CLIP4Caption ++: Multi-CLIP for Video Caption

Mingkang Tang¹,², Zhanyu Wang¹, Zhaoyang Zeng¹, Fengyun Rao¹, Dian Li¹
¹Kandian Content AI Lab, Platform and Content Group, Tencent
²Tsinghua Shenzhen International Graduate School, Tsinghua University
mktang,zhanyuwang@tencent.com

Abstract

This report describes our solution to the VALUE Challenge 2021 in the captioning task. Our solution, named CLIP4Caption++, is built on X-Linear/X-Transformer, which is an advanced model with encoder-decoder architecture. We make the following improvements on the proposed CLIP4Caption++: 1) we utilize three strong pre-trained CLIP models to extract the text-related appearance visual features. 2) we adopt the TSN sampling strategy for data enhancement. 3) we involve the video subtitle information to provide richer semantic information. 4) we design word-level and sentence-level ensemble strategies. Our proposed method achieves 86.5, 148.4, 64.5 CIDEr scores on VATEX, YC2C, and TVC datasets, respectively, which shows the superior performance of our proposed CLIP4Caption++ on all three datasets.

1. Introduction

Video captioning is an advanced multi-modal task that automatically generates a natural language sentence of a given video clip. Currently, numerous SOTA approaches for multi-modal tasks (such as image-text retrieval, image captioning, etc.) adopt the paradigm of "pre-training + fine-tune." Common pre-training tasks include mask language model (MLM), mask region model (MRM), video-text matching (VTM), etc., which have been proven very helpful to improve downstream tasks. The essence of those pre-training tasks is to align the two modalities in the metric space. From this point of view, in feature selection, we utilize multiple strong CLIP pre-training models to extract video appearance features because CLIP's pre-training model has the natural ability to align image and text in its feature space since it has trained on hundreds of millions of image-text pairs. Specifically, we use three CLIP pre-training models for video appearance features, SlowFast pre-training model for video motion features, and discard the S3D and Resnet features that are officially provided. We did not use the pre-training method or add additional data in our whole training process. We also made several effective improvement measures: data enhancement, introducing subtitle information to enhance the feature, an ensemble strategy designed especially for the generation tasks. The proposed CLIP4Caption++ solution won the championship on the VALUE [5] competition captioning task.

2. Methods

Figure 1 illustrates the framework of our proposed CLIP4Caption++ for video captioning. First, we use the pre-trained model of CLIP and SlowFast to extract visual features from origin video (2.1) (higher part in Fig. 1). Second, we sampled the extracted frame features using TSN [12] sampling (2.2). And then, video feature and text, as well as the token type, were embedded by embedding layer for bimodal feature fusion (2.3) (lower part in Fig. 1). For ensemble, we train multiple caption models with the different settings and ensemble the output of multiple models for a final strong result (2.4). Details will be elaborated as follows.

2.1. Feature Extraction

We use the pre-trained CLIP and SlowFast as our video feature extractor. We extract CLIP-based video features at 10 FPS. CLIP [9] was pre-trained on image-text matching with 400 million image-text pairs, making it vital to learn image representations. Furthermore, CLIP has two different architectures for the image encoder. For ResNet architecture, we chose 16x the compute of a ResNet-50, denoted as CLIP(RN50x16); For the Vision Transformers architecture, we chose ViT-B/32 and ViT-B/16 as the image encoder, denoted as CLIP(ViT-B/32) and CLIP(ViT-B/16) separately.

SlowFast [2] was well designed for video recognition, capturing spatial semantics and motion in different frame rates. We chose SlowFast 8x8, ResNet-50 as our feature extractor; the Slow pathway and Fast pathway take 8 and 32 frames as the network input separately, both sampled from a 64-frame raw clip. We sampled each 64-frame raw clip in a sliding time window. The length of the time window is...
1.5 seconds, and the move stride is 0.3 seconds. The sliding
time window allows more clips so that SlowFast can sense
a longer time range of video frames without reducing the
total number of extracted features.

2.2. TSN Sampling Strategy

After feature extraction, we sample the extracted frames
according to TSN [12]. During training, TSN sampling
divides all frames into K segments of equation duration
and then randomly samples one frame from each segment,
thus increasing the sample randomness on the limited frame
length. After using TSN sampling, the frames selected in
each epoch are inconsistent, accordingly achieving data en-
hancement for video data. The TSN sampling is not used in
testing, as we evenly extract K frames from all frame fea-
tures at equal intervals.

2.3. Bimodal Feature Fusion

Frames sampling makes the four different features to the
same frame length so that we can concatenate them to the
final 4096-dim visual video feature. As shown in the Fig. 1,
the visual video feature was embedded by the video embed-
ding layer. We find that the video subtitle can provide rich
semantic information, which can benefit the caption gener-
at ion progress. Thus we also leverage the video subtitle to
enhance our models. The subtitle sentences are tokenized
and embedded by the text embedding layer. Finally, the text
embedding and video embedding are concatenated as the X-
Linear/X-Transformer input. For making the model distin-
guish between these two embeddings, a particular embed-
dding layer, token type embedding layer, was added. First,
we set the video label as 0 and the text label as 1. And
then, the token type embedding layer is used to embedded
video labels and text labels. After that, token type embed-
dings are added to the concatenated video embedding and
text embedding.

2.4. Ensemble Strategy

In order to achieve a more powerful caption result, we
design a word-level ensemble strategy and a sentence-level
strategy for the captioning task. For word-level strategy, we
ensemble the output of all the models in each generation
step. During the generation, the captioning model gener-
ates one word according to the softmax output and takes
the word as the input for the next step. We averaged the
softmax output of all models and chose the word with max
probability in each step. The model selects the correct word
with a greater probability after considering all the results of
multiple models, reducing the impact of incorrect words on
subsequent generated results.

For sentence-level strategy, we utilize the CIDEr met-
ric as the “importance score” of a generated sentence and
select the highest score to compose the final result. Math-
e matically, taking the predicted captions of one video from $n$
Datasets | Model | Public Test | Private Test |
|---------|-------|-------------|--------------|
|         |       | BLEU-4 | ROUGE-L | CIDEr | CIDEr |
| TVC [4] | single model | 13.43 | 35.73 | 61.20 | 59.41 |
|         | ensemble | 15.03 | 36.86 | 66.02 | 64.49 |
| YC2C [14] | single model | 13.03 | 39.78 | 136.01 | 133.98 |
|         | ensemble | 13.60 | 41.43 | 148.91 | 148.39 |
| VATEX [13] | single model | 38.66 | 53.61 | 81.95 | 83.67 |
|         | ensemble | 40.59 | 54.45 | 85.71 | 86.53 |

Table 1. Best single model and multiple model ensemble results on the TVC, YC2C and VATEX test set of public and private.

| Method | CIDEr gain(%) |
|--------|---------------|
| C32+SF | 65.15 -       |
| C32+SF+ST | 65.67 0.8    |
| C32+C16+C50+SF+ST | 69.03 6.0 |
| C32+C16+C50+SF+ST+TSN | 69.23 6.3 |
| C32+C16+C50+SF+ST+TSN+SCST | 80.82 24.1 |
| C32+C16+C50+SF+ST+TSN+SCST+ES | 85.71 31.2 |

Table 2. Performance gain from each component. “C32”, “C16”, “C50”, “SF”, “ST”, “TSN” and “ES” stand for “CLIP(Vit-B/32)”, “CLIP(Vit-B/16)”, “CLIP(RN50x16)”, “SlowFast(8x8)”, “Subtitle”, “TSN sampling strategy” and “ensemble”, respectively.

We follow [7] to conduct a two-stage training for CLIP4Caption++. For the first stage, we train the primitive CLIP4Caption++ with our enhanced video embedding. We employ the X-Linear/X-Transformer attention block to encode enhanced video embedding sequences and further used it in the decoder to mine the interaction information between different modes. Training stage one is done on 4 NVIDIA Tesla P40 GPU graphics, with batch size set to 128 and max epochs set to 100.

For the second stage, we train the caption model with SCST [10]. SCST utilizes the policy-gradient method, which takes the caption Cider in the training stage as a reward score for caption generation, and uses the current model’s reward in the test stage as the reward baseline. Thus, SCST optimizes CIDEr directly and encourages the model to perform consistently in training and testing stages. We initialize the model with the weights from the best model in stage one and set batch size as 20 and total epochs as 60, with a single GPU. Experimental result shows that SCST significantly improved the CIDEr score in stage two.

3. Ablation Study

We report the performance gain brought by each component. All the results were based on X-Linear. As shown in the Table 2. Each strategy contributed to the CIDEr improvement, and the concatenated visual video feature and SCST(Training stage two) provided a significant improvement. The enhanced video feature allows the model to capture more video semantic information, and SCST makes the model directly optimize the CIDEr metric.

3.3. Result

We train multiple models with different frame lengths, combinations of features, and other hyperparameters. We adopt the standard captioning metrics including BLEU-4 [8], ROUGE-L [6], CIDEr [11] to evaluate the performance of the captioning models. We report the single model result and ensemble result on the public test set and validation test set of TVC [4], YC2C [14], VATEX [13] datasets in Table 1. From Table 1 we can find that our ensemble strategy brings significant improvement on all datasets, and the final result ranks 1st in the VALUE competition.

4. Conclusion

This work focuses on better video embedding and proposes the CLIP4Caption++, which improves video embedding by enhanced visual video features and multi-modal text fusion. Besides, we introduce word-level and sentence-level ensemble strategies to ensemble multiple models’ captioning results. Extensive experiments indicate that each strategy contributed to the CIDEr improvement. Our method rank first in the VALUE competition in captioning task.
References

[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929, 2020.

[2] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In Proceedings of the IEEE/CVF international conference on computer vision, pages 6202–6211, 2019.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016.

[4] Jie Lei, Licheng Yu, Tamara L. Berg, and Mohit Bansal. Tvr: A large-scale dataset for video-subtitle moment retrieval. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI, pages 447–463. Springer, 2020.

[5] Linjie Li, Jie Lei, Zhe Gan, Licheng Yu, Yen-Chun Chen, Rohit Pillai, Yu Cheng, Luowei Zhou, Xin Eric Wang, William Yang Wang, et al. Value: A multi-task benchmark for video-and-language understanding evaluation. In 35th Conference on Neural Information Processing Systems (NeurIPS 2021) Track on Datasets and Benchmarks, 2021.

[6] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In Text summarization branches out, pages 74–81, 2004.

[7] Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. X-linear attention networks for image captioning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 10971–10980, 2020.

[8] 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 of the Association for Computational Linguistics, pages 311–318, 2002.

[9] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. arXiv preprint arXiv:2103.00020, 2021.

[10] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 7008–7024, 2017.

[11] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4566–4575, 2015.

[12] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaou 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.

[13] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 4581–4591, 2019.

[14] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In Thirty-Second AAAI Conference on Artificial Intelligence, 2018.