Co-STAR: Collaborative Curriculum Self-Training with Adaptive Regularization for Source-Free Video Domain Adaptation

Supplementary Material

In this Supplementary we provide: (i) additional implementation details, including the network architecture, CLIP zero-shot video classification, and the categorized prompts used in the classification pipeline, (ii) we describe the collaborative pseudo-labeling process using the MatchOrConf scheme, (iii) detailed formulations of Pace Functions explored for curriculum learning, (iv) impact of history buffer size, (v) review of the hyperparameters used for curriculum learning and Adaptive Curriculum Regularization (ACR), along with sensitivity analyses.

9. Additional Implementation Details

Network Architecture. As mentioned in Section 3.1 of the main paper, we employ identical network architectures for both teacher and student models, each consisting of a CLIP visual encoder followed by a Temporal Relation Network (TRN). Our TRN implementation follows Zhou *et al.* [58], maintaining their bottleneck dimension of 256, but increasing the `subsample_num` from 3 to 9 to sample more frame relation combinations at each temporal scale. We use dropout with probability 0.5 after each ReLU activation for regularization. During all experiments, we keep the CLIP encoder frozen and only train the TRN module to prevent overfitting and maintain the encoder's robust visual representations.

CLIP Zero-Shot Video Classification. To generate zero-shot predictions, we leverage CLIP (ViT-B/14) [37]. Action class names are embedded using text prompts (e.g., Look, the human is {class}), and similarities between video and text embeddings are computed. These similarity scores are averaged across multiple text prompts to obtain class distribution probabilities. To reduce computational costs, these probabilities are precomputed before training.

Algorithm 1 shows the details of our zero-shot learning procedure for video classification. Similar to DALL-V [55], we utilize CLIP's vision-language alignment by encoding video frames and text templates, applying normalization, and computing scaled similarities between these modalities to obtain class probabilities. However, while DALL-V ran-

Algorithm 1 Zero-Shot video classification using CLIP

Require: Target domain $\mathcal{D}_t = \{x_i^t\}_{i=1}^{X_t}$, Temperature parameter τ_c = 0.5, CLIP visual encoder G_v, CLIP text encoder G_t

for x_i^t in $\mathcal{D}_t$ **do**
  // Extract visual features
  $v \leftarrow G_v(x_i^t)$
  $v \leftarrow \texttt{normalize}(v)$
  $v \leftarrow \texttt{avg\_p}(v)$ {Average pooling across frames}
  // Compute text-video similarities
  for t in `templates` **do**
    $z_t \leftarrow \texttt{normalize}(G_t(t))$
    $s \leftarrow \texttt{logit\_scale} \times (v \cdot z_t^\top)$
    `append` s to `similarities`
  end for
  $s_{avg} \leftarrow \texttt{mean(similarities)}$
  $p \leftarrow \texttt{softmax}(s_{avg}/\tau_c)$
end for

Ensure: `Class probabilities` p

Scene and Temporal Context
In this scene, something is {}.
In this video, {} *is happening.*
At this moment, the action being performed is {}.
During this scene, something is {}.
This video captures how something is {}.
In this context, the action being shown is {}.
This scene highlights the action of {}.
The action happening right now is {}.
Something is occurring in this scene: {}.
During this time, {} *is taking place.*
Human-Centric Actions
Look, the human is {}.
A human can be observed performing {}.
A human being is seen {} *in this video.*
The individual is currently {}.
This clip shows a person engaged in {}.
This is a human performing {}.
The person in the video is actively {}.
In this video, a human is doing {}.
The human action being performed is {}.
Here, a person is captured {}.
Questions
What is happening in this video: {}*?*
Can you tell what the action {} *is?*
What activity is being demonstrated: {}*?*
What can you observe in this scene: {}*?*
Can you recognize the action {}*?*
What is the person doing: {}*?*
What action is depicted in this video: {}*?*
Do you see what is happening here: {}*?*
What movement is being performed: {}*?*
Can you identify the activity {}*?*
Non-Human-Centric Prompts
The creature is performing {}.
This video shows a being engaged in {}.
The action being done by the creature is {}.
A creature is observed performing {}.
This clip illustrates how a creature is {}.

Table 6. Categorized prompts for CLIP Zero-Shot classification, where the action class name will replace {}.

domly selects one text template per sample [55], we leverage all text templates to compute the similarity scores, enabling more robust predictions through ensemble averaging.

Table 6 lists the 35 text templates (prompts) used in our zero-shot classification, grouped into four categories. These prompts address various scenarios, including contextual descriptions, human and non-human actions, and questions about activities. This diversity enhances the alignment between video features and text templates to improve classification performance.

10. Collabrative pseudo-labeling with MatchOrConf

As per in Section 3.1 of the main paper, we employ the MatchOrConf [56] scheme to determine the collaborative pseudo-label $\tilde{y}$.

Let $\mathcal{M}$ represent the $\arg\max$ operation, and C_s, C_c represent the confidence scores from the teacher, and CLIP, respectively. Then, if $\xi = \{\mathcal{M}(p_{\theta_s}) == \mathcal{M}(p_{\theta_c})\}$, $\tilde{y}$ is assigned as:

$$\tilde{y} = \begin{cases} \mathcal{M}(p_{\theta_s}) & \text{if } \xi, \\ \mathcal{M}(p_{\theta_s}) & \text{if } !\xi \text{ and } C_s \geq \psi_s \text{ and } C_c < \psi_c, \\ \mathcal{M}(p_{\theta_c}) & \text{if } !\xi \text{ and } C_c \geq \psi_c \text{ and } C_s < \psi_s, \\ \mathcal{M}(p_{\theta_s}) & \text{if } !\xi \text{ and } C_s \geq \psi_s \text{ and } C_c \geq \psi_c \text{ and } C_s > C_c, \\ \mathcal{M}(p_{\theta_c}) & \text{if } !\xi \text{ and } C_s \geq \psi_s \text{ and } C_c \geq \psi_c \text{ and } C_c \geq C_s, \\ -1 & \text{otherwise.} \end{cases} \tag{13}$$

where the parameters ψ_s and ψ_c are the minimum confidence thresholds for teacher and CLIP respectively. Following [38], we choose both ψ_s and ψ_c as 0.1.

11. Detailed Formulations of Pace Functions

In Section 5 of the main paper, we evaluate four different pace functions for our curriculum learning. The Exponential Pace Function is defined in Eq. 7, and here we define the rest for completion.

Previous definitions in the paper were as follows: Let r be the reliability score, $e' = e/E$ be the fraction of completed training (where e is the current epoch and E is the maximum number of epochs), and w be the output weight. For all functions, $w_{\min} = 0$ and $w_{\max} = 1$ define the allowed range of weights. The parameter $\beta = 0.6$ is used across all functions, except for the stepwise Pace Function.

Linear Pace Function. This function provides a constant rate of increase throughout training:

$$w = \text{clamp}\left[r \cdot (1 + \beta \cdot e'), w_{\min}, w_{\max}\right] \tag{14}$$

Sigmoid Pace Function. This function offers a smooth, S-shaped transition:

$$w = \text{clamp}\left[r_{\text{eqv}} + \frac{1}{1 + \exp\left(-\beta \cdot (12e' - 6)\right)}, w_{\min}, w_{\max}\right] \tag{15}$$

where $r_{\text{eqv}} = r(w_{\max} - r)$ and the fraction of completed training scaled to $[-6, 6]$ for optimal sigmoid spread.

Stepwise Pace Function. This function provides discrete transitions in sample weights:

$$w = \text{clamp}\left[r + \left(\lfloor n \cdot e' \rfloor \cdot \frac{w_{\max} - r}{n}\right), w_{\min}, w_{\max}\right] \tag{16}$$

where $n = 4$, as suggested in [46], defines the number of discrete steps for weight updates.

12. Impact of History Buffer Size

To evaluate the model's stability, we examine the impact of varying the history buffer size k, which stores historical predictions. As shown in Fig. 4, increasing the buffer size initially improves performance for both adaptation directions. The accuracy stabilizes after $k = 10$, reaching 65.0% for A→H and 32.7% for H→A, suggesting that maintaining the last 10 predictions provides sufficient information for our framework. Further increasing the buffer size does not show additional benefits.

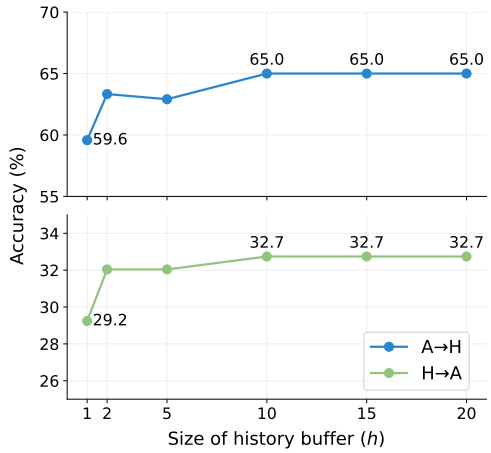

Figure 4. Impact of history buffer size h on adaptation performance. The buffer stores past predictions of the student model to assess prediction stability and stabilizes around $h = 10$.

13. Hyperparameters

The optimized hyperparameters for our curriculum learning and ACR components, determined empirically, are shown in Table 7. To evaluate their robustness, we also tested the sensitivity of these parameters across different ranges, as detailed in Table 8 for curriculum learning and Table 9 for ACR.

For these evaluations, we chose the A → H (ARID → HMDB51) setting due to its larger domain gap and shorter training time compared to other scenarios.

Component	Parameter	Optimal Value
Curriculum Learning	α (Eq. 6)	0.5
	β (Eq. 7)	0.6
ACR	η (Eq. 8)	6
	ρ (Eq. 9)	0.25
	σ (Eq. 11)	0.05
	γ (Eq. 11)	MaxConf
	λ (Eq. 12)	0.2

Table 7. Hyperparameters for Curriculum Learning and ACR.

Parameter	Range	A→H
α	0.25	57.0
	0.5	**60.8**
	0.75	59.6
β	0.2	57.1
	0.4	60.0
	0.6	**60.8**
	0.8	60.5

Table 8. Performance of different parameter ranges for Curriculum Learning. Note that the ACR component was excluded in this analysis to better evaluate the curriculum hyperparameters.

Parameter	Range	A→H
η	1	62.3
	3	63.3
	6	**65.0**
	9	60.8
ρ	0.75	62.5
	0.5	61.6
	0.25	**65.0**
	0.1	63.8
σ	0.07	63.8
	0.05	**65.0**
	0.03	62.9
λ	0.7	59.0
	0.5	62.1
	0.2	**65.0**
	0.1	64.1

Table 9. Performance of different parameter ranges for ACR.

To calculate γ (confidence threshold, Eq. 11 of the main paper), we use the maximum confidence (MaxConf) of samples where the teacher and CLIP predictions agree, i.e.,

$$\gamma = \max_{i \in \mathcal{A}} \{C_s^i\} \tag{17}$$

where $\mathcal{A} = \{i \in \{1, ..., X_t\} : \mathcal{M}(p_{\theta_s}(x_i^t)) = \mathcal{M}(p_{\theta_c}(x_i^t))\}$ represents the set of indices where teacher and CLIP predictions agree, $p_{\theta_s}(x_i^t)$ and $p_{\theta_c}(x_i^t)$ are the predicted probability distributions of the teacher and CLIP models for target sample x_i^t respectively, and C_s^i represents the confidence score of the teacher model for sample i.