VideoCLIP: A Cross-Attention Model for Fast Video-Text Retrieval Task with Image CLIP

Yikang Li
OPPO US Research Center
Palo Alto, CA, US
yikang.li1@oppo.com

Jenhao Hsiao
OPPO US Research Center
Palo Alto, CA, US
mark@oppo.com

Chiuman Ho
OPPO US Research Center
Palo Alto, CA, US
chiuman@oppo.com

ABSTRACT

Video-text retrieval is an essential task in cross-modal information retrieval, i.e., retrieving relevant videos from a large and unlabelled dataset given textual queries. Existing methods that simply pool the image features (e.g., based on the CLIP encoder [14]) from frames to build the video descriptor often result in sub-optimal video-text search accuracy since the information among different modalities is not fully exchanged and aligned. In this paper, we proposed a novel dual-encoder model to address the challenging video-text retrieval problem, which uses a highly efficient cross-attention module to facilitate the information exchange between multiple modalities (i.e., video and text). The proposed *VideoCLIP* is evaluated on two benchmark video-text datasets, MSRVTT and DiDeMo, and the results show that our model can outperform existing state-of-the-art methods while the retrieval speed is much faster than the traditional query-agnostic search model.

CCS CONCEPTS

• **Information systems → Multimedia and multimodal retrieval**; **Video search**.

KEYWORDS

Video-Text Retrieval, Cross-Attention, Query-agnostic, Transformer, CLIP

ACM Reference Format:
Yikang Li, Jenhao Hsiao, and Chiuman Ho. 2022. VideoCLIP: A Cross-Attention Model for Fast Video-Text Retrieval Task with Image CLIP. In *Proceedings of the 2022 International Conference on Multimedia Retrieval (ICMR '22), June 27–30, 2022, Newark, NJ, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3512527.3531429

1 INTRODUCTION

Visual-language learning model, such as video-text retrieval that retrieves a list of relevant videos from a large corpus given text queries, acts as an important role in current multi-modal research and applications. Due to the success of previous algorithms for the cross-modal tasks in the image-text domain [3, 4, 6–8, 14], some researchers [9] suggest that image-text pre-trained model, such as the CLIP model [14] and the ALIGN model [7], benefits the video-text retrieval task as well, which directly leverage the pre-trained CLIP model to encode video frames and corresponding captions.

Though intuitive, directly applying clip-based methods in the original image-text domain to video-text retrieval tasks suffers from several challenges. The first is the video feature representation. Different from the image, the generation of an appropriate video feature representation is not trivial, which should consider both spatial and temporal dimension [6]. Simply pooling the frame features to build a video descriptor would often result in sub-optimal video-text search accuracy. The other challenge is the multi-modal interaction between video and languages. Video-text retrieval is naturally a weakly -supervised learning problem because there are no explicit alignments between the video and text modalities. Traditional embedding approaches, where video and text embeddings were learned independently and aligned in a brute force manner, often lead to an unsatisfactory result. Third, traditional query-dependent models [9], where (video, text) pair is encoded by concatenating and passing into one single network, are prohibitively slow to apply to the entire video corpus since the network needs to recompute for every query. It is thus impractical to apply such a query-dependent model to a real-world video-text retrieval system.

To address the above issues, in this paper, we propose a dual-encoder model for the video-text retrieval task, which is guided by both video and text modalities simultaneously. It can achieve better retrieval accuracy due to a finer fusion mechanism that allows multiple modalities (i.e., video and text) to better exchange information based on the cross-attention module, while also keeping the efficiency in the inference stage because of the use of a query-agnostic search architecture. In summary, the main contributions of this paper include:

- We extend the existing CLIP-based image-text semantic space into a more complex video-text space. The proposed model achieves better video-text retrieval accuracy than state-of-the-art methods [9] in two video-text benchmark datasets: MSRVTT and DiDeMo.
- We propose an efficient cross-attention module that uses a non-patch token as an agent to interchange information between visual/language branches by attention mechanism. The cross-attention module is linear in both computation and memory and can better leverage features from different modalities.
- We adopt a joint training for the proposed query-agnostic and query-dependent search model that is guided by two kinds of losses and is able to scale by pre-computing a video

ICMR '22, June 27–30, 2022, Newark, NJ, USA

ACM ISBN 978-1-4503-9238-9/22/06…$15.00
https://doi.org/10.1145/3512527.3531429

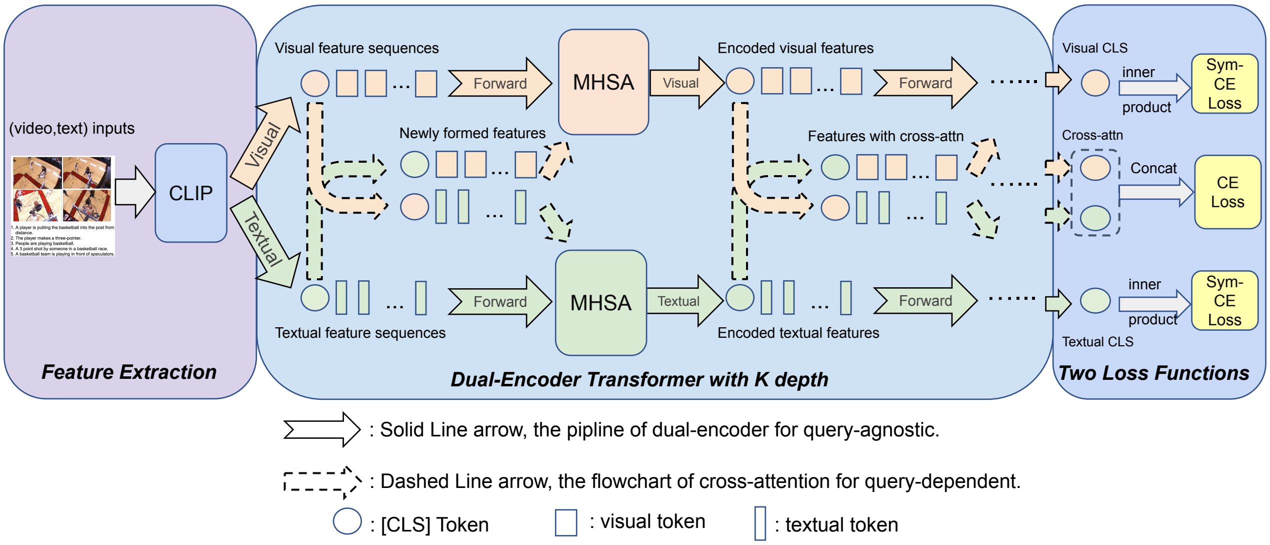
(video,text) inputs
1. A player is putting the basketball into the post from distance.
2. The player makes a three-pointer.
3. People are playing basketball.
4. A 3 point shot by someone in a basketball race.
5. A basketball team is playing in front of speculators.
CLIP
Visual
Textual
Feature Extraction
Visual feature sequences
Forward
MHSA
Visual
Encoded visual features
Forward
Newly formed features
Features with cross-attn
MHSA
Textual
Forward
Textual feature sequences
Encoded textual features
Forward
Dual-Encoder Transformer with K depth
Visual CLS
inner
product
Sym-CE Loss
Cross-attn
Concat
CE Loss
inner
product
Sym-CE Loss
Textual CLS
Two Loss Functions
: Solid Line arrow, the pipline of dual-encoder for query-agnostic.
: Dashed Line arrow, the flowchart of cross-attention for query-dependent.
: [CLS] Token
: visual token
: textual token

Figure 1: The framework of the cross-attention dual-encoder model. Better read in colors.

data index in the inference stage, which makes it extremely efficient in real-world applications.

2 METHODOLOGY

As we mentioned in Section 1, the proposed algorithm takes the frame features as inputs. We utilize the pre-trained image-text encoder CLIP [14] as the frame feature extractor since we want to take the advantage of projecting the images and the corresponding captions into the same semantic space. The proposed algorithm with a cross-attention mechanism is inspired by [2] but we modify the model by combining the cross-attention module with dual-encoder architecture. The losses we utilized are the symmetric cross-entropy and the cross-entropy loss for maintaining the (video, text) pair similarities and learning the classification-like task for the query-agnostic and query-dependent branches respectively.

2.1 Feature Extraction

Since the CLIP [14] model is constructed with dual-encoder architecture, we utilize each encoder to extract the visual and textual features respectively. Different from the original CLIP and the Vision Transformer model which uses the special class token [CLS] as the final output to represent the visual/textual feature, we use all the token outputs of the highest layer of the transformer to represent the video and text features respectively. The features extracted for each frame can be represented as following:

$$x^{v}_{n_{cls,1,\ldots,49}} = \Gamma_v(I_n) \tag{1}$$

where the Γ_v is the CLIP model visual encoder. The dimension of each frame feature is $(50, 512)$ and the 50 equals the number of image tokens, 49, plus the [CLS] token. The 512 is the embedding dimension for each visual token $x^{v}_{n_i}$ where $n \in [1, N]$ that N = number of frames and $i \in [1, 49] + [CLS]$.

A mean pooling along the axis of the number of frames is conducted after the feature extraction. Therefore, the final video and text features can be represented like:

$$\begin{aligned} \mathcal{X}^{v}_{cls,1,\ldots,49} &= Mean\{\Gamma_v(I_1), \ldots, \Gamma_v(I_n)\} \\ \mathcal{X}^{t}_{cls,1,\ldots,76} &= \Gamma_t(T) \end{aligned} \tag{2}$$

where the Γ_t is the CLIP model textual encoder and the output dimension of each textual feature is $(77, 512)$, where the 77 is the pre-defined length of the input sentence and the 512 is the embedding dimension of each textual token $\mathcal{X}^{t}_{n}$. Hence the dimension of the video and the text input for each batch will be $(B, 50, 512)$ and $(B, 77, 512)$ respectively, where the B is the number of batch size.

2.2 Cross-Attention Module

The Cross-Attention Module (CAM) is inspired by [2] that builds a two-branch model with Transformer architecture [5, 15]. In our designed CAM, we have two branches as well, the textual and the visual branch. Different from the dual-encoder model, in which each encoder is isolated from the other, the designed CAM will fuse the [CLS] token from each encoder with the patch tokens from the other branch K times, where the K equals the depth of the Transformer architecture. Besides, the [CLS] token will be projected into the other branch space by a linear projection layer before token fusion.

For the computation of the [CLS] token fusion, the input video features are converted into 3 channels since the video feature contains 4 channels after the feature extraction in the above-mentioned section. We do a dimension flatten along the first and second axis of the video features in Eq.2, which equals to convert the $(B, N, 50, 512)$ to $(B * N, 50, 512)$. Then the [CLS] token fusion can be represented as following:

$$\begin{aligned}\widetilde{\mathcal{X}}^v &= [f^{t-v}(\mathcal{X}^v_{cls})||\mathcal{X}^v_{1,...,49}] \\ \widetilde{\mathcal{X}}^t &= [f^{v-t}(\mathcal{X}^t_{cls})||\mathcal{X}^t_{1,...,76}]\end{aligned} \tag{3}$$

where the $||$ represents the operation of concatenation. The $\widetilde{\mathcal{X}}^v$ and the $\widetilde{\mathcal{X}}^t$ is the new video and text features respectively by concatenating the [CLS] token and the patch tokens from different modalities. The f^{t-v} and f^{v-t} is the projection that maps the [CLS] token from visual to textual branch and vice versa.

The newly formed video and text features are fed into the dual-encoder separately to obtain mutual information. A single encoder has the same architecture as the [2, 15] which can be interpreted as the stack of several layers of Multi-Head Self/Cross-Attention module (**MHSA/MHCA**) with Layer Normalization (**LN**) and residual shortcut. The output of each layer of the encoder can be presented as following:

$$\begin{aligned}\mathcal{Y}^v_{cls} &= \widetilde{\mathcal{X}}^v_{cls} + \mathbf{MHCA}(\mathbf{LN}(\widetilde{\mathcal{X}}^v)) \\ \mathcal{Y}^t_{cls} &= \widetilde{\mathcal{X}}^t_{cls} + \mathbf{MHCA}(\mathbf{LN}(\widetilde{\mathcal{X}}^t))\end{aligned} \tag{4}$$

where the **MHCA** utilize the [CLS] tokens as query instead of the whole sequence. The $\mathcal{Y}^v_{cls}$ and $\mathcal{Y}^t_{cls}$ represents the cross-encoded [CLS] token of the video and text features. At last, the newly encoded [CLS] token is back-projected into the original space and concatenated with the original features to form a new feature. Then the newly formed representation will repeat the steps from Eq.3 to Eq.4 K times, which is the depth of the Transformer architecture, to obtain the new features. The new features can be described as following:

$$\begin{aligned}\mathcal{Z}^v &= \mathbf{Trans}[g^{t-v}(\mathcal{Y}^v_{cls})||\mathcal{X}^v_{1,...,49}] \\ \mathcal{Z}^t &= \mathbf{Trans}[g^{v-t}(\mathcal{Y}^t_{cls})||\mathcal{X}^t_{1,...,76}]\end{aligned} \tag{5}$$

where the **Trans** means the transformer architecture which builds upon on the Eq. 4. The g^{t-v} and g^{v-t} are the back-project functions. For our designed algorithm, we only utilize the [CLS] token from the new features, $\mathcal{Z}^v_{cls}$ and $\mathcal{Z}^t_{cls}$, to do the training and testing which is the same scheme as [2, 7, 9, 14, 15].

Since we emphasize that our model is query-agnostic during the inference process, therefore, we also train the cross-attention model without fusing or inter-changing the [CLS] token in Eq. 3 simultaneously in the training stage like:

$$\breve{\mathcal{X}}^v_{cls} = f^{t-v}(\mathcal{X}^v_{cls}), \breve{\mathcal{X}}^t_{cls} = f^{v-t}(\mathcal{X}^t_{cls}) \tag{6}$$

Then the following computations in Eq. 3, Eq. 4, and Eq. 5 do not contain any mutual information, thus can be regarded as an independent dual-encoder model for the inference procedure.

This process is equivalent to **sharing the weights of the cross-attention module** with the dual-encoder training strategy and the query-agnostic branch will be guided by the information of the query-dependent branch. This learning strategy has two main advantages, performance improvement and the speedups of retrieval task which will be presented in the section 3.

The framework of our proposed method is presented in Fig.1. The cross-attention model pipeline, which contains the computations from Eq.3 to Eq.5, is demonstrated by the dashed line arrow and the dual-encoder flowchart, which does not require any cross-attention mechanism, can be shown by the solid line arrow. In the Fig.1, we use the **MHSA** instead of **MHCA** since the architecture is the same except the **MHCA** utilizes the [CLS] token as query instead. At last, the [CLS] tokens which contain the interchanged cross-attention information will be fed into the Cross-Entropy loss while the [CLS] tokens directly from the dual-encoder model will be computed by the symmetric Cross-Entropy loss. The loss functions can be found in details in the following section. The Fig.1 can have better explanations if reading in colors.

2.3 Loss Function

In this paper, we utilize two kinds of loss functions to train our model, the symmetric cross-entropy loss and the cross-entropy loss.

The symmetric cross-entropy loss is proposed in [17] and it has been widely used for the contrastive learning-based methods recently [7, 9, 14]. In this paper, the symmetric cross-entropy is mainly used for guiding the query-agnostic branch to update that can be presented as:

$$\mathcal{L}_S = -\frac{1}{M}\sum^{M} log \frac{\exp\left(S(v,t)/\tau\right)}{\sum^{M} \exp S(v,t)} \tag{7}$$

where M is the batch size, the τ is the temperature hyper-parameter and S is the similarity matrix of (video, text) pairs.

We utilize the conventional cross-entropy loss for the cross-attention module branch as well. As claimed in several researches [9, 12], the (video, text) pair can be classified into a binary category as "best pair (1)" or "not (0)". Consequently, the retrieval task can be converted into a classification task. For training, we generate the pseudo-binary labels $(0, 1)$ for (video, text) pairs in each batch to represent "paired" or "not". The "not" paired (video, text) is randomly selected from the dataset. Therefore, this conventional cross-entropy loss is designed for the query-dependent branch to update. Then the cross-entropy loss can be presented as:

$$\mathcal{L}_C = -\frac{1}{M}\sum_{i}^{M}[l_i \log(p(v,t)_i) + (1-l_i)(1-p(v,t)_i)] \tag{8}$$

where the $l_i \in \{0, 1\}$ is the binary label for (video,text) pairs and the $p(v, t)$ is the probability of (video,text) pair is the closest pair. So the total loss for our algorithm can be described as:

$$\mathcal{L} = \mathcal{L}_S + \mathcal{L}_C \tag{9}$$

Due to the classification-like task with cross-entropy loss, we concatenate the [CLS] tokens from the output of the cross-attention model branch $\mathcal{Z}^v_{cls}$ and $\mathcal{Z}^t_{cls}$ together and feed it into a Feed-Forward Network (FFN) with softmax activation function.The output probability of the classification-like task can be represented as:

$$p(v,t) = softmax(\mathbf{FFN}([\mathcal{Z}^v_{cls}||\mathcal{Z}^t_{cls}])) \tag{10}$$

Table 1: The results of text-to-video retrieval on MSRVTT

Method	R@1	R@5	R@10	MdR	MnR
JSFusion [19]	10.2	31.2	43.2	13.0	-
HT [13]PT	14.9	40.2	52.8	9.0	-
ActBERT [21]PT	16.3	42.8	56.9	10.0	-
HERO [10]PT	16.8	43.4	57.7	-	-
ClipBERT [9]	22.0	46.8	59.9	6.0	-
CLIP-mean	30.7	53.1	62.6	5.0	38.9
Ours	**33.1**	**56.9**	**69.5**	**4.0**	24.1

Table 2: The results of text-to-video retrieval on DiDeMo

Method	R@1	R@5	R@10	MdR	MnR
S2VT [16]	11.9	33.6	-	13.0	-
FSE [20]	13.9	36.0	-	11.0	-
CE [11]	16.1	41.1	-	8.3	-
ClipBERT [9]	20.4	48.0	60.8	7.0 -	
CLIP-mean	28.4	53.3	64.7	5.0	38.0
Ours	**32.4**	**57.3**	**69.1**	**4.0**	26.2

Table 3: The performance of inference speed on MSRVTT

Method	Time (second)
query-dependent	93918.2
query-agnostic	**111.3**

3 EXPERIMENTS AND SETTINGS

In this paper, we evaluate our designed model on two video-text retrieval benchmark datasets, **MSRVTT** dataset [18] and **DiDeMo** dataset [1]. The **MSRVTT** dataset contains 10K YouTube videos with 200K captions in total and each video clip has 20 related captions. We follow the [9, 19] for training and testing split. The **DiDeMo** dataset contains 10K Flickr videos with 40K sentence descriptions in total. The training and testing scheme is followed by the [9, 20], where all descriptions of one video are concatenated to form one longer caption for retrieval. For our evaluation metrics, we utilize the average recall at K (*R@K*), the mean rank (*MnR*), and the median rank (*MdR*) to demonstrate the performance of all the methods. For all the methods that we compared in this paper, we follow the settings in [9] for extracting video frames.

Settings: For the settings of our proposed algorithm, the K depth of transformer architecture mentioned in Section 2.2 is set to be 12 as a default setting, which is the same in papers [2, 9, 14]. All the settings of the feature extraction of the CLIP follow the default settings in [14] and all the parameters of CLIP are frozen during the training. The total number of training epochs is set to 200 and the batch size in the training stage is 128. The initial learning rate is $1e - 6$ and the AdamW algorithm is used as the optimizer for our training. We conduct all our experiments with PyTorch deep learning tools and all experiments are run and tested on 4 V100 GPUs with 32G memory for each GPU.

1).**MSRVTT**: the text-to-video retrieval task of **MSRVTT** dataset is shown in Table.1. We follow the comparison strategy in [9] and compared our model with **JSFusion** [19], **HT** [13], **ActBERT**[21], **HERO** [10] and **ClipBERT** [9]. The “**PT**" in the table means the pre-trained model. All of the results are cited directly from the original paper of [9] and all of those baseline methods are trained with query-dependent architecture. We also compared our model with the **CLIP-mean** method, which directly using the mean-pooling fusion method on all the CLIP features of frames to represent the video feature. Therefore, the **CLIP-mean** method does not require any training process.

From the Table.1, we can observe that our proposed method outperforms all the other SOTA methods. Surprisingly, the **CLIP-mean** method can even have a better performance than other training-required methods. It demonstrates that a well-defined image-text semantic space plays an important role in the cross-modal retrieval task. Moreover, extending the image-text semantic space into a video-text domain helps the improvement of the cross-modal retrieval task.

2).**DiDeMo**: the text-to-video retrieval task of **DiDeMo** dataset is shown in Table.2. All the baseline methods, **S2VT** [16], **FSE** [20], and **CE** [11], listed in the [9] are utilized for comparison in our paper as well. Same as the evaluation in **MSRVTT** dataset, we also compared all the algorithms with the **CLIP-mean** method and all of the results of the baseline methods are cited directly from the paper [9]. In the Table.2, it shows that our designed method can outperform all other SOTA methods for the long caption to video retrieval as well.

3). **Speed of inference**: we also evaluated the speed of inference on the MSRVTT testing dataset, and the consumed time is listed in Table.3. The evaluation for each branch is done by freezing the corresponding parameters of the other one in our algorithm. From the table, we can see that the query-agnostic branch is much faster than the query-dependent branch in the inference procedure. The query-dependent branch takes more than a whole day to process all the 1000 data which may lead to traversing 1 million ($1000 * 1000$) possible (video, text) pairs while the query-agnostic branch will do the encoding separately which can reduce about 1000 times of the consumed time of the query-dependent branch. Therefore, the query-agnostic method is more practical in real retrieval applications.

4 CONCLUSION

In this paper, we propose an algorithm with cross-attention mechanism to adopt the well-defined image-text semantic space to the 3D video space by leveraging the CLIP features. The evaluation on two benchmark datasets shows that our proposed model can achieve the best performance among other SOTA methods. Additionally, we utilize the cross-attention model based learning to guide the dual-encoder based pipeline in the proposed algorithm and make our model work as query-agnostic in the inference procedure which will save a lot of time compared with query-dependent methods.

REFERENCES

[1] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. 2017. Localizing moments in video with natural language. In *2017 IEEE International Conference on Computer Vision (ICCV)*. 5803–5812.

[2] Chun-Fu Chen, Quanfu Fan, and Rameswar Panda. 2021. Crossvit: Cross-attention multi-scale vision transformer for image classification. *arXiv preprint arXiv:2103.14899* (2021).

[3] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Universal image-text representation learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

[4] Karan Desai and Justin Johnson. 2021. VirTex: Learning Visual Representations from Textual Annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[5] 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).

[6] Jenhao Hsiao, Jiawei Chen, and Chiuman Ho. 2020. GCF-Net: Gated Clip Fusion Network for Video Action Recognition. In *Proceedings of the European Conference on Computer Vision Workshops (ECCV Workshops)*. 699–713.

[7] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. *arXiv preprint arXiv:2102.05918* (2021).

[8] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. 2016. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. https://arxiv.org/abs/1602.07332

[9] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L. Berg, Mohit Bansal, and Jingjing Liu. 2021. Less is More: ClipBERT for Video-and-Language Learningvia Sparse Sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[10] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. 2020. Hero: Hierarchical encoder for video+ language omni-representation pre-training. *arXiv preprint arXiv:2005.00200* (2020).

[11] Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. 2019. Use what you have: Video retrieval using representations from collaborative experts. *arXiv preprint arXiv:1907.13487* (2019).

[12] Antoine Miech, Jean-Baptiste Alayrac, Ivan Laptev, Josef Sivic, and Andrew Zisserman. 2021. Thinking Fast and Slow: Efficient Text-to-Visual Retrieval with Transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 9826–9836.

[13] 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 Conference on Computer Vision and Pattern Recognition (CVPR)*. 2630–2640.

[14] 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. *CoRR* abs/2103.00020 (2021). arXiv:2103.00020 https://arxiv.org/abs/2103.00020

[15] 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.

[16] Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond Mooney, and Kate Saenko. 2014. Translating videos to natural language using deep recurrent neural networks. *arXiv preprint arXiv:1412.4729* (2014).

[17] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. 2019. Symmetric cross entropy for robust learning with noisy labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 322–330.

[18] 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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5288–5296.

[19] Youngjae Yu, Jongseok Kim, and Gunhee Kim. 2018. A joint sequence fusion model for video question answering and retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 471–487.

[20] Bowen Zhang, Hexiang Hu, and Fei Sha. 2018. Cross-modal and hierarchical modeling of video and text. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 374–390.

[21] Linchao Zhu and Yi Yang. 2020. Actbert: Learning global-local video-text representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 8746–8755.