Audio-Enhanced Text-to-Video Retrieval using Text-Conditioned Feature Alignment

Sarah Ibrahimi[1*] Xiaohang Sun[2] Pichao Wang[2] Amanmeet Garg[2]
Ashutosh Sanan[2] Mohamed Omar[2*]
[1] University of Amsterdam [2] Amazon Prime Video
s.ibrahimi@uva.nl, {sunking, wpichao, amanmega, ashsanan}@amazon.com, mkamalmomd@gmail.com

Abstract

Text-to-video retrieval systems have recently made significant progress by utilizing pre-trained models trained on large-scale image-text pairs. However, most of the latest methods primarily focus on the video modality while disregarding the audio signal for this task. Nevertheless, a recent advancement by ECLIP*SE has improved long-range text-to-video retrieval by developing an audiovisual video representation. Nonetheless, the objective of the text-to-video retrieval task is to capture the complementary audio and video information that is pertinent to the text query rather than simply achieving better audio and video alignment. To address this issue, we introduce TEFAL, a* ***TE****Xt-conditioned* ***F****eature* ***AL****ignment method that produces both audio and video representations conditioned on the text query. Instead of using only an audiovisual attention block, which could suppress the audio information relevant to the text query, our approach employs two independent cross-modal attention blocks that enable the text to attend to the audio and video representations separately. Our proposed method's efficacy is demonstrated on four benchmark datasets that include audio: MSR-VTT, LSMDC, VATEX, and Charades, and achieves better than state-of-the-art performance consistently across the four datasets. This is attributed to the additional text-query-conditioned audio representation and the complementary information it adds to the text-query-conditioned video representation.*

1. Introduction

The emergence of online streaming services has led to an enormous and rapidly growing collection of multimedia assets comprising video and audio. Retrieving semantically similar content in these assets is crucial for finding information of interest, making it an important aspect of major streaming platforms. Text-to-video retrieval is a common approach to achieve this goal, whereby video content that best matches a textual description is searched for by learning a joint latent space for text and video representations. This space allows the text modality input to be matched with the video modality, enabling the closest videos to be found through a distance metric.

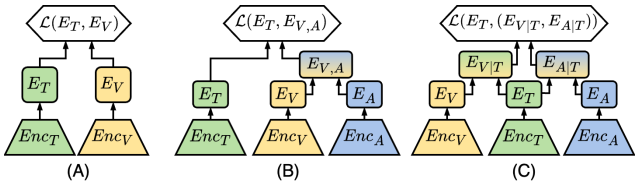


Figure 1. Comparison of our method with the current methods, text-to-video retrieval with (A) video only, (B) audio-video fusion and (C) proposed text-conditioned audio-video alignment (***TEFAL***).

The rise of large-scale transformer models of vision and language has led to the development of many multimodal transformer-based architectures that are commonly evaluated on the text-to-video retrieval task. These architectures can either be pre-trained on large-scale multimodal datasets from scratch or use existing pre-trained models, such as CLIP [29], as starting points or frozen backbones. One of the earliest CLIP-based model architectures, CLIP4Clip [26], shows a significant improvement in performance compared to previous state-of-the-art methods [3, 19, 21] on common text-to-video retrieval benchmarks. By utilizing only a few video frames per video and a simple technique to aggregate the frame embeddings for each video, CLIP4Clip demonstrated the utility of a 2D vision transformer to outperform model architectures using 3D videos as input. Since then, many other works have been built upon this baseline approach with novel ways of cross-attention [16], pretext tasks [12], prompting [20], and other architectural modifications. Recently, significant improvements have been obtained by incorporating post-processing tech-

*This work was done while Sarah Ibrahimi and Mohamed Omar were at Amazon Prime Video.

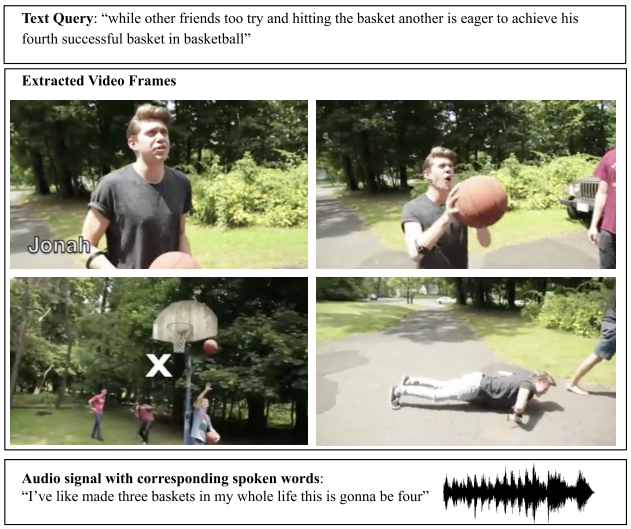


Figure 2. The audio signal can play a significant role in text-to-video retrieval. For instance, consider a textual description that mentions the fourth successful basket, which cannot be retrieved from the visual information alone. However, the spoken words in the video also contain the number four and can be matched to the query. Additionally, sounds of the basketball can further confirm the sport being played.

niques, such as Querybank Normalization [6].

Most of the existing text-to-video retrieval models only consider the correspondence between text and video, as visualized in Figure 1(A), despite the fact that most videos also contain an audio track. As shown in Figure 2, audio can be related to actions, events, objects, and words in the caption, indicating that the audio signal could be beneficial for the task of text-to-video retrieval. Although the use of other modalities for text-to-video retrieval, including audio, has been studied in [33], their results showed no improvement compared to the video-only setting. Recently, ECLIPSE [22] successfully accounted for audio in the video representation by aligning audio and video using cross-modality attention, demonstrating its benefits for long-range text-to-video retrieval.

However, solely embedding the audio features for better video representations may not optimize the objective of text-to-video retrieval, which is to capture complementary audio and video information relevant to the text query. Videos can contain sounds that are not strongly correlated with the visual content, such as a gunshot without a visible gun in the video, as well as other types of off-screen sounds. If audio and video are processed jointly, important audio cues that are relevant to the text query but not visible in the video might be suppressed. To address this issue, we propose TEFAL, a framework that generates text-conditioned representation of both video and audio to achieve effective audio-enhanced text-to-video retrieval, as depicted in Figure 1.

We use the CLIP [29] model as the video and text feature extractor and the AST [15] model as the audio feature extractor in our approach. The text feature serves as a crucial link between the video and audio representations, acting as the query in the calculation for cross-attention. Meanwhile, the video and audio features are used for key and value computations. To simplify matters, the two aligned feature types are combined to produce the final audio-enhanced video representation. We conducted extensive experiments on various datasets that include audio, such as MSR-VTT [41], LSMDC [32], VATEX [39], and Charades [34]. Our proposed audio-enhanced text-conditioned feature alignment method consistently outperforms existing methods. Specifically, our method improves the Recall@1 by over 4% compared to ECLIPSE on the MSR-VTT dataset.

Our key contributions are summarised as follows:

- We propose a text-conditioned feature alignment approach for audio-enhanced text-to-video retrieval. We are the first to do so and we explain why this approach is more suitable for this task than audiovisual alignment. To achieve this, we utilise two independent cross-modal attention blocks for the text to attend to the audio and video representations.

- We conducted extensive experiments on several benchmark datasets that include audio, namely MSR-VTT, LSMDC, VATEX, and Charades. Our results demonstrate state-of-the-art performance in text-to-video retrieval when compared with the best previously published results.

2. Related Work

Our proposed method is closely related to text-to-video retrieval, multimodal video learning and audio-based multimodal learning. In the following we go over some of the main works in these three directions and more comprehensive works are referred to in the survey papers [4, 5].

2.1. Text-to-Video Retrieval

The text-to-video retrieval task has gained significant attention in recent years [3, 10, 11, 14, 18, 19, 24, 25, 27, 33, 40, 44, 46], where the goal is to retrieve relevant videos by a text query from a database of video clips. Prevailing works usually leverage model architectures pre-trained on large-scale text-to-video or text-image datasets [3, 19, 44, 46]. Upon the release of the CLIP model [29], which consists of strong image and text backbones pre-trained on 400M image-text pairs, sparse sampling approaches started to gain popularity and achieved high performance with the release of CLIP4Clip [26]. Since then, many text-to-video retrieval methods have taken CLIP and GPT [7, 12, 20, 27, 30, 31]

backbones as main components and improved by introducing *e.g*. new cross-modal fusion [16] and token selection [25]. Our method also takes advantages of the pre-trained text-video models for robust feature extractions.

2.2. Multimodal Video Learning

Multimodal video models focus on a wide range of tasks, such as visual commonsense reasoning, visual question answering, activity recognition and text-to-video retrieval [42]. Li *et al.* [21] proposed to learn a hierarchical structure where the local context of a video frame is encoded first by a cross-modal transformer, followed by a temporal transformer to learn global video context embeddings. Contrary to works that use dense sampling of video frames and 3D features, ClipBERT [19] introduced a pipeline that uses sparse sampling of video frames and simple 2D visual architectures during training.

2.3. Audio in Multimodal Learning

Including the audio modality has been a topic of research in multimodal works. Merlot Reserve [43] included the audio modality in large scale pre-training and designed a new contrastive mask training task where both text and audio are masked out. The Video-Audio-Text Transformer (VATT) [1] is a convolution-free transformer architecture that can process multiple modalities including audio. Multimodal Versatile Networks [2] presents a self-supervised multimodal learning strategy. For text-to-video retrieval, [23, 28, 40] integrated audio in the pipeline by using NetVLAD as an approach to aggregate audio features. MEE [28] uses a scalar product between the text and audio feature and CE [23] uses a MLP to model the pairwise relation between text and audio. T2VLAD [40] uses a global-local alignment approach where clustering is used for the local alignment without explicitly conditioning on the text.

More recently, Shvetsova *et al.* [33] designed a multimodal fusion transformer that processes input from a combination of modalities. However, for text-to-video retrieval it claims that the fusion of video and audio modalities in their setup is not beneficial compared to only using visual information in a zero-shot setting. To the best of our knowledge, ECLIPSE [22] is the first method to show the benefit of including audio in text-to-video retrieval in combination with strong pre-trained backbones. It introduces a symmetrical type of cross-attention for video and audio to align both modalities and shows its effectiveness in long-range video retrieval. In this work, we do not focus on the alignment between audio and video, but on the alignment between text-audio and text-video simultaneously. By extensive ablation studies, we show that it is not straightforward to combine the audio modality with other modalities for text-to-video retrieval, but we present a simple and effective text-conditioned feature alignment method to boost the performance by using audio.

3. Method

In this section, we describe our proposed method *TEFAL* to achieve **TE**xt-conditioned **F**eature **AL**ignment for audio-enhanced text-to-video retrieval. We evaluated multiple model architectures and the best results across several datasets were achieved by using two independent cross-modal attention blocks for the text to attend to the audio and video representations. The final representation of both the audio and video content are derived independently by conditioning on the text attention weights estimated in the corresponding cross-modal attention blocks. The complete model architecture, training setup and evaluation setup is presented in Figure 3. We explain our key insights in 3.1, followed by our overall architecture in 3.2.

3.1. Key Insight: Text-conditioned Feature Alignment

In most existing works, to achieve the multi-modality fusion, audio-video feature alignment is performed with direct feature fusion [38] and more recently with cross-modal attention mechanisms [22].

For the task of text-to-video retrieval, the goal is to align the text and video features in a joint latent space. However, the text is in general less expressive than the video and corresponds to a subset of the information provided by the video. Therefore, for text-to-video retrieval, the video representation would be better estimated conditioned on the text query to emphasize the aspects of the video which are more relevant to the text.

With these insights, our method, TEFAL, aligns the video frame tokens with text guidance in a text-video cross-attention block. Similar to [16], the video frame embeddings ($\in \mathbb{R}^{F \times D_p}$) are the key and value inputs and the text embedding ($\in \mathbb{R}^{1 \times D_p}$) is the query input to the cross-attention computation, where D_p is the embedding dimension of the respective features. Here, the cross-attention computation aims to condition the video frame tokens representation on the text query to obtain a weighted fusion based on the similarity between text and video frames. More importantly, the text tokens perform weighted token fusion in the frame dimension (F) to select the frames most similar to each of the text tokens.

Further, the audio modality in a video contains information key to identify the video itself. Thus, similar to the previously discussed cross-attention processing of video frames, our method aligns the audio tokens with the text query in a text-audio cross-attention block. Here, the audio embedding ($\in \mathbb{R}^{N_a \times D_p}$) is the key and value input and the same text embedding as above is the query input to the cross-attention computation. The cross-attention computation aims to weigh the tokens in the audio embedding to

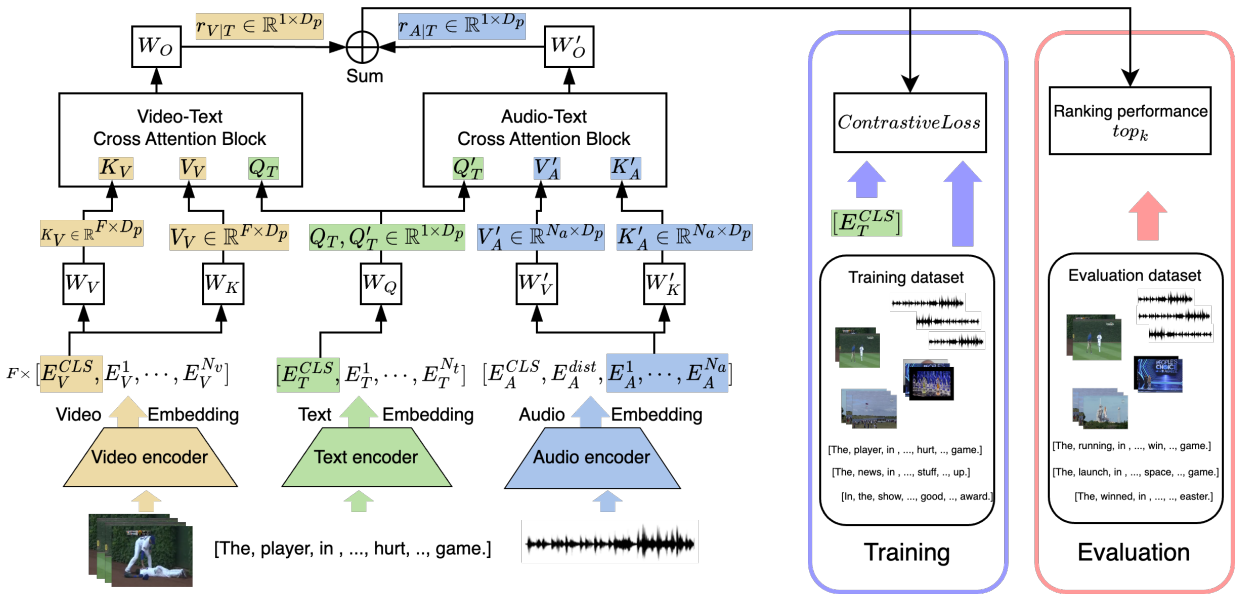


Figure 3. The model architecture of our method *TEFAL* is presented in the left of the figure. The highlighted tokens from the encoders are the F CLS tokens for video frames, one CLS token for the text caption and N_a patch embeddings from the audio encoder. Query, Key and Value projections are created from these tokens and are used in the cross-attention blocks. These cross-attention blocks give us a text-conditioned video embedding and audio embeddings, which are fused through summation and used during training and evaluation, as presented on the right of the figure.

perform weighted fusion conditioned on the text representation. In the text-audio cross-attention, the representation of the audio tokens, corresponding to the input patches from the audio signal spectrogram, are updated conditioned on the text query.

3.2. Overall Architecture

Inspired by the works of [16, 26], we bootstrap from joint image and text models. More specifically, we build on the CLIP [29] model as our text and video frame backbone and Audio Spectrogram Transformer (AST) [15] as our audio backbone to obtain feature embeddings. For a single text-video pair, given a text description T, a video V with F video frames and an audio signal A, we compute their feature embeddings as follows:

Video and Text Features For each video frame v^f we compute features for N_v patches uniformly sampled in the spatial dimension. The text description for a video can vary in length and contains N_t words. The CLIP model outputs a video frame embedding $\in \mathbb{R}^{(N_v+1)\times D}$ and a text embedding $\in \mathbb{R}^{(N_t+1)\times D}$, where the additional token is the class token E_V^{CLS} and E_T^{CLS} for video and text inputs, respectively. As the final video embedding E_V, we use F CLS tokens E_V^{CLS}, one token for each video frame, and for text embedding E_T, one single text token E_T^{CLS} that represents the query embedding.

Audio Features To compute audio features, the Mel-spectrogram features of the audio signal A is passed as input to the encoder. We utilize the recent AST encoder model [15] which demonstrated a strong performance in the audio classification task. From each spectrogram, N_a patches are produced from adjacent overlapping windows to give a final set of $\in \mathbb{R}^{(N_a+2)\times D}$ tokens. For input audio embedding E_A to the cross-attention component, we use N_a patch embeddings and discard the two additional tokens, the CLS token (E_A^{CLS}) and the distillation token (E_A^{dist}).

Text-conditioned Audio and Video Representations The CLS token is well understood to have fused information from all other tokens [15, 35]. We utilize the CLS token as the final feature embedding for the video frame and the text input. The text-video cross-attention selects text-video frame similarity and takes the entire set of F frames $\in \mathbb{R}^{F\times D}$ as input. Similar to the text-video cross-attention, the text-audio cross-attention selectively fuses audio patch embeddings based on the importance of the parts in the audio signal. Thus, we use all the N_a tokens.

The final feature embeddings are projected into a common latent space, where the projections are defined as:

$$\begin{aligned} Q_T &= \text{LN}(E_T^T)W_Q \\ K_V &= \text{LN}(E_V)W_K \\ V_V &= \text{LN}(E_V)W_V \end{aligned} \quad \leftrightarrow \quad \begin{aligned} Q'_T &= \text{LN}(E_T^T)W'_Q \\ K'_A &= \text{LN}(E_A)W'_K \\ V'_A &= \text{LN}(E_A)W'_V \end{aligned} \tag{1}$$

where the learned weight matrices $W_Q, W_K, W_V \in \mathbb{R}^{D\times D_p}$ and $W'_Q, W'_K, W'_V \in \mathbb{R}^{D\times D_p}$ project the final embeddings from $\mathbb{R}^D$ to $\mathbb{R}^{D_p}$. LN stands for LayerNorm.

The text query attends to the audio and video features via the scaled dot product attention (XAttn),

$$\begin{aligned} \text{XAttn}(Q_T, K_V, V_V) &= \text{softmax}\left(\frac{Q_T K_V^T}{\sqrt{D_p}}\right) V_V \\ \text{XAttn}(Q'_T, K'_A, V'_A) &= \text{softmax}\left(\frac{Q'_T {K'_A}^T}{\sqrt{D_p}}\right) V'_A \end{aligned} \tag{2}$$

In the text-video cross-attention, the text query attends to the per-frame video tokens in the key input and selectively fuses based on similarity between text and video tokens. Similarly, in text-audio cross-attention, the text query attends to the audio tokens from the full audio input and fuses based on similarity between text and audio tokens.

To get the final text conditioned audio and video embeddings, we project the cross-attention based output to the final output space via a weight matrix W_O.

$$\begin{aligned} E_{V|T} &= \text{LN}(\text{XAttn}(Q_T, K_V, V_V)W_O) \\ E_{A|T} &= \text{LN}(\text{XAttn}(Q'_T, K'_A, V'_A)W'_O) \end{aligned} \tag{3}$$

The joint embedding $E_{(V,A)|T}$ for a single video is obtained by a simple addition of the text conditioned audio $E_{A|T}$ and video embedding $E_{V|T}$. In Section 4.3.3 we will further elaborate on the effectiveness of this simple fusion method and make an experimental comparison with other potentially suitable fusion methods.

Text-to-Video Retrieval. The final fused embedding $E_{(V,A)|T}$ is compared with the text query embedding E_T, which is the text CLS token E_T^{CLS}, via cosine similarity with the help of the following loss.

Loss. Our model is trained by using the infoNCE loss, a loss which is commonly used for contrastive learning since [36] and later more specifically for training with image-text pairs such as in CLIP [29]. Consider having K text and video-audio embedding pairs $\{(E_T^i, E_{(V,A)|T}^i\}_{i=1}^K$. The infoNCE loss is applied on these pairs where a matching text caption and video are seen as a positive sample and all other caption-video combinations in the batch are seen as negatives. We optimize this loss in a symmetric way by using two losses, a text-to-video ($t2v$) and video-to-text ($v2t$) retrieval loss and taking the sum of these two as the total loss.

$$\mathcal{L}_{t2v} = -\frac{1}{B}\sum_{i=1}^{B} \log \frac{\exp^{s(E_T^i, E_{(V,A)|T}^i)\cdot\tau}}{\sum_{j=1}^{B} \exp^{s(E_T^i, E_{(V,A)|T}^j)\cdot\tau}} \tag{4}$$

$$\mathcal{L}_{v2t} = -\frac{1}{B}\sum_{i=1}^{B} \log \frac{\exp^{s(E_T^i, E_{(V,A)|T}^i)\cdot\tau}}{\sum_{j=1}^{B} \exp^{s(E_T^j, E_{(V,A)|T}^i)\cdot\tau}} \tag{5}$$

$$\mathcal{L} = \mathcal{L}_{t2v} + \mathcal{L}_{v2t} \tag{6}$$

where $s(E_T^i, E_{(V,A)|T}^j)$ is the cosine similarity between the text embedding E_T^i and the fused audiovisual feature $E_{(V,A)|T}^j$, B is the batch size and τ is a learnable scaling parameter, also known as the temperature. By bootstrapping from a pre-trained CLIP model and through our cross-modal attention mechanism, training with this loss enables our model to learn to match a text with its most semantically similar sub-regions of the ground-truth video.

4. Experiments

We perform experiments on benchmark datasets for text-to-video retrieval that include audio tracks, and evaluate our performance following existing literature [16] and report the Recall@1 (R1), Recall@5 (R5), Recall@10 (R10), Median Rank (MdR), and Mean Rank (MnR) scores.

4.1. Datasets

MSR-VTT [41] is considered as the most common dataset for text-to-video retrieval and the videos come with an audio track, consisting of 10,000 web video clips between 10-32 seconds. The dataset has two commonly used splits: one with 7k training videos and one with 9k, resulting in 140k and 180k video-caption pairs, respectively. Both splits use the same evaluation set of 1000 video-caption pairs. We report results on both splits. Similar to [17, 22, 33], we use the audio signals that are provided with the videos. 8,811 of the 10,000 videos have the audio track.

LSMDC [32] contains 118,081 video clips from movies each paired with a single caption description. The lengths of videos range from 2 to 30 seconds. 101,079 videos are used for training, 7,408 for validation and 1,000 for testing, following the setting of X-Pool [16].

VATEX [39] contains 34,991 video clips with multiple captions per video. We follow the HGR [8] split protocol. There are 25,991 videos in the training set, 1,500 videos in the validation set and 1,500 videos in the test set. This dataset is regarded as one long-range video dataset.

Charades [34] contains 9,848 videos with one textual description per video. The average video length is 28 seconds. We follow [22] in their train and test setup.

4.1.1 Data Preprocessing

We leverage pre-trained backbones as baselines to further improve our model's performance. The AST backbone is pre-trained on datasets where all audio files have the same duration. Correspondingly, both the frame shift f_{shift} in the windowed Fourier transform and the target length L_{tar} of the Mel Filter Bank (MFB) features are fixed. The CLIP video encoder, however, is pre-trained on datasets where videos have varying duration, and uniformly sampled frames from videos. To reconcile the mismatch between

Method	Modality	R1 ↑	R5 ↑	R10 ↑	MdR ↓	MnR ↓
X-Pool [16]	V	43.9	72.5	82.3	2.0	14.6
X-Pool [16]	A	5.6	14.7	21.3	109.0	157.4
TEFAL	A + V	**48.1**	**73.8**	**82.8**	2.0	12.1

Table 1. Results on MSR-VTT 7k that confirm the potential of audio as an additional signal for the text-to-video retrieval task. By using audio without the corresponding video, we show that a correspondence between text and audio can be learned. M denotes Modalities. V and A represent Video and Audio, respectively.

the two preprocessings, we adaptively set the frame shift of MFB in audio preprocessing so that MFB features are uniformly sampled from the audio signal and that MFB feature length is fixed across samples. The frame shift used in the filter bank calculation depends on the audio length. Specifically, the frame shift f_{shift} in milliseconds is calculated as $\mathbf{f_{shift}} = n_{frm} \cdot 1000/(sr \cdot L_{tar})$, where n_{frm} is the number of sampled audio frames, sr is the sampling rate in Hz and L_{tar} is the target audio filter bank length.

4.2. Implementation Details

For the AST model, we use a data efficient image transformer DeiT [35] model, pre-trained on ImageNet and AudioSet [13]. We finetune this on the audio files from the MSR-VTT dataset. Whenever a video does not have an audio file, we set the filter bank to a zero vector.

In our implementation, we set the embedding and projection dimensions as 512 ($D = D_p = 512$). We set the number of video frames F to 12 for MSR-VTT and LSMDC and to 32 for VATEX and Charades, similar to X-Pool [16] and ECLIPSE. The final video input to the cross-attention module has 12 or 32 tokens of 512 dimensions depending on the dataset. For the audio signal, we use audio sampling rate $sr = 16k$ and set the target length $L_{tar} = 1024$, which is the same as that used in ImageNet and AudioSet pretraining. The final patch embeddings that are input to the cross-attention block have 1212 tokens of 512 dimensions.

During finetuning we use a simplified strategy compared to the original AST [15] and disable mixup and spectrogram masking. Our models with a ViT-B/32 backbone are trained with batch size 12 on a single A100 GPU and require 1 day of training time. Our models with a ViT-B/16 backbone are trained on 8 V100 GPUs with a batch size of 32.

4.3. Experimental Results and Analysis

4.3.1 Main Results

In Table 1, we present the results that support the motivation of our approach. First we evaluate on text-to-video and text-to-audio retrieval separately on MSR-VTT 7k. Although the results on text-to-audio retrieval are lower, R1 of 5.6% compared to text-to-video retrieval result, R1 of 43.9%, we see that combining the modalities for audio enhanced text-to-video retrieval gives us an improvement of 4% on MSR-VTT 7k, compared to the text-to-video retrieval model only.

Method	R1 ↑	R5 ↑	R10 ↑	MdR ↓	MnR ↓
Everything at once† [33]	-	62.7	75.0	-	-
ALPRO* [20]	33.9	60.7	73.2	3.0	-
CLIP4Clip$_{meanP}$ [26]	42.1	71.9	81.4	2.0	16.2
ECLIPSE$_{meanP}$† [22]	43.2	71.5	81.9	2.0	15.9
CenterCLIP [45]	43.7	71.3	80.8	2	16.9
X-Pool[16]	43.9	72.5	82.3	2.0	14.6
TEFAL	**48.1**	**73.8**	**82.8**	2.0	12.1

Table 2. This table presents the results on MSR-VTT 7k. All models use ViT-B/32 backbones that are pre-trained on WebImageText. * indicates that the model is pre-trained on 5.5M additional text-image and text-video pairs [29]. † indicates that audio is used. - denotes that the value was not reported in the original paper.

Method	R1 ↑	R5 ↑	R10 ↑	MdR ↓	MnR ↓
ViT-B/32 (backbone)					
CLIP4Clip$_{meanP}$ [26]	43.1	70.4	80.8	2.0	15.3
CenterCLIP [45]	44.2	71.6	82.1	2	15.1
ECLIPSE$_{meanP}$† [22]	44.9	71.3	81.6	2.0	15.0
BridgeFormer*[12]	44.9	71.9	80.3	2.0	15.3
X-CLIP[27]	46.1	73.0	83.1	2.0	13.2
X-Pool[16]	46.9	72.8	82.2	2.0	14.3
TS2-Net[25]	47.0	74.5	83.8	2.0	13.0
CAMoE + DSL [9]	47.3	74.6	83.8	2.0	11.9
TEFAL	**49.4**	**75.9**	**83.9**	2.0	12.0
ViT-B/16 (backbone)					
OmniVL* [37]	47.8	74.2	83.8	-	-
TEFAL	49.9	76.2	84.4	2.0	11.4
TEFAL+DSL	50.1	**77.0**	85.4	1.0	10.5
TEFAL+DSL+QB-Norm	**52.0**	76.6	**86.1**	1.0	11.4

Table 3. Results on MSR-VTT 9k split. All works use a CLIP ViT backbone which is pre-trained on WebImageText. Both ViT-B/32 and ViT-B/16 backbones are adopted for evaluation. Post-processing techniques, DSL [9] and QB-Norm [6] are also used to boost the performance. * indicates the use of additional training pairs, more specifically BridgeFormer uses 5.5M additional training pairs and OmniVL uses 14M pairs.

In Tables 2 and 3 we present the results on MSR-VTT 7k and 9k splits respectively and show that our model outperforms current state-of-the-art results. More specifically, for MSR-VTT 7k, our method is 4.9% better in R1 than ECLIPSE [22], the current best method that uses audio for text-to-video retrieval. For MSR-VTT 9k we outperform ECLIPSE by 4.5% for the R1. However, we also outperform other state-of-the-art methods with 2.1%. This performance can be boosted even more with the use of a larger backbone (+0.5%) and QB-Norm (+2%). The supplementary material contains video-to-text retrieval and qualitative results.

In Table 4, 5, and 6, we show that TEFAL also outperforms current state-of-the-art methods with an improvement of R1 of about 1% for LSMDC and VATEX compared to the best previous method and up to 2.4% for Charades. We

Method	R1 ↑	R5 ↑	R10 ↑	MdR ↓	MnR ↓
CLIP4Clip$_{meanP}$ [26]	20.7	38.9	47.2	13.0	65.3
CenterCLIP [45]	21.9	41.1	50.7	10.0	55.6
TS2-Net[25]	23.4	42.3.	50.9	9.0	56.9
X-Pool [16]	25.2	43.7	53.5	8.0	53.2
CAMoE + DSL [9]	25.9	**46.1**	53.7	-	54.4
TEFAL	**26.8**	**46.1**	**56.5**	**7.0**	**44.4**

Table 4. Results on the test split of LSMDC [32]

Method	R1 ↑	R5 ↑	R10 ↑	MdR ↓	MnR ↓
CLIP4Clip$_{seqTransf}$ [26]	55.9	89.2	95.0	1.0	3.9
ECLIPSE$_{meanP}$† [22]	57.8	88.4	94.3	1.0	4.3
TS2-Net[25]	59.1	90.0	95.2	1.0	**3.5**
X-Pool [16]	60.0	90.0	95.0	1.0	3.8
TEFAL	**61.0**	**90.4**	**95.3**	1.0	3.8

Table 5. Results on the test split of VATEX [39]

Method	R1 ↑	R5 ↑	R10 ↑	MdR ↓	MnR ↓
CLIP4Clip$_{meanP}$ [26]	13.9	-	-	-	-
ECLIPSE$_{meanP}$† [22]	15.7	-	-	-	-
X-Pool [16]	16.1	35.2	44.9	14.0	67.2
TEFAL	**18.5**	**37.3**	**48.6**	**11.0**	**60.6**

Table 6. Results on the test split of Charades [34]

Method	Finetuned AST	Adaptive f_{shift}	R1 ↑	R5 ↑	R10 ↑
TEFAL	✓	✓	**49.4**	**75.9**	**83.9**
TEFAL	✓		48.6	74.7	84.1
TEFAL		✓	45.6	72.8	83.2

Table 7. A correct use of the audio encoder is crucial to achieve a good performance. Ablation study results show that finetuning the audio encoder and using an adaptive f_{shift} based on the frame length give the best scores on MSR-VTT 9k.

can conclude that our method can deal well with both short-range and long-range video datasets.

4.3.2 Audio Feature Extraction

We do additional ablation study experiments on MSR-VTT 9k to evaluate the effects of different feature extraction methods for audio. The results are presented in Table 7. In our setup we finetune the audio encoder, since we have seen in the second row of Table 1 that finetuning the audio encoder in the text-to-audio retrieval setup actually helps for this task. We do the following ablation study experiments:
Freezing the AST: in this setup we freeze the AST model and only train the last linear layer that reduces the embedding dimension from 768 to 512. This setup is similar to ECLIPSE [22], since they use the embeddings from the pre-trained audio encoders. We report an improvement of 3.8% of R1 on finetuning vs freezing the AST model.
Fixing the frameshift: a dynamic frame shift has been used in our method to make sure that the Mel Filter Bank maintains a fixed length and will not lose relevant information in case of long videos. We measure the effect of such dynamic frame shift value by fixing it to 10 seconds. The AST model uses a fixed frameshift of 10 seconds for all datasets as well as ECLIPSE [22]. Taking a variable frameshift depending on the audio length improves the performance by 0.8%.

4.3.3 Variations on Multimodal Fusion

We compare our fusion method, addition, with four other fusion types that are visualized in Figure 4. Results are presented in Table 8, where $E_{V|T}$ stands for text-conditioned video embeddings (X-Pool) and $E_{A|T}$ stands for text-conditioned audio embeddings.
(A) Late fusion : In this setup, instead of using text-conditioned audio embeddings, we use the audio embeddings from the audio encoder directly and fuse them with the text-conditioned video embeddings by using addition.

$$E_{(V,A)|T} = E_{V|T} + E_A$$

This model shows a drop of 5.9% in R1 compared to our best fusion technique, which shows the importance of explicit alignment between the text and audio embeddings.
(B) Concatenation : Intuitive concatenation of the text-conditioned audio and text-conditioned video embeddings followed by a fully connected layer might help select the relevant components of each modality for each query. By concatenating the audio and video features, we get an overall embedding dimension of 1024, which is reduced to 512 with a fully connected layer to allow the computation of cosine similarity with each text embedding. But we noticed a large drop in performance of 6.3% in R1 using this method.

$$E_{(V,A)|T} = \mathrm{FC}([E_{V|T}, E_{A|T}])$$

We argue that the linear layer at this stage of the model is possibly causing too many additional parameters to learn an efficient embedding.
(C) Fusion by cross-attention : In this setup, we stack the text-conditioned audio and video embeddings and apply a third cross-attention block which takes the text embedding again as the query and the stacked audio-video embedding as key and value. This results in

$$E_{(V,A)|T} = E_{(V,A)|T} = \mathrm{XAttn}(E_T, [E_{V|T}, E_{A|T}])$$

The R1 score is 3.5% lower than for the best model, which indicates that a third cross-attention block in its current design is not able to capture importance of the audio and video modalities related to the text better than addition.
(D) Stacking audio and video : The cross-attention block in TEFAL learns the weight related to the importance of each frame to the text. In this experiment, instead of using all audio patch embeddings, we use the average of the CLS and DIST tokens from the DEiT model and use this in one cross-attention block together with the video frame embedding. We could see the audio embedding as an additional

Method	Fusion Type	Expression	R1 ↑	R5 ↑	R10 ↑
TEFAL	Addition (best model)	$E_{(V,A)\|T} = E_{V\|T} + E_{A\|T}$	**49.4**	**75.9**	**83.9**
TEFAL	Late Fusion (X-Pool + audio) (A)	$E_{(V,A)\|T} = E_{V\|T} + E_A$	43.5	70.4	80.9
TEFAL	Concatenation (B)	$E_{(V,A)\|T} = \mathrm{FC}([E_{V\|T}, E_{A\|T}])$	43.1	72.1	82.0
TEFAL	Fusion by XAttn (C)	$E_{(V,A)\|T} = \mathrm{XAttn}(E_T, [E_{V\|T}, E_{A\|T}])$	45.9	72.9	81.4
TEFAL	Stacking audio and video (D)	$E_{(V,A)\|T} = \mathrm{XAttn}(E_T, [E_V, E_A])$	46.1	71.8	81.8

Table 8. An evaluation of alternative fusion types for the audio and video embeddings. The visual comparisons of these fusion methods are illustrated in Figure 4.

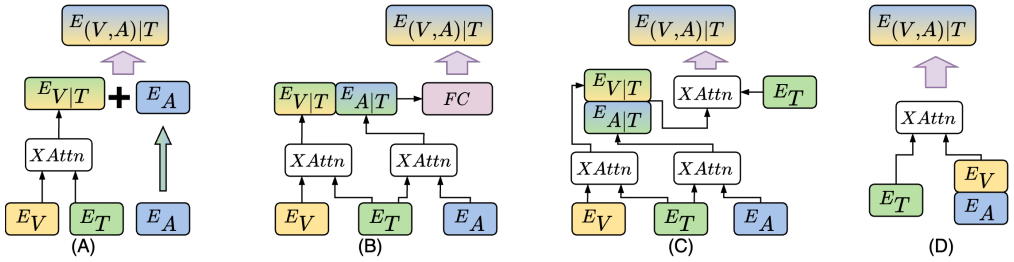


Figure 4. Different fusion methods corresponding to the results in Table 8 with (A) Late fusion, (B) fusion by concatenation in embedding dimension, (C) fusion by stacking the text-conditioned video and audio embeddings (D) fusion by stacking the video and audio embeddings.

frame embedding, by stacking the embeddings of the two modalities in the dimension of the number of the frames.

$$E_{(V,A)|T} = \mathrm{XAttn}(E_T, [E_V, E_A])$$

This leads to an R1 of 46.1%. Our intuition is that the initial embedding spaces of the audio and video frame embeddings are unaligned and therefore require two cross-attention blocks to align them with the text embedding.

From the experiments regarding the fusion of the audio and video embeddings, we can conclude that fusion is not trivial. In fact, alternative fusion methods show lower results than X-Pool, the method that only uses the same video branch as TEFAL and does not leverage audio. The proposed simple addition fusion obtains an R1 of 49.4%, which is the best among all the fusion methods and improves the result by 2.5% compared to the video-only branch.

4.3.4 Applicability at Scale

Our method computes aggregated video and audio embeddings that are both conditioned on text. Therefore, we cannot pre-compute the final audio and video embeddings offline, since the text queries are not known at this point. Similar to X-Pool, we can use TEFAL to re-rank the top $\mathcal{K}$ retrievals of an efficient method, i.e., mean-pooling the frame embeddings. Given $\mathcal{T}$ text queries, $\mathcal{V}$ videos and $\mathcal{A}$ audios, with $\mathcal{A} \leq \mathcal{V}$, instead of a complexity of $\mathcal{O}(\mathcal{TV})$ we get $\mathcal{O}(\mathcal{KT} + \mathcal{V})$, which is the same as X-Pool [16]. To empirically show that this does not result in a significant performance drop, we reduce the search space via approximate nearest neighbor method between the text embedding and the mean-pooled video embeddings, and then perform re-ranking in this reduced search space with TEFAL. Since all test sets have 1,000 to 2,000 videos, we also evaluate on the validation set of LSMDC, which has 7,408 videos, to show the effect of scaling up. In table 9 we notice a very small drop in performance by using TEFAL to re-rank the top 10% retrievals given by mean-pooling, albeit the latter model's low performance.

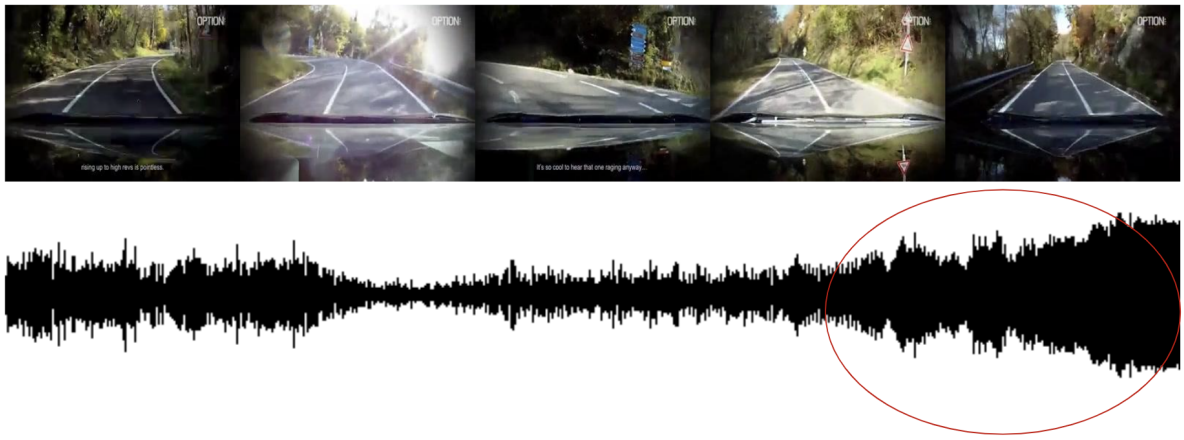
	mean-pool			TEFAL			TEFAL (re-rank)		
Dataset	R1 ↑	R5 ↑	R10 ↑	R1 ↑	R5 ↑	R10 ↑	R1 ↑	R5 ↑	R10 ↑
MSR-VTT-9k	41.7	69.7	78.3	49.4	75.9	83.9	49.4	75.8	83.8
LSMDC (test)	20.9	38.5	48.3	26.8	46.1	56.5	26.7	46.1	56.2
LSMDC (val)	7.8	18.9	25.3	10.2	23.6	31.1	10.2	23.6	31.3
VATEX	53.3	84.9	92.2	61.0	90.4	95.3	61.0	90.3	95.2
Charades	11.8	28.0	36.8	18.5	37.3	48.6	18.6	37.7	49.3

Table 9. Results on all datasets showing the effect of a reduced search (by 90%) space during inference.

4.3.5 Qualitative Results

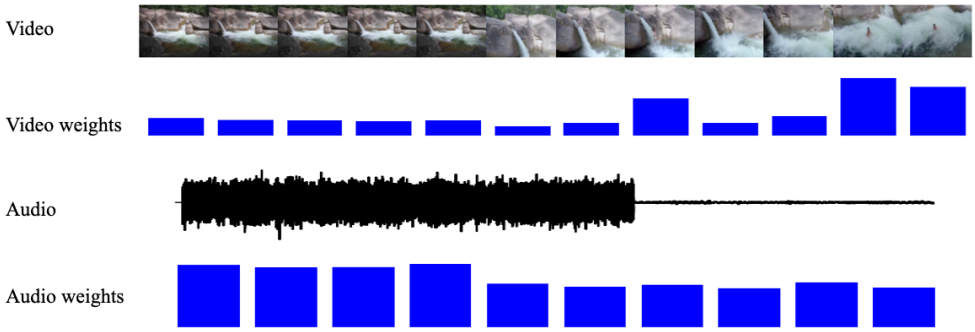
We present an example from the MSR-VTT dataset [41] to highlight how audio provides complementary information to the video to achieve improved text-queried retrieval. More examples are provided in the supplementary material. Additionally, these examples further justify our choice of using text features as the center feature to align video and audio features rather than aligning the audio and video modalities. Shown in Figure 5 is sample 9806 where TEFAL ranks the matched video as the top retrieval result with the help of audio modality, whereas, X-Pool (TEFAL w/o audio) [16] which only utilizes text and video fails to do so.

Rank #1 result in TEFAL, Rank #5 in TEFAL w/o audio



Query: person is driving his black car fast on the street

Figure 5. An example is shown of a description in the text that is not visible, namely the car accelerating, but is audible. Therefore the TEFAL model is able to perform much better on this (Rank 1) than the TEFAL w/o audio model (Rank 5).



Query: a person is swimming in some white water rapids.

Figure 6. This figure shows the weights of frames from the text-video attention block (upper two rows) and the weights of audio tokens aggregated over time from the text-audio block (lower two rows). Video and audio weights emphasize complementary parts of the video.

The query words of this sample are "person is driving his black car fast on the street". The TEFAL w/o audio model ranks the matched video at the fifth place while TEFAL ranks the matched video as the top retrieval. In this example, the keyword "fast" is crucial. However, driving fast is not visible in the frames alone but is cued by the sound of the accelerating engine (encircled in red in the waveform).

4.3.6 Visualization of the Attention Weights

We provide a qualitative example (sample 7152) to illustrate the activated video and audio regions by plotting the attention weights in the $E_{V|T}$ and $E_{A|T}$ blocks in Figure 6. We observe that the latter half of the video stream has higher activation than the first half, whereas, the first half of the audio has higher activation than the latter half, providing complementary information.

5. Conclusion

This paper introduces TEFAL, a novel text-conditioned feature alignment framework for audio-enhanced text-to-video retrieval. Our approach utilises two independent cross-modal attention blocks for the text to attend to the audio and video representations. We are the first to propose this approach for audio-enhanced text-to-video retrieval and explain why it is more suitable for this task than audiovisual alignment. Extensive experiments demonstrate that the text-conditioned feature alignment outperforms audiovisual alignment for audio-enhanced text-to-video retrieval. We attribute this success to our use of an independent cross-modality attention model that develops a representation conditioned on the text and independent of the video content. In the future, we plan to extend our method to other multimodal text-video-audio understanding tasks, such as video captioning and video question answering.

References

[1] Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. VATT: transformers for multimodal self-supervised learning from raw video, audio and text. In *NeurIPS*, 2021.

[2] Jean-Baptiste Alayrac, Adrià Recasens, Rosalia Schneider, Relja Arandjelovic, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. In *NeurIPS*, 2020.

[3] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021.

[4] Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *TPAMI*, 41(2), 2018.

[5] Khaled Bayoudh, Raja Knani, Fayçal Hamdaoui, and Abdellatif Mtibaa. A survey on deep multimodal learning for computer vision: advances, trends, applications, and datasets. *The Visual Computer*, 38(8), 2022.

[6] Simion-Vlad Bogolin, Ioana Croitoru, Hailin Jin, Yang Liu, and Samuel Albanie. Cross modal retrieval with querybank normalisation. In *CVPR*, 2022.

[7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *NeurIPS*, 2020.

[8] Shizhe Chen, Yida Zhao, Qin Jin, and Qi Wu. Fine-grained video-text retrieval with hierarchical graph reasoning. *CVPR*, 2020.

[9] Xingyi Cheng, Hezheng Lin, Xiangyu Wu, F. Yang, and Dong Shen. Improving video-text retrieval by multi-stream corpus alignment and dual softmax loss. *arXiv preprint arXiv:2109.04290*, 2021.

[10] Ioana Croitoru, Simion-Vlad Bogolin, Marius Leordeanu, Hailin Jin, Andrew Zisserman, Samuel Albanie, and Yang Liu. Teachtext: Crossmodal generalized distillation for text-video retrieval. In *ICCV*, 2021.

[11] Jianfeng Dong, Xirong Li, Chaoxi Xu, Xun Yang, Gang Yang, Xun Wang, and Meng Wang. Dual encoding for video retrieval by text. *TPAMI*, 44(8), 2021.

[12] Yuying Ge, Yixiao Ge, Xihui Liu, Dian Li, Ying Shan, Xiaohu Qie, and Ping Luo. Bridging video-text retrieval with multiple choice questions. In *CVPR*, 2022.

[13] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *ICASSP*, 2017.

[14] Simon Ging, Mohammadreza Zolfaghari, Hamed Pirsiavash, and Thomas Brox. Coot: Cooperative hierarchical transformer for video-text representation learning. *NeurIPS*, 2020.

[15] Yuan Gong, Yu-An Chung, and James Glass. Ast: Audio spectrogram transformer. In *Interspeech*, 2021.

[16] Satya Krishna Gorti, Noel Vouitsis, Junwei Ma, Keyvan Golestan, Maksims Volkovs, Animesh Garg, and Guangwei Yu. X-pool: Cross-modal language-video attention for text-video retrieval. In *CVPR*, 2022.

[17] Wangli Hao, Zhaoxiang Zhang, and He Guan. Integrating both visual and audio cues for enhanced video caption. In *AAAI*, 2018.

[18] Fan Hu, Aozhu Chen, Ziyue Wang, Fangming Zhou, Jianfeng Dong, and Xirong Li. Lightweight attentional feature fusion: A new baseline for text-to-video retrieval. In *ECCV*, 2022.

[19] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L. Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. In *CVPR*, 2021.

[20] Dongxu Li, Junnan Li, Hongdong Li, Juan Carlos Niebles, and Steven CH Hoi. Align and prompt: Video-and-language pre-training with entity prompts. In *CVPR*, 2022.

[21] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. Hero: Hierarchical encoder for video+ language omni-representation pre-training. In *EMNLP*, 2020.

[22] Yan-Bo Lin, Jie Lei, Mohit Bansal, and Gedas Bertasius. Eclipse: Efficient long-range video retrieval using sight and sound. In *ECCV*, 2022.

[23] Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. Use what you have: Video retrieval using representations from collaborative experts. In *BMVC*, 2019.

[24] Yu Liu, Huai Chen, Lianghua Huang, Di Chen, Bin Wang, Pan Pan, and Lisheng Wang. Animating images to transfer clip for video-text retrieval. In *SIGIR*, 2022.

[25] Yuqi Liu, Pengfei Xiong, Luhui Xu, Shengming Cao, and Qin Jin. Ts2-net: Token shift and selection transformer for text-video retrieval. In *ECCV*, 2022.

[26] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval. *Neurocomputing*, 508:293–304, 2022.

[27] Yiwei Ma, Guohai Xu, Xiaoshuai Sun, Ming Yan, Ji Zhang, and Rongrong Ji. X-clip: End-to-end multi-grained contrastive learning for video-text retrieval. In *ACM MM*, 2022.

[28] Antoine Miech, Ivan Laptev, and Josef Sivic. Learning a Text-Video Embedding from Imcomplete and Heterogeneous Data. *arXiv preprint arXiv:1804.02516*, 2018.

[29] 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. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[30] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[31] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[32] Anna Rohrbach, Marcus Rohrbach, Niket Tandon, and Bernt Schiele. A dataset for movie description. In *CVPR*, 2015.

[33] Nina Shvetsova, Brian Chen, Andrew Rouditchenko, Samuel Thomas, Brian Kingsbury, Rogério Feris, David Harwath, James R. Glass, and Hilde Kuehne. Everything at once - multi-modal fusion transformer for video retrieval. In *CVPR*, 2022.

[34] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *ECCV*, 2016.

[35] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021.

[36] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[37] Junke Wang, Dongdong Chen, Zuxuan Wu, Chong Luo, Luowei Zhou, Yucheng Zhao, Yujia Xie, Ce Liu, Yu-Gang Jiang, and Lu Yuan. Omnivl: One foundation model for image-language and video-language tasks. In *NeurIPS*, 2022.

[38] Jianren Wang, Zhaoyuan Fang, and Hang Zhao. Alignnet: A unifying approach to audio-visual alignment. In *WACV*, 2020.

[39] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *ICCV*, 2019.

[40] Xiaohan Wang, Linchao Zhu, and Yi Yang. T2vlad: global-local sequence alignment for text-video retrieval. In *CVPR*, 2021.

[41] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. MSR-VTT: A large video description dataset for bridging video and language. In *CVPR*, 2016.

[42] Peng Xu, Xiatian Zhu, and David A Clifton. Multimodal learning with transformers: a survey. *arXiv preprint arXiv:2206.06488*, 2022.

[43] Rowan Zellers, Jiasen Lu, Ximing Lu, Youngjae Yu, Yanpeng Zhao, Mohammadreza Salehi, Aditya Kusupati, Jack Hessel, Ali Farhadi, and Yejin Choi. MERLOT RESERVE: neural script knowledge through vision and language and sound. In *CVPR*, 2022.

[44] Bowen Zhang, Hexiang Hu, and Fei Sha. Cross-modal and hierarchical modeling of video and text. In *ECCV*, 2018.

[45] Shuai Zhao, Linchao Zhu, Xiaohan Wang, and Yi Yang. Centerclip: Token clustering for efficient text-video retrieval. In *SIGIR*, 2022.

[46] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, 2020.