Multi Semantic Feature Fusion Framework for Video Segmentation and Description

Rui Liang [a], Qingxin Zhu

School of Information and Software Engineering, University of Electronic Science and Technology of China, Sichuan Chengdu, 610054, China

[a]1662723894@qq.com

Keywords: Video Semantic Analysis, Video Segmentation and Description, Deep Learning, Multi Feature Fusion.

Abstract. It is a difficult task to make machine understanding video and describe it in natural language. In the reality, videos are much longer than these video clips in research experiments, each video contains multi parts of semantic. It is a challenge work to describe a long video, it requires to control the granularity of the video's semantics, exclude redundancy information and give complete description. This task is very important for video understanding and video retrieving. In the paper, we proposed a framework to solve these problems. The framework consists of two stage: video segmentation and video description, the two stage can divide into five steps, firstly extracts features of video sequence with pre-trained deep learning models, secondly fuse different features of a same frame into a feature vector with a weight vector, thirdly generates a histogram of similarity (HOS) of adjacent frames' feature vectors in sequence, fourthly uses a threshold t to divide the video into short fragments of different semantic, finally uses LSTM networks which take frame sequences' features of each fragment as input and output natural language description for each fragment. Our research handles the 'in-the-wild' long videos, it can enhance the comprehensibility of long video, it is meaningful in the task of understanding and describing video.

1. Introduction

Description of images and videos is a fundamental challenge of computer vision. Recently the description with natural language text has received increased interest, describing both images[1-5]and videos[6-7] with a single sentence. Several previous methods for generating sentence descriptions contains two stage[8-10]which first extracts semantic components (such as subject, verb, object, scene, etc.) in video clips, then generates sentences with a fixed template. Some methods use a two steps way[11-13], the first step is to generate a fixed length vector representation of frames by extracting features from different CNN, the second step is to decode the vector into a sequence of words as the description of the video clip.

Early researches paid more attention on short video clips. For a long video, it is an easy task for most people to tell the segmentation points of different semantics and describe it in natural language, but it's hard for a machine. For the semantic analyzing of open domain videos, the difficulty lies on the understanding of different and abundant semantic elements and determine the boundary of each semantic fragment in videos.

In order to solve long video understanding problem, in this paper, we conducted research on video semantic segmentation and description based on multi features fusion and LSTM networks and proposed a two stage framework. Our approach is inspired by recent breakthroughs reported by Rakshith Shetty and Jorma Laaksonen in video-to-text generation [14], their research won the LSMDC 2015. They applied a multi fusion model for video-to-text generation, in our paper we use different way to fuse the features, in the first stage they are used for semantic segmentation, in the second stage they are used for better generating natural language descriptions, we will discuss the framework in detail in later parts.

2. Approach

In this paper, we proposed a framework (see Figure 1.) to divide video into semantic fragments and describe videos with natural language based on feature fusion and LSTM networks. In this framework we take video frames $(x_1, x_2, \dots, x_n)$ as input, divide the frames into fragments $(x_{11}, x_{12}, \dots, x_{1n_1})$, $(x_{21}, x_{22}, \dots, x_{2n_2})$, …, $(x_{m1}, x_{m2}, \dots, x_{mn_m})$, then try to generate text description of these fragments, and we can get outputs like $(y_{11}, y_{12}, \dots, y_{1k_1})$, $(y_{21}, y_{22}, \dots, y_{2k_2})$,… , $(y_{m1}, y_{m2}, \dots, y_{mk_m})$. The input and output length of all fragments are different naturally, because the length of description depends on the semantics in each fragment. Normally, there are more frames then words in a description. In the framework we estimate the conditional probability of an output sequence $(y_{m1}, y_{m2}, \dots, y_{mk_m})$ for each input fragment $(x_{m1}, x_{m2}, \dots, x_{mn_m})$ i.e.

$$p(y_{m1}, y_{m2}, \dots, y_{mk_m} | x_{m1}, x_{m2}, \dots, x_{mn_m}) \quad (1)$$

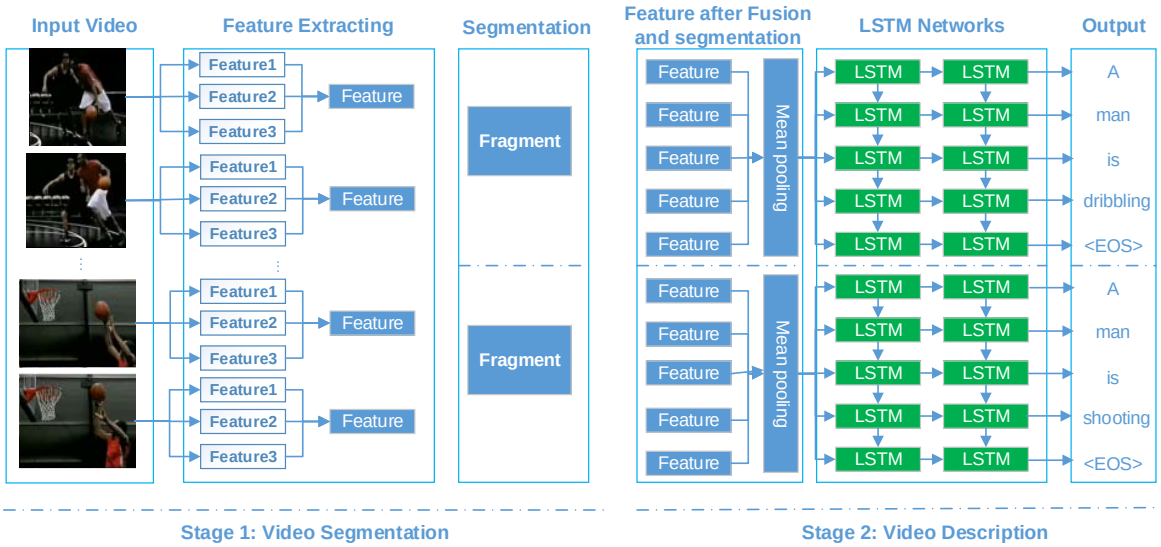

Figure 1. Deep Video Semantic Segmentation and Description Framework

In our framework we use Long Short-Term Memory (LSTM) network to describe the sequence, because of the sequential nature in time, LSTM is suited for generating natural language description of semantic events in videos, and LSTM has achieved great success on language translation [15] and speech recognition [16], in recent years. Our framework consists of two stages. The first stage called segmentation stage, in this stage we extract sequence feature by different models, combine these features to a new feature vector with a weight vector, generate histogram of similarity (HOS) based on the new semantic features of video sequence, divide video into segments by the similarity of adjacent frames. The second stage called description stage, in this stage we use the new feature vector of video fragments divided in the first stage as input of the LSTM network, then we can get natural language descriptions of each fragments. After the two stages we can finally get the segmentation flag and description of each fragments. By processing videos in this way, we can increase the accuracy of video description, and reduce the redundancy information of videos. Next we will look at each of these processing stages in detail.

Sequence feature extracting. We extract image features in video sequence. For the feature extraction we use pre-trained CNNs models on the ImageNet database [17]. We use three famous and effective CNN architectures namely Alexnet and GoogLeNet [18] and 16-layer and 19-layer VGG [19] . The output of these models will be the input to LSTM networks after fusion. In our work, when using Alexnet and VGG model to extract features, we use the output of the fc7 layer which is the second fully connected layer, the fc7 layer is 4096-dimensional, and we followed the suggestion in [20], extracting features of ten regions for each image. When using GoogleNet we used the 5th inception module, then augment these features with two scale levels[21], after average and maximum poolling, finally got 2048-dimensional features[22].

Feature Fusion. In order to use different features extracted by different model to make the description of video much accurate, we have to fuse these features of each frame to get a new feature vector. Consider we used n models to extract features, we can get a set of feature vectors $F_1, F_2, \dots, F_n$. To fuse these feature vector. Firstly, normalization features extracted by different models with sigmoid function, get new feature vectors $F_1', F_2', \dots, F_n'$. Secondly, stitch these vectors to one new feature vector with weight vector $W = (W_1, W_2, \dots, W_n)$, finally we get

$$F_{new} = (W_1, W_2, \dots, W_n) \begin{pmatrix} F_1' \\ F_2' \\ \dots \\ F_n' \end{pmatrix} \tag{2}$$

the weight vector W should be finetuned during the training period.

HOS (histogram of similarity). We take $k(k \geq 2)$ frame features from F_{new} each time, calculate the Mahalanobis distance of any pairs in the k frame features, we get C_k^2 distance, calculate the mean of the distances as d_i , each time we stride p frames before taking k frame features to calculate the d_i, finally we will get a $\left\lfloor \frac{n-k}{p} \right\rfloor$ elements vector D $\left(d_1, d_2, \dots, d_i, \dots, d_{\left\lfloor \frac{n-k}{p} \right\rfloor} \right)$ which we called the HOS, the parameter k, p should be finetuned during the training period.

Video semantic segmentation. The main task is to choose proper segmentation positions, according to the HOS we can easily tell an approximate region where the distance changed rapidly, but for the machine, we should tell it a precise position, so we set a threshold t, when the absolute value of difference of two adjacent distance value if great then t, then we set a segmentation point here.

Video description. After video semantic segmentation, we choose LSTM as the generative model of sentences based on the fusion features described in the previous part. The reason we choose LSTM is based on threes requirements this problem encountered. Firstly, the model can process and generate sentences of arbitrary length. Secondly, the model should be able to learn long-range temporal dependencies. Thirdly, during training, using gradient descent methods, the error signal and its gradients need to propagate a long way back in time without exploding. LSTMs satisfy all three requirements. In our work, we provided a two layer LSTM network, the first LSTM layer receives frame features and encodes them, the second LSTM layer receives the hidden representation(h_t) and decodes it to a sequence of words. During the decoding stage, the model maximizes the log-likelihood of the predicted output sentence of h_t and previous words. Take θ as the parameter and $Y = (y_1, y_2, \dots, y_m)$, the model can formulate as:

$$\theta^* = \underset{\theta}{argmax} \sum_{t=1}^{m} \log p(y_t | h_{n+t-1, y_{t-1}}; \theta) \tag{3}$$

Figure 2 shows the LSTM unit[23] used in our work. A LSTM unit contains a memory cell m, whose value at any timestep t is influenced by the input x, previous output y and previous cell state m_{t-1}. There are three gates, input gate and forgot gate controls the update of m, the output gate controls the output, the forget gate allows the LSTM to forget its previous memory m_{t-1}, and the output gate decides how much of the memory to transfer to the hidden state (m_t).This process is formalized in the equations below:

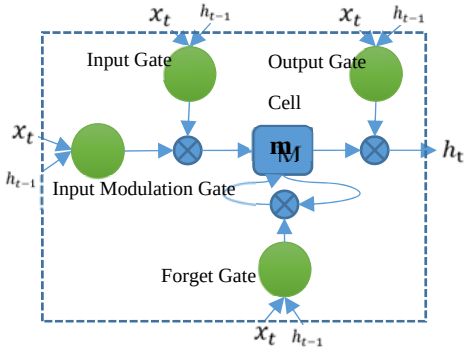

Figure 2. LSTM Unit

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1}) \quad (4)$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1}) \quad (5)$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1}) \quad (6)$$
$$m_t = f_t \odot m_{t-1} + i_t \odot \emptyset(W_{xm}x_t + W_{hm}h_{t-1}) \quad (7)$$
$$h_t = o_t.\emptyset(c_t) \quad (8)$$

3. Conclusion

In this paper, we described a multi semantic feature fusion framework for video segmentation and description. In the reality, videos are long and contain rich semantics. Former model can only handle a short video slice, so we presented a technique which use three pre-trained deep learning models to extract features of video frames, fuse these features to a new feature vector for each frame, generate a HOS of adjacent frames' feature in the video sequence, divide video sequence into several fragments at the point where there is a huge difference in the HOS, input the fusion features of each fragment to an LSTM networks to generate description. Although this novel framework can solve long video understanding problem, but still there is a big gap with the practical application. For a man, he can learn from a single picture to know an object and recognize its' behavior in real world or in videos, in our opinions, the main reason lies on the richness of features, so we should fuse more features to get better result.

Acknowledgements

This paper is financially supported by the National Natural Science Foundation of China (Grant No. 61300192). The Fundamental Research Funds for the Central Universities (Grant No. ZYGX2014J052)

References

[1]. X. Chen and C. L. Zitnick. Learning a recurrent visual representation for image caption generation. CVPR, 2015.
[2]. A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. CVPR, 2015.
[3]. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. CVPR, 2015.
[4]. R. Kiros, R. Salakhutdinov, and R. S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. arXiv:1411.2539, 2014.
[5]. J. Mao, W. Xu, Y. Yang, J. Wang, and A. L. Yuille. Deep captioning with multimodal recurrent neural networks (m-RNN).arXiv:1412.6632, 2014.
[6]. J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach,S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description.In CVPR, 2015.
[7]. M. Rohrbach, W. Qiu, I. Titov, S. Thater, M. Pinkal, and B. Schiele. Translating video content to natural language descriptions. In ICCV, 2013.
[8]. S. Guadarrama, N. Krishnamoorthy, G. Malkarnenkar,S. Venugopalan, R. Mooney, T. Darrell, and K. Saenko.Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shoot recognition.In ICCV, 2013.
[9]. N. Krishnamoorthy, G. Malkarnenkar, R. J. Mooney,K. Saenko, and S. Guadarrama. Generating natural-language video descriptions using text-mined knowledge. In AAAI,July 2013.
[10]. J. Thomason, S. Venugopalan, S. Guadarrama, K. Saenko,and R. J. Mooney. Integrating language and vision to generate natural language descriptions of videos in the wild. In COLING, 2014.
[11]. S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, K. Saenko.Sequence to Sequence -- Video to Text.In ICCV,2015.

[12]. S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach,R. Mooney, and K. Saenko. Translating videos to natural language using deep recurrent neural networks. In NAACL,2015.
[13]. I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In NIPS, 2014.
[14]. S. Rakshith, L. Jorma.Video captioning with recurrent networks based on frame- and video-level features and visual content classification.arXiv:1512.02949v1,2015.
[15]. I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In NIPS, 2014.
[16]. A. Graves and N. Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In ICML, 2014.
[17]. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database.In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009.
[18]. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed,D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. arXiv, abs/1409.4842,2014.
[19]. K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv, abs/1409.1556, 2014.
[20]. A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet classification with deep convolutional neural networks. In NIPS, 2012.
[21]. Y. Gong, L. Wang, R. Guo, and S. Lazebnik. Multi-scale orderless pooling of deep convolutional activation features. arXiv.org:1403.1840, 2014.
[22]. M. Koskela and J. Laaksonen. Convolutional network features for scene recognition. In Proceedings of the 22nd ACM International Conference on Multimedia, Orlando, Florida, 2014.
[23]. Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In Advances in Neural Information Processing Systems (NIPS).