CRET: Cross-Modal Retrieval Transformer for Efficient Text-Video Retrieval

Kaixiang Ji
Ant Group
Hangzhou, Zhejiang, China
kaixiang.jkx@antgroup.com

Jiajia Liu
Ant Group
Hangzhou, Zhejiang, China
lekun.ljj@antgroup.com

Weixiang Hong
Ant Group
Hangzhou, Zhejiang, China
hwx229374@antgroup.com

Liheng Zhong
Ant Group
Hangzhou, Zhejiang, China
zhongliheng.zlh@antgroup.com

Jian Wang
Ant Group
Hangzhou, Zhejiang, China
bobblair.wj@antgroup.com

Jingdong Chen
Ant Group
Hangzhou, Zhejiang, China
jingdongchen.cjd@antgroup.com

Wei Chu
Ant Group
Hangzhou, Zhejiang, China
weichu.cw@antgroup.com

ABSTRACT

Given a text query, the text-to-video retrieval task aims to find the relevant videos in the database. Recently, model-based (MDB) methods have demonstrated superior accuracy than embedding-based (EDB) methods due to their excellent capacity of modeling local video/text correspondences, especially when equipped with large-scale pre-training schemes like ClipBERT. Generally speaking, MDB methods take a text-video pair as input and harness deep models to predict the mutual similarity, while EDB methods first utilize modality-specific encoders to extract embeddings for text and video, then evaluate the distance based on the extracted embeddings. Notably, MDB methods cannot produce explicit representations for text and video, instead, they have to exhaustively pair the query with every database item to predict their mutual similarities in the inference stage, which results in significant inefficiency in practical applications.

In this work, we propose a novel EDB method CRET (Cross-modal REtrieval Transformer), which not only demonstrates promising efficiency in retrieval tasks, but also achieves better accuracy than existing MDB methods. The credits are mainly attributed to our proposed Cross-modal Correspondence Modeling (CCM) module and Gaussian Estimation of Embedding Space (GEES) loss. Specifically, the CCM module is composed by transformer decoders and a set of decoder centers. With the help of the learned decoder centers, the text/video embeddings can be efficiently aligned, without suffering from pairwise model-based inference. Moreover, to balance the information loss and computational overhead when sampling frames from a given video, we present a novel GEES loss, which implicitly conducts dense sampling in the video embedding space, without suffering from heavy computational cost. Extensive experiments show that without pre-training on extra datasets, our proposed CRET outperforms the state-of-the-art MDB methods that were pre-trained on additional datasets, meanwhile still shows promising efficiency in retrieval tasks.

SIGIR '22, July 11–15, 2022, Madrid, Spain.

ACM ISBN 978-1-4503-8732-3/22/07…$15.00
https://doi.org/10.1145/3477495.3531960

CCS CONCEPTS

• **Information systems** → **Multimedia and multimodal retrieval**; Similarity measures.

KEYWORDS

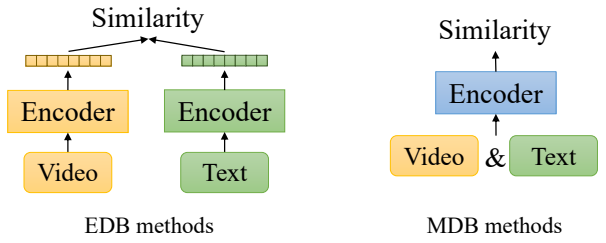
Text-to-video retrieval; Feature alignment; Video frame sampling.

ACM Reference Format:
Kaixiang Ji, Jiajia Liu, Weixiang Hong, Liheng Zhong, Jian Wang, Jingdong Chen, and Wei Chu. 2022. CRET: Cross-Modal Retrieval Transformer for Efficient Text-Video Retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22), July 11–15, 2022, Madrid, Spain.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3477495.3531960

1 INTRODUCTION

Humans are born to perceive the world from multiple modalities such as vision, sound, and touch. Video, as one of the most informative media due to its abundant multimodal content and temporal dynamics, has been used as an ideal testbed to evaluate how well AI perceives. Text-video retrieval systems enable humans to search videos in a simple and natural manner, hence have drawn numerous enthusiasm from multiple research communities. Different from unimodal tasks like image retrieval, text-to-video retrieval operates across different modalities, *i.e.*, it takes text data as the query to retrieve relevant video items. Therefore, text-to-video retrieval is a challenging task since it requires understanding on not only the content of videos and texts, but also their inter-modal correlation [53].

Earlier works often utilize two-stream models to tackle text-to-video retrieval tasks [1, 30, 36, 38]. As shown in Figure 1a left, these



(a) EDB and MDB methods mainly differ at whether explicit embeddings of text/video are generated.

(b) The words or phrases in the caption are connected to the corresponding local visual parts in the video frames.

Figure 1: (a). Embedding-based methods *v.s.* model-based methods. (b). Local visual/linguistic correlation are worth exploiting for achieving accurate retrieval.

methods first extract the features of text/video by modality-specific encoders, then map the cross-modal features to a common feature space. In the inference stage, once a new text query arrives, the text encoder is called to generate the embedding for efficient distance computation with pre-extracted video features. In this paper, we refer to such methods as embedding-based (EDB) methods since the embeddings of both modalities are explicitly generated. Although easy in implementation and efficient in inference, unfortunately, the retrieval accuracy of EDB methods are often inferior to model-based (MDB) methods [16, 33]. As shown in Figure 1a right, MDB methods typically take a text-video pair as input and utilize deep models to model cross-modal correspondence, then predict its similarity score. Thanks to the excellent capacities of cross-modal modeling, MDB approaches usually outperform EDB ones [16], especially when equipped with large-scale pre-training methods like ClipBERT [28].

The devil is in the details! Recent advances [7, 33, 35, 53] reveal that the inferior performances of EDB methods are essentially caused by the lack of modeling local video/text correspondence details. For example, in Table 1 of UniVL [33], the authors present a comparison between two variants of UniVL, *i.e.*, FT-Joint and FT-Align. As shown by the name, FT-Align harnesses a transformer-based cross-modal encoder to align features of both modals, while FT-Joint simply estimates the similarity score as a dot product of video/text embeddings. Consequently, FT-Align achieves significantly better performances than FT-Joint, demonstrating the importance of modeling local correspondences. We also provide an illustrative example in Figure 1b, where the words or phrases in the caption such as "man in a blue shirt", "garlic", "pan with chicken" correspond to the local visual parts in the video frames. Clearly, it is crucial to model such word-to-patch relationship for attaining accurate text-to-video retrieval. To this end, MDB methods are recently proposed to fully exploit the cross-modal correspondences by deep models [28, 33, 59].

Unfortunately, MDB methods are incapable of generating representations separately for texts and videos, yet representation matters for efficient retrieval! Given the explosive growth of modern data, representations are naturally expected to be convenient in storage and efficient in mutual distance computation. However, due to the lack of explicit representations, current MDB methods have to pair the query with every database item and predict the matching scores of all pairs in the inference stage. Such an exhaustive scan over the entire database is computation-intensive and time-consuming, hence hindering the application of MDB methods to practical retrieval scenarios despite their promising accuracy.

In this work, we propose a new EDB method to tackle the text-to-video retrieval problem, *i.e.*, Cross-Modal Retrieval Transformer (CRET), which not only well exploits the cross-modal correspondences and achieves better accuracy than previous MDB methods, but also demonstrates promising efficiency in retrieval tasks. The credits are mainly attributed to our proposed Cross-modal Correspondence Modeling (CCM) module and Gaussian Estimation of Embedding Space (GEES) loss. Specifically, CCM module first separately takes local textual and visual features as inputs, then maps the local text/video features into a common embedding space and minimizes their mutual distances by transformer-based decoders, and finally produces locally-aligned features for both text and video modals. In the inference stage, rather than feeding a video-text pair and exploiting its cross-modal correspondences, the CCM module allows to take unimodal feature as input, and harness the learned query centers as media for modeling video-text correlation. In this way, CCM module is capable to effectively model local cross-modal correspondences, without suffering from the inefficient pairwise inference of MDB methods. Extensive experiments in Section 4 validate the merits and efficacy of the proposed CCM module.

Moreover, the contiguous frames of videos are often semantically abundant. Thus, exhaustively processing every frame is costly and unnecessary, and it is common to summarize a video by sampling several frames from it [28, 30]. Despite the efficiency in computation, frame sampling inevitably results in information loss. It has long been a challenging problem to balance the information loss and computational overhead when sampling frames from a given video [28, 58]. To tackle this problem, we propose GEES loss as a surrogate function for the vanilla Noise Contrastive Estimation (NCE) loss [11, 24], with a Gaussian assumption of video frame-level features distribution. The novel GEES loss makes implicit dense sampling in the input video embedding space, yet remains lightweight in terms of computation. As far as we know, we are the first to propose the estimation of video embedding space distribution and the corresponding sampling strategy.

We evaluate the performance of the proposed CRET on four text-video retrieval datasets. Our method outperforms state-of-the-art EDB methods, as well as MDB ones pre-trained on HowTo100M dataset [38]. We also conduct extensive ablation studies to analyze the variants and merits of our method. Our contributions are summarized as below:

1. We introduce a transformer decoder based Cross-modal Correspondence Modeling (CCM) module to model the correspondences

between text and video modalities. The text/video features are mapped and aligned in a common embedding space, where we harness shared query centers to align the cross-modal correspondence details. By sharing weights and queries of the decoders, CCM realizes effective modeling of local video/text feature correspondences. Experimental results demonstrate that our CCM module significantly boosts the performance in text-to-video retrieval.

2. We propose a novel Gaussian Estimation of Embedding Space (GEES) loss, which serves as a surrogate function of NCE loss under the assumption that video frame-level features follow a multivariate Gaussian distribution. Compared with NCE loss, our GEES loss is not only efficient to compute, but also helpful to ease the negative impact of information loss caused by sparse sampling. The assumption of Gaussian distribution is soundly validated in Section 4.4. Besides, the proposed GEES loss can also benefit to other scenarios involving videos. As a by-product of this paper, we apply it to a video captioning task to demonstrate its generalizability.

3. We present a new embedding-based text-to-video retrieval framework dubbed as CRET, which can perform cross-modal retrieval effectively and efficiently. We conduct extensive experiments to validate the effectiveness of our method on four standard benchmarks and achieve state-of-the-art results. Notably, without pre-training on extra datasets, our approach outperforms state-of-the-art methods pre-trained on HowTo100M dataset.

2 RELATED WORK

Text-video Retrieval. As presented in the introduction, text-video retrieval methods can be categorized into EDB methods and MDB methods, depending on if the explicit embedding is available. Another perspective to group text-video retrieval methods is whether multi-source information in videos are utilized, *i.e.*, collaborative-experts methods [10, 31, 40] and static-frames methods [28, 38, 54]. The former methods harness multi-source information from videos, such as motion, objects, scenes, sounds, faces, speech, OCR, *etc.* The features of these sources are usually extracted by multiple expert models. The latter group of methods only exploits static frames in videos, and leverages CNN [12, 45] or Transformer [8] to extract the features for static frames. The image encoders are fine-tuned during end-to-end training. In general, collaborative-experts methods demonstrate advantageous performances than static-frames ones, thanks to the fusion of multi-source expert features. However, the experts for different sources need separate training, making most collaborative-experts methods unfeasible for end-to-end updating. In this work, we mainly focus on the static-frames methods for their merits in efficiency, the proposed CRET also lies in this category.

Video Frame Sampling. The sampling strategies of video frames are typically dense sampling or sparse sampling. Usually, collaborative-experts methods such as MMT [10] and HiT [30] tend to conduct dense sampling, because they rely on continuous frames to extract expert features such as motion, sounds, *etc*, while static-frames methods often prefer sparse sampling. For example, ClipBERT [28] samples video frames at the frequency of 16 or 8 frames, CLIP4Clip [34] uniformly samples the video 1 frame per second. Generally speaking, dense sampling has advantages in reducing information loss, while sparse sampling is efficient in computation. It is challenging to well balance information loss and computational cost.

Feature Alignment. Feature alignment is observed to be capable of mitigating semantic gaps between different modalities. Existing works conduct cross-modal alignment by either matching [4, 20, 30, 51] or clustering [37]. For example, HiT [30] first encodes video and text separately, then aligns features through cross-modal contrastive matching. MoEE [37] applies NetVLAD [2] to cluster the text features, with the number of clusters equal to that of experts. Then, they estimate the distances of every cluster-expert pair and take the weighted average distance as the final text-video similarity. T2VLAD [50] shares the idea of NetVLAD [2] and performs alignment by clustering text features and video features of multiple experts into a series of common cluster centers. Nevertheless, these methods neglect the alignment of *local* details of video/text modality. In this paper, we highlight the importance of local details and propose the CCM method to model local correspondence of video and text modalities for text-video retrieval task.

3 PROPOSED METHOD

We present the proposed method CRET in this section. Starting with an overview in Section 3.1, we elaborate on the details of CCM and GEES in Section 3.2 and 3.3, respectively. Then, we introduce the batch-wise training strategy in Section 3.4.

3.1 Model overview

Figure 2 illustrates the overall structure of the proposed CRET method. In view of the good performance of self-attention, we utilize transformers [46] as encoders for both the video stream and the text stream. The outputs of both streams consist of two different types of tokens, *i.e.*, global *CLS* tokens containing global features and local tokens containing local features of text/video modal. The local tokens of both modalities are aligned with the proposed Cross-modal Correspondence Modeling module, which is detailed in Section 3.2. The global features of frames are firstly fused by a temporal transformer, then fed to GEES loss together with the *CLS* token features of the text inputs. As presented in Section 3.3, the proposed GEES loss enables us to implicitly conduct dense sampling in the video feature space, meanwhile still preserves the efficiency of sparse sampling.

Text Encoder. The BERT model [6] has shown great generalization capabilities in language representation. We directly apply BERT base-uncased model as our textual backbone. The input sentence is tokenized and padded to a fixed-length sequence. The *CLS* token at the beginning of the text sequence is exploited to gather the global information in the caption. The features can be denoted as $T_i = \mathcal{F}_{BERT}(\mathcal{T}_i)$, where $\mathcal{T}_i$ represents the textual caption of the i-th video-text pair. We split T_i into two parts for different purposes, *i.e.*, the *CLS* embedding $T_i^g = \{T_{ij}\}_{j=0} \in \mathbf{R}^{1\times d}$ and the sequence embeddings $T_i^l = \{T_{ij}\}_{j=1}^{r} \in \mathbf{R}^{r\times d}$, where r is the sequence length. We use g and l as abbreviations for *global* and *local*, d denotes the dimension of the features.

Video Encoder. The video encoder (Figure 2 left part) consists of two components: the spatial encoder encodes the sampled M frames separately, and the temporal encoder fuses frame-level representations into the final temporal features. In light of the success

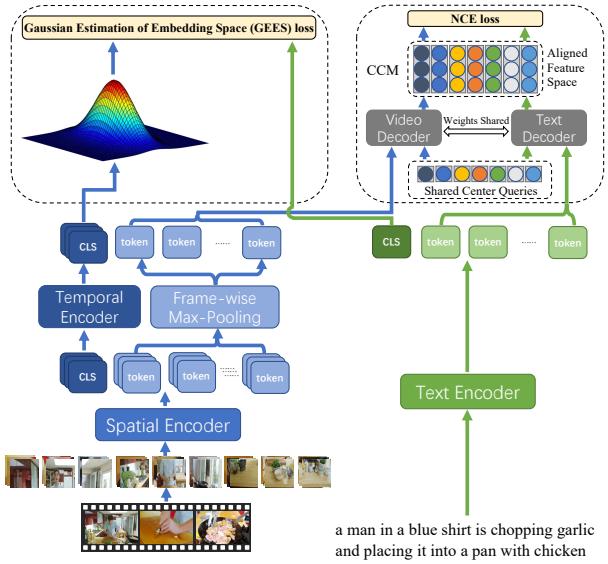


Figure 2: Details of the proposed CRET model, which includes four components, *i.e.*, video encoder, text encoder, Cross-modal Correspondence Modeling (CCM) and Gaussian Estimation of Embedding Space (GEES). (a). As shown in the left part, the encoding of video consists of two stages. First, a spatial encoder samples M frames and encodes each frame separately; Then, the temporal encoder takes the frame-level representation as inputs and conducts temporal fusion. (b). We directly apply BERT base-uncased model as our text encoder, as illustrated in the right part. (c). The outputs of both encoders consist of global *CLS* tokens and a set of local tokens. The local tokens of both modalities are aligned by the CCM module. (d). We assume the features of video frames follow a multivariate Gaussian distribution, and estimate the distribution parameters based on the features of M sampled frames. The GEES loss is computed over the estimated multivariate Gaussian distribution and the global *CLS* token of text modal.

of self-attention for visual feature extraction [9, 46, 56], we adopt the CaiT-S/24 model [47] as our spatial-encoder, which takes p non-overlapping image patches as input. The CaiT model performs self-attention between patches and compiles the set of patch embeddings into a *CLS* embedding. We then leverage a fully connected layer to project the video embeddings to the same dimension as the text embeddings. For a single frame, the output of the CaiT model consists of two parts, a *CLS* token carrying the global feature of the frame and a set of patch embeddings $V_i^l = \{V_{ij}^l\}_{j=1}^M \in \mathbf{R}^{M\times p\times d}$ representing the local features, where p is the number of the patches. We utilize a 3-layers transformer as our temporal encoder. The input tokens for the temporal encoder are all frame-level *CLS* embeddings extracted by the spatial encoder. The final video embeddings from the output of the temporal encoder are represented as $V_i^g = \{V_{ij}^g\}_{j=1}^M \in \mathbf{R}^{M\times d}$.

3.2 Cross-modal Correspondence Modeling

The token-level video/text features contain rich visual and linguistic details, which are supposed to boost the retrieval accuracy if the relationship of these local details can be well exploited. Since the local features of video and text modalities have significant domain gaps, we propose Cross-modal Correspondence Modeling (CCM) module for cross-modal feature alignment from a local perspective.

In detail, we utilize transformer decoders to align the features from video and text modalities, and use C queries $\mathbf{Q} = \{Q_c\}_{c=1}^C$ as common centers of the video/text features, where $Q_c \in \mathbf{R}^{1\times d}$. $\mathbf{Q}$ is randomly initialized and updated during training. Given the local embeddings of text/video modality, the alignment for query Q_c can be expressed as:

$$\mathbf{Z}_{c,j} = softmax(\frac{(Q_c W_j^Q)(EW_j^K)^T}{\sqrt{d_k}})(EW_j^V), \tag{1}$$

where j is the index of the head in multi-head attention layers and $W_j^Q \in \mathbf{R}^{d\times d_k}, W_j^K \in \mathbf{R}^{d\times d_k}, W_j^V \in \mathbf{R}^{d\times d_v}$ are randomly initialized projection matrices. We set $d_k = d_v = d/h$ for each of these parallel attention layers, h is the number of the heads in each multi-head attention layer. These parameters are shared between decoders for both modalities. For video modality, $E = \mathcal{F}_{max-pool}(V_i^l) \in \mathbf{R}^{p\times d}$ where $\mathcal{F}_{max-pool}$ stands for max pooling in the frame channel of V_i^l; for text modality, $E = T_i^l \in \mathbf{R}^{r\times d}$, where T_i^l represents the text local tokens in the i-th text-video pair.

The idea of Equation 1 is to first compute similarity scores between the token features E and the c-th center, and then treat these scores as the weights for the features of text/video modal, so as to improve the salience of the features corresponding to the c-th center. Afterward, we concatenate the aligned features of the multi-head attention layers and project them with another randomly initialized projection matrix:

$$\mathbf{Z}_c = Concat(\mathbf{Z}_{c,1}, \mathbf{Z}_{c,2}, \ldots, \mathbf{Z}_{c,h})W^o, \tag{2}$$

where $W^o \in \mathbf{R}^{hd_v\times d}$.

We denote the aligned video local features as $\mathbf{Z}^v = \{\mathbf{Z}_c^v\}_{c=1}^C \in \mathbf{R}^{C\times d}$, and the aligned textual local features as $\mathbf{Z}^t = \{\mathbf{Z}_c^t\}_{c=1}^C \in \mathbf{R}^{C\times d}$. Note that the text input is usually padded with placeholder tokens to a fixed length. Thus, when processing text modality, we use mask tokens to remove the padded tokens and mitigate the affect caused by padding. The aligned features of video and text modalities are of the same shape and we calculate the similarity score with cosine distance $S_l = cos(\mathbf{Z}^v, \mathbf{Z}^t)$.

3.3 Gaussian Estimation of Embedding Space

Noise Contrastive Estimation (NCE) loss [24] aims to discriminate between positive pairs and those artificially-generated negative pairs. Given a batch that consists of N text-video pairs, the vanilla NCE loss can be written as follows:

$$\mathcal{L}_{NCE} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{M}\sum_{j=1}^{M} -\log(\frac{\exp(<V_{ij}^g, T_i^g>)}{\exp(<V_{ij}^g, T_i^g>) + \sum_{(V^{g\prime}, T^{g\prime})\in\Phi_i}\exp(<V^{g\prime}, T^{g\prime}>)}), \tag{3}$$

where V_{ij}^g is the global features of the j-th frame sampled in the i-th text-video pair, T_i^g is the global features of the caption in the i-th text-video pair, and Φ_i stands for all negative video-text pairs that

are artificially constructed for the i-th text-video pair. Despite the elegant form of NCE loss, its practical performance highly depends on the sampling frequency M. A small M is computationally favored but usually hurts the retrieval performance, while a relatively large M tends to produce more robust NCE loss and improved accuracy at the cost of heavier computational burden.

A natural question to ask is: how to conduct (implicitly) dense sampling, yet remain computationally efficient? To this end, we firstly assume the distribution of frame-level representations, and then estimate the NCE loss over the assumed distribution, so that all latent frame features are implicitly taken into consideration. Specifically, suppose the frame-level features of i-th video follow a multivariate Gaussian distribution:

$$v_i \sim \mathcal{N}(\mu_i, \sigma_i), \tag{4}$$

where μ_i and σ_i stand for a mean vector and covariance matrix. Based on this assumption, we can derive the expectation of Equation 3 as follows:

$$\mathcal{L}_{GEES} = \frac{1}{N}\sum_{i=1}^{N} E_{v_i}[-\log(\frac{\exp(<v_i, T_i^g>)}{\exp(<v_i, T_i^g>) + \sum\limits_{(v\prime, T^g\prime)\in\Phi_i}\exp(<v\prime, T^g\prime>)})]. \tag{5}$$

Since Equation 5 is difficult to compute in its exact form, we further seek an upper bound of Equation 5 as follows:

$$\mathcal{L}_{GEES} \leq -\frac{1}{N}\sum_{i=1}^{N}\{\log[E_{v_i}(\exp(<v_i, T_i^g>))] - \log[\sum_{j=1}^{N} E_{v_i}(\exp(<v_i, T_j^g>))]\}, \tag{6}$$

$$= -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(<T_i^g, \mu_i> + \frac{1}{2}<T_i^g, \sigma_i T_i^g>)}{\sum\limits_{j=1}^{N}\exp(<T_j^g, \mu_i> + \frac{1}{2}<T_j^g, \sigma_i T_j^g>)}, \tag{7}$$

$$= \bar{\mathcal{L}}_{GEES}, \tag{8}$$

where Inequality 6 follows from the Jensen's inequality $E[\log(X)] \leq \log(E[X])$, as the $\log(\cdot)$ is concave. Equation 7 is obtained by leveraging the momentum-generating function $E(e^{tX}) = e^{tE(X)+\frac{1}{2}t^2E(X^2)}$, since v_i is a random Gaussian variable. Essentially, Equation 7 is a surrogate function for the vanilla NCE loss, without suffering from the heavy computation overhead caused by dense sampling. There exist many methods to estimate μ_i and σ_i such as running mean, non-linear mapping, *etc*, for simplicity we adopt a straightforward way to compute the mean vector and covariance matrix, *i.e.*, $\mu_i = \frac{1}{M}\sum_{j=1}^{M} V_{ij}^g$, and $\sigma_i = \frac{1}{M}\sum_{j=1}^{M}(V_{ij}^g - \mu_i)(V_{ij}^g - \mu_i)^{\mathrm{T}}$.

To sum up, the proposed GEES enables us to conduct implicit sampling in the video embedding space, and provides a feasible form of NCE loss that can be efficiently optimized with the stochastic gradient descent (SGD) algorithm. To prove our hypothesis of feature distribution, we validate the assumption in Section 4.4.

3.4 Strategies of Training and Inference

Training. Suppose each batch consists of N text-video pairs. In the training phase, we compute the NCE loss over the output of CCM module by:

$$\mathcal{L}_{CCM} = \frac{1}{N}\sum_{i=1}^{N} -log(\frac{\exp(<\mathbf{Z}_i^v, \mathbf{Z}_i^t>)}{\exp(<\mathbf{Z}_i^v, \mathbf{Z}_i^t>) + \sum\limits_{(\mathbf{Z}^{v\prime}, \mathbf{Z}^{t\prime})\in\Psi_i}\exp(<\mathbf{Z}^{v\prime}, \mathbf{Z}^{t\prime}>)}), \tag{9}$$

where Ψ_i represents all negative text-video pairs that are manually generated for the i-th text-video pair. That is, we treat matched text-video pairs as positive samples, and use all other combinations of texts and videos as negative samples. The network is trained end-to-end by minimizing $\mathcal{L}_{total} = \bar{\mathcal{L}}_{GEES} + \alpha\mathcal{L}_{CCM}$, where α is a hyper-parameter to balance the two losses. We train our network using SGD algorithm.

Inference. Similar to [10, 29, 37, 50], we calculate the overall similarity score as a weighted sum of the global and local similarity scores $S = S_g + \beta S_l$, where $S_l = cos(\mathbf{Z}^v, \mathbf{Z}^t)$ is the cosine distance calculated by the local features, $S_g = cos(V_i^M, T_i^g)$ is the cosine distance calculated by the global features and β is the weight hyper-parameter to adjust the weight of local similarity scores, while $V_i^M = \frac{1}{M}\sum_{j=1}^{M} V_{ij}^g$.

4 EXPERIMENTS

4.1 Experimental Setup

Datasets. We conduct experiments on four benchmark datasets, including **MSRVTT** [52], **LSMDC** [42], **MSVD** [3] and **DiDeMo** [13], to evaluate the performance of our proposed framework on text-to-video retrieval task.

Metrics. For the ease of comparison, we report the experimental results using the standard retrieval metrics, *i.e.*, R@K (recall at rank K, higher is better) [15, 17–19, 21, 22] and MdR [28, 38]. R@K calculates the percentage of test samples for which the correct result is found in the top-K retrieved points to the query sample. For R@K, we report results for R@1, R@5, R@10. The MdR measures the median rank of correct items in the retrieved ranking list, where a lower score indicates a better model.

Implementation Details. We use ImageNet [43] pre-trained CaiT-S/24 model [47] with 24 self-attention layers and 2 class-attention layers as our video spatial encoder, and the pretrained 12-layer BERT-base-uncased model [6] as our text encoder. We train them together with our temporal encoder and CCM module in an end-to-end manner, where the temporal encoder and CCM module are randomly initialized. Adam [26] is adopted as the optimizer, with the initial learning rate set to 5×10^{-5}, and the cosine annealing learning rate scheduler [32] is used. The model is trained for 50 epochs on 8 NVIDIA V100 GPUs with a batch size of 20 per GPU. For the spatial encoder, we resize each frame to 224×224 before feeding to the spatial encoder and set the patch size to 16×16. The final video embedding dimension is 768, in line with the text embedding dimension. We use 8 center queries for our CCM module empirically, and find the number of center queries is insensitive to the results. The CCM module is implemented as a 1 multi-head attention layer with 4 heads. The temporal encoder consists of 3 multi-head attention layers with 3 heads in each layer. We achieve good trade-off between computation cost and performance by sampling 4 frames per video in each dataset. Concretely, we first divide the video into 4 intervals with the same length, then sample 1 frame in each interval randomly. We have one hyper-parameter α in the training stage and one hyper-parameter β in the inference stage. We find that simply setting both α and β to 1 leads to satisfactory accuracy, and the accuracy is insensitive to the choice of fixed α and β.

4.2 Comparison to State-of-the-art

As described in Section 2, despite the advantageous performances of collaborative-experts approaches, they leverage features from

	Weight Initialization		E2E	Type	R@1↑	R@5↑	R@10↑	MdR↓
	Visual	Textual						
MIL-NCE [36]	K+H	G+H	✓	EDB	9.9	24.0	32.4	29.5
JSFusion [54]	I	N.A.	✓	MDB	10.2	31.2	43.2	13.0
HT [38]	I	G	✓	EDB	12.4	36.0	52.0	10.0
HT [38]	I+H	G+H	✓	EDB	14.9	40.2	52.8	9.0
ActBERT [59]	I+H	B+H		MDB	16.3	42.8	56.9	10.0
HiT(appearance-only) [30]	I+H	B+H	✓	EDB	18.2	41.9	55.5	5.0
TACo(R-152) [53]	I+H	B+H	✓	MDB	18.9	46.2	58.8	7.0
UniVL(FT-Joint) [33]	K+H	B+H		EDB	20.6	49.1	62.9	6.0
UniVL(FT-Align) [33]	K+H	B+H		MDB	21.2	49.6	63.1	6.0
ClipBERT [28]	I+C+V	C+V	✓	MDB	22.0	46.8	59.9	6.0
Ours	I	B	✓	EDB	**23.9**	**50.8**	**63.4**	**5.0**

Table 1: Results of text-to-video retrieval on MSRVTT 1K test set, and we follow the split from ClipBERT. The two columns of "Weight Initialization" list the datasets that are exploited to pre-train visual/textual backbones before training on the MSRVTT dataset, where I, K, G, C, V, B, H stand for ImageNet [43], Kinetics [25], GoogleNews [39], COCO Captions [5], Visual Genome Captions [27], BERT-base [48] and HowTo100M [38], respectively. The column "E2E" indicates whether a method can be trained in an end-to-end manner. The column "Type" distinguishes model-based methods (MDB) and embedding-based methods (EDB).

	E2E	Type	R@1↑	R@5↑	R@10↑	MdR↓
HT [38]	✓	EDB	13.0	37.4	52.4	10.0
HT* [38]	✓	EDB	15.5	40.9	55.7	8.0
NoiseE* [1]		EDB	20.3	49.0	63.3	6.0
CLIP4Clip† [34]		EDB	46.2	76.1	84.6	2.0
Ours	✓	EDB	**49.0**	**87.0**	**95.0**	**2.0**

Table 2: Results of text-to-video retrieval on MSVD dataset. ∗ denotes pretraining on HowTo100M dataset, † denotes pretraining on WIT [41].

	E2E	Type	R@1↑	R@5↑	R@10↑	MdR↓
CT-SAN [55]	✓	MDB	5.1	16.3	25.2	46.0
HT [38]	✓	EDB	5.8	18.8	28.4	45.0
HT* [38]	✓	EDB	7.1	19.6	27.9	40.0
NoiseE* [1]		EDB	6.4	19.8	28.4	39.0
JSFusion [54]	✓	MDB	9.1	21.2	**34.1**	36.0
Ours	✓	EDB	**10.0**	**24.9**	33.4	**34.0**

Table 3: Results of text-to-video retrieval on LSMDC dataset. ∗ denotes pretraining on HowTo100M dataset.

multiple experts and usually do not support end-to-end training. Using experts features may further improves the performance, but it is not the focus of this paper. In this section, for the fairness and correctness, we present the comparisons with the state-of-the-art static-frames methods, and all the results of each experiment are obtained by averaging multiple runs, the results in each run are close.

MSRVTT Experiments. The MSRVTT dataset is a general video dataset collected from YouTube with text descriptions, which is composed of 10,000 videos, each with a length that ranges from 10 to 32 seconds and 200,000 captions. We follow the split from ClipBERT [54], which contains 1,000 test text-video pairs. As shown in Table 1, our method outperforms existing works in terms of all metrics, including those methods that pretrained on HowTo100M large scale instructional video dataset. For example, the closest competitor ClipBERT [28] is a recently proposed method that pretrained on two image-text datasets [5, 27]. In contrast, our method does not leverage additional training data. Besides, we also compare with the ablated versions of collaborative-experts methods, *i.e.*, HiT (appearance-only) uses SENet-154 to extract visual features, TACo (R-152) utilizes ResNet-152. Our method still outperforms these two methods. The performance may be further improved by adding multi-experts features, but it is not the focus in this paper.

MSVD Experiments. The MSVD dataset contains 1,970 videos, each with a length that ranges from 1 to 62 seconds. Train, validation and test splits contain 1,200, 100 and 670 videos, respectively. The results are shown in Table 2. It is worth noting that HT [38] and NoiseE [1] are pretrained on HowTo100, and CLIP4Clip [34] is pretrained on the largest multi-modal dataset WIT [41]. Our method improves the state-of-the-art by 2.8%, 10.9% and 10.4% in terms of R@1, R@5 and R@10 metrics, respectively.

LSMDC Experiments. The LSMDC dataset consists of 118,081 videos, each with a length that ranges from 2 to 30 seconds. The videos were extracted from 202 movies. The validation set contains 7,408 videos, and the test set is independent of the training and validation splits and includes 1,000 videos. The results are shown in Table 3. Notably, we achieved 0.9% and 3.7% improvements over the best competitor on R@1, R@5 for text-to-video retrieval task.

DiDeMo Experiments. The DiDeMo dataset contains 10,000 Flickr videos annotated with 40,000 sentences. The results are shown in

	E2E	Type	R@1↑	R@5↑	R@10↑	MdR↓
S2VT [49]	✓	EDB	11.9	33.6	-	13.0
FSE [57]	✓	EDB	13.9	36.0	-	11.0
ClipBERT‡ [28]	✓	MDB	20.4	48.0	60.8	6.0
Ours	✓	EDB	**21.2**	**50.3**	**63.5**	**6.0**

Table 4: Results of text-to-video retrieval on DiDeMo dataset. ‡ denotes pretraining on COCO Captions and Visual Genome Captions.

Method	ClipBERT [28]	Ours
Inference Time	12.5 minutes	20 seconds

Table 5: Total retrieval time for text queries on the MSRVTT 1K test set.

Table 4. Our method consistently improves ClipBERT 2.3% on R@5 for text-to-video retrieval task.

Our method achieves state-of-the-art results on the above four datasets without pretraining on additional video data, and outperforms those methods pretrained on extra datasets (such as HowTo100M). The results obtained in the four datasets above not only firmly validate the efficacy of our method, but also show the capacity of handling videos collected from different domains. In addition, we compare the retrieval efficiency with a representative MDB method (*i.e.*, ClipBERT) in Table 5. Our CRET takes only 20 seconds to retrieve the entire 1k MSRVTT test set, while ClipBERT costs up to 12.5 minutes. The gap in efficiency will further increase when the test set gets larger.

4.3 Ablation Studies

In this section, we conduct experiments to validate the effects of the proposed CCM and GEES loss.

4.3.1 How much does CCM module contribute? We experimentally compare the proposed CCM with T2VLAD [50], which is a representative cross-modal feature alignment method. Though both T2VLAD and CCM align features by clustering *local* features of different modalities into common cluster centers, there exist two key differences between them. On the one hand, T2VLAD aggregates expert features as the weighted sum of the distances to the centers, which shares the similar idea with NetVLAD [2]. As a comparison, our CCM calculates the similarity scores of local text/video features to each center query, and then weights the local features of the text/video modal with these scores to highlight the salience of "local" features. On the other hand, the meaning of "local" in T2VLAD and CCM differs a lot. Specifically, T2VLAD is a collaborative-experts method, where "local" refers to one of all the experts, while "local" in our CCM stands for local regions or words in input videos or captions. Probably this is the reason for the inferior performances of T2VLAD, *i.e.*, it is not originally designed for aligning local region features. As shown in Table 6, our CCM outperforms T2VLAD over all metrics, demonstrating the advantages of the proposed CCM.

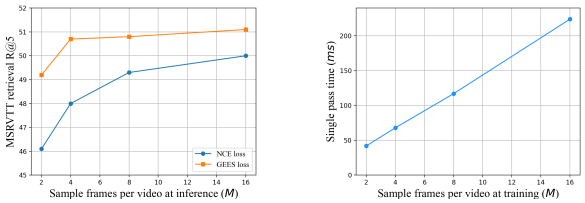
In addition, we also study different design philosophies of our CCM module. Specifically, we investigate how performance varies

Method	Loss	Cross Modal Feature Alignment	R@1↑	R@5↑	R@10↑	MdR↓
CRET	NCE	None	20.3	45.6	54.1	6.0
	GEES	None	21.4	48.2	56.4	6.0
	GEES	T2VLAD [50]	22.1	49.4	58.4	6.0
	GEES	CCM (Ours)	**23.9**	**50.8**	**63.4**	**5.0**

Table 6: The ablation studies on the two modules conducted on the MSRVTT 1K test set. The baseline results are obtained without any modules. We also include other cross modal modeling method to conduct thorough comparisons.

	shared queries	shared weights	R@1↑	R@5↑	R@10↑	MdR↓
CCM			21.6	47.1	57.5	6.0
		✓	22.5	49.5	60.3	6.0
	✓		22.8	47.8	58.9	6.0
	✓	✓	**23.9**	**50.8**	**63.4**	**5.0**

Table 7: The ablation studies on the MSRVTT 1K test set to investigate the different designs of CCM module.



(a) The effectiveness of GEES loss. **(b) Time cost in the training stage.**

Figure 3: The ablation studies on the MSRVTT dataset to investigate the effectiveness of GEES loss and the impact of sampling frames on single pass speed.

when queries and weights of the CCM decoder are shared or not. As shown in Table 7, the one that shares both weights and queries outperforms other variants. The design of shared center queries provides a better alignment to the same semantic topics. More importantly, the feature extraction of video/text can be performed independently. Therefore, our CRET model is still an EDB method.

4.3.2 The effectiveness of GEES loss. To investigate the effectiveness of the GEES loss, we conduct ablation experiments on the MSRVTT dataset, and present the experimental results in Figure 3. As shown in Figure 3a, the X and Y-axis represent sampling frequency per video and R@5 metric. When sampling 2 frames per video, the proposed GEES loss effectively boosts the NCE loss by 3.1%. In cases of other sampling frequencies such as 4, 8 and 16, the GEES loss consistently outperforms the NCE loss. The observations above demonstrate the efficacy and robustness of the proposed GEES loss. In addition, the retrieval R@5 of NCE loss increases faster than that of our GEES loss, with the growing number of sampled frames. This phenomenon indicates that the GEES loss effectively mitigates the information loss with few sampling frames.

Figure 3b demonstrates the comparison of computation cost under different sampling rates during training. The time cost increases almost linearly with the sampling rate M. Our method is robust

Method	GEES	CCM	R@1↑	R@5↑	R@10↑	MdR↓
UniVL (FT-Joint)			*20.6*	*49.1*	*62.9*	*6.0*
			20.7	49.4	62.8	6.0
	✓		21.3	49.8	63.2	6.0
		✓	21.8	50.4	63.6	6.0
	✓	✓	**22.1**	**51.3**	**63.9**	6.0
UniVL (FL-Align)			21.2	49.6	63.1	6.0

Table 8: Ablation studies on the MSRVTT 1K test set. The baseline results are obtained without GEES and CCM. We apply the two methods to the UniVL (FT-Joint) version. We train the baseline results of the UniVL (FT-Joint) with open source code and add our modules to the model.

Method	GEES	BLEU-3	BLEU-4	METEOR	ROUGE-L	CIDEr
UniVL		53.42	41.79	28.94	60.78	50.04
	✓	**54.26**	**42.76**	**29.04**	**61.37**	**50.60**

Table 9: The video captioning results on MSRVTT dataset.

to the sampling rate, thus, the proposed GEES loss reduces the computation amount effectively.

4.3.3 The improvements by CCM and GEES loss on other models. We further couple CCM and GEES loss with UniVL [33] to validate their merits. The experimental results of FT-Joint, a variant of UniVL that belongs to EDB methods, are shown in Table 8. The numbers (highlighted with italics font) in the first row are directly copied from UniVL paper, and the numbers in the second row are reproduced by us running the source code of UniVL [1]. The FT-Joint method enhanced by our CCM module (3rd row) even outperforms its MDB variant FT-Align, demonstrating the effectiveness of the CCM module. Solely applying GEES loss (4th row) can also improves the performance of UniVL, where GEES is implemented using the same sampling frequency as reported in UniVL. The proposed GEES and CCM modules together boost the R@1, R@5 and R@10 of vanilla UniVL method by 1.5, 2.2 and 1.0, as listed in row 5 in Table 8. Notably, our methods are pluggable and can potentially improve other EDB methods.

As a by-product of our work, we also trial GEES on UniVL for the video captioning task. In Table 9, we use the same metrics as used in UniVL, and all metrics increase after adding GEES loss, which demonstrates its generalization ability.

4.4 Validation of Gaussian Assumption

The derivation of GEES loss is based on the assumption that frame-level features of a video follow a multi-variate Gaussian distribution. The validation of this assumption relies on the 3 facts below:

(1) Gaussian random vectors after linear transformation are still Gaussian [44].

(2) For a set of random vectors of any size and dimension larger than 1, we can run the Henze-Zirkler test [14] on it and check the resulted p-value. If the p-value is greater than 0.05, then the set of random vectors are expected to be drawn from a multivariate Gaussian distribution.

[1] https://github.com/microsoft/UniVL

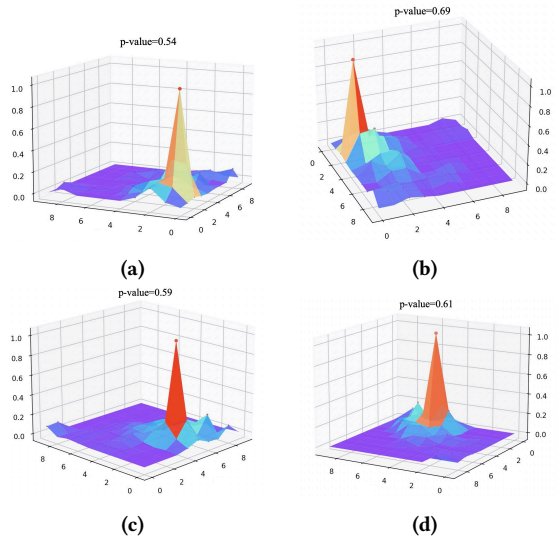


Figure 4: Probability density of video frames features. We utilize the histogram to estimate the probability density. Z-axis represents the probability density.

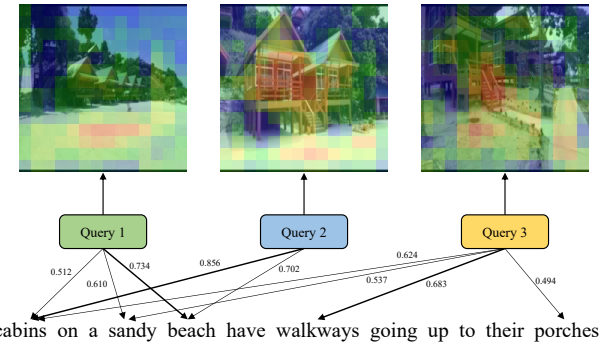


Figure 5: Visualization of the responses to different center queries. The text-video pair in this example is the Video8426 sample in MSRVTT test set. For the simplicity of illustration, we present three of these center queries and visualize the most significant frame for each center query. We use attention scores of the last decoder layer as the weights of different center queries on two modalities. We resize attention weights of video frames to the shape of input size, i.e., 224×224, and draw heatmaps with pseudo color. For text modal, we plot the lines of the words with the dominant weights on each center query and attach the corresponding weights to the lines.

(3) Suppose a random variable X follows a Binomial distribution with the probability p, *i.e.*, a sample of X has p probability to be positive, $1 - p$ probability to be negative. If we randomly draw n samples from X and find all n samples are negative, then at the confidence level of 95%, p is in the range of $[0, \frac{3}{n}]$ [23].

Figure 6: Text-to-video retrieval results on MSRVTT testset. We show top three retrieval results for each query. The upper half of the two retrieval groups are the results that are retrieved with the full model, while the lower half are the retrieval results that are obtained without the CCM module. We use green boxes to represent the video corresponding to the query and red boxes to represent the relevant videos that retrieved.

Given the three facts above, the validation of our Gaussian assumption is straightforward. The idea is to first define a binary event as whether the frame features of a video follow a multivariate Gaussian distribution or not. Suppose this binary event has p and $1-p$ probabilities to be negative and positive, then we can consider a set of video frame features as multiple samples from a Binomial distribution with parameter p. Our goal is to show that the Binomial distribution has a small p with high confidence.

Specifically, for the ease of computation and visualization, we project the video frame vectors from 768 to 2 with a random matrix. The visualization of frame features of several videos are illustrated in Figure 4. Since the entire dataset is large, we randomly select 300 videos from MSRVTT 1K test set to run the Henze-Zirkler test [14] on the projected 2-dimensional feature vectors. It is tenable to do it since linear transformation of Gaussian random vectors are still Gaussian. According to fact 2 above, the Henze-Zirkler test works for set size as 300 and dimension as 2. We found all 300 videos passed the Henze-Zirkler test. Based on fact 3 above, with 95% confidence, we can state that $p \in [0, \frac{3}{n}]$. In other words, with 95% confidence, the frame features of every video in MSRVTT testset follow some multivariate Gaussian distribution with probability at least 0.99, where 0.99 is derived by $1 - \frac{3}{300}$. The validation results prove that our assumption is rational.

4.5 Qualitative Results

In this section, we present qualitative analyses of our proposed model. As shown in Figure 5, we illustrate a text-video pair, the attention heatmaps on sampled frames, and the dominant weights assigned to the words. Specifically, center Query 1 focuses on the beach in the given video and assigns the highest weight to the word "beach" in the caption; center Query 2 concerns more on the cabins as shown by the heatmap, and the word "cabins" has the heaviest weight in the caption; center Query 3 concentrates on the "walkways", which is consistently demonstrated by the visual heatmap and words weights. The observations above illustrate the alignment of the local visual/linguistic parts. In addition, we can also find that different queries focus on different areas and are complementary to each other.

In Figure 6, three text queries and their top 3 retrieval results are presented. The upper and lower halves are results with the full

model and those without the CCM module, respectively. The model can also retrieve related videos without the CCM module, but the results are not accurate due to the lack of local feature alignment, demonstrating the merits of the CCM module.

5 CONCLUSION

In this work, we propose a new embedding-based framework CRET for text-video retrieval. The proposed method well exploits the cross-modal local correspondences between video/text modalities, and demonstrates promising retrieval performance without any pre-training on external video datasets. Moreover, to balance the computational cost and information loss of video frame sampling, we make a Gaussian assumption of the video feature space and propose Gaussian Estimation of Embedding Space (GEES) loss as an upper bound for the vanilla Noise Contrastive Estimation (NCE) loss. With the help of GEES loss, we achieved excellent text-video retrieval performance in terms of both accuracy and efficiency. Extensive experiments demonstrate the advantages of our method against prior state-of-the-art methods. In future, we plan to extend our CCM module and GEES loss to more methods and scenarios, including MDB methods, multi-experts features and pre-training on large-scale video datasets.

REFERENCES

[1] Elad Amrani, Rami Ben-Ari, Daniel Rotman, and Alex Bronstein. 2021. Noise Estimation Using Density Estimation for Self-Supervised Multimodal Learning. In *AAAI*.

[2] Relja Arandjelovic, Petr Gronát, Akihiko Torii, Tomás Pajdla, and Josef Sivic. 2016. NetVLAD: CNN Architecture for Weakly Supervised Place Recognition. In *CVPR*.

[3] David L. Chen and William B. Dolan. 2011. Collecting Highly Parallel Data for Paraphrase Evaluation. In *ACL*.

[4] Shizhe Chen, Yida Zhao, Qin Jin, and Qi Wu. 2020. Fine-Grained Video-Text Retrieval With Hierarchical Graph Reasoning. In *CVPR*.

[5] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. 2015. Microsoft COCO Captions: Data Collection and Evaluation Server. *arXiv* (2015).

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*.

[7] Haiwen Diao, Ying Zhang, Lin Ma, and Huchuan Lu. 2021. Similarity Reasoning and Filtration for Image-Text Matching. In *AAAI*.

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*.

[10] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. 2020. Multi-modal Transformer for Video Retrieval. In *ECCV*.

[11] Michael Gutmann and Aapo Hyvärinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *AISTATS*.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *CVPR*.

[13] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan C. Russell. 2017. Localizing Moments in Video with Natural Language. In *ICCV*.

[14] N. Henze and B. Zirkler. 1990. A class of invariant consistent tests for multivariate normality. *Communications in Statistics - Theory and Methods* 19 (1990).

[15] Weixiang Hong, Yu-Ting Chang, Haifang Qin, Wei-Chih Hung, Yi-Hsuan Tsai, and Ming-Hsuan Yang. 2020. Image Hashing via Linear Discriminant Learning. In *WACV*.

[16] Weixiang Hong, Kaixiang Ji, Jiajia Liu, Jian Wang, Jingdong Chen, and Wei Chu. 2021. GilBERT: Generative Vision-Language Pre-Training for Image-Text Retrieval. In *SIGIR*.

[17] Weixiang Hong, Jingjing Meng, and Junsong Yuan. 2018. Distributed Composite Quantization. In *AAAI*.

[18] Weixiang Hong, Jingjing Meng, and Junsong Yuan. 2018. Tensorized projection for high-dimensional binary embedding. In *AAAI*.

[19] Weixiang Hong, Xueyan Tang, Jingjing Meng, and Junsong Yuan. 2019. Asymmetric Mapping Quantization for Nearest Neighbor Search. *T-PAMI* (2019).

[20] Weixiang Hong, Zhenzhen Wang, Ming Yang, and Junsong Yuan. 2018. Conditional Generative Adversarial Network for Structured Domain Adaptation. In *CVPR*.

[21] Weixiang Hong and Junsong Yuan. 2018. Fried Binary Embedding: From High-Dimensional Visual Features to High-Dimensional Binary Codes. In *T-IP*.

[22] Weixiang Hong, Junsong Yuan, and Sreyasee Das Bhattacharjee. 2017. Fried Binary Embedding for High-Dimensional Visual Features. In *CVPR*.

[23] B. D. Jovanovic and P. S. Levy. 1997. A Look at the Rule of Three. *The American Statistician* 51, 2 (1997), 137–139.

[24] Rafal Józefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. 2016. Exploring the Limits of Language Modeling. *arXiv* (2016).

[25] Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. 2017. The Kinetics Human Action Video Dataset. *arXiv* (2017).

[26] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.

[27] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. 2017. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *IJCV* (2017).

[28] 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 Learning via Sparse Sampling. *arXiv* (2021).

[29] 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. In *EMNLP*.

[30] Song Liu, Haoqi Fan, Shengsheng Qian, Yiru Chen, Wenkui Ding, and Zhongyuan Wang. 2021. HiT: Hierarchical Transformer with Momentum Contrast for Video-Text Retrieval. *arXiv* (2021).

[31] Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. 2019. Use What You Have: Video retrieval using representations from collaborative experts. In *BMVC*.

[32] Ilya Loshchilov and Frank Hutter. 2017. SGDR: Stochastic Gradient Descent with Warm Restarts. In *ICLR*.

[33] 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* (2020).

[34] 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* (2021).

[35] Nicola Messina, Giuseppe Amato, Andrea Esuli, Fabrizio Falchi, Claudio Gennaro, and Stéphane Marchand-Maillet. 2020. Fine-grained Visual Textual Alignment for Cross-Modal Retrieval using Transformer Encoders. *arXiv* (2020).

[36] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. 2020. End-to-End Learning of Visual Representations From Uncurated Instructional Videos. In *CVPR*.

[37] Antoine Miech, Ivan Laptev, and Josef Sivic. 2018. Learning a Text-Video Embedding from Incomplete and Heterogeneous Data. *arXiv* (2018).

[38] 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 *ICCV*.

[39] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv* (2013).

[40] Mandela Patrick, Po-Yao Huang, Yuki Markus Asano, Florian Metze, Alexander G. Hauptmann, João F. Henriques, and Andrea Vedaldi. 2021. Support-set bottlenecks for video-text representation learning. In *ICLR*.

[41] 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. In *ICML*.

[42] Anna Rohrbach, Marcus Rohrbach, and Bernt Schiele. 2015. The Long-Short Story of Movie Description. In *GCPR*.

[43] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. 2015. ImageNet Large Scale Visual Recognition Challenge. *IJCV* (2015).

[44] G. Strang. 1993. Introduction to linear algebra. *Wellesley-Cambridge Press Wellesley* (1993).

[45] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. 2017. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. In *AAAI*.

[46] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. 2021. Training data-efficient image transformers & distillation through attention. In *ICML*.
[47] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. 2021. Going deeper with Image Transformers. *arXiv* (2021).
[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *NIPS*.
[49] Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond J. Mooney, and Kate Saenko. 2015. Translating Videos to Natural Language Using Deep Recurrent Neural Networks. In *NAACL-HLT*.
[50] Xiaohan Wang, Linchao Zhu, and Yi Yang. 2021. T2VLAD: Global-Local Sequence Alignment for Text-Video Retrieval. *arXiv* (2021).
[51] Yu Wu, Linchao Zhu, Yan Yan, and Yi Yang. 2019. Dual Attention Matching for Audio-Visual Event Localization. In *ICCV*.
[52] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. MSR-VTT: A Large Video Description Dataset for Bridging Video and Language. In *CVPR*.
[53] Jianwei Yang, Yonatan Bisk, and Jianfeng Gao. 2021. TACo: Token-aware Cascade Contrastive Learning for Video-Text Alignment. *arXiv* (2021).
[54] Youngjae Yu, Jongseok Kim, and Gunhee Kim. 2018. A Joint Sequence Fusion Model for Video Question Answering and Retrieval. In *ECCV*.
[55] Youngjae Yu, Hyungjin Ko, Jongwook Choi, and Gunhee Kim. 2017. End-to-End Concept Word Detection for Video Captioning, Retrieval, and Question Answering. In *CVPR*.
[56] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Francis E. H. Tay, Jiashi Feng, and Shuicheng Yan. 2021. Tokens-to-Token ViT: Training Vision Transformers from Scratch on ImageNet. *arXiv* (2021).
[57] Bowen Zhang, Hexiang Hu, and Fei Sha. 2018. Cross-Modal and Hierarchical Modeling of Video and Text. *arXiv* (2018).
[58] Yuan Zhi, Zhan Tong, Limin Wang, and Gangshan Wu. 2021. MGSampler: An Explainable Sampling Strategy for Video Action Recognition. *arXiv* (2021).
[59] Linchao Zhu and Yi Yang. 2020. ActBERT: Learning Global-Local Video-Text Representations. In *CVPR*.