Tencent-MVSE: A Large-Scale Benchmark Dataset for Multi-Modal Video Similarity Evaluation

Zhaoyang Zeng, Yongsheng Luo, Zhenhua Liu, Fengyun Rao, Dian Li, Weidong Guo, Zhen Wen
QQ Browser Lab, Tencent
{zhaoyanzeng,yongshenluo,edinliu,fengyunrao,goodli,weidongguo,zhenzwen}@tencent.com

Abstract

Multi-modal video similarity evaluation is important for video recommendation systems such as video de-duplication, relevance matching, ranking, and diversity control. However, there still lacks a benchmark dataset that can support supervised training and accurate evaluation. In this paper, we propose the Tencent-MVSE dataset, which is the first benchmark dataset for the multi-modal video similarity evaluation task. The Tencent-MVSE dataset contains video pairs similarity annotations, and diverse metadata including Chinese title, automatic speech recognition (ASR) text, as well as human-annotated categories/tags. We provide a simple baseline with a multi-modal Transformer architecture to perform supervised multi-modal video similarity evaluation. We also explore pre-training strategies to make use of the unpaired data. The whole dataset as well as our baseline will be released to promote the development of the multi-modal video similarity evaluation. The dataset has been released in https://tencent-mvse.github.io/.

1. Introduction

Recent years have witnessed the rapid development of online video-sharing platforms. More and more platforms such as YouTube, Youku, iQIYI, Tencent Video, and TikTok have emerged to become a crucial part of our daily life. To satisfy the diverse requirements of users, these platforms implement complicated video recommendation systems to perform diverse tasks, including video de-duplication, relevance matching, ranking, diversity control, etc. All of these applications rely on effective similarity evaluation algorithms that process a thorough understanding of video contents.

The "similarity" of video content is reflected in multiple modalities, including visual content and metadata. Figure 1 shows some examples of video pairs that might be "similar". For the first example, the two videos have similar visual contents about "playing football". Videos from the second pair have different visual contents, while they are both crosstalks acted by the same troupe according to their titles. For the third example, the two videos have different visual contents and titles, while they share "similar" visual and text information related to the same game. Since the similarity exists in such diverse manners, in real application scenarios, video similarity should be evaluated by considering multi-modal information. Inspired by the recent success in the field of natural language processing and computer vision, large-scale labeled datasets are mandatory to advance research progress. However, when creating a video similarity benchmark dataset, the multiple modalities bring significant challenges for data annotation and evaluation.

Figure 1. Some examples of similar video pairs. The video pairs in the three rows are similar in their visual contents, titles, and semantic information.

Learning video representations for similarity evaluation requires the supervision of video pairs similarity. Most existing approaches learn video representations via multi-label classification by using the semantic tags as supervi-

sion [1, 19, 33]. The tags summarize the videos from various semantic levels and perceptions, and thus can briefly estimate the similarities of video pairs. However, in real video recommendation systems, these semantic tags can not satisfy the higher precision requirements. CDML [23] and GCML [22] try to involve user behaviours to estimate the video pair similarity. Their idea is conceptually aligned with collaborative filtering, where many users implicitly collaborate to filter relevant items. However, user behavior relevance is affected by many factors, not only video content. What's worse, user behavior differs in different platforms. Moreover, there does not exist a video similarity evaluation benchmark in the research community. Such restrictions greatly limit the development of multi-modal video similarity evaluation.

In this paper, we propose a large-scale Tencent-MVSE dataset, which is the first benchmark dataset for the multi-modal video similarity evaluation task, to promote the development of multi-modal video similarity evaluation. We collect $135,705$ video pairs, and finely annotate their similarity scores. A detailed specification for the similarity annotation is provided to make sure that the annotated similarity score aligns with the human's perception. We provide videos as well as rich metadata including Chinese titles, automatic speech recognition (ASR) text, and human-annotated categories and tags to support the evaluation of the video similarity in a multi-modal manner. The annotated video pairs data is separated into a *pairwise* split, a *test-dev* split, and a *test-std* split for supervised training, validation, and final evaluation, respectively. In addition, we also collect a *pointwise* split, which contains 1 million individual videos with video frames and metadata. The collected *pointwise* split is to encourage researchers to explore advanced annotation-free approaches by leveraging more accessible unlabeled data. Compared with existing video understanding datasets with language annotations, the Tencent-MVSE dataset has two main characteristics. First, Tencent-MVSE regards video-text as a whole item, and annotates the similarity between items, while existing datasets [29, 39, 42] focus on exploring the relation between video and text. Second, Tencent-MVSE provides 328 categories and $64,903$ tags, which is much larger than existing datasets [1, 19, 21, 33]. All the categories and tags are manually annotated by humans to guarantee high quality. The Tencent-MVSE dataset has been validated in the competition of one of the leading international data mining conferences. It enabled hundreds of participants to implement innovative methods of measuring.

Except for the collection of the Tencent-MVSE dataset, we also provide a simple baseline for the multi-modal video similarity evaluation task. Inspired by the great success of vision-language understanding approaches such as UNITER [6], VL-BERT [34], SOHO [16] and VideoBERT [36], we adopt the advanced single-stream multi-modal Transformer (MMT) as the base model architecture. Taking the concatenation of sentence token embeddings and video frame features as input, the MMT learns joint video-text embeddings for the input video-text item by using the multi-modal attention mechanism. The annotated similarity scores are utilized as the supervision signal to optimize the embedding cosine distance between video pairs by mean squared error (MSE) loss. The joint video-text embeddings learned through this method have the rich discriminate ability, and thus can evaluate the video similarity much preciser. Additionally, inspired by the effective pre-training strategies of recent works [6, 10, 16, 34, 36], we attempt to leverage the *pointwise* split to perform multi-modal pre-training for MMT. We adopt widely-used masked language modeling (MLM), masked frame modeling (MFM), and video-text matching (VTM) pre-training tasks to pre-train the MMT. Our results show that all the pre-training strategies can boost the model performance by a large margin, which reveals the potential of the annotation-free data.

Summarily, the contributions of this paper include:

- We collect and annotate the Tencent-MVSE dataset, which is the first multi-modal video similarity evaluation benchmark in the research community;
- We build a simple baseline which adopts the advantages Transformer for multi-modal learning, and conducts sufficient ablation experiments to show the effectiveness of each module;
- We adopt the advanced multi-modal pre-training strategies to mine the potential of the MMT model. The experiment results demonstrate the effectiveness of the pre-training strategies on the multi-modal video similarity evaluation task.

2. Related Work

2.1. Video Understanding Datasets

The development of video understanding research should be credited with large-scale datasets. HMDB51 [21], UCF-101 [33], Sport1M [18] and Thumous [17] are early datasets that provide video-wise tags, and are all widely used video classification benchmarks. Kinetics [19] is a much larger dataset that contains over 300K videos clips and 400 categories. ActivityNet [4] provides segment-level action annotation, and enables intelligence to perform temporal action detection. The above datasets only focus on human action and sports scenarios, while the real-world videos have much richer semantics. YouTube-8M provides 8 million videos belonging to $4,800$ classes, and its great scale and diversity can support robust representation learning.

Later on, researchers find that if we want to perform more human-like video understanding, we need to bridge

the gap between video and language. To this end, several video datasets with language annotation are proposed. YouCook [8], MSR-VTT [42], VATEX [39] and STAR [41] are video datasets with human-written sentences annotations, which can support video captioning and video retrieval tasks. HowTo100M [29] is the largest video-text dataset, which consists of 1.3 million video clips with ASR text annotation. The authors show that the HowTo100M [29] can help learn robust video and text embeddings, and can greatly boost the performance on video retrieval tasks.

Although lots of videos understanding datasets are proposed, there is still no dataset specially designed for the video similarity evaluation task. Our proposed Tencent-MVSE dataset is the first video similarity evaluation benchmark dataset. With the video pairs similarity annotations, researchers can perform supervised training and accurate evaluation. Tencent-MVSE also provides rich metadata for supporting multi-modal and multi-task learning.

2.2. Vision-Language Pre-training

In these few years, a great number of works try to explore improving vision-language understanding by self-supervised pre-training, and achieve great success in a series of image-text tasks (*e.g.* VQA [3], VCR [43], NLVR [35], STAR [41], Image Retrieval [12]) and video-text tasks (*e.g.* Video QA [25], Video Captioning [42], Video Retrieval [20]). Among these, most works adopt the single-stream architecture to jointly learn the inter-modal and intra-modal relation between vision and language domains. UNITER [6] adopt vision BUTD feature [2] as input, proposes several pre-training tasks, and shows their effectiveness on several image-text downstream tasks. VL-BERT [34] attaches the whole BUTD feature extraction network to the multi-modality model, and makes the whole network trainable. SOHO [16] breaks out the limitation of bounding box annotation, uses a simple CNN backbone to produce grid feature, and shows that the end-to-end training of the whole network can produce great results.

Except for image-related tasks, VideoBERT [36] studies the pre-training on video-text tasks, proposes to use discrete tokens to represent video frames by clustering, and applies mask-predict strategy on video features. HERO [25] proposes video-subtitle matching and frame order modeling pre-training strategies to capture the temporal alignment between multiple modalities. CLIPBERT [24] explores the end-to-end training strategy for video-text pre-training and demonstrates that even using less clips can perform better.

In this paper, we follow the widely used self-supervised pre-training approaches to pre-train on the Tencent-MVSE *pointwise* split, and find that it can bring significant improvement to the multi-modal video similarity evaluation task. Details of the pre-training will be explained in Sec 4.2.

3. The Tencent-MVSE Dataset

To promote the development of multi-modal video similarity evaluation research and application, we build the Tencent-MVSE dataset, which is the first benchmark for the multi-modal video similarity evaluation task. The Tencent-MVSE dataset provides similarity scores for video pairs, as well as rich metadata, including Chinese titles, ASR text, and human-annotated categories and tags. In this section, we will introduce how we build this dataset in detail.

3.1. Data Collection

We collect the video data from Tencent Kandian[1], which is a large-scale feeds recommendation platform in China. Tencent Kandian receives hundreds of thousands of PGC (Professionally-generated Content) short videos and more other videos every day. We only select the PGC videos since their qualities are much better. We first fetch 1 million short videos on the Tencent Kandian service to construct the *pointwise* split. Such an amount of videos can ensure that the type distribution is consistent with the online system. Among the fetched videos, short videos less than 60 seconds are mainly selected, since they are popular on mobile devices. We then fetch another 2 million short videos as a gallery for pairwise annotation and finally annotate $135,705$ video pairs. The annotation details will be explained in 3.2. The annotated pairs are separated into a *pairwise* split, a *test-dev* split, and a *test-std* for supervised training, validation, and final evaluation, respectively. We ensure there is no video overlapping among the training and testing splits to avoid data leaks, and all splits have consistent data distributions.

For each video, we provide rich information, including Chinese title, ASR text, pre-extracted frame features, and human-annotated categories and tags. The title is written by the author of the video, and the ASR text is generated using Tencent Cloud ASR API[2] based on the audio. We exclude categories and tags in the *test-dev* split and the *test-std* split since we do not want to introduce any manual annotation in the testing stage. Limited by the copyright of the original data, we only provide video id[3] as the link to the corresponding raw videos.

3.2. Data Annotation

3.2.1 Video Category and Tag Annotation

We provide both category and tag annotation for each training video. The categories summarize the type of videos, and the tags indicate the concepts of video contents. We

[1]https://kandian.qq.com/

[2]https://cloud.tencent.com/document/api/1093/35636

[3]The video id can be used to build the URL with the pattern https://kandianshare.html5.qq.com/v3/video/{id}

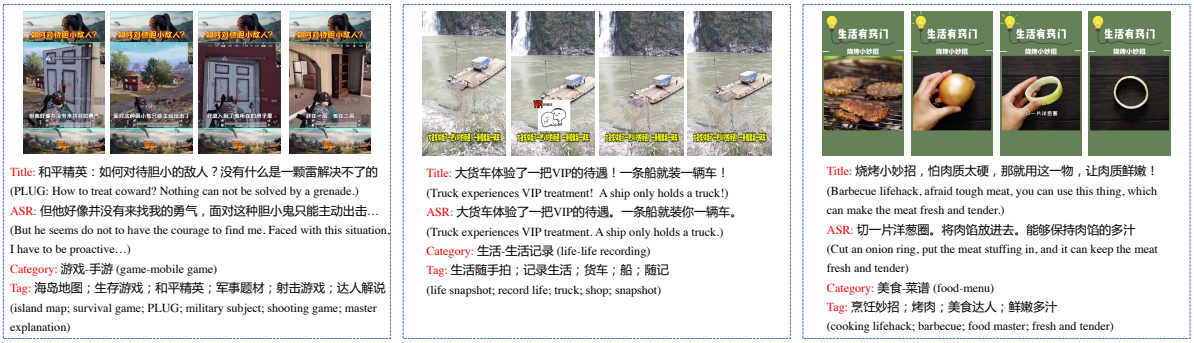

Figure 2. Some examples of the Tencent-MVSE dataset. All videos in the dataset contain video frames, Chinese title, ASR text, and several semantic tags. The pairwise data are annotated with similarity scores.

similarity degree	annotation specifications	examples
strongly similar (1.0)	the subjects are consistent and the core elements (such as the IP, characters, actions, scenes) are the same	both are the same movie/series/shows, similar scenario or the same actor
		both are live video streaming, similar shows or the same streamer
		both are beauty makeup videos, similar makeup or the same streamer
		both are sport videos, the same kind of sport and the same country
		both are cooking videos, similar foods or the same chef
weakly similar (0.5)	the subjects are consistent, and the core elements are slightly different	both are the same movie/series/shows, different subject or actor or role
		both are live video streaming, relevant content but different streamers
		both are beauty makeup videos, different streamers and diverse makeup
		both are sport videos, the same kind of sport but different countries or matches
		both are cooking videos, different foods and chefs
not similar (0.0)	the subjects are not consistent, or the subjects are consistent but the core elements are greatly different	both are movie/series/shows, different subjects and no common actor
		both are sport videos, different types of sport
		both are game videos, different types of game
		different places and people

Table 1. A simplified annotation specification for multi-modal video similarity

totally define 328 categories, which can be further classified into 29 super-categories, and 64,903 tags. We build the categories and tags vocabularies by first mining from user searching queries and knowledge graphs behind a large-scale video-sharing platform, then verifying by humans. Each video belongs to exactly one category and may have one or several tags. Annotators are hired to manually label the videos, each of which is asked to select the categories and tags after watching the videos (including the video titles and the videos themselves). Figure 2 shows some samples of the annotated videos.

3.2.2 Multi-modal Video Similarity Annotation

Multi-modal video similarity can measure the semantic similarity between the contents of two videos, which requires ground-truth similarity for each video pair. However, it is difficult for human beings to accurately decide the similarity score. Inspired by the semantic textual similarity (STS) task [5], we define three similarity degrees and design detailed annotation specifications. For each video pair, we invite ten annotators to select the similarity degrees based on the specification after watching both videos and titles.

A simplified version is listed in Table 1. We define three similarity degrees, including "strongly similar", "weakly similar" and "not similar", whose similarity scores are 1.0, 0.5, 0.0, respectively. For each video pair, we consider the average of all annotated scores as its final similarity score.

In the real world, most video pairs are categorized as "not similar", forming a crucially long-tailed classification. To create a benchmark, however, we should maintain a relatively balanced class distribution. To this end, we select the candidate video pairs according to the following procedures. First, We train three video embedding models for video pairs selection. The three models are all trained by multi-label classification tasks supervised by tags following [1]. The three models take video, title, and video+title as input, respectively, so that samples can be summarized from different perceptions, which grants more diversity to the candidate video pairs. Then, we randomly sample query videos in the gallery with 2 million videos. For each embedding model above, we retrieve the top 200 similar candidate videos based on their cosine distances, and randomly sample three videos from the top 50, 50-100, and 100-200 results, respectively. According to our observation, very few similar samples exist in the top 100-200 list. The three sampling ranges can roughly denote the three respective similarity degrees, and result in candidate video pairs in relatively balanced distribution. Given the nine videos from three models, the query videos as well as their retrieved videos

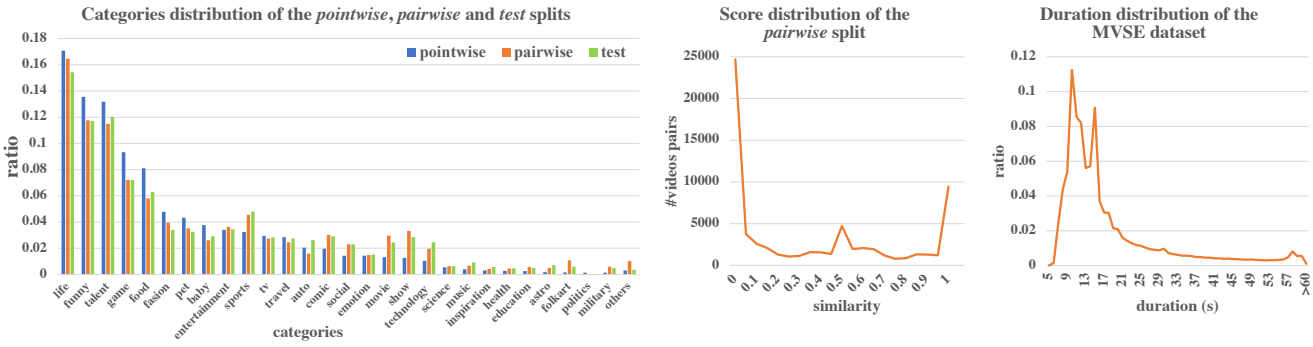

Figure 3. The data analysis of our proposed Tencent-MVSE Dataset.

Dataset	#Videos	#Clips	Duration(hs)	#Cats	#Tags	#Text	Text Type	Source
MSR-VTT [42]	7.2K	10K	40	257		200K	Caption	YouTube
YouCook II [45]	2K	14K	176	89		14K	Caption	YouTube
ActivityNet Captions [19]	20K	100K	849	200		100K	Dense Caption	YouTube
TGIF [26]	102K	126K	103	-		126K	Caption	Tumblr
LSMDC [31]	200	128K	150	-		128K	Movie Description	Movies
How2 [32]	13.2K	185K	298	-		185K	Subtitle	YouTube
VATEX [39]	41.3K	41.3K	115	600		825K	English & Chinese Caption	YouTube
HowTo100M [29]	1.2M	136M	134K	-		136M	ASR text	YouTube
YouTube-8M [1]	8.3M	8.3M	500K	4,800		-	-	YouTube
Tencent-MVSE	1.1M	1.1M	5,805	328	64,903	2.3M	Chinese Title & ASR text	Kandian

Table 2. Data analysis and comparison between Tencent-MVSE and other video understanding datasets. Tencent-MVSE provides the largest scale of human-annotated categories and tags, and author-written titles.

are randomly selected for annotation.

After filtering out the low-quality annotations that have variance greater than 0.25, we finally obtain 135,705 annotated video pairs. Here the 0.25 threshold is decided by the variance of the array that has five 1.0 scores and five 0.5 scores. We conduct 10-folds cross-validation on the annotations where the current annotation is considered as the prediction and the average of the others is regarded as ground truth. The 10-folds average Spearman's Rank Correlation is 0.9096, which could be recognized as a human score. This justifies that annotations among different annotators have a strong correlation, and thus are reliable.

3.3. Data Statistic

We split the annotated video pairs into *pairwise*, *test-dev* and *test-std* splits. The *pairwise* split contains 63,613 videos and 67,854 video pairs, which is used for training. The *test-dev* split contains 31,514 videos and 27,161 video pairs, which is used for validation. The *test-std* split contains 43,027 videos and 40,726 video pairs, which is used for evaluation. The *test-dev* spilt and the *test-std* split have 10,581 same videos, and all videos in the *pairwise* do not appear in the testing splits.

Figure 3 shows the categories, score and duration distribution of the Tencent-MVSE dataset. We provide 328 categories and 64,903 tags. The 328 categories belong to 29 super-categories, following the distribution illustrated in Figure 3(a). The category distribution of the three splits is consistent and can reflect the real distribution of the online system. From the annotation similarity distribution shown in Figure 3(b), we find that except for the video pairs that have 1.0 or 0.0 similarity scores, the score distribution is relatively balanced. The duration of Tencent-MVSE dataset is 5,805 hours, where 90% of the videos possess duration ranging from 7-35 seconds, as shown in Figure 3(c).

Table 2 shows the the statistic of Tencent-MVSE along with other video datasets. The tag system of the Tencent-MVSE dataset is the largest in the research community. Compared with YouTube-8M whose tags are generated by the YouTube video annotation system, our annotated categories and tags are based on manual annotation followed by a sophisticated processing procedure, which is, therefore, more representative and reliable.

3.4. Data Pre-processing

We extract the video frames in 1 FPS. We adopt three kinds of typical models to extract the video frame features. The first model is ResNet-50 [15], which is a classical image classification model trained on ImageNet dataset [9]. We follow the standard strategy to pre-process the frames by resizing the short edges to 256 pixels, and then cropping the center 224×224 square region. The second model

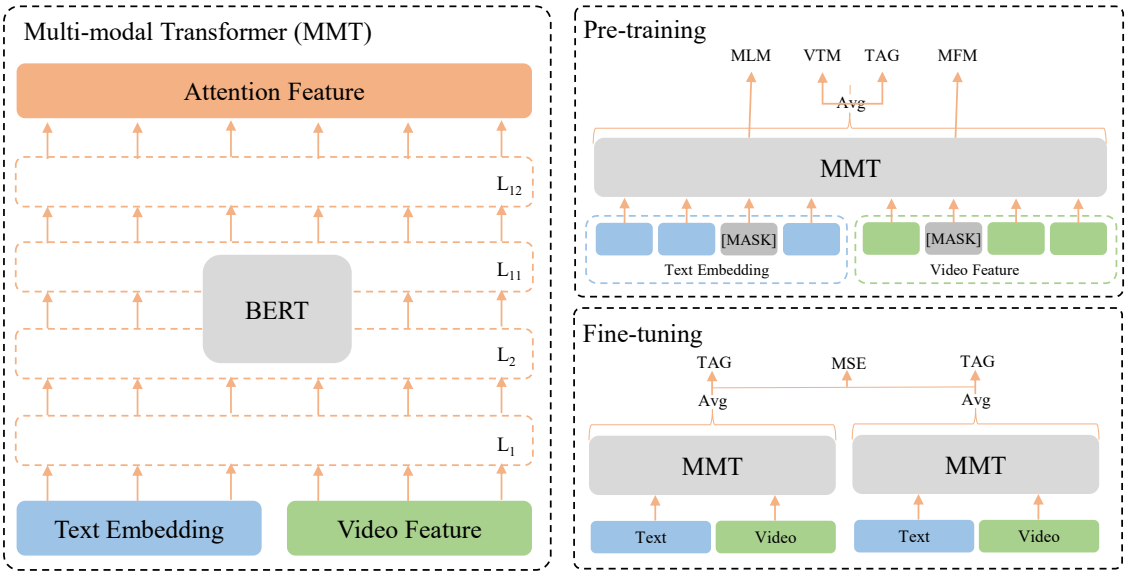

Figure 4. The overview of our proposed MMT framework. The left part shows the MMT model architecture, which takes the concatenation of text and video features as input, and output the multi-modal attention features. The right part shows the overview pipeline of the pre-training and the fine-tuning.

we use is EfficientNet-B3 [37], which has stronger performance than ResNet-50 on the ImageNet dataset [9]. For EfficientNet-B3, we resize the short edges of input frames to 300, and crop the center 300×300 regions. The third network we adopt is CLIP [30]. CLIP is a large-scale pre-training model under the supervision of natural language, which can bridge the gap between vision and language domains. It adopt ViT [11] as the image backbone. We pre-process the input frames by resizing their short edges to 256, then cropping the center 256×256 regions. The specific CLIP model we use is ViT-B/32.

The effectiveness of these three kinds of features is presented in Sec 5.3. The comparison between their performance illustrates the representation ability of classification features affects the final multi-modal understanding results. We do not adopt the models that are specifically designed for video understanding tasks (*e.g.* SlowFast [13], S3D [44]) because of the inconsistent of sampling FPS. All the features are also released to researches to reduce the time costing of fetching data.

4. Baselines

We propose a simple multi-modality Transformer (MMT) for joint video-text embedding learning. The overview framework of MMT is shown in Figure 4.

4.1. Model Architecture

The MMT takes as input the video frame features and text tokens. Given a video feature sequence, we use a fully-connected layer to project the features into a commonly hidden space with dimension d, followed by a LayerNorm layer. For text input, we follow the pre-processing strategy of BERT [10] to first use word-piece to tokenize the sentence and then use an embedding layer to embed the token sequence into d dimension.The text feature is then concatenated with the video feature according to the sequence length. We add a [CLS] token to the start of the sequence and add a [SEP] token to indicate the end of the sentence.

We use a 12-layer Transformer [38] as a multi-modal encoder, whose parameters are inherited from public accessible pre-trained models. The average pooling of the attention features is subsequently encoded by a linear layer into the target embedding dimension. In this paper, we set the target embedding dimension to 256 in all our experiments. Such feature output of the linear layer is considered as the joint video-text embedding.

4.2. Pre-training

We adopt three pre-training tasks to make use of the large-scale *pointwise* split, including masked language modeling (MLM), video text matching (VTM), and masked frame modeling (MFM). For the MLM task, we follow BERT [10] to randomly mask the text. Each token has a 15% probability of being masked. If a token is masked, it has an 80% probability of being replaced by a [MASK] token, 10% probability of being replaced by another randomly token, and 10% probability of being kept. Given the original tokens as ground truth labels, the model is required to predict the masked tokens in a self-supervised manner.

For the VTM task, we consider the input video and text as positive pairs. For each video, we randomly sample a text from another video to construct a negative pair. The ra-

tio of positive pairs and negative pairs is set as 1:1 in each batch. We take the average pooling of the attention features to perform a 2-way classification to predict whether the input video-text pair is positive or negative.

For the MFM task, we randomly mask the video frames features by [MASK] token embeddings. We gather the attention features of masked frames and feed them into a linear layer to project them to the same dimension of the input frame features. We follow [25] to adopt noise contrastive estimation (NCE) loss by considering the original frames features as ground truths, and other frame features inside the same batch as negative distractors.

Except for the three self-supervised pre-training tasks, we also take the average pooling of the attention features to perform category and tag classification. We transfer the annotated categories and tags into one-hot vectors for supervision and adopt "pem cls" [27] loss function since it can better handle the long-tail label distribution compared with typically used cross-entropy loss and is shown effective in [14]. We denote such classification tasks as TAG in the rest of this paper. We experimentally set the loss weight of tag and category classification to 1.0 and 0.1, respectively. Besides, since the tags and categories are also provided by the *pairwise* split, we also adopt TAG in the fine-tuning stage and find that it can benefit the final results.

When performing pre-training, we use the pre-trained BERT parameters to initialize the word embedding, Transformer, and the MLM prediction layer. And when conducting fine-tuning, we use the pre-trained parameters to initialize the whole network except the final projection layer.

5. Experiments

5.1. Implementations

We perform pre-training on the *pointwise* split, use the *pairwise* split for supervised fine-tuning, and *test-dev* and *test-std* splits for evaluation. For pre-training, we perform the experiments on 8 NVIDIA A100 GPUs and set the batch size to 32 videos per GPU. We use AdamW optimizer since it has been proven effective for Transformer-based models. We pre-train the models for 20 epochs and set the initial learning rate to 5e-5. We adopt the linear learning rate decay strategy with 2 warmup epochs. When fine-tuning on the *pairwise* split, we train the model on 2 NVIDIA A100 GPUs. We fine-tune the model for 10 epochs with 1 warmup epoch. The other hyper-parameters are kept the same as pre-training. The pre-training progress costs about 3 hours, and the fine-tuning progress costs about 40 minutes.

For the input videos, we extract the video features by the method described in Sec 3.4. We limit the maximum video frame length to 32. If the video frame length is greater than 32, we will select the first 32 frames. We limit the maximum text length of titles and ASR text to 32 and 128, respectively.

5.2. Evaluation Metric

The core insight of video embedding is to serve the recommendation, ranking, matching tasks. These tasks can all be viewed as ranking problems, which are only sensitive to the relative similarity scores. Therefore, we follow [5] to adopt Spearman's Rank Correlation as the evaluation metric. The computation of Spearman's Rank Correlation is

$$r_s = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}, \tag{1}$$

where d_i indicates the rank difference between the predicted rank and the original rank for each observation, and n is the number of observations. For simplification, we consider the cosine distance between given video pairs in embedding space as the similarity score and compute the Spearman's Rank Correlation score of the aforementioned similarity score for the final evaluation.

5.3. Ablation Studies

5.3.1 Modality Selection

Our baseline model takes three kinds of modalities as input, which are video features, title, and ASR text. We evaluate the effectiveness of each modality and its combinations. In this ablation study, we adopt the settings of using EfficientNet-B3 [37] features and BERT [10] architecture. The experiment results are reported in Table 3. We mainly compare the performance on the *test-dev* split. We first adopt a single modality as input, and we observe that the video-only model achieves the highest score with 0.6046, the title-only model achieves the second-highest score with 0.5696, while the ASR-only model presents inferior performance. Such single modality ablation results show that visual information plays the most important role in the video similarity evaluation task, which is also consistent with humans' perception. Then we evaluate "video+title" and "video+ASR" models, and discover that both title and ASR information can boost the performance. We combine all three modalities and notice that such a model can achieve a 0.7561 score, which outperforms all previous results. Later, we try to leverage tags and categories for multi-task training and find that the tags can boost the performance to 0.7778, and categories can further boost the performance to 0.7825. Such ablation studies show that all the information provided can contribute to the multi-modal video similarity task.

5.3.2 Video Feature Selection

The representation ability of video features also may affect the final performance. We try three feature extractors, including ResNet-50 [15], EfficientNet-B3 [37], and CLIP [30]. Table 4 demonstrates the baseline performance of these visual features. Empirical results justify that the representation ability of video features will affect the performance greatly. Among the three kinds of features, the

Video	Title	ASR	Tag	Category	test-dev	test-std
✓					0.6046	0.6014
	✓				0.5696	0.5577
		✓			0.1989	0.1940
✓	✓				0.7539	0.7525
✓		✓			0.5816	0.5724
✓	✓	✓			0.7561	0.7512
✓	✓	✓	✓		0.7778	0.7734
✓	✓	✓	✓	✓	**0.7825**	**0.7787**

Table 3. Ablation studies on multiple modalities.

Visual feature	Transformer initialization	test-dev	test-std
R50 [15]	BERT [10]	0.7496	0.7442
EFN-B3 [37]		0.7825	0.7787
CLIP [30]		0.8014	0.8004
R50 [15]	RoBERTa [28]	0.7480	0.7403
EFN-B3 [37]		0.7849	0.7805
CLIP [30]		0.8003	0.8006
R50 [15]	MacBERT [7]	0.7441	0.7399
EFN-B3 [37]		0.7840	0.7776
CLIP [30]		0.8017	0.8006

Table 4. Ablation studies on visual features and Transformer initialization.

CLIP features outperform the EfficientNet-B3 features and the ResNet-50 features by 0.02 and 0.05 on both *test-dev* and *test-std* splits, respectively.

5.3.3 Transformer Initialization

We study three variants of Transformer initialization, including the original BERT [10], the RoBERTa [28], and the MacBERT [7]. These three models vary in their pre-training strategies. We select these three initialization models because there are already open-source implementation and pre-trained model [40], which can greatly reduce our experiment cost. From Table 4 we can find that the three initialization models achieve comparable results.

5.3.4 Pre-Training

Many vision-language works [6, 10, 16, 24, 25, 36] show the effectiveness of pre-training strategies, and have already proposed several novel pre-training tasks. In this ablation study, we investigate three widely used self-pretraining tasks, including masked language modeling (MLM), video text matching (VTM), masked frame modeling (MFM). In addition, we also integrate category and tag classification tasks by using the annotated categories and tags from the *pointwise* split as supervision. We adopt CLIP [30] as the video feature extractor, and BERT [10] as the Transformer architecture. We simply adopt the same loss weights for all the pre-training tasks following [16]. In the fine-tuning stage, we use the best setting in Sec. 5.3.1.

We first apply MLM for pre-training since it is proved effective in many vision-language works [6, 16, 34]. From the first two lines of Table 5 we can find that pre-training with MLM can improve the final downstream performance from 0.8014 to 0.8164. Next, since in previous study we find that tags and categories classification tasks can bring great improvement, we then adopt TAG. From Table 5 we find that the TAG task can boost the final performance to 0.8268 on *test-dev*. Then we conduct the VTM task. From Table 5 we can find that the adding VTM task also boosts the performance to 0.8276. Finally, we evaluate the MFM task. We find that the MFM task can also improve the final performance to 0.8289 on the *test-dev* split.

Pre-training Tasks	test-dev	test-std
-	0.8014	0.8004
MLM	0.8164	0.8168
TAG + MLM	0.8268	0.8246
TAG + MLM + VTM	0.8276	0.8261
TAG + MLM + VTM + MFM	**0.8289**	**0.8250**

Table 5. Ablation studies on pre-training.

Pre-training	Fine-tuning	test-dev	test-std
MLM+VTM+MFM	-	0.8119	0.8089
MLM+VTM+MFM	TAG	0.8196	0.8167
TAG+MLM+VTM+MFM	-	0.8193	0.8153
TAG+MLM+VTM+MFM	TAG	0.8289	0.8250

Table 6. Ablation studies on TAG in pre-training and fine-tuning.

5.3.5 TAG

The categories and tags can provide rich semantic information for multi-modal video similarity evaluation. However, they require a large labeling cost and thus maybe not be easy to achieve in similar application scenarios. We conduct an ablation experiment to study the effectiveness of TAG in both the pre-training and the fine-tuning stages under the best setting of previous studies. The ablation experiments on TAG can be found in Table 6. We find that the TAG can bring about 0.007 and 0.008 performance gain on the *test-dev* split when integrating with the pre-training and the fine-tuning stages, respectively. When incorporated with other pre-training tasks, the contribution of TAG is weakened. We encourage researchers to explore approaches that can be free from categories and tags.

6. Conclusion

We introduce the Tencent-MVSE dataset, which is the first large-scale benchmark dataset for the multi-modal video similarity evaluation task. The annotated video similarity scores can help evaluate the similarity of video pairs, and thus benefit the video recommendation system in many real application scenarios. We involve an advanced Transformer to construct a multi-modal understanding baseline, and also explore several self-supervised pre-training strategies to improve the Tencent-MVSE performance. We hope that such a benchmark can attract more researchers to study it, and make more improvements to video understanding.

References

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016. 2, 4, 5

[2] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, pages 6077–6086, 2018. 3

[3] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *ICCV*, pages 2425–2433, 2015. 3

[4] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, pages 961–970, 2015. 2

[5] Daniel Cer, Mona Diab, Eneko Agirre, Inigo Lopez-Gazpio, and Lucia Specia. Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. *arXiv preprint arXiv:1708.00055*, 2017. 4, 7

[6] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *ECCV*, pages 104–120. Springer, 2020. 2, 3, 8

[7] Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. Revisiting pre-trained models for chinese natural language processing. *arXiv preprint arXiv:2004.13922*, 2020. 8

[8] Pradipto Das, Chenliang Xu, Richard F Doell, and Jason J Corso. A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching. In *CVPR*, pages 2634–2641, 2013. 3

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. Ieee, 2009. 5, 6

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186, 2019. 2, 6, 7, 8

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020. 6

[12] Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: Improving visual-semantic embeddings with hard negatives. *arXiv preprint arXiv:1707.05612*, 2017. 3

[13] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, pages 6202–6211, 2019. 6

[14] Daya Guo and Zhaoyang Zeng. Multi-modal representation learning for video advertisement content structuring. In *ACM Multimedia*, pages 4770–4774, 2021. 7

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 5, 7, 8

[16] Zhicheng Huang, Zhaoyang Zeng, Yupan Huang, Bei Liu, Dongmei Fu, and Jianlong Fu. Seeing out of the box: End-to-end pre-training for vision-language representation learning. In *CVPR*, pages 12976–12985, 2021. 2, 3, 8

[17] Haroon Idrees, Amir R Zamir, Yu-Gang Jiang, Alex Gorban, Ivan Laptev, Rahul Sukthankar, and Mubarak Shah. The thumos challenge on action recognition for videos "in the wild". *Computer Vision and Image Understanding*, 155:1–23, 2017. 2

[18] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, pages 1725–1732, 2014. 2

[19] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 2, 5

[20] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, pages 706–715, 2017. 3

[21] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *ICCV*, pages 2556–2563. IEEE, 2011. 2

[22] Hyodong Lee, Joonseok Lee, Joe Yue-Hei Ng, and Paul Natsev. Large scale video representation learning via relational graph clustering. In *CVPR*, pages 6807–6816, 2020. 2

[23] Joonseok Lee, Sami Abu-El-Haija, Balakrishnan Varadarajan, and Apostol Natsev. Collaborative deep metric learning for video understanding. In *SIGKDD*, pages 481–490, 2018. 2

[24] 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*, pages 7331–7341, 2021. 3, 8

[25] 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*, pages 2046–2065, 2020. 3, 7, 8

[26] Yuncheng Li, Yale Song, Liangliang Cao, Joel Tetreault, Larry Goldberg, Alejandro Jaimes, and Jiebo Luo. Tgif: A new dataset and benchmark on animated gif description. In *CVPR*, pages 4641–4650, 2016. 5

[27] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. In *ICCV*, pages 3889–3898, 2019. 7

[28] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 8

[29] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic.

Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, pages 2630–2640, 2019. 2, 3, 5

[30] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021. 6, 7, 8

[31] Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Christopher Pal, Hugo Larochelle, Aaron Courville, and Bernt Schiele. Movie description. *IJCV*, 123(1):94–120, 2017. 5

[32] Ramon Sanabria, Ozan Caglayan, Shruti Palaskar, Desmond Elliott, Loïc Barrault, Lucia Specia, and Florian Metze. How2: a large-scale dataset for multimodal language understanding. *arXiv preprint arXiv:1811.00347*, 2018. 5

[33] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 2

[34] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. In *ICLR*, 2019. 2, 3, 8

[35] Alane Suhr, Mike Lewis, James Yeh, and Yoav Artzi. A corpus of natural language for visual reasoning. In *ACL*, pages 217–223, 2017. 3

[36] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, pages 7464–7473, 2019. 2, 3, 8

[37] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*, pages 6105–6114. PMLR, 2019. 6, 7, 8

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017. 6

[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*, pages 4581–4591, 2019. 2, 3, 5

[40] Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. Transformers: State-of-the-art natural language processing. In *EMNLP*, pages 38–45, 2020. 8

[41] Bo Wu, Shoubin Yu, Tenenbaum Joshua B Chen, Zhenfang, and Chuang Gan. Star: A benchmark for situated reasoning in real-world videos. In *NeurIPS*, 2021. 3

[42] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, pages 5288–5296, 2016. 2, 3, 5

[43] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *CVPR*, pages 6720–6731, 2019. 3

[44] Da Zhang, Xiyang Dai, Xin Wang, and Yuan-Fang Wang. S3d: single shot multi-span detector via fully 3d convolutional networks. *arXiv preprint arXiv:1807.08069*, 2018. 6

[45] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, 2018. 5