CLIP4Caption: CLIP for Video Caption

Mingkang Tang[1,2], Zhanyu Wang[1],Zhenhua Liu[1], Fengyun Rao[1], Dian Li[1], Xiu Li[2]
[1]Kandian Content AI Lab, Platform and Content Group, Tencent
[2]Shenzhen International Graduate School, Tsinghua University, China
{mktang,zhanyuwang,edinliu,fengyunrao,goodli}@tencent.com,li.xiu@sz.tsinghua.edu.cn

ABSTRACT

Video captioning is a challenging task since it requires generating sentences describing various diverse and complex videos. Existing video captioning models lack adequate visual representation due to the neglect of the existence of gaps between videos and texts. To bridge this gap, in this paper, we propose a CLIP4Caption framework that improves video captioning based on a CLIP-enhanced video-text matching network (VTM). This framework is taking full advantage of the information from both vision and language and enforcing the model to learn strongly text-correlated video features for text generation. Besides, unlike most existing models using LSTM or GRU as the sentence decoder, we adopt a Transformer structured decoder network to effectively learn the long-range visual and language dependency. Additionally, we introduce a novel ensemble strategy for captioning tasks. Experimental results demonstrate the effectiveness of our method on two datasets: 1) on MSR-VTT dataset, our method achieved a new state-of-the-art result with a significant gain of up to 10% in CIDEr; 2) on the private test data, our method ranking 2nd place in the ACM MM multimedia grand challenge 2021: Pre-training for Video Understanding Challenge. It is noted that our model is only trained on the MSR-VTT dataset.

CCS CONCEPTS

• **Computing methodologies** → **Neural networks**.

KEYWORDS

video caption, video-text matching, pre-train, transformer

ACM Reference Format:
Mingkang Tang[1,2], Zhanyu Wang[1],Zhenhua Liu[1], Fengyun Rao[1], Dian Li[1], Xiu Li[2]. 2021. CLIP4Caption: CLIP for Video Caption. In *Proceedings of the 29th ACM International Conference on Multimedia (MM '21), October 20–24, 2021, Virtual Event, China.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3474085.3479207

1 INTRODUCTION

Describing video content is a labor-intensive task for humans. Therefore, computer scientists have put much effort into connecting human language with visual information to develop a system that automatically describes videos using natural language sentences. The advancement of video captioning enhances various applications in reality, e.g., automatic video subtitling, aid to visually impaired person, human-computer interaction, and improving online video search or retrieval [1].

Early research in video captioning used template-based methods [8, 27, 36], which aligns predicted words with the pre-defined template. S2VT [31] proposed a LSTM [9] based sequence-to-sequence model for video captioning. Since then, numerous sequence learning methods, which adopt encoder-decoder architecture to generate caption flexibly, were introduced [13, 19, 20, 33, 37, 38]. [37] propose an attention-based approach that takes into account both the local and global temporal structure of videos to product descriptions. RecNet [33] proposed a reconstruction network that leverages both the video-to-text and text-to-sentence flows for video captioning. In recent years' study, some researchers also successfully use vision-language (VL) pretraining for VL understanding, which has made significant progress in the downstream task of image captioning [11, 12, 40].

All the methods mentioned above build their video encoder with a CNN-based network, lacking adequate visual representation since they only take advantage of the information from vision modality. In this paper, we introduce a video-text matching network which empowered by a well-pretrained CLIP [26] model to learn the video embeddings taking fully advantage from both vision and language modality.

Specially, we first pre-train a video-text matching model to obtain a text-correlated video embeddings, and then we taken those enhanced video embedding as input to fine-tune in a well pre-trained transformer decoder network. It is noted that our transformer decoder is initialized by the part of weights of the pretrained Uni-VL [15] model. Extensive experiments demonstrate that our methodology outperforms state-of-the-art video captioning methods [28] on the MSR-VTT dataset [35]. Additionally, our methodology ranks 2nd in the ACM MM grand challenge 2021: Pre-training for Video Understanding Challenge, in the first track of pre-training for video captioning.

The main contributions of this work are summarized as follows:

• We utilize a CLIP-enhanced video-text matching network to enforce our model to learn strongly correlated video and text features for text generation.

• We leverage the weights of well pre-trained video and language model Uni-VL while greatly simplified its structure to better-fitting video captioning tasks.

• We design a novel ensemble mechanism for video captioning.

• We extensively validate our model on the most widely used MSR-VTT dataset. The results indicate that our framework outperforms multiple state-of-the-art methods in video captioning, exhibiting the great potential of our framework for this challenging task.

MM '21, October 20–24, 2021, Virtual Event, China.

ACM ISBN 978-1-4503-8651-7/21/10.
https://doi.org/10.1145/3474085.3479207

2 METHODOLOGY

Figure 1 illustrates the framework of our proposed CLIP4Caption for video captioning. We train our system in two stage. First we pre-train a video-text matching network on MSR-VTT dataset to obtain better visual representation(2.1) (lower part in Fig. 1). Second, we take our pre-trained matching network as the video feature extractor in fine-tuning stage (upper part in FIg.1). A sequence of frames embedding was inputted to video encoder, connected with a decoder which generated the text (2.2). For ensemble, we train multiple caption models with different layers of encoder and decoder, and ensemble all the generated captioning text for a final strong result (2.3). Details will be elaborated as follows.

2.1 Video-text matching pre-training

As the CLIP4Clip model transferred from CLIP [26] has demonstrated outstanding performance in the video-text retrieval task, we pre-train our video-text matching network (VTM) upon CLIP4Clip. CLIP4Clip extracts frames of images from the video at 1 FPS, the input video frames for each epoch come from the video's fixed position. We improve the frames sampling method to the TSN sampling[34], which divides the video into K splits and randomly samples one frame in each split, thus increasing the sample randomness on the limited data set. After the TSN sampling, input frames are encoded by the pre-trained CLIP (ViT-B/32) video encoder [7] with 12 layers and the patch size 32, since CLIP is adequate for the image-text retrieval task. Take the input video frames of video as $v_i = \{v_i^1, v_i^2, \ldots, v_i^{|v_i|}\}$, the video frames embedding can denote as $f_i = \{f_i^1, f_i^2, \ldots, f_i^{|v_i|}\}$.

A 12-layer 512-wide model with eight attention heads Transformer [29] encoder is used as text encoder, whose weights originated from the pre-trained CLIP text encoder. Following CLIP and CLIP4Clip, the [EOS] token's activations of the highest layer of the transformer are used as the feature representation of the input text. For the input text s_j, corresponding text embedding is denoted as t_j.

After video encoding, we use a mean pooling layer to aggregate the embedding of all frames and obtain a average frame embedding. Then, the similarity function can be defined for video-text matching. Similar to CLIP4Clip, we adopt cosine similarity to measure similarity between joint video frames embedding $\hat{f_i}$ and text embedding t_j, as formulated by:

$$s(\hat{f_i}, t_j) = \frac{t_j{}^\top \hat{f_i}}{\|t_j\|\|\hat{f_i}\|}. \tag{1}$$

Video-text matching is trained in a self-supervised way. Given a batch of N video-text pairs, VTM generates a N × N similarities and the optimization goal is to maximize the similarity between paired video-text and maximize the similarity of unpaired text. Therefore, the loss function is defined as follows:

$$\mathcal{L}_{v2t} = -\frac{1}{N}\sum_{i}^{N} \log \frac{\exp(s(\hat{f_i}, t_i))}{\sum_j^N \exp(s(\hat{f_i}, t_j))}, \tag{2}$$

$$\mathcal{L}_{t2v} = -\frac{1}{N}\sum_{i}^{N} \log \frac{\exp(s(\hat{f_i}, t_i))}{\sum_j^N \exp(s(\hat{f_j}, t_i))}, \tag{3}$$

$$\mathcal{L} = \mathcal{L}_{v2t} + \mathcal{L}_{t2v}, \tag{4}$$

where $\mathcal{L}_{v2t}$ and $\mathcal{L}_{t2v}$ denotes the loss function of video-to-text and text-to-video respectively.

We use the output of ViT video encoder, a sequence of frames embedding, as our video visual representation. Each frame is mapped to 512d visual feature, resulting n×512 dynamic features $F_{v_i} = ViT(v_i)$ for each video, where n is the number of frames.

2.2 Fine-tune on video captioning

In fine-tuning stage, we leverage the well pre-trained model Uni-VL and fine-tune the encoder-decoder architecture of Uni-VL on video captioning with MSR-VTT dataset. Uni-VL is a two-stream video and language pre-training model. During Uni-VL's pre-training, text and video are input to text encoder and video encoder, respectively, and a cross encoder aligns the text embedding and video embedding. In our fine-tuning stage, we discard the text encoder and cross encoder since the input video of our dataset without related transcripts. The fine-tuning stage on total pre-trained Uni-VL layers is difficult since the MSR-VTT dataset (for fine-tuning stage) is relatively tiny to the Uni-VL's pre-training dataset HowTo100M [17]. CLIP4Caption, therefore, train effortless and prevent over-fitting through reducing the number of Transformer layers.

As described above, our captioning model is composed of the Transformer based Video Encoder and Decoder, the strongly text-correlated video feature F_v is input to one-layer Transformer Video Encoder(TE) to obtain the enhanced feature $f_{ve} = TE(F_v)$, and then fed into a three-layer Transformer Decoder (TD) to produce caption $t = TD(F_{ve})$ for each video. We initialize TE and TD with the weights pre-trained in Uni-VL and train the model with only cross-entropy loss:

$$\mathcal{L}_{ce} = -\sum_{t=1}^{L} \log p_t(S_t), \tag{5}$$

where L is the max length of caption sentence, p_t is the probability of predicted word at time t, and S_t is the sentence which has generated at time t.

2.3 Ensemble strategy

The single model is not strong enough for a great predicted result. In order to obtain a more powerful caption result, we design a novel metric-based voting strategy for captioning task. We utilise the captioning evaluation metrics, such as BLEU4, CIDEr, SPICE, etc., as the "importance score" of a generated sentences and select the sentence with highest score to compose the final result.

Mathematically, Considering the predicted captions of one video from n different models as T_i, the importance score for ith caption S_i using single metric can be calculated by assuming the rest of predicted captions $[T_j]_{j\neq i}$ as "ground-truth" captions:

$$S_i = metric(ref = [T_1, \ldots, T_{i-1}, T_{i+1}, \ldots, T_n], hpy = T_i), \tag{6}$$

where $i \in [1, n]$ and $metric(\cdot)$ is the captioning metric. The predicted caption with biggest score S is selected as the final output. Since captioning task often uses multiple metrics, and the value range of each metric is inconsistent, we use the maximum value of each metric to normalize it [5]. Considering multiple metrics, the

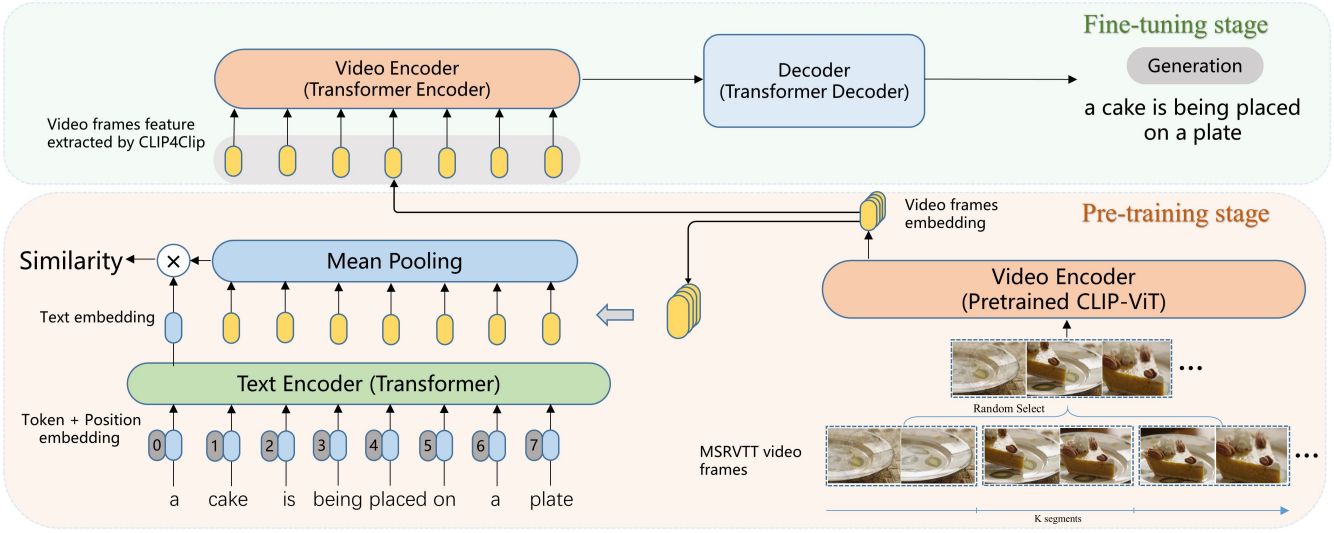

Figure 1: An Overview of our proposed CLIP4Caption framework comprises two training stages: a video-text matching pre-training stage and a video caption fine-tuning stage. In the pre-training stage, we build our video-text matching network upon a CLIP-based Video Encoder and a BERT-based Text Encoder, and it compares the similarity of video and text features for the match. In the fine-tuning stage, we take the strongly text-correlated video feature as input and fine-tuning our transformer decoder network for captioning task.

overall metric can be calculated as:

$$metric_{overall} = (\frac{metric_1}{metric_{1b}} + \frac{metric_2}{metric_{2b}} + \cdots + \frac{metric_M}{metric_{Mb}})/M, \quad (7)$$

where M denotes the number of metrics used for calculating $metric_{overall}$, $metric_{ib}$ denotes the best numeric value of the specific metric $metric_i$. And the importance score of multiple metrics is:

$$\hat{S}_i = metric_{overall}(ref = [T_1, \ldots, T_{i-1}, T_{i+1}, \ldots, T_n], hpy = T_i), \quad (8)$$

3 RESULTS

3.1 Dataset split

On account of the significant difference between Video Understanding Challenge pre-training dataset Auto-captions on GIF (ACTION) [18] and the MSR-VTT data set, we only use MSR-VTT as our training dataset. The MSR-VTT dataset covers a broad range of video content categories, with a total video time of about 50 hours, 10000 videos, and 20 descriptions per video. For pre-training results in Table 1 and Table 2, we report our results on the split of MSR-VTT Training-9K[16], which was used in CLIP4Clip. For fine-tuning results in Table 3 , we used the MSR-VTT's standard split (MSR-VTT Training-6K), i.e., 6,512, 498, and 2,990 clips for training, validation, and testing for comparision with state-of-the-art methods. We also used total MSRVTT (MSR-VTT Training-10K) to train more models for our ensemble results in Table 4, and evalute our ensemble mechanism on Video Understanding Challenge test set.

3.2 Pre-training result

Video-text matching pre-training is done on 8 NVIDIA Tesla P40 GPU graphics, with batch size set to 512 and max epochs set to 5.

Table 1: Pre-training results of video-text matching on MSR-VTT Training-9K. In the column R_{T2V}@K denotes text-to-video recall at rank K. For VTM, max frames K means K splits for TSN sampling.

Methods	max frames	R_{T2V}@1	R_{T2V}@5	R_{T2V}@10
CLIP4Clip	12	42.6	70.1	80.2
VTM	12	42.7	70.1	**81.2**
VTM	6	44	70.6	80
VTM	3	**44.5**	**71.1**	79.7

Table 2: Pre-training results of video-text matching on MSR-VTT Training-9K. In the column R_{V2T}@K denote video-to-text recall at rank K. For VTM, max frames K means K splits for TSN sampling.

Methods	max frames	R_{V2T}@1	R_{V2T}@5	R_{V2T}@10
CLIP4Clip	12	43.4	70.2	81.7
VTM	12	43.4	70.5	**82.2**
VTM	6	43.5	71.1	81.4
VTM	3	**44.3**	**71.7**	81.7

We use standard retrieval metrics: recall at rank K, denoted as R@K, to evaluate the performance of our pre-trained network. We report the metrics of R@1, R@5, R@10 for both text-to-video retrieval and video-to-text retrieval. The VTM result with different TSN split is shown in Table 1 and Table 2. Our VTM outperformed origin CLIP4Clip, and TSN sampling with 3 splits perform better in R@1 and R@5. Given this, we used VTM with 3 splits TSN as our feature

Table 3: Comparison on MSR-VTT dataset. "B@", "R", "M", "C", "S" denotes "BLEU-4", "ROUGE-L", "METEOR", "CIDEr", "SPICE" respectively. † indicates the results are quoted from the published literature [22]. For other methods in comparison, their results are obtained by re-running the publicly released codes or models on MSR-VTT dataset using the same training-test partition as our method.

Methods	B@4	R	M	C	S
$REVnetv3-RL^{\dagger}$ [10]	42.4	62.3	28.1	53.2	-
$ORG-TRL^{\dagger}$ [39]	43.6	62.1	28.8	50.9	-
$CST\_GT\_None^{\dagger}$ [25]	44.1	62.4	29.1	49.7	-
$SCN-LSTM+sampling^{\dagger}$ [5]	43.8	62.4	28.9	51.4	
$topic-guided^{\dagger}$ [6]	44.9	62.8	29.6	51.8	-
$GFN-POS\_RL(IR+M)^{\dagger}$ [32]	41.3	62.1	28.7	53.4	-
$VNS-GRU^{\dagger}$ [4]	46.0	63.3	29.5	52.0	-
$MSAN^{\dagger}_{f+o+c}$ [28]	46.8	-	29.5	52.4	-
$AVSSN$ [23]	42.8	61.7	28.8	46.9	-
$SemSynAN$ [24]	44.3	62.5	28.8	50.1	6.3
$Uni-VL$ [15]	42.2	61.21	28.8	49.9	6.5
$CLIP4Caption$	46.1	63.7	30.7	57.7	7.6
$CLIP4Caption(ensemble)$	**47.2**	**64.8**	**31.2**	**60.0**	**7.9**

extractor, and retrained the VTM in MSR-VTT Training-6K in the fine-tuning stage for better visual representation.

3.3 Fine-tuning result

In the fine-tuning stage, we used the pre-trained VTM on MSR-VTT Training-6K as our feature extractor. To be consistent with Uni-VL, we set the maximum frame length to 20. Caption fine-tuning is done on 4 GPUs, making the batch size 1024 and the total epochs 30.

In Table 3, we report the standard captioning metrics, BLEU-4 [21], ROUGE-L [14], METEOR [3], CIDEr [30], SPICE [2] of our proposed CLIP4Caption, and other state-of-the-art methods for video captioning on MSR-VTT dataset. As shown in Table 3, our pre-training stage learns the powerful text-correlated visual representation for text generation, significantly improving all the metrics upon Uni-VL. CLIP4Caption achieved a new state-of-the-art result with a significant gains of up to 10% in the CIDEr score.

3.4 Ensemble result

We vary dataset split and the layers of the Transformer to train more models. We used two splits for the pre-training stage, the standard dataset split MSR-VTT Training-6K, and the other dataset split MSR-VTT Training-9K, with 9000 videos for training and 1000 videos for validation. As a result, two VTM features are obtained: VTM-6K-feature and VTM-9K-feature.

We adopted three splits for the fine-tuning stage, MSR-VTT Training-6K and MSR-VTT Training-9K as used in the pre-training stage, and MSR-VTT Training-10k, which used the total MSR-VTT dataset. At the same time, we also combined the different layers of Transformer layers. For the visual encoder, we tried 1-layer, 3-layer, and 6-layer Transformer encoders. As for the decoder, we tried the number of Transformer layers from 1 to 6. These combinations

Table 4: Ensemble results of video captioning on the Video Understanding Challenge test set. "single model" means the best result using a single model, while "k models" indicate the ensemble strategy uses results from k models for a better result.

Methods	B@4	M	C	S
CLIP4Caption(single model)	22.76	17.89	26.93	5.93
CLIP4Caption(9 models)	23.61	18.69	29.35	6.67
CLIP4Caption(17 models)	**23.78**	19.18	30.39	7.14
CLIP4Caption(47 models)	23.42	19.40	30.56	7.47
CLIP4Caption(59 models)	23.67	**19.63**	**31.19**	**7.53**

produced a lot of captioning results. We eliminated some of the results that were not effective on the MSR-VTT validation dataset and applied ensemble strategy in other results.

We Validate our proposed strategy in Video Understanding Challenging test dataset. The metric used for the ensemble is SPICE and BLEU-4, because of the poor performance using other metrics. The experimental results are shown in Table 3. Compared with the best result of a single model, our ensemble strategy has significantly improved the metrics on the test dataset, and ranks 2nd in the Video Understanding Challenge. Furthermore, the result is positively correlated with the number of results we use, which means we can use this ensemble strategy to improve our results by continuously training enough models.

4 CONCLUSION

In this work, we focus on learning better visual representation for text generation and improving video captioning with video and language pre-training models. We propose the CLIP4Caption, a two-stage language and video pre-training-based video caption solution. For better visual representation, we adopt the pre-training stage to learn strongly text-correlated video features. Also, to improve video captioning, we make use of Uni-VL pre-trained weights to initialize our encoder-decoder-based captioning architecture and fine-tune the model in MSR-VTT dataset. Besides, we introduce a novel ensemble strategy to ensemble multiple models' captioning results by using captioning metrics. Extensive experiments indicate that our proposed CLIP4Caption significantly outperforms the current state-of-the-art method and ranks 2nd in the Video Understanding Challenge test dataset.

5 ACKNOWLEDGMENTS

This research was partly supported by the National Natural Science Foundation of China (Grant No. 41876098), and Shenzhen Science and Technology Project (Grant No. JCYJ20200109143041798).

REFERENCES

[1] Soheyla Amirian, Khaled Rasheed, Thiab R Taha, and Hamid R Arabnia. 2020. Automatic image and video caption generation with deep learning: A concise review and algorithmic overlap. *IEEE Access* 8 (2020), 218386–218400.
[2] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. Spice: Semantic propositional image caption evaluation. In *European conference on computer vision*. Springer, 382–398.
[3] Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings*

of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization. 65–72.

[4] Haoran Chen, Jianmin Li, and Xiaolin Hu. 2020. Delving deeper into the decoder for video captioning. *arXiv preprint arXiv:2001.05614* (2020).

[5] Haoran Chen, Ke Lin, Alexander Maye, Jianmin Li, and Xiaolin Hu. 2020. A Semantics-Assisted Video Captioning Model Trained with Scheduled Sampling. *Frontiers in Robotics and AI* 7 (2020).

[6] Shizhe Chen, Qin Jin, Jia Chen, and Alexander G Hauptmann. 2019. Generating video descriptions with latent topic guidance. *IEEE Transactions on Multimedia* 21, 9 (2019), 2407–2418.

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

[8] Sergio Guadarrama, Niveda Krishnamoorthy, Girish Malkarnenkar, Subhashini Venugopalan, Raymond Mooney, Trevor Darrell, and Kate Saenko. 2013. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *Proceedings of the IEEE international conference on computer vision*. 2712–2719.

[9] S. Hochreiter and J. Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9, 8 (1997), 1735–1780.

[10] Huidong Li, Dandan Song, Lejian Liao, and Cuimei Peng. 2019. Revnet: Bring reviewing into video captioning for a better description. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 1312–1317.

[11] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. 2020. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*. Springer, 121–137.

[12] Yehao Li, Yingwei Pan, Ting Yao, Jingwen Chen, and Tao Mei. 2021. Scheduled Sampling in Vision-Language Pretraining with Decoupled Encoder-Decoder Network. *arXiv preprint arXiv:2101.11562* (2021).

[13] Yehao Li, Ting Yao, Yingwei Pan, Hongyang Chao, and Tao Mei. 2018. Jointly Localizing and Describing Events for Dense Video Captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

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

[15] Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. 2020. Univl: A unified video and language pre-training model for multimodal understanding and generation. *arXiv preprint arXiv:2002.06353* (2020).

[16] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. 2021. Clip4clip: An empirical study of clip for end to end video clip retrieval. *arXiv preprint arXiv:2104.08860* (2021).

[17] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. 2019. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2630–2640.

[18] Yingwei Pan, Yehao Li, Jianjie Luo, Jun Xu, Ting Yao, and Tao Mei. 2020. Auto-captions on GIF: A Large-scale Video-sentence Dataset for Vision-language Pre-training. *arXiv preprint arXiv:2007.02375* (2020).

[19] Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. 2016. Jointly modeling embedding and translation to bridge video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4594–4602.

[20] Yingwei Pan, Ting Yao, Houqiang Li, and Tao Mei. 2017. Video captioning with transferred semantic attributes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 6504–6512.

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

[22] Jesus Perez-Martin, Benjamin Bustos, Silvio Jamil F Guimarães, Ivan Sipiran, Jorge Pérez, and Grethel Coello Said. 2021. Bridging Vision and Language from the Video-to-Text Perspective: A Comprehensive Review. *arXiv preprint arXiv:2103.14785* (2021).

[23] Jesus Perez-Martin, Benjamin Bustos, and Jorge Pérez. 2021. Attentive visual semantic specialized network for video captioning. In *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 5767–5774.

[24] Jesus Perez-Martin, Benjamin Bustos, and Jorge Pérez. 2021. Improving video captioning with temporal composition of a visual-syntactic embedding. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 3039–3049.

[25] Sang Phan, Gustav Eje Henter, Yusuke Miyao, and Shin'ichi Satoh. 2017. Consensus-based sequence training for video captioning. *arXiv preprint arXiv:1712.09532* (2017).

[26] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. arXiv:2103.00020 [cs.CV]

[27] Marcus Rohrbach, Wei Qiu, Ivan Titov, Stefan Thater, Manfred Pinkal, and Bernt Schiele. 2013. Translating video content to natural language descriptions. In *Proceedings of the IEEE international conference on computer vision*. 433–440.

[28] Liang Sun, Bing Li, Chunfeng Yuan, Zhengjun Zha, and Weiming Hu. 2019. Multimodal Semantic Attention Network for Video Captioning. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 1300–1305.

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

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

[31] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. 2015. Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision*. 4534–4542.

[32] Bairui Wang, Lin Ma, Wei Zhang, Wenhao Jiang, Jingwen Wang, and Wei Liu. 2019. Controllable video captioning with pos sequence guidance based on gated fusion network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2641–2650.

[33] Bairui Wang, Lin Ma, Wei Zhang, and Wei Liu. 2018. Reconstruction network for video captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7622–7631.

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

[35] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5288–5296.

[36] Ran Xu, Caiming Xiong, Wei Chen, and Jason Corso. 2015. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 29.

[37] Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville. 2015. Describing videos by exploiting temporal structure. In *Proceedings of the IEEE international conference on computer vision*. 4507–4515.

[38] Haonan Yu, Jiang Wang, Zhiheng Huang, Yi Yang, and Wei Xu. 2016. Video paragraph captioning using hierarchical recurrent neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4584–4593.

[39] Ziqi Zhang, Yaya Shi, Chunfeng Yuan, Bing Li, Peijin Wang, Weiming Hu, and Zheng-Jun Zha. 2020. Object relational graph with teacher-recommended learning for video captioning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 13278–13288.

[40] Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason Corso, and Jianfeng Gao. 2020. Unified vision-language pre-training for image captioning and vqa. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 13041–13049.