# ACTIVE TEST-TIME ADAPTATION FOR TEMPORAL ACTION SEGMENTATION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Temporal action segmentation (TAS) models often suffer from performance degradation during deployment due to temporal distribution shifts and boundary ambiguity. Active test-time adaptation has shown promising results in image recognition by strategically querying sparse human feedback for adaptation. However, extending this paradigm to TAS is challenging due to long temporal span and action boundary ambiguity in procedural videos. To address these challenges, we present the first active test-time adaptation framework for TAS to strategically query sparse human annotations to ground the adaptation process. Our approach is based on a principled analysis of prediction dynamics, revealing that model hesitation under temporal smoothing constraints manifests as informative entropy peaks at action boundaries. We then propose directional boundary overwrite with agnostic and semantic variants that convert local feedback into adaptation labels, offering different trade-offs between annotation effort and supervision reliability. Our results demonstrate that annotating a few key temporal anchors significantly improves performance, establishing a new foundation for active adaptation of TAS models.

## 1 INTRODUCTION

Temporal Action Segmentation (TAS) (Farha & Gall, 2019; Yi et al., 2021; Ding et al., 2023; Liu et al., 2023) is a video understanding task that seeks to assign dense, frame-level action labels to untrimmed videos. Modern deep learning architectures (Farha & Gall, 2019; Yi et al., 2021; Liu et al., 2023) perform well, but their effectiveness often deteriorates during testing due to the distribution shift between training and deployment conditions. In TAS, this shift manifests primarily along the temporal dimension, including changes in action duration and boundary ambiguity. To bridge this gap, Test-time Adaptation (TTA) (Liang et al., 2020; Sun et al., 2020; Wang et al., 2021) has emerged as a promising solution, enabling models to adjust to new data distributions during inference. Standard TTA relies on fully unsupervised objectives, such as entropy minimization (Wang et al., 2021; Niu et al., 2022; 2023; Seto et al., 2024) or self-training with pseudo-labels (Liang et al., 2020; Hou & Zheng, 2021). However, as studied in (Gui et al., 2024), such empirical or heuristic designs can be susceptible to confirmation bias, whereby incorrect model predictions are reinforced.

Recently, active TTA (Gui et al., 2024; Chen et al., 2024) has been proposed as a test-time learning paradigm that integrates principles from active learning (Ren et al., 2021) to allow deployed models to selectively query human annotators. By requesting limited supervision under a strict annotation budget, active TTA proves to be a powerful mechanism to mitigate confirmation biases. Ultimately, the central challenge lies in maximizing adaptation performance while remaining within a minimal annotation budget. Extending active TTA to TAS requires accounting for the temporal structure of predictions and the cost of annotating long videos. Querying every frame is impractical, yet feedback at a few carefully chosen locations may help correct the segmentation beyond those locations. Moreover, annotation effort depends on both the number of queries and the information requested at each query. This paper studies how to select and use sparse human feedback for adapting TAS models at test time. We address three challenges: identifying query locations, constructing useful supervision from limited feedback, and adapting the model when most frames remain unlabeled.

The *first challenge* is to identify informative query locations among the thousands of frames in an untrimmed video. To address this, we analyze frame-level predictions under the temporal smoothing loss commonly used in TAS (Farha & Gall, 2019). The smoothing term is designed to encourage



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

temporally consistent predictions between consecutive frames, but can cause the model's "hesitation" to appear as a series of peaks in prediction entropy concentrated around action boundaries. This motivates us to exploit the uncertainty near action transitions for active annotation. To avoid explicitly locating individual local entropy peaks, we use predicted boundaries as readily identifiable approximations to their locations, yielding a compact candidate set for active querying.

The *second challenge* is how to extract effective supervision from the minimal annotation effort. Asking annotators to locate exact ground-truth boundaries entails the fine temporal localization required for dense frame-level annotation, which is time-consuming and complicated by ambiguous action transitions (Ma et al., 2020). We introduce two directional boundary overwrite schemes that convert a local judgment about a predicted boundary into adaptation labels. Specifically, our agnostic overwrite checks if the preceding action ends at the boundary and uses predicted labels for adaptation. Accordingly, we analyze boundary entropy profiles to select boundaries that provide comparatively reliable predicted labels for adaptation. In contrast, our semantic overwrite additionally corrects action label errors, requiring careful inspection of the visual context. This introduces higher cognitive load and incurs greater annotation time and effort than agnostic overwrite.

The *third challenge* is how to integrate sparse feedback for adaptation as no direct learning signal is provided for most frames of the sequence. In this regard, we apply sparse semantic supervision to the original TAS loss, *i.e.*, computing classification loss solely on the actively selected query frames while keeping the standard temporal smoothing loss unchanged. The annotations on these active query frames provide targeted supervision at uncertain or ambiguous regions. The temporal smoothing loss then propagates these corrections across the video, mitigating temporal drift and boundary misalignment without exhaustive labeling.

Building on the challenges and our proposed solutions, here are our contributions: 1) To the best of our knowledge, this work is the first to propose an active test-time adaptation framework for TAS. 2) We provide an analytical interpretation connecting temporal smoothing with entropy-based boundary selection, supporting the use of boundary frames for annotation. 3) We propose a directional boundary overwrite strategy that transforms minimal human feedback into effective supervisory signals for adaptation. 4) We demonstrate that sparse supervision at key temporal anchors is sufficient to significantly improve the robustness of TAS models during test-time adaptation.

## 2 Related Work

**Temporal Action Segmentation.** Temporal Action Segmentation (TAS) (Ding et al., 2023) has been studied under a wide range of problem settings. Prior work explores diverse backbone architectures and model designs, including convolutional and attention based frameworks such as MS-TCN (Farha & Gall, 2019), ASFormer (Yi et al., 2021), and diffusion based approaches like DiffAct (Liu et al., 2023). TAS methods also differ substantially in their supervision regimes and label space assumptions, spanning fully supervised settings (Singhania et al., 2023; Lu & Elhamifar, 2024), semi-supervised variants (Ding et al., 2025; Singhania et al., 2022), weakly supervised learning (Richard et al., 2018; Ding & Yao, 2022b), unsupervised learning (Kumar et al., 2022; Sener & Yao, 2018; Bueno-Benito & Dimiccoli, 2025), and more recently active TAS (Su & Elhamifar, 2024). Beyond supervision, several works investigate alternative input processing and training paradigms, such as incremental learning (Ding et al., 2024), dataset condensation (Ding et al., 2025; Gauthier-Villar et al., 2026), as well as online TAS for streaming scenarios (Shen & Elhamifar, 2024; Zhong et al., 2024). Despite this breadth of research, comparatively little attention has been paid to the challenges associated with test-time deployment of TAS models under temporal distribution shifts.

**Test-time Adaptation.** Test-time adaptation (TTA) updates model parameters during inference to mitigate data distribution shift on test set. Early attempts mainly recalibrate the running statistics in normalization layers (Li et al., 2017; Schneider et al., 2020). To handle more complex distribution changes, later methods update a wider part of the network using self-supervised signals (Wang et al., 2021; Niu et al., 2022; 2023; Seto et al., 2024). However, entropy minimization can produce overly sharp decision boundaries, which is undesirable for maintaining stable temporal behavior (Ding & Yao, 2022b). Other approaches impose stronger priors via additional models (Prabhudesai et al., 2023; Lin et al., 2025) or auxiliary tasks (Sun et al., 2020; Kundu et al., 2022; Gandelsman et al., 2022), but this typically introduces a more complex and computationally heavier pipeline. We tackle test-time adaptation from an active learning perspective, where a small amount of manual effort can yield

substantial gains. Following the taxonomy outlined in (Liang et al., 2025), our setup corresponds precisely to test-time domain adaptation where adaptation is performed on full test data prior to evaluation. Several works (Gui et al., 2024; Lee et al., 2025) study active TTA, but mainly focuse on querying class labels for images. In contrast, we introduce the setup to TAS, where the long temporal span and ambiguity of actions pose additional challenges.

# 3 Active Test-time Adaptation for TAS

## 3.1 Task Definition

**Temporal Action Segmentation.** Let $\mathcal{M}$ denote a pre-trained temporal action segmentation model that predicts frame-level action labels for an untrimmed procedural video $v = \{x^1, x^2, \ldots, x^T\}$, where $T$ is the total number of frames. The model is trained under a per-frame classification formulation with the corresponding loss defined as:

$$\mathcal{L}_{\text{cls}}(x, y) = \frac{1}{T} \sum_{t} -\log(p^t(y^t)), \tag{1}$$

where $p^t(\cdot) = \mathcal{M}(x^t)$ denotes the predicted class probability at frame $t$ and $y^t \in [1, ..., C]$ is the corresponding action label. To encourage temporal consistency between consecutive predictions, a temporal smoothing loss is additionally employed:

$$\mathcal{L}_{\text{sm}}(x) = \frac{1}{TC} \sum_{t,c} \tilde{\Delta}^2_{t,c}, \quad \tilde{\Delta}_{t,c} = \begin{cases} \Delta_{t,c} : \Delta_{t,c} \leq \tau \\ \tau \quad : \text{otherwise} \end{cases} \quad \text{and} \quad \Delta_{t,c} = \left| \log p^t(c) - \log p^{t-1}(c) \right|, \tag{2}$$

where the truncation parameter $\tau$ is typically set to 4, as per (Farha & Gall, 2019). The overall training objective combines the classification and smoothing losses with a trade-off parameter $\lambda$:

$$\mathcal{L}_{\text{tas}} = \mathcal{L}_{\text{cls}}(x, y) + \lambda \mathcal{L}_{\text{sm}}(x). \tag{3}$$

**Active Test-time Adaptation Setting.** Active test-time adaptation aims to update the model $\mathcal{M}$ during inference in order to improve segmentation performance on test video sequences, which typically differ in distribution from the training data. In this setting, the model is allowed to actively query a limited number of annotations under a constrained labeling budget. Concretely, the model $\mathcal{M}$ first generates prediction labels $y_{\text{pred}}$ from its initial outputs on the test videos. These labels will then be refined to form adaptation labels as $y_{\text{adpt}}$, while a small set of queried annotations provide ground-truth labels $y_{\text{gt}}$ that correct errors and anchor the adaptation process. Adaptation is performed over the full test sequences prior to evaluation. This setting introduces key questions on how to effectively acquire and exploit limited supervision for adaptation.

In this work, we study the task by addressing three fundamental questions: i) which frames should be selected for annotation (Sec. 3.2), ii) how the selected frames should be labeled (Sec. 3.3), and iii) how the acquired supervision should be incorporated into the learning process to facilitate adaptation (Sec. 3.4).

## 3.2 Selection via Smoothing Loss Analysis

We begin with an analytical examination of the unique smoothing regularization term in TAS and its implications for deciding which frames should be selected for active labeling,

**Smooth Temporal Transition.** For intuition, we simplify the smoothing term in Eq. (2) by omitting the logarithmic transformation and truncation parameter $\tau$: The function is written as:

$$\mathcal{L}_{\text{sm}} = \sum_{t=1}^{T} ||p^t - p^{t-1}||^2. \tag{4}$$

During optimization, the partial derivative of $\mathcal{L}_{\text{sm}}$ with respect to the probability vector $p^t$ is derived as follows:

$$\frac{\partial \mathcal{L}_{\text{sm}}}{\partial p^t} = \frac{\partial}{\partial p^t} \left[ \|p^t - p^{t-1}\|^2 + \|p^{t+1} - p^t\|^2 \right] = -2 \, [p^{t+1} - 2p^t + p^{t-1}]. \tag{5}$$

Setting this derivative to zero yields the following equilibrium condition:

$$p^t = \frac{1}{2}\left(p^{t+1} + p^{t-1}\right). \tag{6}$$

This demonstrates that the optimal value for $p^t$ is the arithmetic mean of its immediate temporal neighbors, where a smooth transition across frames is induced by a zero second-order temporal derivative. Consequently, across a transition between two actions, smoothing can produce a gradual change in class probabilities, as illustrated in Fig. 1. A similar transitional effect has also been empirically observed and leveraged by (Ding & Yao, 2022a).

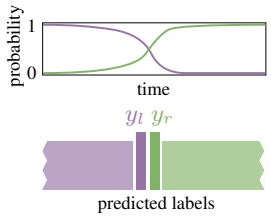



Figure 1: Visualization of the smooth transition effect under the smoothing loss. The top plots the evolution of class probabilities over time, where the smoothing loss encourages a gradual crossover between the outgoing and incoming actions. The bottom represents the resulting predicted labels.

**Entropy near a Transition.** For any frame $t$ located within the transition region between the segment of two consecutive actions $y_l$ and $y_r$,[1] the predicted class probability under the smoothing constraint can be approximated as:

$$p^t = \alpha(t)\, y_l + (1 - \alpha(t))\, y_r, \quad \alpha(t) \in [0, 1]. \tag{7}$$

Here, with a slight abuse of notation, we let $\{y_l, y_r\} \in \mathbb{R}^C$ denote the one-hot vector representation of the corresponding action classes. Under the effect of the smoothing loss, the mixing coefficient $\alpha(t)$ tends to vary monotonically across the region and typically approaches $0.5$ at the boundary. The frame-wise entropy therefore reduces to a binary form given by:

$$H(t) = -\alpha(t)\log\alpha(t) - (1 - \alpha(t))\log(1 - \alpha(t)). \tag{8}$$

Accordingly, for the frame $t_b$ attaining the maximum entropy, *i.e.*, $t_b = \arg\max_t H(t)$, we expect either $y^{t_b} \neq y^{t_b+1}$ or $y^{t_b-1} \neq y^{t_b}$, indicating that $t_b$ is immediately adjacent to an action boundary, either before or after the transition. This observation that the local entropy peaks tend to coincide with action boundaries motivates our selection of boundary frames for annotation.

In practice, action boundaries can be directly identified from the predicted label sequence, making boundary detection straightforward and eliminating the need for entropy peak estimation. For ease of notation, we specify $t_b$ as the first frame immediately following the change in label prediction:

$$t_b = \{t \mid y^{t-1} \neq y^t\}. \tag{9}$$

Accordingly, the candidate set for active query is $\mathcal{B} = \{t_b^n\}_{n=1}^N$ with $N$ boundaries where $N \ll T$.

### 3.3 Labeling by Directional Boundary Overwrite

For a selected boundary $t_b \in \mathcal{B}$, the initial prediction assigns different action labels to frames $t_b - 1$ and $t_b$. We convert feedback about this transition into adaptation labels using the directional boundary overwrite in Fig. 2. From here on, we denote the initial predicted labels on test video as $y_{\text{pred}}$, the labels used for adaptation as $y_{\text{adpt}}$ and the ground truth action label as $y_{\text{gt}}$.

**Agnostic Overwrite.** The agnostic query asks which predicted action should occupy both frames adjacent to the boundary, without asking the annotator to name an action class. Specifically, let $o(t_b) \in \{0, 1\}$ denote whether the left action $y_l$ terminates at its last frame, $t_b - 1$. $o(t_b) = 0$ indicates that the left action should extend forward to $t_b$; while $o(t_b) = 1$ indicates that the right action should extend backward to $t_b - 1$. Using this oracle, we first perform an agnostic overwrite, which essentially propagates the label across the boundary based solely on temporal direction (Fig. 2①). We define the propagated label for overwrite as:

$$\mathcal{O}_{\text{agn}} = (1 - o(t_b))\, y_{\text{pred}}^{t_b-1} + o(t_b)\, y_{\text{pred}}^{t_b}, \tag{10}$$

and overwrite the labels of both adjacent frames at the boundary with the propagated value:

$$y_{\text{adpt}}^{t_b-1} = y_{\text{adpt}}^{t_b} = \mathcal{O}_{\text{agn}}, \tag{11}$$

[1]The two-class case is used because it matches how ground truth is defined in TAS: each boundary marks a transition from one action class to another. More discussions are provided in Appendix A.

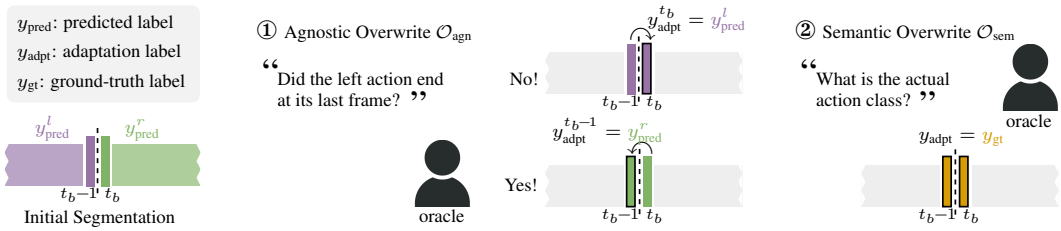



Figure 2: Overview of the proposed directional boundary overwrite for labeling. Given the initial predicted boundary between two action labels ($y^l_{\text{pred}}$ and $y^r_{\text{pred}}$), a refinement to frames at $t_b$–1 and $t_b$ can be achieved via oracle queries: ① Agnostic overwrite resolves temporal ambiguity by checking if the left action has concluded. A “No” triggers a one-step forward adaptation label overwrite using the preceding label ($y^{t_b}_{\text{adpt}} = y^l_{\text{pred}}$); otherwise, the left frame is one-step backward overwritten by the succeeding label ($y^{t_b-1}_{\text{adpt}} = y^r_{\text{pred}}$). ② Semantic overwrite addresses classification errors by querying the oracle the true action class for adaptation ($y_{\text{adpt}} = y_{\text{gt}}$), resulting in a definitive label correction for the boundary. These adaptation labels are used to supervise the adaptation.

which realizes a one-step left/right shift of the predicted label along the temporal axis. The intuition that this benefits adaptation is because assigning a shared label to two boundary-adjacent frames provides a local consistency signal that guides the model to learn more discriminative feature representations and improved overall class separability.

**Reliability of $\mathcal{O}_{\text{agn}}$.** Agnostic overwrite uses predicted action classes $y_{\text{pred}}$ as adaptation labels, which can be noisy. We probe this by the entropy ranking of predicted boundaries. With a slight abuse of notation, let $\mathcal{B} = (t_b^1, t_b^2, \ldots, t_b^N)$ denote the $N$ predicted boundaries of a test video ordered by increasing entropy, such that $H(t_b^1) \leq H(t_b^2) \leq \cdots \leq H(t_b^N)$. We refer to boundaries in the lower, middle, and upper portions of this per-video ranking as low-, medium-, and high-entropy boundaries, respectively. For agnostic overwrite to be useful, a boundary should offer scope for temporal correction while its predicted action labels remain sufficiently reliable to propagate. The three entropy ranges reflect different balances between these two factors:

**- Low entropy**: Relative to other predicted boundaries in the video, these predictions are more concentrated on a dominant class. The class used for overwrite may be more reliable, but there may be less boundary uncertainty for a directional correction to resolve.

**- Medium entropy**: These boundaries show intermediate predictive uncertainty relative to the other boundaries in the video. They may provide scope for correcting the transition while retaining sufficiently reliable predicted classes for overwrite.

**- High entropy**: These boundaries have the most diffuse predictions within the video. Although they may offer scope for correction, uncertainty about the predicted action classes may make their overwrite labels less reliable.

These characteristics suggest that medium-entropy boundaries are better suited for agnostic overwrite: they provide scope for temporal correction while retaining comparatively reliable semantic labels. Given a query ratio $\eta$, we select $\eta N$ boundaries from the middle of the ranking list. We provide additional visualizations and discussions in Appendix A.

**Semantic Overwrite.** Agnostic overwrite cannot correct an erroneous action identity because it propagates one of the model's predicted labels. Semantic overwrite addresses this limitation by additionally asking the annotator for the action class to propagate across the selected boundary, as illustrated in Fig. 2②. Given the directional response $o(t_b)$, the propagated label is

$$\mathcal{O}_{\text{sem}} = y_{\text{gt}}^{t_b-1+o(t_b)}, \tag{12}$$

where $y^t_{\text{gt}}$ is the ground-truth action label at frame $t$, and assign:

$$y^{t_b-1}_{\text{adpt}} = y^{t_b}_{\text{adpt}} = \mathcal{O}_{\text{sem}}. \tag{13}$$

**Annotation Efforts.** In this work, we query the same selected boundary set $\mathcal{B}_{\text{sel}} \subseteq \mathcal{B}$ for both overwrite schemes. Given a selection ratio $\eta$, each scheme requests one annotation at each of $|\mathcal{B}_{\text{sel}}| = \lfloor \eta |\mathcal{B}| \rfloor$ boundaries. The corresponding frame query ratio is $\beta = |\mathcal{B}_{\text{sel}}|/|\mathcal{Y}|$, where $|\mathcal{Y}|$ is the

**Algorithm 1** Active Test-time Adaptation for TAS

**Input:** Model $\mathcal{M}$, test video $v$, budget $\eta$
**Output:** Adapted model $\mathcal{M}'$

Segment $p^{1:T} = \mathcal{M}(v)$ ▷ *initial inference*
Detect boundaries $\mathcal{B}$ via Eq. (9) ▷ *frame selection*
Rank boundary $t_b$ from $\mathcal{B}$ by entropy $H(t_b)$
Construct $\mathcal{B}_{\text{sel}}$ of size $\eta|\mathcal{B}|$
**for** $t_b \in \mathcal{B}_{\text{sel}}$ **do** ▷ *boundary overwrite*
    Query oracle $o(t_b) \in \{0, 1\}$,
    Get $y_{\text{adpt}}$ by $\begin{cases}\text{agnostic overwrite via Eq. (10)}\\ \text{semantic overwrite via Eq. (12)}\end{cases}$
**end for**
**for** $e \in \{1, \cdots, E\}$ **do** ▷ *model adaptation*
    Update $\mathcal{M}'$ with $y_{\text{adpt}}$ via Eq. (14)
**end for**

total number of frames. However, two schemes differ in the information requested from the annotator. Agnostic overwrite asks only how the boundary should move and inherits the action identity from the model. Semantic overwrite obtains that identity from the annotator, allowing it to correct class errors but requiring a more involved judgment for each query. Thus, equal numbers of queried boundaries do not imply equal human effort. We measure this difference in annotation time in Sec. 4.3.

### 3.4 Adaptation with Sparse Semantic Supervision

During test-time adaptation, the model parameters of $\mathcal{M}$ are updated using the sparse supervision derived from directional boundary overwrite. Algorithm 1 summarizes the overall pipeline. We employ the same training objective as Eq. (3), but apply the classification loss to the boundary-adjacent frames with directional overwrite, while enforcing the smoothing loss over all frames. Specifically:

$$\mathcal{L}_{\text{adpt}} = \mathcal{L}_{\text{cls}}(x^{\{t_b-1,t_b\}}, y_{\text{adpt}}^{\{t_b-1,t_b\}}) + \lambda \mathcal{L}_{\text{sm}}(x), \quad t_b \in \mathcal{B}_{\text{sel}}, \tag{14}$$

where $t_b$ denotes selected boundary and $y_{\text{adpt}}$ the updated labels.

**Remarks.** Our sparse semantic supervision applies the cross-entropy loss only to actively labeled boundary frames for two reasons. First, it ensures that supervision is based on more reliable or manually verified labels, avoiding the risk of reinforcing errors from noisy pseudo labels in non-boundary frames. Second, the smoothing loss effectively propagates information from boundary frames to their neighbors, a detailed illustration is provided in Appendix B.

## 4 Experiments

### 4.1 Evaluation Settings

**Datasets.** We evaluate our approach on three common TAS benchmarks. **Breakfast** (Kuehne et al., 2014) comprises 1,712 breakfast preparation videos. There are 10 activities and a total of 48 action classes. **GTEA** (Fathi et al., 2011) contains 28 videos of seven kitchen activities composing 11 different actions. **50Salads** (Stein & McKenna, 2013) includes 50 videos of salad preparation with 19 action classes, characterized by long action sequences. We use standard pre-computed I3D (Carreira & Zisserman, 2017) features as inputs.

**Evaluation Measures.** We adopt commonly used evaluation metrics, covering both frame-level classification accuracy and segment-level temporal consistency. We report frame-wise accuracy (**Acc**) to measure per-frame labeling correctness, the edit score (**Edit**) and the **F1** scores under thresholds of 10%, 25%, and 50% to quantify the quality at segment level.

**Adaptation Protocols.** Adaptation is performed on the test set of each dataset while respecting the original training and evaluation splits, with results reported as averaged over standard splits. The total query budget for each dataset is determined by the number of predicted boundaries $|\mathcal{B}|$

Table 1: Performance comparison of our framework for active TTA across three benchmarks using two common backbones. Best performances are highlighted in bold for each setup ($\mathcal{O}_{\text{agn}}$ and $\mathcal{O}_{\text{sem}}$).

| | $\mathcal{O}_{\text{agn}}$ | $\mathcal{O}_{\text{sem}}$ | Breakfast (Kuehne et al., 2014) | | | | | GTEA (Fathi et al., 2011) | | | | | 50Salads (Stein & McKenna, 2013) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Acc | Edit | F1 @ {10, 25, 50} | | | Acc | Edit | F1 @ {10, 25, 50} | | | Acc | Edit | F1 @ {10, 25, 50} | | |
| MS-TCN (Farha & Gall, 2019) | | | | | | | | | | | | | | | | | |
| Baseline | - | - | 65.4 | 67.5 | 68.5 | 63.0 | 49.6 | 74.1 | 78.3 | 83.1 | 80.7 | 67.1 | 74.9 | 59.5 | 64.0 | 60.4 | 51.8 |
| Label ratio $\beta(\eta\|\mathcal{B}\|/\|\mathcal{Y}\|)$ | | | 0.15% | | | | | 1.25% | | | | | 0.20% | | | | |
| Equi (Rana & Rawat, 2023) | ✓ | - | 59.8 | 62.0 | 61.6 | 54.5 | 40.0 | 71.0 | 76.3 | 80.8 | 77.7 | 62.1 | 71.8 | **64.0** | **69.3** | **66.1** | 54.1 |
| Coreset (Sener & Savarese, 2017) | ✓ | - | 64.2 | 65.9 | 64.2 | 55.4 | 39.5 | 71.2 | 74.2 | 81.2 | 77.8 | 62.0 | 75.7 | 62.5 | 68.3 | 64.8 | **56.7** |
| Ours | ✓ | - | **73.7** | **73.2** | **74.3** | **69.9** | **57.6** | **75.1** | **78.9** | **82.3** | **80.0** | **67.4** | **76.3** | 62.7 | 68.2 | 64.6 | 56.1 |
| Equi (Rana & Rawat, 2023) | - | ✓ | 70.1 | 67.9 | 67.5 | 60.1 | 45.7 | **75.5** | 77.4 | 80.1 | 78.0 | 64.8 | **76.9** | **63.8** | 69.4 | 65.7 | **56.7** |
| Coreset (Sener & Savarese, 2017) | - | ✓ | 69.3 | 70.8 | 70.3 | 64.8 | 51.4 | 74.6 | 75.0 | 79.0 | 75.5 | 64.5 | 75.9 | 63.4 | 67.7 | 63.4 | 54.9 |
| Ours | - | ✓ | **75.7** | **74.7** | **75.8** | **71.6** | **60.4** | 75.3 | **78.9** | 82.4 | **80.1** | **67.5** | 76.7 | 63.5 | **69.6** | **66.2** | 56.4 |
| SimATTA (Gui et al., 2024) | - | ✓ | 74.1 | 73.5 | 74.1 | 70.8 | 58.9 | 74.4 | 77.7 | 81.6 | 79.8 | 63.6 | 75.8 | 62.6 | 68.4 | 65.1 | 55.9 |
| BiTTA (Lee et al., 2025) | - | ✓ | 65.9 | 69.8 | 71.4 | 65.5 | 52.7 | 73.9 | 78.4 | **82.6** | **80.1** | 67.1 | 74.9 | 60.8 | 65.8 | 62.4 | 53.1 |
| Label ratio | | | 100% | | | | | 100% | | | | | 100% | | | | |
| Full | - | - | 84.5 | 82.3 | 84.8 | 81.6 | 72.6 | 80.9 | 82.3 | 86.6 | 84.7 | 72.1 | 79.6 | 65.5 | 71.3 | 68.2 | 59.6 |
| ASFormer (Yi et al., 2021) | | | | | | | | | | | | | | | | | |
| Baseline | - | - | 72.8 | 75.2 | 76.2 | 71.0 | 58.9 | 76.5 | 82.5 | 87.7 | 84.9 | 74.5 | 76.7 | 69.3 | 74.7 | 70.1 | 60.9 |
| Label ratio $\beta(\eta\|\mathcal{B}\|/\|\mathcal{Y}\|)$ | | | 0.16% | | | | | 1.45% | | | | | 0.18% | | | | |
| Equi (Rana & Rawat, 2023) | ✓ | - | 64.2 | 65.9 | 64.2 | 55.4 | 39.5 | 71.8 | 80.8 | 85.3 | 81.7 | 66.3 | 74.4 | 67.2 | 75.3 | 73.1 | 60.9 |
| Coreset (Sener & Savarese, 2017) | ✓ | - | 71.0 | 73.9 | 74.4 | 68.2 | 53.9 | 73.6 | 83.1 | 87.3 | 83.1 | 69.9 | 74.9 | 69.4 | 76.5 | 73.2 | 61.8 |
| Ours | ✓ | - | **77.3** | **76.2** | **79.0** | **74.2** | **61.5** | **75.3** | **84.8** | **89.1** | **87.3** | **73.7** | **80.5** | **73.1** | **80.2** | **77.4** | **67.7** |
| Equi (Rana & Rawat, 2023) | - | ✓ | 76.3 | 73.1 | 74.1 | 66.8 | 51.3 | 74.4 | 82.2 | 86.5 | 84.1 | 71.1 | 80.7 | 74.0 | 80.2 | 76.6 | 66.7 |
| Coreset (Sener & Savarese, 2017) | - | ✓ | 76.8 | 75.9 | 79.0 | 73.3 | 60.0 | 75.1 | 84.8 | 88.8 | 85.9 | 73.4 | 80.7 | **74.5** | 80.1 | 76.9 | 67.5 |
| Ours | - | ✓ | **79.1** | **77.4** | **80.2** | **76.0** | **63.0** | **79.6** | **85.5** | **90.1** | **88.4** | **77.2** | **81.4** | 74.1 | **80.9** | **78.8** | **68.0** |
| SimATTA (Gui et al., 2024) | - | ✓ | 75.5 | 76.6 | 79.3 | 74.5 | 61.2 | 77.8 | 83.3 | 89.5 | 87.4 | 75.1 | 80.6 | 74.0 | 80.5 | 77.9 | 67.5 |
| BiTTA (Lee et al., 2025) | - | ✓ | 73.0 | 75.5 | 77.0 | 71.4 | 59.3 | 75.6 | 83.1 | 88.4 | 85.6 | 74.0 | 77.4 | 70.5 | 76.7 | 73.6 | 62.1 |
| Label ratio | | | 100% | | | | | 100% | | | | | 100% | | | | |
| Full | - | - | 84.6 | 84.8 | 85.8 | 82.4 | 73.7 | 92.6 | 95.5 | 97.3 | 97.1 | 94.4 | 87.2 | 80.1 | 85.5 | 82.4 | 75.5 |

produced during the initial segmentation stage, which is backbone dependent. In practice, with the selection ratio $\eta = 0.5$, we work with very low annotation ratios of approximately $\beta = 0.15\%$ for Breakfast, $1.25\%$ for GTEA and $0.2\%$ for 50Salads with MS-TCN (Farha & Gall, 2019). While for ASFormer (Yi et al., 2021), they are $0.16\%$, $1.45\%$ and $0.18\%$, respectively.

**Implementation.** We integrate our approach with MS-TCN (Farha & Gall, 2019) and ASFormer (Yi et al., 2021) as backbones. The adaptation learning rate is kept identical to that used during original model training as per (Farha & Gall, 2019; Yi et al., 2021), except for GTEA and 50Salads with MS-TCN, where it is set to a smaller rate of $1e{-}5$ to stabilize the adaptation. The adaptation epoch is set to $E = 5$ for all datasets. The adaptation is conducted on full set of test videos prior to final evaluation. All experiments are conducted on a single NVIDIA A40 GPU.

**Baselines.** As the first work on active TTA for TAS, direct comparisons are limited. We therefore adapt data selection methods, Equi (Rana & Rawat, 2023) and Coreset (Sener & Savarese, 2017) and combine with our overwrite strategies. We further adapt active TTA methods from image domain, SimATTA (Gui et al., 2024) and BiTTA (Lee et al., 2025) to match our setting. For fair comparison, we select exactly the same number of frames for each method, matching the selection budget $\beta$ (or $\eta|\mathcal{B}|/|\mathcal{Y}|$). We also include an upper-bound setting using full ground truth frame labels for adaptation. More details are provided in Appendix C.

### 4.2 Effectiveness

Table 1 summarizes the quantitative results on common TAS benchmarks using two representative backbones. We make the following observations.

**Frame Selection Strategy.** Across all three datasets and both segmentation backbones, our approach shows consistent performance gains over the Equi (Rana & Rawat, 2023) and Coreset (Sener & Savarese, 2017) baselines under the same agnostic overwrite setting. Specifically, on the Breakfast dataset with MS-TCN (Farha & Gall, 2019), our method reaches 73.7% accuracy, substantially outperforming Equi (59.8%) and Coreset (64.2%). Likewise, on GTEA with ASFormer, we observe boosts in edit score of 4.0% (80.8% to 84.8%) and 1.7% (83.1% to 84.8%). In some cases, approaches like Equi and Coreset tend to underperform when combined with the agnostic overwrite, and can even degrade performance relative to the original baseline. On Breakfast with MS-TCN, accuracy

drops from 65.4% to 59.8% with Equi and to 64.2% with Coreset. A similar pattern occurs with ASFormer (Yi et al., 2021), where Equi reduces accuracy from a 72.8% baseline to 64.2%. This degradation likely occurs because these methods either rely on fixed geometric distributions or feature diversity rather than the model's actual predictive state. As a result, they tend to select low-information frames which, when overwritten without semantic, may introduce label noise rather than refinement. An exception is Equi's competitive performance on 50Salads with MS-TCN. One possible explanation is that 50Salads has lower variances in segment lengths (0.03) compared with Breakfast (0.14). This may allow Equi's uniformly spaced queries to cover more action segments. Other than that, our method circumvents this degradation and achieves a substantial performance boost. Furthermore, agnostic overwrite with weaker signals already matches or surpasses Equi and Coreset with semantic labels.

**Comparison with existing active TTA methods.** With semantic annotation provided, our approach consistently outperforms BiTTA (Lee et al., 2025) and SimATTA (Gui et al., 2024) across multiple benchmarks and different backbones, demonstrating the effectiveness and generalizability of our design. We further observe two additional findings. First, BiTTA (Lee et al., 2025) only achieves performance comparable to the baseline, suggesting that the gains from queried frames are limited. In fact, other than queried supervision, BiTTA also relies on "stable" pseudo-labels, which may constrain the temporal propagation of supervision and consequently weaken the effect of active queries (see Appendix C for more analysis). Secondly, SimATTA (Gui et al., 2024) clusters frames according to prediction entropy without leveraging any pseudo-labels, and consistently outperforms BiTTA by a clear margin. However, it remains inferior to our semantic overwrite strategy, further validating the effectiveness of our boundary-centric active query design. On a different note, the consistent gains in Edit score suggest that sparse supervision improves the predicted action sequence beyond the queried frames.

### 4.3 Ablation and Analysis

**Agnostic vs Semantic Annotation.** Our annotation user study in Table 2 shows that semantic annotation on average takes 1.8× longer than agnostic annotation (16.3s vs. 9.0s) per candidate frame and has higher variance (4.7s vs. 1.4s). This difference indicates greater cognitive load and lower consistency across annotators (more details on user study are provided in Appendix D). Despite the higher cost, semantic supervision yields limited additional benefit. For instance, on Breakfast with MS-TCN (see Table 1), agnostic overwrite improves average performance by 7%, whereas semantic overwrite provides only a further 1.9% gain, highlighting agnostic overwrite as a more cost-effective option.

Table 2: User study on annotation cost for two different settings. Semantic annotation requires longer time and higher cognitive load than agnostic annotation.

| | $\mathcal{O}_{\text{agn}}$ | $\mathcal{O}_{\text{sem}}$ |
|---|---|---|
| Time (s) | $9.0 \pm 1.4$ | $16.3 \pm 4.7$ |

**Performance vs Budget.** To better characterize the trade-off between performance gains and annotation budget, we study the effect of the selection ratio $\eta$ on both agnostic ($\mathcal{O}_{\text{agn}}$) and semantic ($\mathcal{O}_{\text{sem}}$) overwrite strategies using the ASFormer backbone on 50Salads dataset. The results in Table 3 reveal several trends. For the agnostic overwrite, increasing $\eta$ from 0.25 to 0.5 yields substantial and consistent improvements across all evaluation metrics. On average, this corresponds to an accuracy gain of around 5%, indicating that a moderate increase in boundary annotations is sufficient to realize most of the benefits of agnostic adaptation. However, further increasing the selection ratio to $\eta = 1$ does not lead to consistent gains. Based on these observations, we adopted $\eta = 0.5$ as the default ratio, which offers a favorable balance between performance improvement and annotation efficiency.

Table 3: Performance on 50Salads under varying $\eta$.

| $\eta$ | $\beta$ | $O_{\text{agn}}$ | $O_{\text{sem}}$ | Acc | Edit | F1 @ {10, 25, 50} |
|---|---|---|---|---|---|---|
| 0.25 | 0.09% | ✓ | - | 75.3 | 68.7 | 75.4 / 72.7 / 62.0 |
| 0.5 | 0.18% | ✓ | - | **80.5** | **73.1** | **80.2 / 77.4 / 67.7** |
| 1.0 | 0.35% | ✓ | - | 80.4 | 72.2 | 79.8 / 76.7 / 67.5 |
| 0.25 | 0.09% | - | ✓ | 76.8 | 69.8 | 77.4 / 74.7 / 63.4 |
| 0.5 | 0.18% | - | ✓ | 81.4 | 74.1 | 80.9 / 78.8 / 68.0 |
| 1.0 | 0.35% | - | ✓ | **85.3** | **77.5** | **83.8 / 82.3 / 73.4** |

In contrast, semantic overwrite has a stronger dependence on the annotation budget. As $\eta$ increases, $\mathcal{O}_{\text{sem}}$ surpasses its agnostic counterpart, with significant improvements when all boundary frames are

Table 4: Impact of entropy range on agnostic overwrite on Breakfast dataset. Selecting the medium-range boundaries achieves the best performance.

| Low→Medium→High | Acc | Edit | F1 @ {10, 25, 50} | | |
|---|---|---|---|---|---|
| | 65.4 | 67.5 | 68.5 | 63.0 | 49.6 |
| 0.5 | 71.3 | 70.3 | 71.9 | 66.6 | 53.8 |
| 0.5 | **73.7** | **73.2** | **74.3** | **69.9** | **57.6** |
| 0.5 | 72.0 | 70.5 | 72.7 | 67.8 | 55.1 |
| 1 | 71.7 | 71.3 | 72.8 | 69.0 | 57.6 |

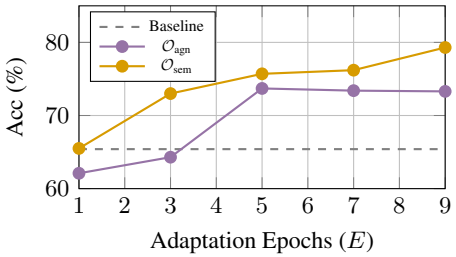



Figure 3: Accuracy on Breakfast with MS-TCN vs adaptation epochs.

annotated. These results suggest that semantic supervision scales effectively with budget, as explicit class-level corrections can resolve misclassifications that agnostic overwrite alone cannot resolve.

In conclusion, we recommend setting $\eta = 0.5$ for agnostic overwrite, as it delivers strong gains with minimal annotation effort. When higher accuracy is desired and annotation resources allow, increasing $\eta$ and adopting semantic overwrite constitutes a reliable strategy for achieving additional performance gains.

**Entropy Ranges for Selection.** To examine how entropy-based boundary selection influences the agnostic overwrite, we conduct a sensitivity analysis in Table 4. Specifically, boundary frames are ranked on a per-video basis according to their entropy values, and a fixed budget $\eta = 0.5$ is allocated to select frames from the ranked list. Overall, agnostic overwrite consistently outperforms the no-adaptation baseline (65.4% Acc) regardless of which portion of the ranked list is selected. However, the magnitude of the improvement can be affected by the targeted entropy range. Among the considered settings, selecting frames from the medium-entropy region, corresponding to the middle 50% of the ranked list, yields the most pronounced gains. This configuration achieves 73.7% in Acc and 73.2% in Edit, representing the best performance across all reported metrics.

A comparison of the distribution extremes further reveals that prioritizing high-entropy frames is more effective than selecting low-entropy ones, with 72.0% versus 71.3% in Acc, respectively. This is because higher-entropy boundaries provide greater scope for adaptation, whereas low-entropy boundaries tend to be relatively well modeled in both temporal alignment and semantic prediction. In contrast, overwriting all boundary frames ($\eta = 1$, last row) leads to a slight performance degradation relative to the mid-range selection, indicating that overwriting boundaries with high entropy may introduce less reliable supervision. This observation is consistent with our analysis in Sec. 3.2, which suggests frames with medium entropy are most suitable for agnostic overwrite.

**Adaptation Epochs.** Fig. 3 presents the Acc on Breakfast dataset with the MS-TCN backbone, plotted against the number of adaptation epochs ($E$). For $\mathcal{O}_{\text{agn}}$, Acc in the initial epochs falls below the baseline, but rises sharply by the 5-th epoch. This likely reflects the time required for the sparse training objective to take effect and meaningfully adjust boundaries. The performance peaks at the 5-th epoch and stabilizes thereafter. In contrast, $\mathcal{O}_{\text{sem}}$ steadily improves over the baseline throughout. Considering these trends and the efficiency of test-time adaptation, we set the total number of adaptation epochs to 5.

## 5 Conclusion

This paper presents the first active TTA framework for temporal action segmentation that leverages sparse feedback at boundaries. By analyzing the relationship between temporal smoothing loss and model predictive entropy, we first identify boundary frames as query candidates that capture temporal uncertainty. We then propose directional boundary overwrite with agnostic and semantic variants that offer different trade-offs between annotation effort and supervision reliability. Combined with sparse semantic supervision and temporal smoothing loss, these local corrections can propagate across the sequence, enhancing temporal alignment without the need for dense labeling. Experiments across common benchmarks and backbones demonstrate that sparse boundary feedback can substantially improve segmentation performance, supporting the efficiency and practicality of boundary-centric active adaptation of action segmentation models during deployment.

## AI USE STATEMENT

Generative AI tools were used to assist with drafting and refining the manuscript, including improving clarity, organization, and language, and to support literature discovery by identifying potentially relevant related work. The authors reviewed and edited the generated text, verified the cited sources, and take full responsibility for the content and accuracy of the paper.

## ETHICS STATEMENT

This work studies human-assisted test-time adaptation for temporal action segmentation using established research benchmarks. The adaptation experiments simulate annotator feedback using existing ground-truth annotations. We also conduct a user study to compare the time required for agnostic and semantic annotation. These timing measurements characterize annotation effort; they do not establish the reliability of human feedback in deployment. Details about the user study are provided in Appendix D.

## REPRODUCIBILITY STATEMENT

Section 3 describes the query selection procedure, annotation schemes, and adaptation objective, with the overall pipeline summarized in Algorithm 1. Implementation details, including learning rates, query ratios, adaptation epochs, and computational hardware, are provided in Sec. 4.1 to ensure reproducibility. Appendix C additionally provides details of the adapted baselines, while Appendix D describes the annotation user study and its interfaces.

## REFERENCES

Elena Bueno-Benito and Mariella Dimiccoli. Clot: Closed loop optimal transport for unsupervised action segmentation. In *ICCV*, 2025.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.

Yaofo Chen, Shuaicheng Niu, Yaowei Wang, Shoukai Xu, Hengjie Song, and Mingkui Tan. Towards robust and efficient cloud-edge elastic model adaptation via selective entropy distillation. In *ICLR*, 2024.

Guodong Ding and Angela Yao. Leveraging action affinity and continuity for semi-supervised temporal action segmentation. In *ECCV*, 2022a.

Guodong Ding and Angela Yao. Temporal action segmentation with high-level complex activity labels. *TMM*, 2022b.

Guodong Ding, Fadime Sener, and Angela Yao. Temporal action segmentation: An analysis of modern techniques. *IEEE TPAMI*, 2023.

Guodong Ding, Hans Golong, and Angela Yao. Coherent temporal synthesis for incremental action segmentation. In *CVPR*, 2024.

Guodong Ding, Rongyu Chen, and Angela Yao. Condensing action segmentation datasets via generative network inversion. In *CVPR*, 2025.

Yazan Abu Farha and Jurgen Gall. Ms-tcn: Multi-stage temporal convolutional network for action segmentation. In *CVPR*, 2019.

Alireza Fathi, Xiaofeng Ren, and James M Rehg. Learning to recognize objects in egocentric activities. In *CVPR*, 2011.

Yossi Gandelsman, Yu Sun, Xinlei Chen, and Alexei Efros. Test-time training with masked autoencoders. *NeurIPS*, 2022.

Artheme Gauthier-Villar, Guodong Ding, and Angela Yao. Adaptive latent trajectory anchoring for action segmentation dataset condensation. In *ECCV*, 2026.

Shurui Gui, Xiner Li, and Shuiwang Ji. Active test-time adaptation: Theoretical analyses and an algorithm. In *ICLR*, 2024.

Yunzhong Hou and Liang Zheng. Visualizing adapted knowledge in domain transfer. In *CVPR*, 2021.

Hilde Kuehne, Ali Arslan, and Thomas Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *CVPR*, 2014.

Sateesh Kumar, Sanjay Haresh, Awais Ahmed, Andrey Konin, M Zeeshan Zia, and Quoc-Huy Tran. Unsupervised action segmentation by joint representation learning and online clustering. In *CVPR*, 2022.

Jogendra Nath Kundu, Suvaansh Bhambri, Akshay Kulkarni, Hiran Sarkar, Varun Jampani, and R Venkatesh Babu. Concurrent subsidiary supervision for unsupervised source-free domain adaptation. In *ECCV*, 2022.

Taeckyung Lee, Sorn Chottananurak, Junsu Kim, Jinwoo Shin, Taesik Gong, and Sung-Ju Lee. Test-time adaptation with binary feedback. *arXiv preprint arXiv:2505.18514*, 2025.

Yanghao Li, Naiyan Wang, Jianping Shi, Jiaying Liu, and Xiaodi Hou. Revisiting batch normalization for practical domain adaptation. In *ICLR Workshop*, 2017.

Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *ICML*, 2020.

Jian Liang, Ran He, and Tieniu Tan. A comprehensive survey on test-time adaptation under distribution shifts. *International Journal of Computer Vision*, 133(1):31–64, 2025.

Qiuxia Lin, Rongyu Chen, Kerui Gu, and Angela Yao. Semantics-aware test-time adaptation for 3D human pose estimation. In *ICML*, 2025.

Daochang Liu, Qiyue Li, Anh-Dung Dinh, Tingting Jiang, Mubarak Shah, and Chang Xu. Diffusion action segmentation. In *ICCV*, 2023.

Zijia Lu and Ehsan Elhamifar. Fact: Frame-action cross-attention temporal modeling for efficient action segmentation. In *CVPR*, 2024.

Fan Ma, Linchao Zhu, Yi Yang, Shengxin Zha, Gourab Kundu, Matt Feiszli, and Zheng Shou. Sf-net: Single-frame supervision for temporal action localization. In *ECCV*, 2020.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *ICML*, 2022.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. In *ICLR*, 2023.

Mihir Prabhudesai, Tsung-Wei Ke, Alex Li, Deepak Pathak, and Katerina Fragkiadaki. Diffusion-tta: Test-time adaptation of discriminative models via generative feedback. *NeurIPS*, 2023.

Aayush J Rana and Yogesh S Rawat. Hybrid active learning via deep clustering for video action detection. In *CVPR*, 2023.

Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54(9):1–40, 2021.

Alexander Richard, Hilde Kuehne, and Juergen Gall. Action sets: Weakly supervised action segmentation without ordering constraints. In *CVPR*, 2018.

Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving robustness against common corruptions by covariate shift adaptation. *NeurIPS*, 2020.

Fadime Sener and Angela Yao. Unsupervised learning and segmentation of complex activities from video. In *CVPR*, 2018.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017.

Skyler Seto, Barry-John Theobald, Federico Danieli, Navdeep Jaitly, and Dan Busbridge. Realm: Robust entropy adaptive loss minimization for improved single-sample test-time adaptation. In *WACV*, 2024.

Yuhan Shen and Ehsan Elhamifar. Progress-aware online action segmentation for egocentric procedural task videos. In *CVPR*, 2024.

Dipika Singhania, Rahul Rahaman, and Angela Yao. Iterative contrast-classify for semi-supervised temporal action segmentation. In *AAAI*, 2022.

Dipika Singhania, Rahul Rahaman, and Angela Yao. C2f-tcn: A framework for semi-and fully-supervised temporal action segmentation. *IEEE TPAMI*, 2023.

Sebastian Stein and Stephen J McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *UbiComp*, 2013.

Yuhao Su and Ehsan Elhamifar. Two-stage active learning for efficient temporal action segmentation. In *ECCV*, 2024.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *ICML*, 2020.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *ICLR*, 2021.

Fangqiu Yi, Hongyu Wen, and Tingting Jiang. Asformer: Transformer for action segmentation. In *BMVC*, 2021.

Qing Zhong, Guodong Ding, and Angela Yao. Onlinetas: An online baseline for temporal action segmentation. *NeurIPS*, 2024.

## A Entropy Profile Analysis

**Why two-class setting.** Our analysis in Sec. 3.2 is intended to provide intuition for the emergence of uncertainty near temporal transitions under smoothing regularization. It is not intended as a formal proof of query optimality or adaptation guarantees, since real TAS predictions may involve multi-class uncertainty, non-smooth transitions, and model-specific effects. The two-class case is chosen because it is representative of the ground-truth formulation where each boundary corresponds to a transition between two action classes in TAS.

In fact, on Breakfast, over 90% of frames have a top-1 + top-2 probability sum greater than 0.9 (and 96% over 0.8), indicating that most cases are effectively close to a two-class setting. Predictions for a sample Breakfast video are plotted in Fig. 4, which supports this observation.

**Profile analysis.** Our entropy profiles are **NOT** defined by hard thresholds; instead, they are characterized based on their relative ranking within each sequence. In this context:

- *Low-entropy* predictions are more likely to be dominated by two classes with a clear gap between the top-1 and top-2 probabilities, indicating a relatively stable transition.

- *Medium-entropy* predictions may show a smaller top-1 vs. top-2 gap, suggesting increased ambiguity between competing classes.

- *High-entropy* predictions spread probability mass across multiple classes, indicating diffuse uncertainty rather than a well-defined boundary.

This can be observed in Fig. 4, and we further provide a numerical example for the three-class case in Table 5.

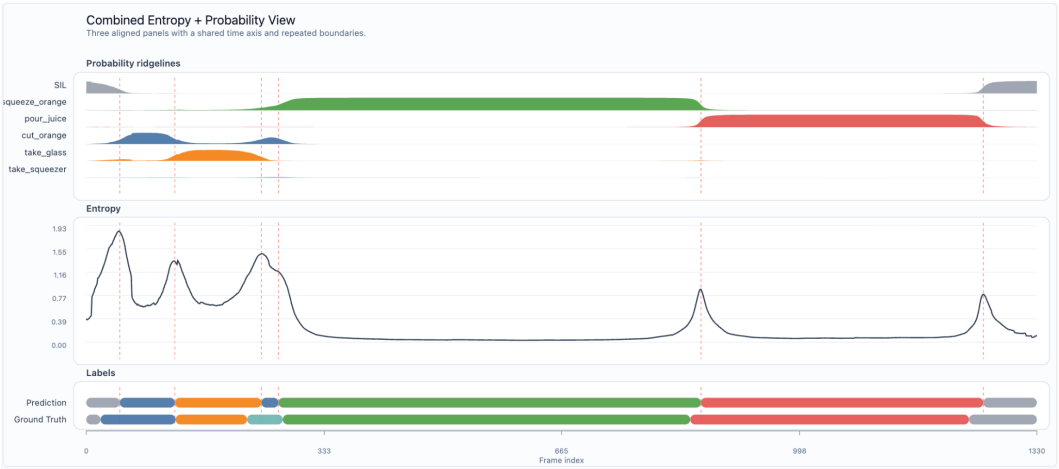



Figure 4: Class probability ridgelines and entropy for video sequence "P15_webcam01_P15_juice" from Breakfast. Several observations can be made: 1) local entropy maxima are generally well aligned with the predicted action boundaries; 2) in most cases, the uncertainty is dominated by confusion between two action classes; 3) higher boundary entropy values correspond to greater semantic confusion.

Table 5: Example probability distributions and corresponding entropy values under different uncertainty levels.

| | Low | Med | High |
|---|---|---|---|
| Prob | $[0.3, 0.7, 0]$ | $[0.45, 0.55, 0]$ | $[0.35, 0.45, 0.1]$ |
| Ent | 0.611 | 0.688 | 0.957 |

**Selection example.** To illustrate the actual selection, consider a video with 10 predicted labels whose entropy values are ranked from lowest to highest: $b_1 < b_2 < \cdots < b_{10}$. Under a $\eta = 50\%$ annotation budget (5 boundaries), we have:

*Low-entropy* boundaries: $b_1 - b_5$
*Medium-entropy* boundaries: $b_3 - b_7$
*High-entropy* boundaries: $b_6 - b_{10}$

With this scheme, the profiles naturally adapt to other videos and datasets without requiring fixed thresholds.

## B Local propagation of boundary supervision

Here, we detail the label propagation effect induced by our sparse semantic supervision.

Consider an unlabeled frame $t \notin \mathcal{B}$. Since no cross entropy term is applied at frame $t$, its prediction $p^t$ is optimized solely through the smoothing loss. The first order optimality condition yields:

$$\frac{\partial \mathcal{L}_{\text{sm}}}{\partial p^t} = 2\lambda(p^t - p^{t-1}) + 2\lambda(p^t - p^{t+1}) = 0 \tag{15}$$

which implies:

$$p^t = \frac{1}{2}(p^{t-1} + p^{t+1}) \tag{16}$$

Thus, at optimality, the prediction at an unlabeled frame equals the average of its temporal neighbors. Repeated application of this condition shows that predictions at unlabeled frames correspond to a discrete harmonic extension determined by neighboring supervised boundary frames.

Since boundary frames are directly constrained by the cross entropy loss to match their ground truth labels, their predictions act as anchor points. The smoothing loss enforces gradual variation between

Table 6: Performance comparison of Pseduo-labeling and BiTTA on Breakfast with MS-TCN.

| Method | Acc | Edit | F1@{10, 25, 50} | | |
|---|---|---|---|---|---|
| Baseline | 65.4 | 67.5 | 68.5 | 63.0 | 49.6 |
| Pseudo-label | 65.5 | 68.9 | 71.3 | 65.4 | 52.2 |
| BiTTA | 65.9 | 69.8 | 71.4 | 65.5 | 52.7 |

anchors, causing label information to propagate to neighboring unlabeled frames. Furthermore, because each update depends on local averaging, the influence of a boundary frame decreases with increasing temporal distance. An illustrative example is provided in Fig. 5.

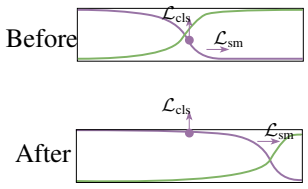



Figure 5: Illustration of local propagation effect under sparse boundary supervision. Initially ("Before"), high predictive uncertainty results in a misaligned transition. Upon providing supervision at the identified boundary frame (indicated by the purple dot), $\mathcal{L}_{\text{cls}}$ pulls the prediction to the correct label, while $\mathcal{L}_{\text{sm}}$ propagates this correction to neighboring frames, preserving the semantic anchor and effectively realigning the decision boundary to the correct temporal position ("After").

## C Baseline & More experiments

In this section, we provide additional details for the adapted baselines used in comparison.

**SimATTA.** SimATTA (Gui et al., 2024) was originally designed for an online setting. To adapt it to our offline setup, we first cluster frame samples within each video according to prediction entropy, with the number of clusters matched to the number of action boundaries per video for fair comparison. We then randomly sample one frame from each cluster for semantic annotation. The annotated frames are subsequently used for 5 epochs of adaptation.

**BiTTA.** For BiTTA (Lee et al., 2025), we follow the original frame selection protocol by performing five rounds of MC dropout to estimate prediction uncertainty and selecting the most uncertain frames for annotation, while using the remaining "stable" frames for pseudo-label supervision. Different from the original setting in (Lee et al., 2025), which relies on binary feedback and reinforcement learning, we directly provide ground-truth labels for the queried frames to ensure a fair comparison. As shown in Table 1, BiTTA performs only marginally better than the original baseline. We hypothesize that this is due to the reliance on pseudo-label supervision for the stable samples. To further investigate this effect, we additionally evaluate a pseudo-labeling variant and report the results in Table 6.

The baseline and pseudo-labeling variant achieve similar performance, suggesting the presence of confirmation bias in pseudo-labels. In addition, the comparison to BiTTA suggests that once "stable" samples are supervised through pseudo-labels, they can constrain how effectively sparse active supervision propagates along the temporal axis. This observation further supports our design choice of sparse semantic supervision during adaptation.

**Unsupervised TTA methods.** Existing unsupervised TTA methods (e.g., Tent, SAR, EATA) primarily rely on entropy minimization by updating normalization layers (such as BatchNorm) to align with test-time statistics, under the assumption that these layers act as a bridge between internal representations and input distributions. However, in TAS, the commonly used backbones (MS-TCN and ASFormer) do not include normalization layers that track input statistics. As a result, the distributional adaptation mechanism leveraged by these methods is not applicable in our setting.

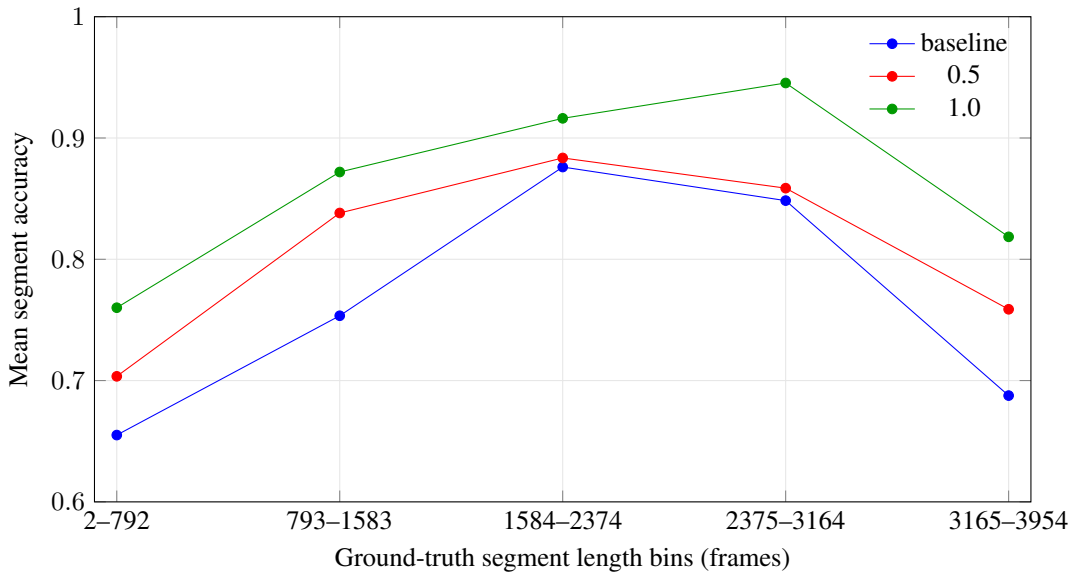



Figure 6: Mean segment accuracy across ground-truth action-duration bins. 1) Applying semantic overwrite at all boundaries (1.0) consistently improves performance over the baseline by comparable margins regardless of action durations (green vs blue); 2) Applying it to half of the boundaries (0.5) also yields consistent gains, though the improvement is smaller for mid-length actions (red vs blue).

**Impact of action lengths.** We further study the impact of action lengths during adaptation by dividing the actions into different duration bins and report the Acc for original and semantic overwrite (with ratio 0.5 and 1.0) in Fig. 6. The observations we make are as follows:

- Applying semantic overwrite on all boundaries (1.0) consistently improves accuracy across all action durations with comparable gains, indicating that the effectiveness of our method is not strongly dependent on action length.

- Applying semantic overwrite on only half of the boundaries (0.5) also yields consistent improvements across durations. However, the gains are relatively smaller for mid-length actions. A possible reason is that these segments already achieve relatively high baseline accuracy compared to other lengths, as performance increases with duration before decreasing for very long actions (the blue curve in Fig. 6). As a result, their boundary predictions tend to be confident and have moderate or low entropy and are less likely to be selected under the 0.5 ratio, which focuses on medium-entropy samples.

## D Annotation User Study

**User Study Setup.** We conduct a user study on the Breakfast (Kuehne et al., 2014) to compare the annotation efficiency of agnostic overwrite ($\mathcal{O}_{\text{agn}}$) and semantic overwrite ($\mathcal{O}_{\text{sem}}$). A total of five annotators participate in the study to annotate 20 videos each. The interfaces used for the user study are provided in Figs. 7 and 8. For each trial, annotators are presented with a video clip together with the predicted temporal action segmentation results and a queried boundary region selected by our method.

Under agnostic annotation (Fig. 7), annotators are asked to determine whether the queried boundary is correct and indicate the overwrite direction without specifying the action category. Under semantic annotation (Fig. 8), annotators are additionally required to provide the semantic action label for the corrected segment. All annotators are given the same set of queried samples for both settings to ensure fair comparison.

We record the annotation time for each query and report the average time and standard deviation across all annotators. Before the formal study, annotators complete a short tutorial session with

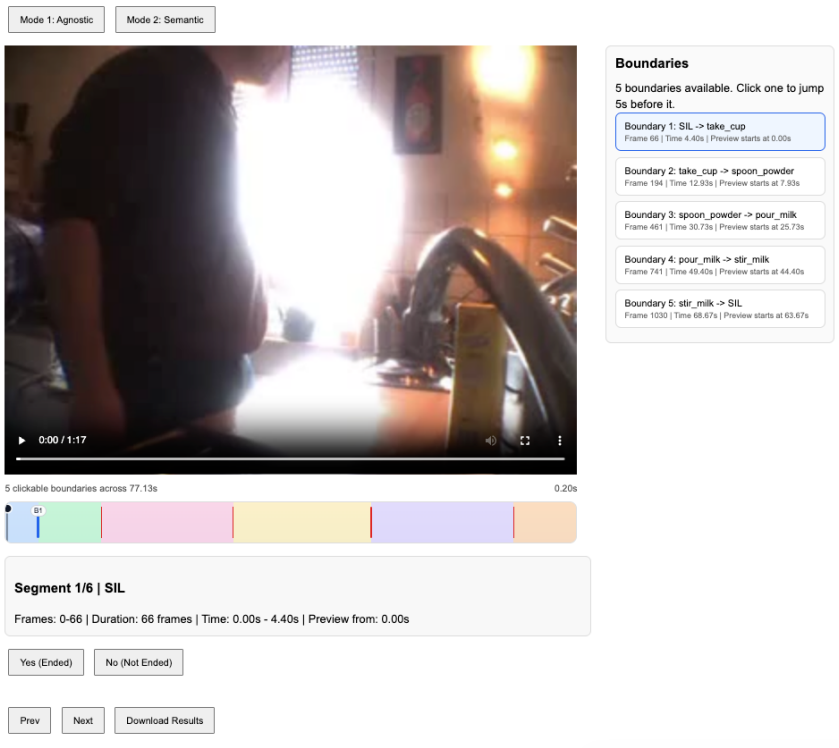



Figure 7: User interface for agnostic annotation.

several practice examples to familiarize themselves with the annotation interface and overwrite protocols.

## E Comparison to Active TAS

Perhaps the closest work to ours is (Su & Elhamifar, 2024), which proposes a two-stage active learning framework for TAS. Their method identifies informative frames by measuring the alignment cost between unlabeled videos and learned action prototypes. Their approach requires an initial learning of discriminative action prototypes via contrastive objectives. In contrast, our approach addresses test-time adaptation (TTA), where constructing such prototypes is infeasible due to efficiency constraints. Moreover, while their framework relies on a computationally intensive combination of greedy search and dynamic time warping (DTW), our method employs a lightweight, entropy-based criterion to efficiently identify critical temporal boundaries with minimal latency.

## F Limitation

While our approach effectively adapts TAS models at test time, it currently operates under a single-round active labeling scheme. An interesting direction would be to extend it to a multi-round querying setup with a fixed labeling budget, which could iteratively refine its temporal boundaries. Such an extension may also introduce challenges such as optimizing budget allocation across query rounds and increased adaptation overhead of repeated querying and model updates.

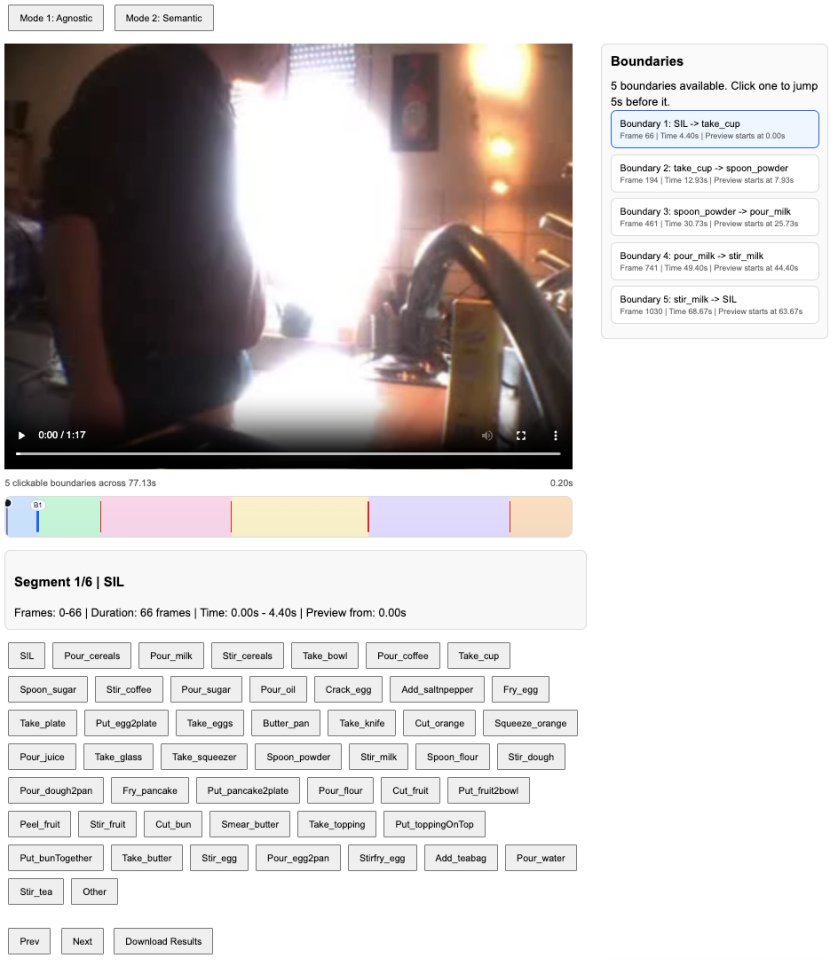



Figure 8: User interface for semantic annotation.