CLOP: Video-and-Language Pre-Training with Knowledge Regularizations

Guohao Li  
liguohao@baidu.com  
Baidu Inc.  
Beijing, China

Hu Yang  
yanghu03@baidu.com  
Baidu Inc.  
Beijing, China

Feng He  
hfeng07@baidu.com  
Baidu Inc.  
Beijing, China

Zhifan Feng  
fengzhifan@baidu.com  
Baidu Inc.  
Beijing, China

Yajuan Lyu  
ivyajuan@baidu.com  
Baidu Inc.  
Beijing, China

Hua Wu  
wu_hua@baidu.com  
Baidu Inc.  
Beijing, China

Haifeng Wang  
wanghaifeng@baidu.com  
Baidu Inc.  
Beijing, China

ABSTRACT

Video-and-language pre-training has shown promising results for learning generalizable representations. Most existing approaches usually model video and text in an implicit manner, without considering explicit structural representations of the multi-modal content. We denote such form of representations as "structural knowledge", which express rich semantics of multiple granularities. There are related works that propose object-aware approaches to inject similar knowledge as inputs. However, the existing methods usually fail to effectively utilize such knowledge as "regularizations" to shape a superior cross-modal representation space. To this end, we propose a Cross-modal Knowledge-enhanced Pre-training (CLOP) method with Knowledge Regularizations. There are two key designs of ours: 1) a simple yet effective Structural Knowledge Prediction (SKP) task to pull together the latent representations of similar videos; and 2) a novel Knowledge-guided sampling approach for Contrastive Learning (KCL) to push apart cross-modal hard negative samples. We evaluate our method on four text-video retrieval tasks and one multi-choice QA task. The experiments show clear improvements, outperforming prior works by a substantial margin. Besides, we provide ablations and insights of how our methods affect the latent representation space, demonstrating the value of incorporating knowledge regularizations into video-and-language pre-training.

KEYWORDS

video-and-language pre-training, knowledge, contrastive learning

ACM Reference Format:

Guohao Li, Hu Yang, Feng He, Zhifan Feng, Yajuan Lyu, Hua Wu, and Haifeng Wang. 2022. CLOP: Video-and-Language Pre-Training with Knowledge Regularizations. In Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisboa, Portugal. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3503161.3548346

1 INTRODUCTION

Learning generalizable visual-and-language representations requires a comprehensive understanding of visual and textual inputs, as well as the semantic correlations between these modalities. Recently, the research community has witnessed a burst of progress on image-text pre-training [8, 22, 25, 26, 33, 40, 46, 59], but pre-training for video-text is still in its infancy.

The dominant paradigm of video-text pre-training is to train powerful encoders (for video and text) with several self-supervised proxy tasks [3, 23, 34, 43]. The canonical proxy tasks include “conditioned masked modeling” (predicting masked inputs from multi-modal context), “video-text contrastive learning” (discriminating positive samples from negatives), etc. Essentially, the ultimate goal of this paradigm is to learn a scalable cross-modal representation space, where the video-text representations should reside properly and can be easily generalized to downstream tasks.

Most existing approaches usually model video and text in an implicit manner, without considering explicit structural representations of the multi-modal content. We denote such form of representations as "structural knowledge", which express rich semantics of multiple granularities. There are related works that incorporate such knowledge as inputs or masking constraints. For example, the object-level regional features [1] are widely used in image-text pre-training [8, 25, 27, 33, 59]. The Ernie-Vil [59] incorporates knowledge by masking the semantical units in the text. Several video-text pre-training works propose object-aware approaches to model visual and textual inputs [50, 62]. However, the methods...
usually fail to utilize such knowledge as “regularizations” to effectively regularize the learning process, meanwhile do not reveal clear evidence towards a superior representation space. The lack of knowledge regularizations may limit the potential for better representations.

To this end, we propose incorporating the structural knowledge to regularize the cross-modal representation learning from two aspects: (1) **Pull together similar videos with knowledge objective.** We regularize the video representation learning through a structural knowledge prediction objective, which essentially attracts videos of similar semantics together in the representation space. (2) **Push apart cross-modal hard negatives with knowledge guidance.** We augment the cross-modal contrastive objective with a knowledge-guided hard negative sampler, so that the model can efficiently learn desired semantics with subtle variations.

Specifically, we propose a Cross-modal knOwledge-enhanced Pre-training (CLOP) method with Knowledge Regularization, consisting of two key components as follows:

1. A simple yet effective **Structural Knowledge Prediction (SKP) Proxy Task.** It first represents multi-modal content as a generalized form of “structural knowledge”. Then, a pre-train task is hereby proposed to model video representations with the exploited “structural knowledge” as pseudo-labels. Essentially, the pseudo-labels act as learnable semantic centers in the representation space.

2. A novel sampling approach for **Knowledge-Guided Contrastive Learning (KCL) with Hard Negatives.** This module augments the contrastive pre-train task by a light-weight multi-level hard negative sampling method: a) The instance-level sampling retrieves mutually similar samples to form batches of hard negatives; b) Based on that, the batch-level sampling selects eligible batches with the aid of the aforementioned knowledge.

We evaluate our method on four standard text-video retrieval tasks and one multi-choice QA task. Our experiments show clear improvements in performance across all tasks and datasets considered, outperforming prior works by a substantial margin. Besides, we provide detailed ablation analysis and insights of how our methods affect the latent representation space in the view of alignment and uniformity [52]. The analysis demonstrates that our knowledge regularizations promote the uniformity property to help data points better “spreading” on the representation space. It results in better separability of the representations and benefits the cross-modal retrieval tasks.

To summarize, we make the following contributions:

1. We propose a video-text pre-training paradigm to improve the representation learning with knowledge regularizations, which consists of a novel SKP prediction task and a KCL hard negative sampling approach.
2. We demonstrate clear improvements of our methods on five downstream tasks, achieving state-of-the-art performance compared to prior works.
3. We conduct in-depth analysis of the representation distributions of our model and provide intuitive explanations to the mechanism of our method.

## 2 RELATED WORK

### 2.1 Video-and-Language Pre-training

Since the release of several large-scale video-text datasets [3, 38], there have been a lot of works leveraging these corpora for video-text pre-training, including alleviating data noise [37, 56], designing better model architectures [23, 34, 55, 62] or proposing new objectives [23, 39].

Briefly, the dominant paradigm of video-language pre-training is to train powerful transformer-based encoders with various self-supervised proxy tasks. The canonical pre-train objectives include conditioned masked modeling and cross-modal video-text matching [8, 22]. As for the video-text matching, the existing works usually leverage cross-modal contrastive objectives to learn the semantic alignments. Besides, there are other works modeling frame orders [23] or using generative objectives [59].

Beyond the existing works, we propose a novel method to incorporate knowledge regularizations for video-text pre-training, and show its effectiveness through abundant experiments and insights.

### 2.2 Incorporating Structural Knowledge

The knowledge-enhanced pre-training methods first emerge in the NLP fields. The researchers incorporate structural knowledge in different aspects, such as augmenting raw textual data [28], integrating knowledge embeddings [31, 61], or constraining masking objectives [45].

For the video-text scenario, there are pioneering works trying to incorporate object-level structural knowledge into pre-training. More recently, OA-Trans [50] explicitly utilizes the object tags and bounding boxes in the training process. ALPRO [21] proposes to obtain visually-grounded entities by a standalone prompter without requiring off-the-shelf object detectors. However, most of the existing works are limited to use only object-level information as model inputs, ignoring the structural relationships (e.g., subject-predicate knowledge, etc.).

In contrast, our work expands the symbolic concepts (e.g., objects, relations, etc.) to a generalized form of “structural knowledge” by integrating various structural relationships.
2.3 Contrastive Learning with Hard Negatives

The key idea behind contrastive learning [19] is to learn a general feature function by discriminating positive and negative samples - making the positive pairs attracted and the negative pairs separated [13, 42, 48]. This learning paradigm has been proved very successful in represetnsing visual data [7, 16], natural languages [32], graphs [15, 44], etc.

Hard negatives have been proved essential for contrastive learning systems [4, 58]. In the computer vision domain, related works propose various methods to explore negatives, such as maintaining a memory queue to enrich negatives [16], designing sampling distributions [41] or generating negatives in latent space [18]. UNIMO [26] constructs hard textual negatives by re-writing sentences for image-text pre-training. Recently, several works propose to retrieve nearest-neighbor pairs as contrastive negatives for dense text retrieval [54] or video-text pre-training [56].

In this work, we propose a knowledge-guided approach for contrastive learning with hard negatives (KCL). Our method designs a multi-level sampling method to realize an instance-level hard negative sampling and a controllable batch-level sampling to emphasize desired semantics.

3 OUR METHOD

In this work, we propose a Cross-modal, knOwledge-enhanced Pre-training (CLOP) method. The overall framework is illustrated in Fig. 2.

3.1 Architecture Overview

We use transformer [49] as the backbone of our model, and preserve several canonical pre-train tasks. Specifically, we follow the architecture of HERO [23] to establish a base pre-training framework. Our base model mainly consists of a shared multi-modal encoder for representing texts and videos, four proxy tasks including Masked Language Modeling (MLM), Masked Frame Modeling (MFM), Frame Order Modeling (FOM) and Video-Text Matching (VTM). The VTM task learns from contrastive samples using hinge-based triplet loss. Considering the cost of data loading and the efficiency of computing, we follow [24] to pre-extract 2D + 3D video features [11, 40] as input, instead of feeding the model with raw video as several recent end-to-end arts [3, 20] do.

As shown in Fig. 2, we illustrate the overview of our method. The video and text pairs are fed into transformer-based encoders to obtain latent representations. We exploit the cues of the text and video to build an explicit form of "structural knowledge representation", including entities, actions, and relations. Based on that, we design two knowledge regularization components. The SKP proxy task pulls together the videos of similar semantics with multi-label classification, where the “structural knowledge” serves as pseudo-labels to supervise the learning process (Sec. 3.2). As the SKP module learns a general understanding of videos, it can update and complement the “structural knowledge” with its predicted results (marked with red dashed line boxes in Fig. 2) when it converges. The KCL module performs sampling for contrastive learning to push apart the cross-modal hard negatives under knowledge guidance (Sec. 3.3). During the batch-level sampling, the aforementioned knowledge specifies a partition of semantic spaces and guides the model to select proper batch anchors to emphasize the learning of desired semantics. According to the anchors, the instance-level sampling procedure assembles batches of hard negatives (mutually similar samples) for contrastive learning. Altogether, these two knowledge regularizations collaborate together to form a superior cross-modal representation space.

3.2 Structural Knowledge Prediction as Proxy Task

This module represents multi-modal data as a generalized symbolic form of "structural knowledge". Meanwhile, a cross-modal prediction task is hereby proposed to model video representations with the exploited “structural knowledge".
### 3.2.1 Structural Knowledge Representations.

We represent multimodal raw data as symbolic concepts, then expand these concepts by incorporating “structural knowledge”, i.e., exploiting the structural relationships (e.g., subject-predicate knowledge, superordinate concept knowledge, etc.) to build a generalized form of representation.

For a sample \( e = (o, t) \), we first collect two types of symbolic concepts from the textual and visual inputs. In detail, for the textual side, we utilize the Stanford CoreNLP Toolkit [36] and Stanford Part-Of-Speech Tagger [47] to extract all meaningful concepts (e.g., nouns and verbs) from the text. For the visual side, we generate scene graphs from videos with [14] and remain the names of recognized objects as concepts. Altogether, we get a list of atom concepts \( \{ c_i \} = \{ c_0, c_1, \cdots \} \) for each sample.

Afterward, we exploit various structural relationships to expand the atom concepts. Generally, we denote each structural relationship as an operation \( G_{rel}(\cdot) \) to transform a set of concepts into a new set of concepts. For example, the subject-predicate (SP) relationship is important for discriminating fine-grained semantics when there are multiple subjects (agents) but a single predicate (action). We incorporate this kind of knowledge by create another type of concepts (subject-predicate phrases) by combining each predicate with its corresponding subject, i.e., \( G_{sp}(\{ c_i \}) = \{ c_i^{sp}, c_i^{sp}, \cdots \} \). We also use the as-is relationship \( G_{as}(\{ c_i \}) = \{ c_i \} \) to remain the atom concepts. More structural relationships could be considered to further enhance the representation, such as linking them with outside concepts based on external KBs, etc. In the end, we build the “structural knowledge” representations by merging the output of every transform operation: \( o = \{ o_j \} = \bigcup G_{rel}(\{ c_i \}), rel \in \{ sp, as, \cdots \} \).

Overall, the “structural knowledge” representations hold the following benefits:

1. Universal. Representations of different semantic granularity (e.g., entities, actions, subject-predicate relations, etc.) are unified in the flexible form of explicit “structural knowledge”.
2. Extendable. It is possible to link the multi-modal content with outside structural resources (e.g., superordinate concept), thus expanding the realm of knowledge.

#### 3.2.2 Structural Knowledge Prediction Proxy Task.

We regard the aforementioned structural knowledge as pseudo-labels, and propose a simple yet effective structural knowledge prediction (SKP) proxy task to regularize the representation of videos.

Given the “structural knowledge” representation \( o_e \) for each sample \( e = (o, t) \) in the pre-train dataset \( D = \{ e_0, e_1, \cdots, e_{|D|-1} \} \), we first collect the most frequently (e.g., top 15000) appeared concepts across the pre-train dataset, i.e., \( O = \text{topK}(\{ o | o \in o_e, \forall e_1 \in D \}) \). Then, each video \( o \) is fed into the encoder \( f(\cdot) \) and a Multi-Layer Perceptron (MLP) to predict the probability of “structural knowledge” targets: \( P(y_i|v; \theta) = \sigma(\text{MLP}(f(o))) \).

The SKP task is formulated as a multi-label classification with binary cross-entropy loss:

\[
L^{SKP} = - \sum \left[ y_{i}^{*} \log P(y_{i}|v; \theta) + (1 - y_{i}^{*}) \log(1 - P(y_{i}|v; \theta)) \right],
\]

where \( y_{i}^{*} \) is the ground-truth label indicating whether concept \( O_i \) exists in \( o_e \), \( \theta \) represents the neural network parameters.

Note that, our SKP task is obviously different from existing pre-train tasks in several aspects:

1. Compared with other prediction tasks, e.g., Masked Language Modeling (MLM) [9], we learn visual representations with cross-modal cues, instead of modeling contextual language. Besides, the prediction targets are beyond the word vocabulary.
2. Compared with other cross-modal tasks, e.g., visual-text matching [33], we construct flexible prediction targets to learn semantic alignments with multiple granularities.

As a proxy pre-train task, the SKP task regularizes the embedding space by pulling together the videos of similar semantics, where the “structural knowledge” specifies learnable semantic centers. A detailed analysis of how the SKP task affects the embedding space is provided in the experiments section (Sec. 4.4).

### 3.3 Knowledge-Guided Contrastive Learning with Hard Negatives

We propose a knowledge-guided sampling approach for contrastive learning (KCL) with hard negatives. It is a two-stage sampling procedure: 1) For the batch-level, the model is guided to select proper anchor samples for every batch to emphasize the learning of desired semantics. 2) For the instance-level, according to the batch anchors, we assemble batches of hard negatives (mutually similar samples) from the whole training dataset for contrastive learning.

We first introduce the instance-level hard negative sampling for contrastive learning as a basis, then elaborate on the knowledge-guided batch-level sampling.

#### 3.3.1 Contrastive Learning with Instance-Level Hard Negative Sampling.

The hard negative samples are more informative than the easy ones for learning a good embedding space [4, 58]. Inspired by recent works [54, 56], we implement a custom sampling procedure to gather batches of hard negatives from the training set, by retrieving clusters of mutually similar samples.

The similarity of two individual samples is measured based on their latent representations in a common embedding space. For each sample \( e = (o, t) \), its embedding is calculated as the average of its visual and textual representations. We adopt a light-weight “self-distilled, online-updating” approach to maintain the embedding of each sample:

1. Self-distilled. The embeddings are computed by the model itself without requiring any extra resources;
2. Online-updating. We update the embeddings on-the-fly during the previous training epoch, thus introducing no computational overhead.

At the start of each training epoch, we first sample \( L \) (the number of batches per epoch) samples from the training set, in which each sample is regarded as the representative anchor of each batch. Then, we collect each batch of samples according to the anchor by retrieving its \( k \) (a number larger than the batch size) nearest neighbors \(^1\) from the training set and randomly choose \( N \) (the batch size) neighboring samples to finally form a batch. In this way, we gather \( L \) hard negative batches, where each sample is a hard negative of each other in the same batch.

Note that during the sampling procedure, a part of training samples may appear zero or multiple times in one epoch. To ensure that

\(^1\)We use faiss [17] for efficient k-nearest neighbor searching.
every sample embedding is available for nearest neighbor searching in the next epoch, we organize the missing training samples into random batches \( \mathcal{R} \). Combined with the hard negative batches \( \mathcal{C} \), we randomly shuffle the order of the \( \mathcal{C} + \mathcal{R} \) batches for this epoch. The detailed sampling procedure is described in Algorithm 1.

**Algorithm 1** Training with KCL Hard Negative Sampling.

**Input:**
1) all samples \( \mathcal{D} = \{e_0, e_1, \ldots , e_{|\mathcal{D}|-1}\} \), where \( e_i = (v_i, t_i) \);  
2) the hyper-parameter batch-size \( N \);  
3) the encoder \( f(\cdot) \).

**Output:** A list of batches \( \mathcal{B} = \{B_0, B_1, \ldots , B_{|\mathcal{B}|-1}\} \), where each \( B_i \) consists of \( N \) samples: \( \{e_{b_0}, e_{b_1}, \ldots , e_{b_{N-1}}\} \)

1: \( M \) is an empty embedding memory cache.  
2: for all epoch do  
3: \( M \) is not empty then  
4: \% batch-level sampling  
5: Draw \( L \) anchors \( \mathcal{A} = \{e_{\alpha_0}, \ldots , e_{\alpha_{L-1}}\} \) from \( \mathcal{D} \) as in Sec. 3.3.2  
6: \% instance-level sampling  
7: for all anchor \( i \in \{0, \ldots , L-1\} \) do  
8: Retriever a cluster of samples using nearest neighboring:  
9: \( \mathcal{G}_i \sim \text{KNN}(M[e_{\alpha_i}], k = 2N; M[\cdot]) \), where \( |\mathcal{G}_i| = N \)  
10: end for  
11: \% add the non-visited samples  
12: \( \mathcal{R} = \text{Random\_Assemble\_Batches}(\mathcal{D} - \{e | e \in \mathcal{G}_i, 0 \leq i < L\}) \)  
13: \( \mathcal{B} = \text{Random\_Assemble\_Batches}(\mathcal{D}) \)  
14: end if  
15: for all batch \( B_i \in \mathcal{B} \) do  
16: \( \forall j \in \{0, \ldots , N-1\} \)  
17: Forward: infer embeddings \( \{(e_{b_{ij}}, z^0_{b_{ij}}, z^1_{b_{ij}})\} = f(\{e_{b_{ij}}\}) \)  
18: Update \( M[e_{b_{ij}}] = (z^0_{b_{ij}}, z^1_{b_{ij}}) / 2 \)  
19: Backward: update the encoder parameters \( f(\cdot) \).  
20: end for  
21: Build KNN retriever index with \( M \).  
22: end for

In the training stage, the textual and visual input are fed into the encoder \( f(\cdot) \) to obtain the normalized representations \( \{(z^0_{ij}, z^1_{ij})\} = f(\{e_{ij}\}) \). For a batch of features \( \{(e_{b_{ij}}, z^0_{b_{ij}}, z^1_{b_{ij}}), \ldots , (e_{b_{iN-1}}, z^0_{b_{iN-1}}, z^1_{b_{iN-1}})\} \), every sample is a contrastive negative of the others. We compute the symmetrical hinge-based triplet loss for a batch as follows:

\[
L_{ij}^{(12)} = \max(0, \delta + \text{sim}(z^1_{ij}, z^2_{ij}) - \text{sim}(z^0_{ij}, z^2_{ij})), \quad (2)
\]

\[
L_{ij}^{(v21)} = \max(0, \delta + \text{sim}(z^0_{ij}, z^1_{ij}) - \text{sim}(z^0_{ij}, z^2_{ij})), \quad (3)
\]

\[
L^{(KCL)} = \frac{1}{N} \sum_{ij} (L_{ij}^{(12)} + L_{ij}^{(v21)}), \quad \forall i, j \in \{0, \ldots , N-1\} \quad (4)
\]

where \( \delta \) is the hyper-parameter margin, \( \text{sim} \) operator computes the cosine similarity of two normalized features.

### 3.3.2 Knowledge-Guided Batch-Level Sampling

In the previous section, we have developed an instance-level hard negative sampling method for gathering samples in a batch. It constructs each batch with mutually similar samples, in other words, each batch is assembled around a specific "topic". However, we notice that there is a "distribution discrepancy" issue for the sampling procedure:

(1) Redundancy. A considerable number of "topics" are frequently sampled during pre-training but are usually unimportant for video understanding, e.g., abstract scenes, boring landscape views, etc.;

(2) Insufficiency. In contrast with redundancy, there are numerous "topics" very important for video understanding but not sufficiently learned, e.g., human actions, events, etc.

Based on these observations, we propose a knowledge-guided batch-level sampling component to address the "distribution discrepancy" issue. This component augments the instance-level sampling algorithm of Sec. 3.3.1 as shown in Alg. 1 line 5. It modulates the sampling distribution of batch anchors to emphasize the desired but insufficient learning "topics", and suppress the inessential but redundant learning "topics". In this procedure, we use the prediction results of the aforementioned SKP task (Sec. 3.2) to identify the "topics" of each batch. In this sense, we denote the specified collections of "topics" (desired semantics) as knowledge to guide the batch sampling and model learning.

In detail, we adopt a heuristic approach to emphasize the human-action-related semantics. When sampling batch anchors, we increase the sampling probabilities for the anchors that involve human actions, at the same time preventing the selection of irrelevant anchors. We validate the feasibility of our instance-level and batch-level sampling methods against random negative sampling for contrastive learning in the ablations (Sec. 4.3). A qualitative visualization of our KCL sampling are shown in Fig. 3.

We close the section by pointing out that, our SKP (Sec. 3.2) and KCL (Sec. 3.3) knowledge regularizations collaborate together to encourage a superior cross-modal representation space (Sec. 4.4).

### 4 EXPERIMENTS

In this section, we first describe the overall experiment settings in Sec. 4.1, then present the main results on several end tasks in Sec. 4.2. We also provide in-depth analysis of our method through ablations studies (Sec. 4.3), insights of the embedding space (Sec. 4.4) and visualized examples (Fig. 4).
4.1 Experiment Settings

Our experiment follows the conventional “pretrain + finetune” two-stage paradigm. We pre-train our models without/with our knowledge regularization techniques (i.e., base/full variants of our method), then fine-tune on downstream tasks with the pre-trained weights as model initialization.

Our method only takes effect in the pre-train stage and not the fine-tune stage. The changes in the resulting finetune scores clearly reflect the effectiveness of our proposed method. Additionally, we also provide zero-shot results, which should be more straightforward for evaluating pre-training strategies.

Pre-Training Dataset. We perform video-text pre-training on the recently released large-scale WebVid2M dataset [3]. The WebVid2M dataset consists of 2.5 million video-text pairs scraped from the web. Despite that the dataset is an order of magnitude smaller than the Howto100M [38] dataset, it demonstrates better alignments between text semantics and video content.

End Tasks. We evaluate our methods on four retrieval tasks and one multi-choice QA task.

The MSRVTT [57] is a well-known dataset for text-video retrieval. This dataset consists of 10K videos, where each video is associated with 20 text descriptions. We follow the official 1k-A split [60] for training and evaluation.

The MSVD [5] dataset consists of 1,970 videos with about 40 text descriptions per video. This dataset is split into 1,200 videos for training, 100 videos for validation, and 670 videos for testing. We report the text-video retrieval results on the standard 670 test videos.

The VATEX [53] dataset is a large-scale, high-quality multilingual video-text dataset. This dataset contains 34,991 videos, of which 25,991 videos are for training, 1K for validation, and 6K for testing. Since the test annotations are not available, we follow the widely-used HGR’s [6] protocol which splits the original validation set into 1,500 test videos and 1,500 validation videos. We only use the English annotations for fair comparisons.

The DiDeMo [2] dataset contains 10K flickr videos and 40K localized text annotations. We follow [3, 20, 29] to evaluate paragraph-to-video retrieval task, where text annotations of the same video are concatenated into a single query. We report the results without the ground-truth localization proposals (as described in [3]).

The MSRVTT multi-choice test [60] is formulated as a multi-choice QA task, which requires the model to select the correct (most relevant) answer from 5 textual candidates based on a given video. This task shares the MSRVTT training videos and evaluates video-text relevance similar to retrieval tasks.

Evaluation Metrics. We report the Average Recall at rank K (R@K, K ∈ {1, 5, 10}) and the Median Rank (MedR) metrics for retrieval tasks. The higher R@K and the lower MedR indicate better performance. For the MSRVTT multi-choice test, we report the accuracy (Acc.). Note that, in order to reduce the variance in finetuning experiment, the scores are averaged from three repeated experiments with casually selected random seeds.

| Method       | Years | Pretrain Dataset | AveR | R@1  | R@5  | R@10 | MedR |
|--------------|-------|------------------|------|------|------|------|------|
| ActBERT [62] | CVPR’20 | [136M] HowTo100M | 38.7 | 16.3 | 42.8 | 56.9 | 10.0 |
| UniVL [34]   | Arxiv’20 | [136M] HowTo100M | 44.6 | 21.2 | 46.9 | 63.1 | 6.0  |
| MMT [12]     | ECCV’20 | [136M] HowTo100M | 51.1 | 26.6 | 57.1 | 69.6 | 4.0  |
| SupportSet [39] | ICLR’21 | [136M] HowTo100M | 52.6 | 30.1 | 58.5 | 69.3 | 3.0  |
| ClipBERT [20] | ICCV’21 | [6.5M] COCO, VisGenome | 42.9 | 22.0 | 46.8 | 59.9 | 6.0  |
| VideoCLIP [56] | EMNLP’21 | [136M] HowTo100M | 51.0 | 30.9 | 55.4 | 66.8 | -    |
| Frozen [3]   | ICCV’21 | [5.5M] WebVid2M, CC3M | 53.7 | 31.0 | 59.5 | 70.5 | 3.0  |
| ALPRO [21]   | Arxiv’21 | [5.5M] WebVid2M, CC3M | 55.9 | 33.9 | 60.7 | 73.2 | 3.0  |
| OA-Trans [50] | Arxiv’21 | [2.5M] WebVid2M | 55.4 | 32.7 | 60.9 | 72.5 | 3.0  |
| [ours] base  | -     | [2.5M] WebVid2M | 57.8 | 34.0 | 64.3 | 75.0 | 3.0  |
| [ours] full  | -     | [2.5M] WebVid2M | 58.2 | 33.7 | 64.7 | 76.3 | 3.0  |

| Method       | AveR | R@1  | R@5  | R@10 | MedR  |
|--------------|------|------|------|------|-------|
| - zeroshot - |      |      |      |      |       |
| Frozen [3]   | 36.6 | 18.7 | 39.5 | 51.6 | 10.0  |
| [ours] base  | -    | 36.2 | 18.7 | 39.8 | 50.0  |
| [ours] full  | -    | 38.7 | 21.3 | 42.4 | 52.5  | 9.0  |

Implementation Details. We pre-train a 6-layer 12-head transformer (initialized with Roberta [30] weights) as the video and text encoder with five proxy tasks (SKP and KCL, plus other three tasks following [23]). The multiple pre-train tasks are switched at random for each batch. We use 4 A100 GPUs and pre-train for about 6 days, with batch size 128 per GPU and learning rate 2e-5. Please refer to the Supplementary Material for more implementation details.
4.2 Overall Results on End Tasks
The overall results on five downstream tasks are listed in Tab. 1-5. We provide two model variants of ours in these tables. As the name indicates, the [ours] base variant represents the base model we built on top of, while the [ours] full represents the full model with our knowledge regularization techniques (SKP + KCL). The direct comparisons between the two variants can provide convincing evidence for assessing the effects of our proposed method.

Compared with the base model, we obtain on average +2.9% fine-tune and +3.2% zero-shot AveR@{1, 5, 10} boosts across the MSRVTT (Tab. 1), MSVD (Tab. 2), VATEX (Tab. 3) and DiDeMo (Tab. 4) retrieval tasks. Besides retrieval, we also conduct a quick evaluation of the MSRVTT multi-choice test in Tab. 5. In general, our method shows clear improvements across all the tasks above and outperforms the state-of-the-art methods by a substantial margin. The overall results demonstrate the effectiveness of our method and prove it generalizes well across various datasets.

Table 3: VATEX HGR-1k5 test set.

| Method      | AveR | R@1  | R@5  | R@10 | MedR |
|-------------|------|------|------|------|------|
| HGR [6]     | 64.0 | 35.1 | 73.5 | 83.5 | 2.0  |
| SupportSet [39] | 72.0 | 44.6 | 81.8 | 89.5 | 1.0  |
| SupportSet-PT [30] | 72.9 | 45.9 | 82.4 | 90.4 | 1.0  |
| [ours] base | 78.1 | 54.8 | 87.0 | 92.6 | 1.0  |
| [ours] full | 79.7 | 56.6 | 88.6 | 93.9 | 1.0  |

- zeroshot -
| [ours] base | 33.3 | 19.5 | 37.2 | 43.1 | 32.0 |
| [ours] full | 35.8 | 22.8 | 39.5 | 45.1 | 19.0 |

Table 4: DiDeMo test set.

| Method       | AveR | R@1  | R@5  | R@10 | MedR |
|--------------|------|------|------|------|------|
| CE [29]      | -    | 16.1 | 41.1 | -    | 8.0  |
| ClipBert [20] | 40.5 | 20.4 | 44.5 | 56.7 | 7.0  |
| Frozen [3]   | 54.4 | 31.0 | 59.8 | 72.4 | 3.0  |
| [ours] base  | 51.5 | 27.8 | 57.2 | 69.6 | 4.0  |
| [ours] full  | 56.2 | 32.7 | 62.7 | 73.3 | 3.0  |

- zeroshot -
| [ours] base | 36.9 | 16.2 | 41.1 | 53.4 | 9.0  |
| [ours] full | 40.8 | 20.2 | 45.5 | 56.7 | 8.0  |

Table 5: MSRVTT Multi-choice QA.

| Method     | Accuracy |
|------------|----------|
| JSFusion [60] | 83.4     |
| ActBERT [62] | 85.7     |
| ClipBert [20] | 88.2     |
| VideoCLIP [56] | 92.1     |
| Ours       | 95.6     |

We notice that, a line of CLIP-based methods [10, 35] have achieved remarkable performance on the retrieval tasks. However, they use the model architecture and weights from CLIP [40], which is pre-trained on 400 million image-text pairs (150x larger than WebVid2M [3]), thus are not suitable to compare in our settings.

4.3 Ablation Studies
As shown in Tab. 6, we conduct extensive ablation experiments to study the effects of each component in our method. Considering the heavy computational cost, unless specified, we use a subset of WebVid data (500K / 2.5M) to pre-train our model in ablation studies. We provide the average recall AveR@{1, 5, 10} results on the VATEX HGR-1k5 test set in finetuning and zero-shot settings. Note that, the finetune results are averaged from three repeated experiments with casually selected random seeds.

4.3.1 Effects of the Model Components. As shown in Tab. 6 (1), both the SKP and KCL components obtain obvious improvements over the base model (+1.52% and +1.04%) for finetuning. Putting them together achieves even better performance (+1.99%). As for zero-shot evaluation, it shows less improvement (+0.18%) for SKP but greater improvement (+2.59%) for KCL. We think the reason is that, KCL directly optimizes the cross-modal relevance similar to the retrieval end task, while the SKP task improves the potential to be exploited with finetuning for superior representations.

4.3.2 Effects of the Structural Knowledge for SKP Task. The SKP[∗] represents SKP variants with different structural knowledge representations. [t] only uses the atom concepts from the text; [t + v] uses both the textual and visual atom concepts; [t + v + s] expands all the atom concepts with structural relationships, e.g., the subject-predicate (SP) knowledge. As shown in Tab. 6 (2), adding the video-side concepts improves zero-shot performance (+1.48%); incorporating the structural relationships brings extra +0.37% finetune performance gains.

4.3.3 Effects of the Strategies for KCL sampling. The KCL[v1] and KCL[v2] represent the negative sampling in Sec. 3.3.1 and Sec. 3.3.2. As shown in Tab. 6 (3), the relative +0.38% and +1.51% improvements support the effectiveness of the knowledge-guided batch-level sampling. Overall, our method outperforms the base model by a significant margin (+2.72% and +6.68%).

Table 6: Ablation results on the VATEX HGR-1k5 test set.

| Ablations                        | Finetune | Zero-shot |
|----------------------------------|----------|-----------|
|                                  | AveR     | Boost     |
| - Base model                     | 78.14    | +2.22     |
| 1) - Model components            |          |           |
| - SKP[t]                         | 77.44    | +1.52     |
| - K[ v1]                         | 76.96    | +1.04     |
| - K[ v1] + SKP[t]                | 77.91    | +1.99     |
| 2) - SKP task                    |          |           |
| - K[ v1] + SKP[t + v]            | 77.89    | +1.97     |
| - K[ v1] + SKP[t + v + s]        | 78.26    | +2.34     |
| 3) - KCL sampling                |          |           |
| - K[ v2] + SKP[t + v + s]        | 78.64    | +2.72     |
|                                  | 35.27    | +6.68     |

4.4 Insights of the Representation Space
To get further insights, we conduct a quantitative analysis of how our methods affect the latent representation space in the view of alignment and uniformity [52].
The alignment metric measures the “closeness” of positive pairs (i.e., similar samples should have similar features). We make slight modifications to adapt the formulations to the cross-modal scenario, as the expected distance between positive video-text pairs:

\[
\mathcal{L}_{\text{align}}(f; \alpha) \triangleq \mathbb{E}_{(v, t) \sim p_{\text{pos}}} \| f(v) - f(t) \|_2^2, \quad \alpha > 0.
\] (5)

The uniformity metric indicates the “spreading” degree of the data points on the hypersphere. We measure this property for text and videos, respectively. It is defined as the logarithm of the average pairwise Gaussian potential:

\[
\mathcal{L}_{\text{unif}}^{(m)}(f; \beta) \triangleq \log \mathbb{E}_{(x, y) \sim p_m} [e^{-\beta \| f(x) - f(y) \|_2^2}], \quad \beta > 0.
\] (6)

where \( m \in \{\text{vis}, \text{txt}\} \).

We use these metrics to quantitatively analyze the representation distributions of our model. The lower \( \mathcal{L}_{\text{align}} \) and \( \mathcal{L}_{\text{unif}} \) (or equivalently as the y axis in Fig. 5, the higher \( -\mathcal{L}_{\text{align}} \) and \( -\mathcal{L}_{\text{unif}} \)) indicate higher quality of representation space. A recent work [51] identifies that there is a compromise between these two properties.

As shown in Fig. 5, we illustrate the zero-shot alignment and uniformity of our model on three retrieval tasks and have several interesting insights as follows:

(1) **Our method promotes the uniformity to benefit the cross-modal retrieval.** As shown in Fig. 5 (b) and Tab. 6 (1), we observe a clear correlation of the uniformity enhancements and the retrieval performance boosts. The model seems to find an effective way to improve the separability for retrieval on the hypersphere: optimizing the uniformity property to help data points better “spreading” on the representation space, even at the cost of hurting alignment to some extent.

(2) **The SKP and KCL regularizations collaborate to work.** Our SKP task attracts similar videos to learnable semantic centers. When combined with KCL which efficiently pushes apart cross-modal hard negatives, they together achieve better uniformity of the representation space, as shown in Fig. 5 (b). A consistent uniformity improvement is observed across different retrieval datasets.

**Overall, our knowledge regularizations encourage the model to fully utilize the representation space for better separability, thus benefiting the cross-modal retrieval tasks.**

### 5 CONCLUSIONS

In this work, we propose CLOP, a video-and-language pre-training method with Knowledge Regularizations. Our method consists of a simple yet effective Structural Knowledge Prediction (SKP) proxy task and a novel sampling approach for Knowledge-Guided Contrastive Learning (KCL) with hard negatives. The extensive experiments validate the effectiveness of our method, showing that CLOP outperforms the state-of-the-art methods on five downstream tasks. An in-depth analysis shows evidence that, our knowledge regularizations promote the uniformity property to form a superior cross-modal representation space.
[50] Alex Jinpeng Wang, Yixiao Ge, Guanyu Cai, Ru Yan, Xudong Lin, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. 2021. Object-aware Video-language Pre-training for Retrieval. *arXiv preprint arXiv:2112.00656* (2021).

[51] Feng Wang and Huagong Liu. 2021. Understanding the behaviour of contrastive loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2495–2504.

[52] Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*. PMLR, 9929–9939.

[53] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. 2019. Vatex: A large-scale, high-quality multilingual dataset for video- and language research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4581–4591.

[54] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate nearest neighbor contrastive learning for dense text retrieval. (2021).

[55] Hu Xu, Gargi Ghosh, Po-Yao Huang, Prahal Arora, Masoumeh Aminzadeh, Christoph Feichtenhofer, Florian Metze, and Luke Zettlemoyer. 2021. VLM: Task-agnostic Video-Language Model Pre-training for Video Understanding. *arXiv preprint arXiv:2105.09996* (2021).

[56] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. 2021. VideoCLIP: Contrastive Pre-training for Zero-shot Video-Text Understanding. *arXiv preprint arXiv:2109.14084* (2021).

[57] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. M3r-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5288–5296.

[58] Hong Xuan, Abby Stylianou, Xiaotong Liu, and Robert Pless. 2020. Hard negative examples are hard, but useful. In *European Conference on Computer Vision*. Springer, 126–142.

[59] Fei Yu, Jiji Tang, Weichong Yin, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2020. Ernie-vil: Knowledge enhanced vision-language representations through scene graph. *arXiv preprint arXiv:2006.16934* (2020).

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

[61] Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced language representation with informative entities. *arXiv preprint arXiv:1905.07129* (2019).

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