-learning based procedure to train the network sequentially.

CIDEnt-RL. [25] proposes an entailment-enhanced reward that corrects phrase-matching based CIDEr to only allow for logically-implied partial matches.

4.4. Comparison Results

As shown in Table 1, our REVnet optimized with cross-entropy loss have achieved comparable results with the state-of-the-art methods. It can be noticed that all the three different implementations of REVnet bring a significant improvement over the baseline-XE method. Then we pick $\mathrm{REVnet}_{\mathrm{v3}}$ to be optimized with REINFORCE algorithm ($\mathrm{REVnet}_{\mathrm{v3}}$-RL). The performance of $\mathrm{REVnet}_{\mathrm{v3}}$-RL is further improved, and outperforms all the other algorithms listed in the table on three metrics (BLUE-4, ROUGE-L, and ROUGE-L).

It should be noticed that our $\mathrm{REVnet}_{\mathrm{v3}}$-RL's performance on METEOR is not that good, even worse than $\mathrm{REVnet}_{\mathrm{v3}}$-XE. There may be two reasons for this phenomenon. Firstly, our reward of REINFORCE algorithm is the final CIDEr score of the whole generated sentence, which has few contributions on METEOR score promotion. Secondly, our models enhance the connection between adjacent words, while this is further emphasized with a reinforcement learning optimization method. It can restrain the semantic expression and the recall rate of the whole sentence.

Despite underperformance on METEOR, $\mathrm{REVnet}_{\mathrm{v3}}$-RL get an obvious improvement on BLEU-4 metric, which is the most popular evaluation metric for machine translation. It demonstrates that our generated sentence is closer to the ground-truth and more coherent. Relationship of adjacent words is better modeled and enhanced. Competitive captions generated by our model are shown in Fig. 2, where the given examples are selected from the test set of MSR-VTT. As we can see, our generated sentences are more accurate descriptions about visual content of the video. It is obvious that the generated results of our REVnet matches the ground-truth captions better than the baseline method, and have a more accurate expression in the details.

5. CONCLUSION

In this paper, we propose three forms of REVnet to reconstruct the previous hidden state in the caption generation process. Our proposed REVnet gets a new state-of-the-art on three metrics(BLEU, ROUGE-L, CIDEr). The REVnet can bring backward flow into the caption generation process and help to better utilize the semantic fragments extracted by the encoder. Competitive results have shown the effectiveness of our proposed method.

6. ACKNOWLEDGMENTS

This work was supported by National Key Research and Development Program of China (Grant No.2016YFB1000902).

7. REFERENCES

[1] Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond Mooney, and Kate Saenko, "Translating videos to natural language using deep recurrent neural networks," *Computer Science*, 2015.

[2] Subhashini Venugopalan, Marcus Rohrbach, Jeff Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko, "Sequence to sequence – video to text," in *IEEE International Conference on Computer Vision*, 2015, pp. 4534–4542.

[3] Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville, "Describing videos by exploiting temporal structure," vol. 53, pp. 199–211, 2015.

[4] Atsuhiro Kojima, Takeshi Tamura, and Kunio Fukunaga, "Natural language description of human activities from video images based on concept hierarchy of actions," *International Journal of Computer Vision*, vol. 50, no. 2, pp. 171–184, 2002.

[5] Marcus Rohrbach, Wei Qiu, Ivan Titov, Stefan Thater, Manfred Pinkal, and Bernt Schiele, "Translating video content to natural language descriptions," in *IEEE International Conference on Computer Vision*, 2013, pp. 433–440.

[6] Sergio Guadarrama, Niveda Krishnamoorthy, Girish Malkarnenkar, Subhashini Venugopalan, Raymond Mooney, Trevor Darrell, and Kate Saenko, "Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition," in *IEEE International Conference on Computer Vision*, 2014, pp. 2712–2719.

[7] Girish Kulkarni, Visruth Premraj, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C Berg, and Tamara L Berg, "Baby talk: Understanding and generating simple image descriptions," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1601–1608.

[8] Youngjae Yu, Hyungjin Ko, Jongwook Choi, and Gunhee Kim, "Video captioning and retrieval models with semantic attention," 2016.

[9] M. Schuster and K. K. Paliwal, *Bidirectional recurrent neural networks*, IEEE Press, 1997.

[10] Bairui Wang, Lin Ma, Wei Zhang, and Wei Liu, "Reconstruction network for video captioning," 2018.

[11] Xinpeng Chen, Lin Ma, Wenhao Jiang, Jian Yao, and Wei Liu, "Regularizing rnns for caption generation by reconstructing the past with the present," 2018.

[12] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba, "Sequence level training with recurrent neural networks," *Computer Science*, 2015.

[13] Ronald J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine Learning*, vol. 8, no. 3-4, pp. 229–256, 1992.

[14] Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel, "Self-critical sequence training for image captioning," 2016.

[15] Dzmitry Bahdanau, Philemon Brakel, Kelvin Xu, Anirudh Goyal, Ryan Lowe, Joelle Pineau, Aaron Courville, and Yoshua Bengio, "An actor-critic algorithm for sequence prediction," 2016.

[16] Li Zhang, Flood Sung, Feng Liu, Tao Xiang, Shaogang Gong, Yongxin Yang, and Timothy M. Hospedales, "Actor-critic sequence training for image captioning," 2017.

[17] Jun Xu, Tao Mei, Ting Yao, and Yong Rui, "Msr-vtt: A large video description dataset for bridging video and language," in *Computer Vision and Pattern Recognition*, 2016, pp. 5288–5296.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.

[19] Saining Xie, Ross Girshick, Piotr Dollr, Zhuowen Tu, and Kaiming He, "Aggregated residual transformations for deep neural networks," *arXiv preprint arXiv:1611.05431*, 2016.

[20] Xinlei Chen, Hao Fang, Tsung Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollar, and C. Lawrence Zitnick, "Microsoft coco captions: Data collection and evaluation server," *Computer Science*, 2015.

[21] Qin Jin, Jia Chen, Shizhe Chen, Yifan Xiong, and Alexander Hauptmann, "Describing videos using multimodal fusion," in *ACM on Multimedia Conference*, 2016, pp. 1087–1091.

[22] Rakshith Shetty and Jorma Laaksonen, "Frame- and segment-level features and candidate pool evaluation for video caption generation," 2016.

[23] Vasili Ramanishka, Abir Das, Huk Park Dong, Subhashini Venugopalan, Lisa Anne Hendricks, Marcus Rohrbach, and Kate Saenko, "Multimodal video description," in *ACM on Multimedia Conference*, 2016, pp. 1092–1096.

[24] Yangyu Chen, Shuhui Wang, Weigang Zhang, and Qingming Huang, "Less is more: Picking informative frames for video captioning," 2018.

[25] Ramakanth Pasunuru and Mohit Bansal, "Reinforced video captioning with entailment rewards," 2017.