FitCLIP: Refining Large-Scale Pretrained Image-Text Models for Zero-Shot Video Understanding Tasks

Santiago Castro*1
https://santi.uy/

Fabian Caba Heilbron2
https://fabiancaba.com/

1 University of Michigan
2 Adobe Research

Abstract

Large-scale pretrained image-text models have shown incredible zero-shot performance in a handful of tasks, including video ones such as action recognition and text-to-video retrieval. However, these models have not been adapted to video, mainly because they do not account for the time dimension but also because video frames are different from the typical images (e.g., containing motion blur, less sharpness). In this paper, we present a fine-tuning strategy to refine these large-scale pretrained image-text models for zero-shot video understanding tasks. We show that by carefully adapting these models we obtain considerable improvements on two zero-shot Action Recognition tasks and three zero-shot Text-to-video Retrieval tasks. The code is available at https://github.com/bryant1410/fitclip

1 Introduction

Imagine is winter season and our quest is to develop an auto-tagging system that recognizes all the activities in our winter vacation footage. Luckily, there have been tremendous advances in the action recognition community [3, 7, 54]. For instance, we could leverage one of the existing models that recognize up to 700 human actions [31]. Sadly, it turns out that our family’s favorite activity, sledding, is not on the list of categories that these models can recognize. In a traditional supervised setting, we would have to collect many sledding examples to train a new model. Such a process is labor-intensive, costly to create, and difficult to scale to recognize further new activities. Instead, zero-shot models [6, 32, 47] can alleviate such a burden by enabling recognition of unseen concepts.

Large pre-trained image-text models, such as CLIP [45] and ALIGN [29], have shown outstanding zero-shot capabilities on a handful of visual tasks, including video tasks such as Action Recognition and Text-to-Video Retrieval. Such models have overcome the limitations of traditional zero-shot learning algorithms by using abundant images (on the internet) with (free) natural language supervision. Despite their remarkable zero-shot performance in video tasks, there is room for improvement to close the image-to-video domain gap. For

© 2022. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.

*Work done as an intern at Adobe Research.
instance, recent studies have shown that fine-tuning CLIP yields significant improvements in target video tasks [36, 56]. Unfortunately, fine-tuning and improving performance in a target dataset comes with a cost: harshly penalizing the model’s zero-shot capabilities [59].

There have been multiple efforts to train video-language models that can be employed for various downstream video understanding tasks. Even though these approaches use video data, their zero-shot capabilities remain poor compared to those exhibited by CLIP [39]. It would be unfair not to mention that video-language pretraining methods either train with clean yet two orders of magnitude smaller datasets [8], or with large datasets with unaligned natural language supervision [43]. The alternative is to scale up further the amount of unaligned natural language supervision abundant on internet videos. In comparison, ALIGN [29] (in the image space) has shown the ability to cope with noisy supervision by scaling up to the billion-samples scale. However, replicating such experiments with video data would only be possible for selected (if any) industrial players.

This work introduces FitCLIP, a fine-tuning strategy to adapt large-scale image-text pre-trained models for zero-shot video understanding tasks. The goal of FitCLIP is to retain the knowledge of CLIP [45] while gently adapting and learning how video data looks. Our method leverages relatively small labeled and extensive pseudo-labeled video data to train a student network. To validate the effectiveness of FitCLIP, we designed and set zero-shot benchmarks for two popular video understanding tasks: action recognition and text-to-video retrieval. Our experiments empirically validate the effectiveness of distillation to better train and fine-tune multimodal video models and show that FitCLIP establishes a new state-of-the-art for zero-shot video recognition and retrieval. Our design incorporates weight-space ensembling in a strategic manner, which has not been explored before, as far as we know.

Contributions. Our key idea is to develop a method to refine large-scale pretrained image-language models to zero-shot video use-cases. Our work brings two contributions:

(1) We introduce FitCLIP, a refinement strategy, and model for zero-shot video understanding. The model leverages abundant knowledge in large-scale image models and a distillation strategy to learn new video knowledge. We describe FitCLIP in (Section 3).

(2) We evaluate FitCLIP and competitive baselines in a newly designed zero-shot benchmark (Section 4). Our experiments include results for two sets of video understanding tasks, action recognition, and text-to-video retrieval, where we show the value of FitCLIP (Section 5).

2 Related Work

Zero-shot Video Understanding. Multiple zero-shot methods have been proposed to tackle popular tasks such as action recognition [1, 4], text-to-video retrieval [60], and localization-related tasks [28, 66]. Most of the zero-shot action recognition literature either follows an attribute-based approach or leverages word embedding to transfer knowledge [6, 8, 18, 19, 28, 34, 37]. Differently, in the text-to-video retrieval task, zero-shot methods leverage large-scale natural language supervision to pre-train video-language models. After pretraining, these models can then be employed and tested in text-to-video retrieval tasks. Similar to [4, 60], our work leverages natural language supervision from video titles to unlock zero-shot capabilities. However, we focus on adapting already well-trained image-text models to videos rather than learning a video-language model from scratch.

One of our goals is to establish a benchmark for zero-shot action recognition and text-to-video retrieval. Previous efforts have devoted insightful analyses to creating true zero-shot evaluation for action recognition [24]. These efforts are valuable for the traditional zero-shot
setting where methods use a close vocabulary of (seen) actions, but they do not fit when zero-shot models learn with natural language supervision. Instead, we follow standard (full) tests on popular action recognition datasets and well-established text-to-video retrieval datasets.

**Visual-Language Pretraining.** Pretraining visual models with natural language became a popular learning strategy in the image domain \cite{973, 108, 124, 145, 150}. The idea of matching images with text dates back to the late 90s when Mori *et al.* trained models to predict nouns and adjectives from image-text pairs \cite{973}. Others modernized this idea using large-scale datasets to train CNNs \cite{124}. However, only recently, Radford *et al.* took this idea to the next level \cite{145}. They trained CLIP, a dual image-text encoder, with more than 400M images and text descriptions using a contrastive objective \cite{142}. Our work builds upon CLIP and adapts it to video use-cases while preserving its zero-shot capabilities.

Video-language pretraining also gained traction in the video space. Despite the progress, it has been hard for video-language methods to compete in zero-shot settings with image-language pre-trained models. We argue this is due to the limited availability of videos with clean (and aligned) natural language supervision. For instance, Frozen in Time \cite{194} trains a transformer-based architecture on the WebVid dataset, which contains 2.5M humanly curated video-title pairs. The dataset is at least two orders of magnitude smaller than the dataset to pre-train CLIP \cite{145}. The importance of large and diverse data emerges when we compare Frozen in Time with CLIP in zero-shot video tasks. Others \cite{158, 159} have trained with the relatively larger HowTo100M dataset, which contains 100M unaligned video-text pairs. Still, the zero-shot capabilities of these models remain subpar to what CLIP can provide. Our approach, FitCLIP, leverages the WebVid dataset \cite{194} as a rich source to adapt CLIP for zero-shot video understanding tasks.

**Refining Large-scale Image Models.** DistInit \cite{48} explored distilling image models for video. More recently, CLIP’s strong visual representation inspired multiple researchers to explore its usage for video tasks \cite{10, 16, 46, 126, 127, 128, 129}. CLIP4Clip, for instance, proposed a straightforward strategy to fine-tune CLIP for the text-to-video retrieval task \cite{127}. Surprisingly, their simple method sets a new state-of-the-art in various datasets. Similarly, ActionCLIP introduced a novel paradigm to action recognition harnessing CLIP’s general visual knowledge \cite{129}. While existing approaches effectively boost performance on target datasets, and tasks, they have not show to preserve the original CLIP zero-shot capabilities (also based on early experiments we ran).

### 3 Method: FitCLIP

Our goal is to train a model that expands and complements large image-language models \cite{124, 129} for zero-shot video (see Figure 1). To do so, we introduce FitCLIP, a refinement strategy that leverages small labeled and large pseudo-labeled data together with existing knowledge acquired from large image-text pairs. FitCLIP includes two steps. The first step trains a model, in a Teacher-Student fashion, leveraging both: labeled video-text pairs and pseudo-labels generated by a teacher model. The second step fuses the existing knowledge of the teacher, a large-scale pre-trained image-language model, with the student trained on video data. We call the resulting model the same as our refinement strategy, FitCLIP.
3.1 Teacher-Student Fine-tuning

Our goal is to train a model using video-text pairs while leveraging knowledge from image-language representations. One alternative is to reuse image-language encoder weights and fine-tune them in a target dataset [36, 56]. Such an approach is effective in boosting performance for in-distribution datasets but tends to fail at preserving the zero-shot capabilities of the original model’s weights due to catastrophic forgetting [16]. Instead, we focus on gently refining the original image-language model’s weights by incorporating a two-fold strategy. We use a small sample of labeled data to avoid model drift [46] (because of using a much smaller batch size and less diverse dataset), and we regularize the learning process by adding pseudo-labels generated with the original image-language model. Note that our strategy shares intuitions with the Knowledge Distillation literature [25], where a Teacher-Student analogy is used to describe the process of training a Student with priors derived from a strong Teacher model. Figure 1 (step 1) illustrates the process to train our Student model.

**Data Subsets.** Our fine-tuning strategy relies upon two subsets of data: a small labeled dataset of video-text pairs and an unlabeled (unaligned) set of video-text candidate pairs. The labeled subset contains a collection of videos matched with one text describing their visual content. These video-text pairs are of high quality and made by a human. The unlabeled subset also contains a list of videos and a list of text descriptions. However, the match between a video and the best describing text does not exist in this subset.

**Teacher Model.** The goal of the teacher is to provide soft pseudo-labels on unlabeled sets of video-text candidate pairs. We adopt CLIP [45] as a teacher. CLIP includes an image encoder and a text encoder, which were trained to predict the correct pairing of image-text pairs using a contrastive objective [42]. In practice, we use CLIP to compute the similarity between a subset of videos (within a large unlabeled set) and a set of candidate texts. Given that CLIP only takes individual images as input, we pass $N$ frames from the video through its visual encoder and mean-pool the outputs into a single visual feature. We then use these similarity scores as target soft pseudo-labels.

**Student Model.** We aim to train a student model that learns from video-text pairs and
distills knowledge from large pretrained image-language models. As the student, we choose
the same dual architecture proposed by CLIP [45]. To train the model, we leverage two types
of supervision: samples from the manually labeled video-text pairs dataset and soft pseudo-
labels from the unlabeled set. Like the Teacher model, the student’s visual stream takes
N frames from each video and mean-pool the resulting representations into a single feature.

**Student’s Training Objective.** We train the student model with two losses: a loss to learn
from labeled samples, and a loss to distill the teacher knowledge via pseudo-labels. Given a
video-text pair denoted \((v, t)\) our student’s dual encoder extracts a video representation \(z_v\) and
a text representation \(z_t\). For labeled samples, we use the InfoNCE [42] loss to learn a video-
text correspondence. We follow [4, 60] and minimize the text-to-video and video-to-text
correspondence. We follow [4, 60] and minimize the text-to-video and video-to-text
contrastive losses:

\[
L_{v2t} = \sum_{(v, t) \in B_l} \log \frac{e^{z_v \cdot z_t^+}}{\sigma} \frac{1}{\sum_{z_v \in \{z_v^+, z_v^-\}}} e^{z_v \cdot z_t^-} / \sigma
\]

(1)

\[
L_{t2v} = \sum_{(v, t) \in B_j} \log \frac{e^{z_t \cdot z_v^+}}{\sigma} \frac{1}{\sum_{z_t \in \{z_t^+, z_t^-\}}} e^{z_t \cdot z_v^-} / \sigma
\]

(2)

where \(\sigma\) is the temperature hyper-parameter, \(B_l\) is a batch of video-text pairs, \(z_v^+\) the pos-
itive text for the candidate video \(z_v\), \(z_v^+\) the positive video for candidate text \(z_t\), and \(\{z_v^+, z_v^-\}\)
the negatives sets to contrast the candidate video and text representations. Then \((L_{v2t} + L_{t2v})\)
is the final labeled (contrastive) loss.

To distill knowledge from soft pseudo-labels generated by the teacher, we use the teacher’s
predictions as pseudo-labels [45] and minimize the cross-entry of the student’s scores relative
to those from the teacher:

\[
L_{\text{distill,v2t}} = \sum_{(v, t) \in B_l} \frac{e^{x_v \cdot x_t}}{\sum_{x_v \in T} e^{x_v \cdot x_t}} \log \frac{e^{z_v \cdot z_t}}{\sum_{z_v \in T} e^{z_v \cdot z_t}}
\]

(3)

\[
L_{\text{distill,t2v}} = \sum_{(v, t) \in B_j} \frac{e^{x_t \cdot x_v}}{\sum_{x_t \in V} e^{x_t \cdot x_v}} \log \frac{e^{z_t \cdot z_v}}{\sum_{z_t \in V} e^{z_t \cdot z_v}}
\]

(4)

where \(x_v\) and \(x_t\) are the teacher’s video and text representations, and \(V\) and \(T\) are the sets
of videos and texts in the batch.

Our final objective combines the contrastive and distillation losses as in Equation (5). We scale the distillation loss, with \(\lambda\), to prevent over-fitting to noisy pseudo-labels.

\[
\mathcal{L} = \lambda (L_{\text{distill,v2t}} + L_{\text{distill,t2v}}) + (1 - \lambda) (L_{v2t} + L_{t2v})
\]

(5)

### 3.2 Fusing Teacher-Student Knowledge

We aimed to train a competent student compared to the teacher. However, it is hard to
compete with the 400M image-text pairs that were used to train CLIP [45]. Therefore, our
goal is, instead, to fuse both: the general visual knowledge encapsulated by the teacher and
the video-specific properties learned by the student. There are multiple ways to ensemble
models [41]; however, given that our fine-tuning strategy gently adapts the teacher to video
use cases, we can leverage elegant weight-space ensembling techniques [20, 59]. We follow
the same approach in [59] to linearly combine the teacher and student weights (by \(\alpha\) and create our final model, FitCLIP.
3.3 FitCLIP’s Implementation Details

We uniformly sample $N = 4$ frames from each video, similarly to TSN [55]. The Teacher and Student models both use a ViT-B/16 architecture initialized with OpenAI’s publicly released weights [45]. We empirically set $\lambda = 10^{-4}$ to smooth the training process (note the labeled and pseudo-labeled loss magnitudes may be wildly different). We consistently use $\sigma = 0.05$ as temperature value. At training time, we randomly crop the frames to a size of $224 \times 224$, and perform random horizontal flips. We use the AdamW optimizer with a learning rate equal to $3 \times 10^{-5}$. We use the same tokenizer as in CLIP [45]. We conduct our experiments using 8x A100 (40GB) GPUs. We use 4.5K labeled videos, randomly sampled from the WebVid-2.5M dataset [4], to compute the losses in Equations (1) and (2). The entire WebVid-2.5M dataset is used to compute the distillation losses – Equations (3) and (4). We choose the (labeled) validation loss in the WebVid-2.5M dataset as a criterion to select the best student models. Finally, to fuse the teacher and the student weights, we use $\alpha = 0.4$. We encourage the reader to read the supplementary material for analyses to some of the hyperparameter values. We wrote our code on Python using PyTorch [43] and Lightning [14].

4 Zero-shot Video Understanding Benchmark

4.1 Baselines

CLIP [45]. This model has been pre-trained with the WIT dataset [49], which contains about 400M image-text pairs. We re-implement the zero-shot inference of this baseline model. To deal with video, we encode $N = 4$ uniformly sampled frames per video and average their features to obtain the final video representation. In all our experiments, we use the publicly released CLIP ViT-B/16 [13] model. Note that our CLIP adaptation is equivalent to ActionCLIP [56] (see Supplementary Material).

CLIP4Clip [36]. This method proposes changes on top of CLIP. In particular, they propose something the authors call post-pretraining that fine-tunes CLIP on the category “Food and Entertaining” (380k videos) from the HowTo100M [38] dataset. The authors have not provided this checkpoint, so we cannot evaluate it on our benchmarks. Still, we decide to include the results they report. Nevertheless, note the evaluation conditions are not the same to constitute a fair comparison (e.g., the authors sample more than 4 frames per video clip).

Frozen in Time [4] (Frozen). This model was pre-trained leveraging video-text pairs from the WebVid dataset. There are multiple versions pre-trained versions for this model, including one that leverages the well-curated CC3M image-text pairs dataset. In our (main) experiments, we use the model that trains using the WebVid-2.5M, COCO, and CC3M dataset (note this is much less data than CLIP’s pretraining dataset). Results for other versions of Frozen in Time can be found in the supplementary material.

VideoCLIP [60]. This baseline uses a Transformer [53] on top of a frozen HowTo100M-pre-trained S3D [65] video model from MIL-NCE [39] and a fine-tuned BERT [11] text model. This method trains on HowTo100M. A notable difference is that VideoCLIP samples 32 clips of size 32 frames (1024 frames) for each video, while we sample only 4 frames for each video.

VIOLET [17]. This method uses a video-language transformer trained end-to-end by masking discrete visual tokens. The authors use multiple training datasets including CC3M and WebVid.
BridgeFormer [21] (BF). This model leverages a multimodal encoder on top of the unimodal encoders and a method that masks the main verb and nouns as a form of multiple-choice questions as a pre-text task. The authors find this method to be more sample efficient than vanilla NCE.

4.2 Zero-shot Tasks and Datasets

Action Recognition. Our goal is to classify a video with one of \(C\) possible action classes. To do so, we form pretext language queries with predefined prompts. An illustrative example is the following prompt: "a video of a person \(\{c_i\}\)", where \(c_i\) is the \(i\)-th class out of the \(C\) candidate action categories. Given the visual representation of the target video, we compute its similarity with the language feature of each candidate action class prompt. We predict the action class by selecting the visual-text pair with the highest similarity. We report the top-1 and top-5 accuracy. We evaluate zero-shot action recognition in two datasets:

- *Moments in Time (MiT)* [40] consists of 3-second YouTube clips that capture the dynamics of actions performed by varied subjects including animals and humans. The dataset includes 339 categories and 33,900 validation videos.

- *UCF101* [48] contains 101 action classes. Our zero-shot experiments in this dataset aim to classify all the 1794 available test videos from the split 1.

Text-to-video Retrieval. Given a text query, the goal of text-to-video retrieval is to find a video, from a collection, that visually matches the text description. Given that the concept of classes does not exist in this task, previous methods [4, 36] denote experiments as zero-shot when the visual-language models are not fine-tuned on the downstream datasets. To measure performance, we report recall at \(k = \{1, 5, 10\}\) and the median ranking (MdR). We evaluate zero-shot text-to-video retrieval in three datasets:

- *MSR-VTT* [61] contains video clips with a duration of up to 30 seconds paired with captions. We adopt the 1K-A test split [64], which contains 1,000 video-text pairs.

- *YouCook2* [67] comprises challenging cooking videos depicting fine-grained human actions. We test on 3305 clip-text pairs [39].

- *DiDeMo* [2] contains mostly unedited video clips from Flickr. We follow [4, 33, 35] and cast a video-paragraph retrieval problem. We evaluate on 4021 test samples.

5 Experimental Results

In this section, we conduct zero-shot experiments in two popular video understanding tasks, and then a diagnostic analysis of FitCLIP. First, we study the performance of the zero-shot baselines described in Section 4.1 in the task of action recognition. The second analysis summarizes the baseline performance in diverse datasets for text-to-video retrieval. We run diagnostic experiments to validate the importance of fusing the teacher knowledge, as in [59], to a competent zero-shot model. Finally, we run performance analyses on FitCLIP that study per-class gains in the action recognition task, and the shift in ranking distributions for the text-to-video retrieval tasks.
5.1 Zero-shot Action Recognition Results

We compare the zero-shot performance of FitCLIP and different baselines using two popular action recognition datasets. We describe the results and provide our analysis.

**Analysis on Moments in Time.** Table 1a summarizes the zero-shot results in the moments in time dataset. To establish a reference point, we also report VATT [1], the state-of-the-art using full supervision. In this dataset, FitCLIP outperforms both baselines, CLIP and Frozen, by a significant margin. It is noteworthy that CLIP, without seeing video data at training time, still outperforms Frozen by 11% at top-5 accuracy. Despite CLIP’s good performance, we observe that FitCLIP further improves performance by 4.3% (top-5) setting a new state-of-the-art in this dataset. While FitCLIP achieves outstanding zero-shot results, e.g. 44.6% top-5 accuracy, there is still an ample gap with respect to approaches that leverage supervision from the target dataset.

**Analysis on UCF101.** Table 1b shows the results on the UCF101 zero-shot benchmark. FitCLIP outperforms CLIP at Top 5 accuracy and slightly underperforms at Top 1. All the findings remain consistent: a not-so-large gap between the best zero-shot and supervised approaches and Frozen under-performing with respect to CLIP-based methods. We attribute FitCLIP and CLIP close performance (when looking at both top-1 and top-5) due to the characteristics of UCF101, which contains a lot of common actions, including many sport-related actions. These types of actions often appear in photographs, and chances are, they are well-represented in CLIP training set.

<table>
<thead>
<tr>
<th>Method</th>
<th>Top 1</th>
<th>Top 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supervised</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VATT [1]</td>
<td>41.1</td>
<td>67.7</td>
</tr>
<tr>
<td>Zero-shot</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frozen</td>
<td>14.0</td>
<td>31.8</td>
</tr>
<tr>
<td>CLIP</td>
<td>19.9</td>
<td>40.3</td>
</tr>
<tr>
<td>FitCLIP</td>
<td>21.8</td>
<td>44.6</td>
</tr>
</tbody>
</table>

(a) Moments in Time (MiT)

<table>
<thead>
<tr>
<th>Method</th>
<th>Top 1</th>
<th>Top 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supervised</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SMART [23]</td>
<td>98.6</td>
<td>–</td>
</tr>
<tr>
<td>Zero-shot</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frozen</td>
<td>51.9</td>
<td>76.1</td>
</tr>
<tr>
<td>BF [21]</td>
<td>51.1</td>
<td>–</td>
</tr>
<tr>
<td>CLIP</td>
<td>74.5</td>
<td>94.3</td>
</tr>
<tr>
<td>FitCLIP</td>
<td>73.3</td>
<td>95.3</td>
</tr>
</tbody>
</table>

(b) UCF101

Table 1: **Zero-shot action recognition results.** (a) FitCLIP improves performance upon CLIP, and significantly outperforms Frozen. (b) FitCLIP shows slight improvements upon CLIP; Frozen lags behind in terms of zero-shot performance. Reported numbers in both tables are percentages and compute the top-1 and top-5 accuracy.

5.2 Zero-shot Text-to-video Retrieval

To compare FitCLIP and the baselines, here we report the experimental results and analysis for the text-to-video retrieval task.

**Analysis on MSR-VTT.** Table 2a summarizes results in the MSR-VTT dataset. We observe that Frozen performance is poor compared to that of CLIP and FitCLIP. Even though Frozen was trained on video data with similar properties to MSR-VTT, it is hard for this model to compete with the general knowledge encoded in CLIP-like models. We observe FitCLIP consistently improves performance upon CLIP across all the retrieval metrics. These results suggest that FitCLIP captures complementary video-language information that CLIP lacks. Concerning the gap to reach the performance of the best-supervised approach, CAMoE [9].
Table 2: Zero-shot text-to-video retrieval results. In all datasets, FitCLIP improves upon CLIP by significant margins. (a) FitCLIP’s shows the best zero-shot results though there is an important gap with the supervised state of the art. (b) In this dataset, YouCook2, FitCLIP exhibits the largest gap concerning fully supervised approaches and with VideoCLIP, that pretrained on HowTo100M. We attribute this result to the fine-grained nature of the dataset. (c) FitCLIP shows consistent boosts upon CLIP even for the DiDeMo (paragraph-retrieval) task, which includes long language queries. R@k denotes recall at the top-k = {1, 5, 10} predictions, and MdR refers to the Median Ranking metric.

FitCLIP does not lag that behind. Even though there is a 16.1% gap at R@10, we see that FitCLIP closely approaches supervised performance at the MdR metric.

Analysis on YouCook2. We report zero-shot results for the YouCook2 dataset in Table 2b. From the get-go, we observe the difficulty of this dataset. Even the state-of-the-art, TACo [63], struggles to achieve more than 30% R@1. While we observe that FitCLIP’s performance consistently outperforms other zero-shot baselines, we have observed a large overall gap between our method and those that are supervised or pretrained on HowTo100M [38] (VideoCLIP in the table). We hypothesize this is due to the fine-grained nature of the language descriptions contained in YouCook2 and HowTo100M. Moreover, lots of videos in this dataset are captured from an egocentric view.

Analysis on DiDeMo. Table 2c summarizes the results in DiDeMo’s paragraph retrieval task. First, we observe that the performance of Frozen, VIOLET [17], and BridgeFormer [21] approach the one achieved by CLIP in this dataset. Contrary to other datasets, DiDeMo contains unedited, human-centric footage that shares commonalities with the WebVid dataset used to train Frozen. Conversely, FitCLIP, which leverages both: the knowledge from CLIP and the WebVid dataset, achieves the best performance overall. For completeness, we report the CAMoE’s supervised performance [9], which is 15.9% better than FitCLIP, the most competitive zero-shot alternative.

The results on these three datasets empirically demonstrate the value of FitCLIP to push the limits of zero-shot text-to-video retrieval. FitCLIP establishes a new state-of-the-art in zero-shot text-to-video retrieval across three different datasets. Despite such a milestone, there is still room for improvement, especially in fine-grained datasets such as YouCook2. We hope this benchmark promotes more work on zero-shot text-to-video retrieval.

5.3 Diagnostic Analysis

Impact of Fusing the Teacher-Student Knowledge (Table 3). One of the key properties of FitCLIP is the ability to incorporate the student learning from video data, and knowledge of the CLIP teacher. Here we report the performance of both our Student and Teacher
Table 3: **Impact of fusing teacher-student knowledge.** \( \Delta \) denotes the absolute difference in performance between FitCLIP and the Teacher model. We report the top-1 accuracy for the zero-shot action recognition datasets, and the top-5 recall for the zero-shot text-to-video retrieval ones. We observe that even though the Student model is weaker than the Teacher, it still provides complementary information to FitCLIP, yielding consistent improvements (\( \Delta \)) across datasets. Full results with all the metrics in the supplementary material.

<table>
<thead>
<tr>
<th></th>
<th>Action Recognition</th>
<th></th>
<th>Text-to-video Retrieval</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>UCF101</td>
<td>MiT</td>
<td>MSR-VTT</td>
<td>YouCook2</td>
</tr>
<tr>
<td>Teacher (CLIP)</td>
<td>74.5</td>
<td>19.9</td>
<td>55.1</td>
<td>14.6</td>
</tr>
<tr>
<td>Student</td>
<td>64.7</td>
<td>17.7</td>
<td>52.6</td>
<td>9.7</td>
</tr>
<tr>
<td>FitCLIP</td>
<td>73.3</td>
<td>21.8</td>
<td>59.8</td>
<td>15.5</td>
</tr>
<tr>
<td>( \Delta )</td>
<td>↓1.2</td>
<td>↑1.9</td>
<td>↑4.7</td>
<td>↑0.9</td>
</tr>
<tr>
<td>Err. rate red.</td>
<td>↓4.7</td>
<td>↑2.4</td>
<td>↑10.5</td>
<td>↑1.1</td>
</tr>
</tbody>
</table>

(Clip) and contrast that with the final zero-shot performance obtained with FitCLIP. Table 3 summarizes the results. We observe that although the Student’s performance remains inferior to that of the Teacher, it is close enough in various datasets, *e.g.* MiT, MSR-VTT, and DiDeMo. \( \Delta \) denotes the difference in performance between FitCLIP and the teacher and indirectly measures the contribution of the student learning. We observe that improvements are consistent across all tasks and datasets. These results suggest that the Student effectively passes complementary information to the teacher after weight assembling.

**Additional Ablations.** Due to space limitations, we include additional analysis in the *Supplementary Material.* We compare the properties of FitCLIP vs. CLIP, do a deep-dive on the impact of fusing the Teacher-Student knowledge, ablate weight-ensembling parameters, and report comparisons with additional methods trained on HowTo100M.

## 6 Conclusions

This paper presents a fine-tuning strategy to adapt large-scale image-text pre-trained models for zero-shot video understanding tasks, dubbed FitCLIP. FitCLIP performs well on zero-shot settings for three Text-to-Video Retrieval and two Action Recognition tasks that we evaluated. We show the importance of doing the weight-space ensembling step of our method to keep or improve the teacher’s robust performance across different datasets, even when the student was trained on different data. We highlight our method introduces no extra inference costs while improving CLIP results overall.

## Acknowledgements

We thank Christine Feak for revising a draft of this document. Santiago thanks Adobe Research for providing financial support to continue working on this project after the internship finished.
References


