Postprint: Video-to-Text Research Based on Deep Bidirectional Models and Feature Fusion

Authors: Ning Peiyang, Shi Jinglun, Zhang Rongfeng, Qiu Wei

Date: 2018-11-29T00:00:00+00:00

Abstract

Automatic generation of natural language descriptions for videos is a highly challenging research hotspot. Methods based on deep BLSTM models and CNN features can learn global spatio-temporal correlation information from video sequences. To address the issues of low accuracy and high computational complexity in video-to-text conversion, a deep BMGU model is proposed, which improves computational efficiency while maintaining the structural advantages of deep BLSTM models; additionally, late fusion of CNN features from original video frames with CNN features from videos preprocessed with Haar features is performed, thereby increasing the diversity of training features and consequently enhancing the experimental performance of video-to-natural-language conversion. On the M-VAD and MPII-MD datasets, compared with the original S2VT model, the proposed method improves the METEOR scores from 6.7 and 7.1 to 8.0 and 8.3, respectively. The results demonstrate that the proposed method effectively improves the accuracy and language description quality of the original S2VT model.

Full Text

Research on Video Description Based on Deep Bidirectional Model and Feature Fusion

Ning Peiyang, Shi Jinglun, Zhang Rongfeng, Qiu Wei
(School of Electronic & Information Engineering, South China University of Technology, Guangzhou 510640, China)

Abstract: Automatically generating natural language descriptions for videos is a challenging research focus. Methods based on deep bidirectional LSTM (DBLSTM) models and CNN features can learn global spatiotemporal correlation information from video sequences. To address the issues of low accuracy and high computational complexity in video-to-text conversion, this paper proposes

a deep bidirectional minimal gated unit (BMGU) model that improves computational efficiency while maintaining the structural advantages of deep BLSTM models. Additionally, the CNN features of original video frames are fused with CNN features of video frames preprocessed with Haar features, increasing the diversity of training features and thereby improving the effectiveness of video-to-natural-language conversion. On the M-VAD and MPII-MD datasets, compared to the original S2VT model, the proposed method improves METEOR scores from 6.7 and 7.1 to 8.0 and 8.3, respectively. The results demonstrate that the proposed method effectively improves the accuracy and language description quality of the original S2VT model.

Keywords: video to text; deep bidirectional model; Haar feature; feature fusion; convolutional neural networks

0 Introduction

Video captioning, also known as automatically generating natural language descriptions for videos, primarily involves understanding and analyzing video content to extract useful semantic information, and then associating this semantic information with the application context to convert video frame sequences into natural language descriptions. This technology has high application value and practical significance in various fields such as intelligent security, human-computer interaction, and video retrieval.

With the gradual extension of deep learning into many areas of computer vision, video-to-text methods represented by S2VT (Sequence to Sequence-Video to Text) have significantly outperformed previous non-deep learning methods in performance, but several aspects still require improvement. For instance, to obtain semantic information contained in video frames, CNN models are generally used to extract convolutional features from video frames, which contain spatial information of the frames. However, video frames in video description datasets often have complex backgrounds (containing multiple objects), and the performance of some CNNs models in extracting features from such frames decreases, leading to inaccurate natural language descriptions from video-to-text methods.

Additionally, LSTM is the core model of the S2VT method. It improves upon the gate-less structure of RNNs (Recurrent Neural Networks) by adopting a structure with three gates and two hidden states, effectively overcoming the problems of gradient vanishing or explosion, and thus possesses good capability for learning and modeling long sequence information. However, LSTM adds a large number of parameters, reducing computational efficiency and making it unsuitable for applications with high real-time requirements and strict computational constraints. Furthermore, recent experiments by Chung et al. have shown that more gate structures do not necessarily mean better final performance; some simpler RNN models can achieve better results than LSTM while reducing computational complexity.

To address the issues of low description accuracy and high computational complexity in the S2VT method, this paper proposes a video-to-text approach based on deep bidirectional recurrent neural networks and Haar features. Specifically: First, to address the problem that the S2VT model's unidirectional LSTM-based encoding layer cannot fully utilize temporal information from both previous and future frames, a deep bidirectional LSTM-based video-to-text method is proposed to learn global temporal correlation information. Second, to address the issue that complex backgrounds in video frames affect the extraction of features from main objects, a video frame enhancement method based on Haar feature preprocessing is proposed. Before extracting implicit features from video frames using convolutional neural networks such as VGG, Haar features are extracted to preprocess the video frames to suppress complex background information and enhance main object information. Third, to address the high computational complexity of deep BLSTM, a deep BMGU-based video-to-text method is proposed. Experiments show that this simplified model-based approach can effectively improve computational efficiency while achieving natural language description performance comparable to the deep BLSTM model.

1 Principles of Video Captioning and the S2VT Model

Video captioning can be mathematically formulated as: given a video frame sequence $X(x_1, x_2, ..., x_t, ..., x_n)$, generating the conditional probability of a word sequence $Y(y_1, y_2, ..., y_t, ..., y_m)$ that summarizes the video' s semantic information, i.e.,

$$p(Y|X) = p(y_1, y_2, ..., y_m|x_1, x_2, ..., x_n) \quad (1)$$

where the frame sequence length n and word sequence length m are variable, generally with $n \neq m$ and $n > m$. Video-to-text methods based on recurrent neural networks construct an 'encoder-decoder' model to achieve joint modeling of frame sequences and word sequences using implicit features. Accordingly, Equation (1) can be rewritten as:

$$p(Y|X) = \prod_{t=1}^{m} p(y_t|h_{n+t}) \quad (2)$$

S2VT is a classic LSTM-based video-to-text model that can generate natural language sentences to describe events occurring in videos.

[Figure 1: see original paper] Schematic diagram of S2VT model

As shown in Figure 1, the S2VT model obtains convolutional features (CNNs features) of the input video sequence through the VGG-16 network, then inputs the feature sequence chronologically into the first LSTM layer for feature modeling. In the second LSTM layer, the LSTM network learns the mapping relationship between frame sequences and word sequences to complete the joint

modeling of features and language. Additionally, <Pad> in the figure indicates using all-zero vectors as input to pad corresponding position information, while inputting <BOS> signals that the frame sequence input is complete, instructing the model to switch from the encoding phase to the decoding phase (i.e., begin predicting word sequences). <EOS> indicates that the S2VT model has finished outputting the predicted word sequence.

LSTM is the core algorithm for the S2VT model to implement the function in Equation (2). Specifically, assuming the input variable at time t is x_t, the corresponding hidden layer state parameter is h_t, and the memory cell state is c_t, the formulas in the LSTM unit at time t are as follows [5]:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \quad (3)$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \quad (4)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \quad (5)$$

$$g_t = \phi(W_{xg}x_t + W_{hg}h_{t-1} + b_g) \quad (6)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \quad (7)$$

$$h_t = o_t \odot \phi(c_t) \quad (8)$$

In Equations (3)-(8), i, f, o, g represent the input gate, forget gate, output gate, and input modulation gate of LSTM, respectively, with corresponding bias vectors b_i, b_f, b_c, b_g. $h_t \in \mathbb{R}^n$ represents n hidden state parameters. W_{ab} ($a \in \{x, h\}$, $b \in \{i, h, o, g\}$) denotes the weight matrix from input or hidden layer state parameter a to gate b. $\sigma(x)$ is the sigmoid function, $\phi(x)$ is the hyperbolic tangent function, and $\odot$ is element-wise product operation. Through Equations (3)-(8), the S2VT model iteratively calculates the hidden layer parameters $h_1, h_2, ..., h_t, ..., h_n$, and further computes the conditional probability $p(y_t|h_{n+t})$ of hidden layer parameters with respect to word y_t ($t = 1, 2, ..., m$), thereby obtaining the predicted word sequence.

2 Proposed Improved Methods

2.1 Video-to-Text Method Based on DBLSTM and Haar Feature Preprocessing

First, to address the insufficient utilization of video frame features by the unidirectional LSTM encoding layer in the S2VT model, we improve the method by employing deep bidirectional LSTM (DBLSTM) networks. The schematic diagram of the video-to-text method based on deep bidirectional LSTM is shown in Figure 2 [Figure 2: see original paper].

[Figure 2: see original paper] Schematic diagram of video-to-text based on depth bidirectional LSTM

A BLSTM consists of one LSTM that transmits information forward and another that transmits information backward. Two BLSTMs are then connected as shown in Figure 3 [Figure 3: see original paper] to form a DBLSTM. For the fusion of hidden layer state parameters from the two LSTM layers, the following method is generally used [7]:

$$h_t = H(W_{xh}x_t + W_{hh}h_{t-1} + b_h) \quad (9)$$

$$\overrightarrow{h_t} = H(W_{xh}x_t + W_{hh}\overrightarrow{h_{t-1}} + b_h) \quad (10)$$

$$\overleftarrow{h_t} = H(W_{xh}x_t + W_{hh}\overleftarrow{h_{t+1}} + b_h) \quad (11)$$

$$y_t = W_{\overrightarrow{h}y}\overrightarrow{h_t} + W_{\overleftarrow{h}y}\overleftarrow{h_t} + b_y \quad (12)$$

where $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ are the hidden layer state parameters of the forward and backward transmission layers at time t, respectively, $H(x)$ is the activation function of LSTM, and y_t is the fused output. The meanings of other parameters are similar to those in Equations (3)-(8).

By stacking two BLSTM networks to form a deep BLSTM network, this improved deep RNN network structure has the following advantages: First, BLSTM can learn not only information from previous frames but also information from future frames. Through correlation learning of both past and future frames and contextual association learning, global temporal information in videos can be utilized to enhance the learning of video-sentence pairs, thereby improving the accuracy of video captioning. This effectively overcomes the limitation of unidirectional LSTM that can only utilize information from previous frames. Second, in deep neural networks, widening and deepening the network are two main directions for optimizing and improving model performance. Corresponding to the spatial depth of CNNs, LSTM is a network with temporal depth. Bidirectional LSTM (BLSTM) is deeper in time than unidirectional LSTM, thus BLSTM enhances temporal dependencies compared to LSTM, and deep BLSTM further strengthens these temporal dependencies. Increasing temporal depth adds network parameters and enhances the association learning between video and natural language during training, thereby further improving the performance of video-to-text (as intuitively reflected by higher METEOR scores). However, deeper networks contain more parameters, which also increases computational complexity.

Second, to address the issue that complex video backgrounds may affect CNN' s extraction of main object features, we separate video frames by RGB channels, perform first-order Haar wavelet filtering on each channel to remove detail information, and finally recombine them to obtain video frames containing Haar features. The process is shown in Figure 3 [Figure 3: see original paper].

[Figure 3: see original paper] Flowchart of extracting Haar features from video frames

As can be seen from comparing the images before and after processing in Figure 3, the main object information in the video frames is preserved and enhanced after Haar feature extraction, while relatively complex background information is weakened, thereby providing certain semantic information. Extracting CNNs features from these video frames yields CNNs features containing Haar features, which are new features different from the CNNs features of original video frames. Meanwhile, since effective feature fusion can often improve the accuracy and language quality of video captioning [11], this paper also fuses the CNNs features of original video frames with the CNNs features containing Haar features, thereby increasing the variety of training features and enhancing the richness of video feature learning to optimize learning effectiveness. As shown in Figure 5 [Figure 5: see original paper], this paper fuses the CNNs features of original video frames with the CNNs features containing Haar features to improve the performance of video-to-natural-language conversion.

2.2 Video-to-Text Method Based on DBMGU and Haar Feature Preprocessing

Applying the DBLSTM model to video-to-text improves performance, but it should be noted that, as mentioned earlier, the significant increase in model parameters leads to higher computational complexity, often increasing training time and making it unsuitable for real-time applications. To address these issues, we utilize the simplicity of the MGU (Minimal Gated Unit) model to reduce computational parameters and training time. Since extracting Haar features from video frames can often provide good semantic information, we propose a video-to-natural-language method based on the DBMGU model and feature fusion.

The Minimal Gated Unit (MGU) is a simplified RNNs model with only one gate structure, hence the name 'Minimal Gated Unit' . The calculation formulas for the MGU unit at time t are as follows (the symbols have the same meaning as in Equations (3)-(8)):

$$f_t = \sigma(W_{fh}h_{t-1} + W_{fx}x_t + b_f) \quad (13)$$

$$\tilde{h_t} = \phi(W_{hh}(f_t \odot h_{t-1}) + W_{hx}x_t + b_h) \quad (14)$$

$$h_t = (1 - f_t) \odot h_{t-1} + f_t \odot \tilde{h_t} \quad (15)$$

The MGU model has far fewer parameters than LSTM (approximately half under the same conditions), and theoretically its computational complexity is significantly lower than LSTM, thus effectively reducing computational overhead and improving training speed. Second, Chung et al.'s research shows that RNNs networks with gate structures generally achieve significantly better experimental results than simple RNNs networks using only hyperbolic tangent functions without gate structures [6]. The MGU model follows this conclusion, retaining the necessary single gate structure to ensure effective learning of sequential data.

To intuitively demonstrate the difference in computational complexity within one time step between LSTM and MGU, referring to the research of Zhou Guobing et al. [10], we illustrate their unit structures in Figure 4 [Figure 4: see original paper].

[Figure 4: see original paper] Calculating process of LSTM and MGU within a time step

The DBMGU is constructed in the same way as DBLSTM. The video-to-text method based on the DBMGU model and feature fusion replaces the DBLSTM model with the DBMGU model, while also using features preprocessed with Haar features to improve model effectiveness for comparison purposes.

3 Experimental Results and Analysis

3.1 Experimental Setup

This paper implements the experimental model using the Caffe [12] (Convolutional Architecture for Fast Feature Embedding) deep learning framework. The model is trained and tested independently on two mainstream video captioning datasets: M-VAD and MPII-MD.

To fully utilize sample information, during training we comprehensively consider the length of frame sequences and word sequences for each sample, adaptively extracting an appropriate number of video frames; during testing, we sample video frames based only on the length of each sample' s frame sequence. For the comparison and evaluation of annotated sentences and sentences generated by our method, we use the METEOR evaluation metric as an objective evaluation criterion for the output sentences of our method.

METEOR [13] is an evaluation metric proposed by Lavir et al. in 2004 after discovering the significance of recall in evaluation metrics. Their research shows that metrics considering recall have higher correlation with human judgment compared to those based solely on precision. Therefore, the METEOR evaluation metric is often used as a reference for evaluation in machine translation, image captioning, video captioning, and other fields. For example, Rohrbach et al. [14] used METEOR as an objective evaluation metric in their video captioning research.

The main hyperparameters for model training are shown in Table 1 . Additionally, the initial learning rate is 0.01, and the learning rate adjustment method is: reduce the learning rate to half every 20,000 iterations; the training optimization method is SGD with Mini-Batch, momentum set to 0.9; the regularization method is Dropout.

Hyper parameter of model training

Referring to the experimental analysis method of Rohrbach et al. [15], we evaluate model performance through two perspectives: METEOR metric evaluation

and comparative analysis of annotated sentences and sentences generated by the two methods.

First, between 40,000 and 60,000 iterations, we evaluate models at even thousand iterations and compile the relationship between METEOR evaluation scores and iteration numbers for the two methods, as shown in Figure 6 [Figure 6: see original paper].

[Figure 6: see original paper] Relationship between METEOR evaluation scores and iterations of two methods

3.2 Experimental Results Analysis

Analysis of Figure 6 shows that the METEOR evaluation scores of both methods are relatively stable, indicating that they can converge well under different datasets. We compile the peak METEOR evaluation scores of the two methods from Figure 6 and compare them with the peak METEOR evaluation scores of other video-to-text methods, with results shown in Table 2 . METEOR is measured in %, with higher values indicating better performance.

METEOR Evaluation of M-VAD and MPII-MD datasets

Analysis of Table 2 shows that for the M-VAD dataset, the two methods proposed in this paper improve the METEOR score from 6.7% of the original S2VT model to 7.9% and 8.0%, respectively. Similarly, on the MPII-MD dataset, the two methods improve the score from 7.1% of the original S2VT model to 8.1% and 8.3%, respectively. On one hand, the METEOR evaluation scores of both methods are higher than those of previous video-to-text methods, demonstrating that the organic combination of deep bidirectional models and feature fusion can improve the accuracy and language quality of video captioning. On the other hand, the two methods proposed in this paper use DBLSTM and DBMGU models respectively for video feature modeling. Using the DBMGU model, compared to the DBLSTM model, the METEOR score not only does not decrease but even slightly increases, indicating that although the DBMGU model has nearly half the parameters of the DBLSTM model, the language effect of the generated sentences is similar to that of DBLSTM. More importantly, the DBMGU model can effectively reduce computational complexity, lower computational overhead, and thus improve video-to-text speed.

This paper conducts a comparative analysis of annotated sentences and sentences generated by the two methods. Due to space limitations, three examples from each dataset are selected as illustrations, as shown in Figure 7 [Figure 7: see original paper].

[Figure 7: see original paper] Video description examples from M-VAD and MPII-MD datasets

As can be seen, the sentences generated by both methods are not only accurate in description but also contain more detailed information and increased

language richness compared to the annotated sentences, verifying that the proposed methods can effectively improve the accuracy and language quality of video captioning.

4 Conclusion

To address the issue of low description accuracy in the S2VT method, this paper proposes a video-to-natural-language method based on DBMGU and feature fusion, building upon the DBLSTM and feature fusion approach. The proposed method effectively improves the accuracy and language quality of the original S2VT model. The DBMGU model has only about half the number of parameters as the DBLSTM model, reducing computational overhead and improving computational speed, yet achieving language description performance comparable to the DBLSTM model, making the proposed method applicable to a wide range of scenarios. Of course, the current work still has some limitations. In future research, we will further improve the decoding model and language model aspects of the S2VT method.

References

[1] Kojima A, Izumi M, Tamura T, et al. Generating natural language description of human behavior from video images [C]// Proc of the 15th International Conference on Pattern Recognition. Washington DC: IEEE Computer Science, 2000: 728-731.

[2] Venugopalan S, Rohrbach M, Donahue J, et al. Sequence to sequence-video to text [C]// Proc of IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2015: 4534-4542.

[3] Donahue J, Hendricks L A, Rohrbach M, et al. Long-term recurrent convolutional networks for visual recognition and description [J]. IEEE Trans on Pattern Analysis and Machine Intelligence. 2017, 39 (4): 677-691.

[4] Jozefowicz R, Zaremba W, Sutskever I. An empirical exploration of recurrent network architectures [C]// Proc of the 32nd International Conference on Machine Learning. New York: ACM Press, 2015: 2342-2350.

[5] Hochreiter S, Schmidhuber J. Long short-term memory [J]. Neural Computation. 1997, 9 (8): 1735-1780.

[6] Chung J, Gulcehre C, Cho K H, et al. Empirical evaluation of gated recurrent neural networks on sequence modeling [EB/OL]. (2015) [2018-06-01]. https://arxiv.org/abs/1412.3555.

[7] Graves A, Jaitly N, Mohamed A R. Hybrid speech recognition with Deep Bidirectional LSTM [C]// Proc of IEEE Workshop on Automatic Speech Recognition and Understanding. Piscataway, NJ: IEEE Press, 2013: 273-278.

[8] Papageorgiou C P, Oren M, Poggio T. A general framework for object detection [C]// Proc of IEEE International Conference on Computer Vision. Piscataway, NJ, IEEE Press, 2002: 555-562.

[9] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition [EB/OL]. (2014) [2018-06-01]. https://arxiv.org/abs/1409.1556.

[10] Zhou Guobing, Wu Jianxin, Zhang Chenlin, et al. Minimal gated unit for recurrent neural networks [J]. International Journal of Automation and Computing. 2016, 13 (3): 226-234.

[11] 梁锐, 朱清新, 廖淑娇, 等. 基于多特征融合的深度视频自然语言描述方法 [J]. 计算机应用. 2017, 37 (4): 1179-1184. (Liang Rui, Zhu Qingxin, Liao Shujiao, et al. Deep natural language description method for video based on multi-feature fusion [J]. Journal of Computer Applications. 2017, 37 (4): 1179-1184.)

[12] Jia Yangqing, Shelhamer E, Donahue J, et al. Caffe: convolutional architecture for fast feature embedding [C]// Proc of the 22nd ACM International Conference on Multimedia. New York: ACM Press, 2014: 675-678.

[13] Denkowski M, Lavie A. Meteor universal: language specific translation evaluation for any target language [C]// Proc of the 9th Workshop on Statistical Machine Translation. Cambridge, MA: MIT Press, 2014: 376-380.

[14] Rohrbach A, Rohrbach M, Schiele B. The long-short story of movie description [C]// Proc of German Conference on Pattern Recognition. Berlin: Springer, 2015: 209-221.

[15] Rohrbach A, Rohrbach M, Tandon N, et al. A dataset for movie description [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. New York: IEEE Press, 2015: 3202-3212.

[16] Venugopalan S, Xu Huijuan, Donahue J, et al. Translating videos to natural language using deep recurrent neural networks [C]// Proc of Annual Conference of the North American Chapter of the ACL. Cambridge, MA: MIT Press, 2015: 1494-1504.

Note: Figure translations are in progress. See original paper for figures.