PEAL: PROBABILISTIC EDIT ALIGNMENT LOSS FOR DISCRETE SEQUENCE PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Discrete sequence prediction requires models to identify the correct elements and capture their sequential dependencies. However, position-wise cross-entropy (CE) can encourage local corrections that disrupt a correct subsequence when it is shifted relative to the target—a limitation we call CE's myopia. We introduce Probabilistic Edit Alignment Loss (PEAL), which formulates alignment between predicted and target sequences as cost optimization over monotone edit paths. Substitution costs decrease with the probability assigned to the aligned target class, while insertion and deletion incur fixed penalties. PEAL aggregates the costs of all valid paths using a differentiable soft minimum. These edit paths allow correctly ordered subsequences to remain aligned despite positional shifts. Our gradient analysis shows that the resulting gradient is a Gibbs-weighted combination of path-cost gradients, with lower-cost paths receiving larger weights. We further establish conditional empirical H-consistency: under specified conditions, any model minimizing empirical PEAL risk within a fixed model class H is also optimal for structured prediction within that class, as measured by empirical edit-distance risk. Experiments on automatic speech recognition, temporal action segmentation, and long-term action anticipation demonstrate improvements across multiple backbone architectures. Further analyses associate these gains with better relative-order preservation and longer correctly aligned subsequences.

1 INTRODUCTION

Real-world information about physical environments, human behavior, and social interactions is often captured through continuous multimodal observations. Interpreting these observations supports decision-making and task execution in industrial production, everyday life, and scientific research. In many such applications, intelligent systems need to recover or anticipate sequences of meaningful discrete elements. For example, automatic speech recognition (ASR) transcribes speech signals into ordered text (Panayotov et al., 2015; Ravanelli et al., 2021). Temporal action segmentation (TAS) recovers sequences of action segments from untrimmed videos (Farha & Gall, 2019; Behrmann et al., 2022; Bahrami et al., 2026). Long-term action anticipation (LTA) predicts future action sequences from previously observed video (Chu et al., 2026; Zhao et al., 2024; Wasim et al., 2025; Zatsarynna et al., 2025). Despite differences in input modality and output vocabulary, these tasks can all be viewed as instances of the discrete sequence prediction problem conditioned on observed signals. This requires structured prediction capability: predicting the correct elements and capturing the temporal, semantic, and other dependencies among them to produce a coherent discrete sequence.

However, a misalignment exists between prevalent training objectives and the structured prediction capabilities required by discrete sequence prediction tasks. Model training commonly relies on cross-entropy (CE) supervision. CE penalizes position-wise mismatches even when a predicted subsequence is correct in both content and internal structure but is not positionally aligned with the ground truth. The resulting element-wise corrections can disrupt this correct subsequence and undermine the model's acquired ability to capture such structure. We refer to this limitation as *myopia*. In Figure 1, the prediction contains the correct subsequence "rinse apple → dry apple → pick up knife." However, this subsequence is not positionally aligned with the ground truth. CE therefore encourages replacing the displaced action "dry apple" with the reference action "rinse apple", which can reduce the loss while removing a correct action and introducing a repetition, thereby disrupting

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.

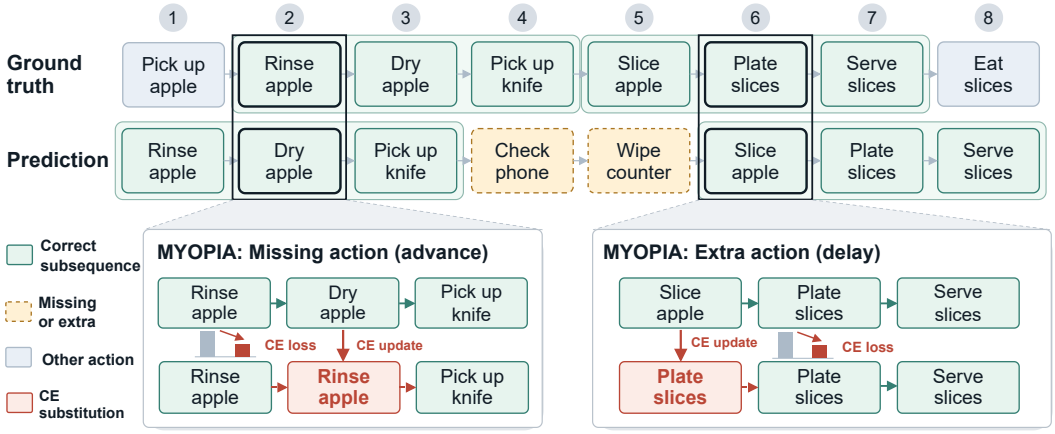

Figure 1: Myopia of CE in LTA. CE encourages the red-highlighted substitutions to restore position-wise agreement, reducing loss while removing correct actions and introducing repetitions.

the structure already captured by the model. Similarly, at the sixth position, replacing "slice apple" with the reference action "plate slices" can reduce CE while disrupting the correct subsequence "slice apple → plate slices → serve slices". The underlying limitation lies in what position-wise CE rewards. At each position, the loss depends only on the probability assigned to the reference element. Two output distributions with the same reference-element probability therefore incur the same local loss, even if one favors an element belonging to a correct but displaced subsequence and the other favors an unrelated element. The local loss thus makes no explicit distinction between correct content at a displaced position and incorrect content. Increasing the reference-element probability is rewarded without separately accounting for the structural contribution of the element being suppressed.

We address this limitation with the Probabilistic Edit Alignment Loss (PEAL). PEAL formulates alignment between predicted and target sequences as a path-level cost optimization problem over valid edit paths on an edit-state grid. Each valid monotone edit path represents one possible alignment and consists of three types of operations: insertion, deletion, and substitution. The cost of a path is the sum of its operation costs. Substitution costs depend on the model prediction and decrease as the probability assigned to the aligned target class increases, whereas insertion and deletion incur fixed penalties. PEAL aggregates the costs of all valid paths with a differentiable soft-minimum operator, yielding a smooth objective for sequence-level alignment. A key property of PEAL is that lower-cost alignment paths receive greater weights in the gradient. We establish this property by expressing the gradient of the soft-minimum loss as a Gibbs-weighted combination of path-cost gradients. In this path-level training signal, insertion and deletion steps can absorb positional shifts, while substitution costs evaluate content agreement between aligned elements. Their combination allows correctly ordered subsequences to remain aligned even when they do not occupy the target positions.

To guarantee that a PEAL-optimal model is also optimal for structured prediction, we establish conditional empirical H-consistency (Zhang & Agarwal, 2020; Awasthi et al., 2022) with PEAL as the surrogate loss and edit distance as the target loss. Here, H denotes the fixed model class. Under our theorem's conditions, PEAL satisfies this consistency criterion: as empirical PEAL excess risk approaches zero, empirical edit-distance excess risk also approaches zero within H. Consequently, a model attaining the empirical PEAL optimum in H is also optimal under the edit-distance criterion for structured prediction. Experiments on ASR, TAS, and LTA demonstrate consistent gains across multiple modalities and task formulations. Our results show that training objectives that model sequence-level alignment help models produce more coherent output sequences. Our main contributions are as follows:

1. We propose PEAL, which uses differentiable soft minimization to guide models toward lower-cost sequence alignments and improve structured prediction capability.
2. We analyze conditions under which minimizing PEAL guarantees model optimality within a fixed model class, as measured by edit distance.
3. Experiments on ASR, TAS, and LTA show that PEAL improves structured prediction capability across modalities and task formulations.

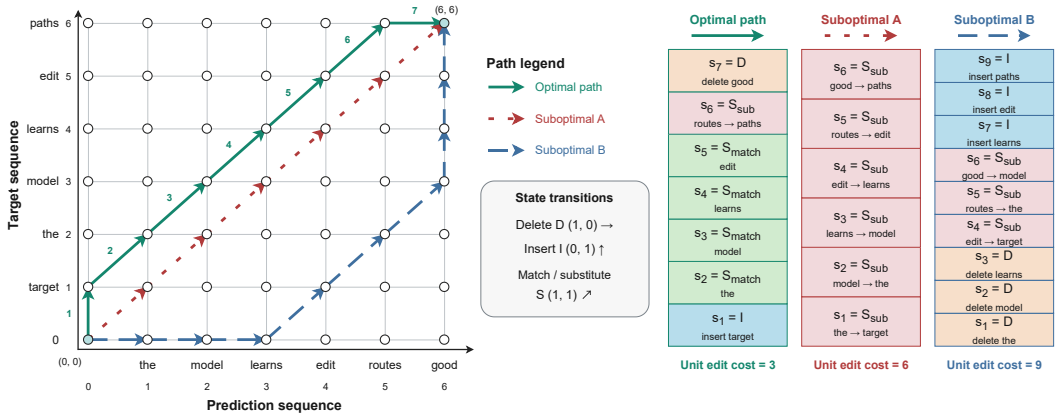

Figure 2: Optimal and suboptimal alignment paths on the edit-state grid. The green path contains one insertion, four matches, one substitution, and one deletion, attaining the minimum unit edit cost of 3. The red dotted and blue dashed paths have costs of 6 and 9, respectively.

2 Probabilistic Edit Alignment Loss

2.1 Notation and Preliminaries

Lemma 1 (Differentiable soft-minimum operator). For K real values $(a_1, \ldots, a_K)$, the following operator provides a differentiable approximation to their minimum (Cuturi & Blondel, 2017; Mensch & Blondel, 2018):

$$\operatorname{softmin}_\tau(a_1, \ldots, a_K) = -\tau \log \sum_{k=1}^{K} \exp\left(-\frac{a_k}{\tau}\right), \qquad \tau > 0. \tag{1}$$

Here, $\tau > 0$ controls the smoothness of the approximation.

Sequence Notation. Given an input x, let $y = (y_1, \ldots, y_m)$ denote the target sequence. The model h_θ produces a probability distribution over C classes at each of n prediction positions, yielding $Z = h_\theta(x) = (p_{i,c}) \in \mathbb{R}^{n \times C}$ where $p_{i,c}$ is the probability assigned to class c at position i. The corresponding predicted discrete sequence is denoted by $\widehat{y} = (\widehat{y}_1, \ldots, \widehat{y}_n)$.

Edit Path. For the predicted and target sequences $\widehat{y}$ and y, we formulate their alignment as a monotone path on an edit-state grid. The horizontal and vertical coordinates index the predicted and target sequences, respectively. A state (i, j) indicates that the first i predicted elements and the first j target elements have been processed. Thus, $(0, 0)$ is the initial state, where no elements have been processed, and (n, m) is the terminal state, where both sequences have been fully processed. Let $\Pi_{n,m}$ denote the set of edit paths from the initial state $(0, 0)$ to the terminal state (n, m). Each path $\pi = (s_1, \ldots, s_T) \in \Pi_{n,m}$ consists of steps $s_t \in \{D, I, S\}$ satisfying $\sum_{t=1}^{T} s_t = (n, m)$. In detail, a deletion step $D = (1, 0)$ consumes a predicted element without advancing in the target sequence. An insertion step $I = (0, 1)$ advances in the target sequence without consuming a predicted element, accounting for a target element missing from the prediction. A substitution step $S = (1, 1)$ aligns one predicted element with one target element. It represents either an exact match, denoted by S_{match}, or a mismatch, denoted by S_{sub}. These two cases share the same displacement but may have different costs. Since all steps have nonnegative displacements, an edit path allows elements to be aligned across different positions while preserving their relative order.

Figure 2 illustrates the alignment between the predicted sequence "the model learns edit routes good" and the target sequence "target the model learns edit paths". The green path is optimal:

$$(0,0) \xrightarrow{I} (0,1) \xrightarrow{S_{\text{match}}} (1,2) \xrightarrow{S_{\text{match}}} (2,3) \xrightarrow{S_{\text{match}}} (3,4) \xrightarrow{S_{\text{match}}} (4,5) \xrightarrow{S_{\text{sub}}} (5,6) \xrightarrow{D} (6,6).$$

It inserts "target", matches the four-word subsequence "the model learns edit", replaces "routes" with "paths", and deletes the extra word "good". Under the conventional unit-cost edit distance, where matches have cost 0 and insertions, deletions, and mismatches have cost 1, this path has cost 3.

2.2 FORMULATION OF PEAL

PEAL formulates the alignment between the predicted and target sequences as a path-level cost optimization problem over the valid edit paths in $\Pi_{n,m}$, where each path is evaluated by its prediction-dependent cost. Direct hard minimization is nondifferentiable when the minimum-cost path changes. Other alternatives, such as squared-ℓ_2 regularization and perturbation-based smoothing (Mensch & Blondel, 2018; Berthet et al., 2020), either modify the aggregation of path costs or introduce additional sampling overhead. We therefore replace the hard minimum with the differentiable soft-minimum operator introduced in Lemma 1 (Cuturi & Blondel, 2017; Mensch & Blondel, 2018) and define the per-sample Probabilistic Edit Alignment Loss (PEAL) as

$$L_\tau(Z, y) = -\tau \log \sum_{\pi \in \Pi_{n,m}} \exp\left[-\frac{C_p(\pi; Z, y)}{\tau}\right], \qquad \tau > 0. \tag{2}$$

As $\tau \to 0$, PEAL approaches the minimum-cost alignment. Here, $C_p(\pi; Z, y)$ denotes the total cost of edit path π, obtained by summing the local costs of its insertion, deletion, and substitution steps:

$$C_p(\pi; Z, y) = \sum_{t \in \mathcal{T}_D(\pi)} c_{\text{del}} + \sum_{t \in \mathcal{T}_I(\pi)} c_{\text{ins}} + \sum_{t \in \mathcal{T}_S(\pi)} q_{i_t, j_t}. \tag{3}$$

where $\mathcal{T}_D(\pi) = \{t : s_t = D\}, \mathcal{T}_I(\pi) = \{t : s_t = I\}, \mathcal{T}_S(\pi) = \{t : s_t = S\}$ indicate the index sets of deletion, insertion, and substitution steps of the edit path $\pi = (s_1, \ldots, s_T)$.

The cost of each edit step should reflect whether an operation aligns a prediction with a target element. A substitution step directly aligns a prediction position i with a target position j, covering both exact matches and mismatches. Its cost should continuously measure their disagreement and decrease as the probability assigned to the target class y_j increases. Therefore, we define the substitution cost as $q_{i,j} = (1 - p_{i,y_j})^\alpha, \alpha > 0$. This cost lies in $[0, 1]$, approaches zero for a high-confidence match, and approaches one when the target class receives little or no probability. The exponent α controls the curvature of the confidence-dependent penalty. An insertion step accounts for a target element missing from the prediction, while a deletion step accounts for a surplus predicted element. Since neither operation aligns a prediction with a target label, we assign each a fixed cost independent of the model probabilities as $c_{\text{ins}} > 0$ and $c_{\text{del}} > 0$ and set both to 1 by default. This value matches the maximum possible substitution cost and places all three edit operations on the same cost scale.

By assigning costs to complete edit paths and aggregating them with a differentiable soft minimum, PEAL evaluates each prediction in the context of a sequence-level alignment. The resulting training signal therefore depends on the correspondence induced by the entire path rather than on independent position-wise matches. Insertion and deletion steps account for missing and surplus elements, respectively, allowing a correctly ordered subsequence to remain aligned despite positional shifts. The soft minimum provides a differentiable surrogate for the hard minimum while giving greater influence to lower-cost paths, thereby guiding model updates toward predictions that support globally coherent alignments. PEAL also exhibits two key properties, which we verify in the next sections:

1. **Alignment-guided gradients.** PEAL guides model updates toward lower-cost alignment paths.
2. **Conditional optimality guarantees.** Under the stated assumptions, minimizing standalone PEAL guarantees model optimality within a fixed model class, as measured by edit distance.

We use PEAL in two training strategies. **PEAL-Joint** combines PEAL with the baseline objective: $\mathcal{L}_{\text{Joint}} = (1 - \lambda)\mathcal{L}_{\text{other}} + \lambda\mathcal{L}_{\text{PEAL}}$, where $\mathcal{L}_{\text{other}}$ denotes the baseline objective and $0 < \lambda < 1$ controls PEAL's weight. **PEAL-FT** first trains with $\mathcal{L}_{\text{other}}$ and then fine-tunes with PEAL.

2.3 DIFFERENTIABLE ALIGNMENT AND GRADIENT ANALYSIS

We analyze the gradient of PEAL with respect to prediction probabilities to explain how it guides model updates toward lower-cost alignment paths. Differentiating L_τ with respect to $p_{i,c}$ gives

$$\frac{\partial L_\tau}{\partial p_{i,c}} = \sum_{\pi \in \Pi_{n,m}} \underbrace{\frac{\exp[-C_p(\pi; Z, y)/\tau]}{\sum_{\pi' \in \Pi_{n,m}} \exp[-C_p(\pi'; Z, y)/\tau]}}_{P_\tau(\pi | Z, y)} \frac{\partial C_p(\pi; Z, y)}{\partial p_{i,c}}. \tag{4}$$

Here, $P_\tau(\pi \mid Z, y)$ weights each path's contribution to the gradient. These weights follow a Gibbs distribution, so a smaller path cost $C_p(\pi; Z, y)$ gives a larger weight. The insertion and deletion costs are constants and have zero derivatives with respect to $p_{i,c}$. Only substitution costs contribute directly to the path-cost gradient. Let $I_{i,c}(\pi)$ indicate whether path π contains a substitution step that aligns prediction position i with a target position j whose label satisfies $y_j = c$. The indicator takes value 1 when such a step is present and 0 otherwise. Since $q_{i,j} = (1 - p_{i,y_j})^\alpha$, we obtain

$$\frac{\partial C_p(\pi; Z, y)}{\partial p_{i,c}} = -\alpha(1 - p_{i,c})^{\alpha-1} I_{i,c}(\pi). \tag{5}$$

Combining this expression with the Gibbs-weighted path gradient gives

$$\frac{\partial L_\tau}{\partial p_{i,c}} = \sum_{\pi \in \Pi_{n,m}} P_\tau(\pi \mid Z, y) \frac{\partial C_p(\pi; Z, y)}{\partial p_{i,c}} = -\alpha(1 - p_{i,c})^{\alpha-1} \sum_{\pi \in \Pi_{n,m}} P_\tau(\pi \mid Z, y) I_{i,c}(\pi). \tag{6}$$

The final expression shows that the gradient at $p_{i,c}$ is determined by the sensitivity of the substitution cost to $p_{i,c}$ and the Gibbs-weighted mass of paths that support the corresponding alignment. A path contributes to this derivative only when it aligns prediction position i with a target element whose class is c. Among such paths, lower-cost paths receive larger Gibbs weights and therefore contribute more strongly to the gradient. Since the derivative is negative whenever this weighted alignment mass is nonzero, these paths provide a probability-space signal favoring a higher probability for class c. Although insertion and deletion costs have no direct derivatives with respect to prediction probabilities, they still affect the gradient indirectly by changing the total path costs and, consequently, the Gibbs weights. Appendix H provides the corresponding derivation.

2.4 Conditional Optimality Guarantees for PEAL

We establish a conditional optimality guarantee for PEAL: any model that globally minimizes PEAL is also an optimal model for structured prediction. We formulate this guarantee using conditional empirical H-consistency, where H denotes the fixed set of models realizable under a given model architecture. Specifically, a surrogate loss is conditionally empirically H-consistent with respect to a target loss if driving its risk toward the infimum over H necessarily drives the target risk toward its infimum over the same class (Zhang & Agarwal, 2020; Awasthi et al., 2022).

Under this framework, PEAL serves as the surrogate loss. We use edit distance as the target loss because model optimality should be defined according to a task-level criterion. Edit distance measures the cost of an optimal alignment between a predicted and a target sequence. It therefore captures discrepancies in both their elements and their relationships, providing a sequence-level measure of structured prediction quality (Levenshtein, 1966; Chu et al., 2026; Zhao et al., 2024).

Given a fixed sample set, we define empirical PEAL risk and empirical edit-distance risk of a model h. The former is the average normalized PEAL loss, while the latter is the average normalized edit distance between predictions and targets. Since H-consistency concerns convergence to the best performance attainable within H, we measure each model's distance from the corresponding optimum using excess risk. Specifically, we introduce excess risk, defined as the difference between a model's empirical risk and its infimum over H. We denote their corresponding excess risks by $\widehat{\mathcal{E}}_{\mathrm{S},\tau,H}(h)$ and $\widehat{\mathcal{E}}_{\mathrm{E},H}(h)$, respectively. Under the above definitions, we say that PEAL is conditionally empirically H-consistent with respect to edit distance if $\widehat{\mathcal{E}}_{\mathrm{S},\tau,H}(h) \to 0$ implies $\widehat{\mathcal{E}}_{\mathrm{E},H}(h) \to 0$.

Theorem 1 (Conditional empirical H-consistency). Consider N equally weighted samples, with prediction and target lengths both equal to n, and unit insertion and deletion costs. Let H be nonempty, and suppose every $h \in H$ satisfies $\max_c p_{h,i,c}(x) \geq 1 - \xi$ at every sample and prediction position, for some $0 < \xi < 1/2$. If the approximation error $\beta_{\xi,\tau}$, defined in Appendix J.3, satisfies $\beta_{\xi,\tau} < 1/(Nn)$, then every $h \in H$ satisfies

$$\widehat{\mathcal{E}}_{\mathrm{E},H}(h) \leq \frac{1}{1 - Nn\beta_{\xi,\tau}} \widehat{\mathcal{E}}_{\mathrm{S},\tau,H}(h). \tag{7}$$

This formula shows that any model that minimizes empirical PEAL risk also minimizes empirical edit-distance risk within the same hypothesis class H. This establishes conditional empirical H-consistency, linking PEAL optimality to model optimality under the edit-distance criterion. Since edit distance measures discrepancies in sequence content and structure, this result provides a theoretical

Table 1: Results on Ego4D dataset for LTA and on the Gronings dataset for ASR.

	Ego4D LTA					Gronings ASR	
	SlowFast		Qwen-3.5-4B			XLS-R-300M	
Objective	Verb ↓	Noun ↓	Verb ↓	Noun ↓	Action ↓	WER ↓	CER ↓
CE	0.7546	0.7909	0.6951	**0.6861**	0.8652	0.4198	0.1089
CTC	–	–	–	–	–	0.3253	0.0822
LS-Joint	0.7279	0.7694	0.6908	0.6868	0.8623	0.3289	0.0848
FL-Joint	0.7242	0.7713	0.6914	0.6865	0.8622	0.3228	0.0815
AXE-Joint	0.7140	0.7682	0.6907	0.6866	0.8620	0.3114	0.0798
PEAL-FT	0.6913	0.7700	**0.6584**	0.6905	0.8583	0.2915	**0.0704**
PEAL-Joint	**0.6875**	**0.7628**	0.6605	0.6878	**0.8552**	**0.2913**	0.0715

Table 2: Complete evaluation results on split 1 of the Breakfast dataset for TAS.

Objective	TAS Edit ↑	F1@10 ↑	F1@25 ↑	F1@50 ↑	Avg F1 ↑	Acc. ↑
CE	75.310	74.527	68.853	57.861	67.080	63.159
LS-Joint	75.284	75.124	68.986	57.819	67.310	64.453
FL-Joint	75.842	75.900	69.868	59.151	68.306	66.926
AXE-Joint	74.593	75.081	68.903	57.311	67.098	64.114
PEAL-Joint	**76.108**	**76.779**	**71.215**	**60.322**	**69.439**	**67.190**

basis for using PEAL to promote structured prediction capability. The proof, approximation error, and further applicability conditions are provided in the Appendices I–L.

3 Experiments

3.1 Experimental Setup

Tasks. For LTA, we evaluate a Transformer using SlowFast features Grauman et al. (2022); Feichtenhofer et al. (2019) and Qwen-3.5-4B Team (2026) on Ego4D. For TAS, we evaluate UVAST-S Behrmann et al. (2022) on Breakfast Kuehne et al. (2014). For ASR, we fine-tune Wav2Vec2-XLS-R-300M Babu et al. (2022) on a 24-minute labeled subset of Gronings (Bartelds et al., 2023). Further details are provided in Appendix C.

Comparison losses. We compare PEAL with Label Smoothing (LS), Focal Loss (FL), and Aligned Cross Entropy (AXE) (Szegedy et al., 2016; Lin et al., 2017; Ghazvininejad et al., 2020). We consider two training strategies. Joint training combines each auxiliary loss with CE for LTA and TAS, and with connectionist temporal classification (CTC) for ASR (Graves et al., 2006).

3.2 Main Results across Backbones and Tasks

Across backbones. Table 1 shows that PEAL improves sequence-level prediction across both the SlowFast-based Transformer and Qwen-3.5-4B. These results suggest that PEAL's benefits extend across different backbone architectures.

Across tasks. Tables 1 and 2 show gains across LTA, TAS, and ASR, suggesting that PEAL supports structured prediction across different task formulations. On TAS, PEAL improves structured prediction at multiple levels: edit score reflects sequence-level action ordering, while segmental F1 scores and frame-wise accuracy capture segment overlap and frame-wise classification. PEAL and baseline losses jointly preserve sequence structure while retaining local classification and segmentation supervision. Both PEAL-Joint and PEAL-FT improve performance on LTA and ASR, showing that PEAL is effective when combined with the baseline objective or applied during a fine-tuning stage.

3.3 Ablation Studies

We ablate hyperparameters and edit operations using the SlowFast and Qwen-3.5-4B models on LTA to analyze their roles in PEAL. Additional ablations on Breakfast are reported in Appendix D.

Table 3: Operation-cost and temperature ablations on the Ego4D dataset.

(a) Operation-cost ablation

α	Verb ↓	Noun ↓	c	Verb ↓	Noun ↓
1.0	0.6875	**0.7628**	0.1	0.7582	0.7847
1.5	**0.6865**	0.7705	0.5	0.6921	0.7643
2.0	0.6981	0.7680	1.0	**0.6875**	**0.7628**
2.5	0.7015	0.7641	1.5	0.7156	0.7702
3.0	0.7045	0.7703	3.0	0.7185	0.7691

(b) Temperature ablation

τ	Verb ↓	Noun ↓
0.01	0.6878	0.7622
0.05	0.6941	**0.7551**
0.1	0.6875	0.7628
0.2	**0.6845**	0.7642
0.5	0.6910	0.7653

Table 4: Edit-operation ablation on the Ego4D dataset. I, D, and S denote insertion, deletion, and substitution.

	SlowFast		Qwen-3.5-4B		
Objective	Verb ↓	Noun ↓	Verb ↓	Noun ↓	Action ↓
I+D	0.7599	0.7978	0.6793	0.7046	0.8721
S / I+S / D+S	0.7202	0.7823	0.6794	0.6890	0.8653
PEAL-FT	0.6913	0.7700	**0.6584**	0.6905	0.8583
PEAL-Joint	**0.6875**	**0.7628**	0.6605	**0.6878**	**0.8552**

Table 5: Correlations of CE and PEAL losses with edit distance.

		Pearson		Spearman	
Train	Head	CE	PEAL	CE	PEAL
CE	Noun	0.775	**0.911**	0.827	**0.920**
CE	Verb	0.755	**0.839**	0.733	**0.832**
PEAL	Noun	0.743	**0.984**	0.797	**0.977**
PEAL	Verb	0.649	**0.967**	0.627	**0.960**

Operation-cost ablation. Table 3(a) examines the local cost exponent α and insertion/deletion costs. The default $\alpha = 1.0$ balances verb and noun performance.

Temperature ablation. Table 3(b) examines the effect of PEAL's temperature parameter τ. Across $\tau \in \{0.01, 0.05, 0.1, 0.2, 0.5\}$, PEAL retains improvements over CE on both metrics, suggesting that its gains are not confined to one temperature setting.

Edit-operation ablation on the Ego4D dataset. Table 4 compares combinations of insertion (I), deletion (D), and substitution (S) on Qwen-3.5-4B and SlowFast. All variants use the joint training strategy. For equal-length predictions and targets, every complete path must contain equal numbers of insertions and deletions. The $I + S$, $D + S$, and S variants therefore admit only the diagonal path, preventing them from accommodating positional shifts. With only $I + D$, path costs are independent of prediction probabilities, so the PEAL term provides no gradient and joint training reduces to a scaled CE objective. Full PEAL outperforms all ablated variants across both backbones, supporting the importance of all three edit operations for effective sequence alignment.

3.4 Effectiveness Analysis

Relative-Order Preservation and Alignment Continuity. We examine whether PEAL preserves ordered content and continuous correct subsequences using four complementary metrics in Figure 3. The colored violin shapes show the per-sample distributions for CE and PEAL, with wider regions indicating higher density. Normalized longest common subsequence (LCS) measures the normalized length of the longest sequence of matching elements appearing in the same order in prediction and target, allowing nonconsecutive matches (a1). Normalized order inconsistency measures the proportion of matched element pairs in the prediction whose order disagrees with the target (a2). For example, given target $ABCD$, prediction $CBAD$ contains three reversed pairs, (C, B), (C, A), and (B, A), among six pairs, giving a score of $3/6 = 0.5$. Lower values indicate fewer ordering errors. The longest correctly aligned segment counts consecutive matching steps along the evaluated edit alignment, with any insertion, deletion, or mismatch interrupting the segment (b1). The maximum residual error-block length counts the longest uninterrupted sequence of nonmatching edit operations between correct matches or at sequence boundaries (b2). Longer correct segments indicate greater alignment continuity, whereas shorter error blocks indicate less concentrated errors.

These results suggest that PEAL improves structured prediction by better preserving ordered subsequences. PEAL increases the mean normalized LCS for verbs and reduces mean order inconsistency for both verbs and nouns, indicating that predictions retain more correctly ordered verb content and exhibit fewer ordering violations in both heads. PEAL also increases the mean longest cor-

rectly aligned segment for both heads, showing that correct predictions form longer continuous subsequences. These improvements are consistent with PEAL's alignment-guided gradients, which give greater weight to lower-cost paths and guide models toward lower-cost sequence alignments, improving their structured prediction capability. Together, the results suggest that PEAL's gains reflect improved relative-order preservation and subsequence continuity.

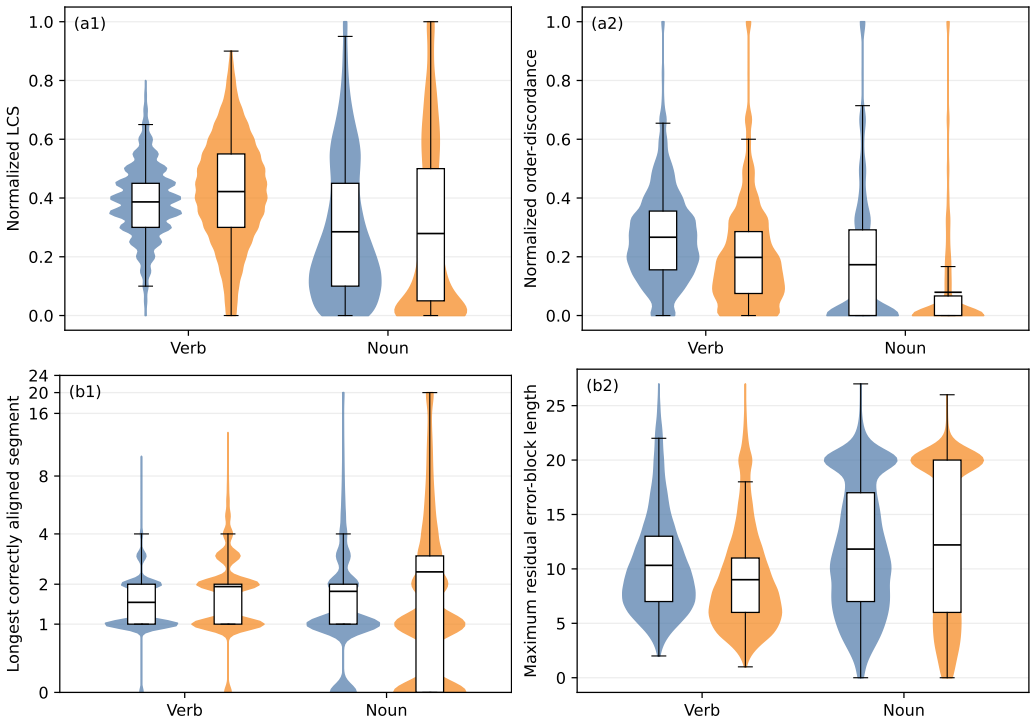

Figure 3: Sequence order and alignment quality under CE and PEAL-Joint on the Ego4D dataset. (a1) Normalized LCS. (a2) Normalized order inconsistency. (b1) Longest correctly aligned segment. (b2) Maximum residual error-block length. Blue and orange violins show the per-sample metric distributions for CE and PEAL-Joint. Wider violin regions indicate higher sample density. White boxes span the interquartile range (25th–75th percentiles), black horizontal lines mark the means.

Correlation analysis. To test whether PEAL better reflects sequence-level errors than CE, we compare each loss with edit distance. For each head under CE and PEAL training, we compute per-sample edit distance and evaluate both CE and standalone PEAL on the same model outputs. We then calculate Pearson and Spearman correlations separately for CE versus edit distance and PEAL versus edit distance. Table 5 shows that PEAL has higher correlations with edit distance than CE for both heads under either training objective. Thus, PEAL more closely tracks variation in sequence-level errors, even when the predictions come from a CE-trained model. This finding is consistent with PEAL's path-based cost, which accounts for positional shifts when evaluating agreement between predicted and target sequences.

Retention of Frequent Adjacent Action Pairs. To assess whether PEAL preserves local action structure, we rank adjacent action pairs by their frequency in the dataset and select the top 80%. For each selected pair, we measure its retention rate in sequences predicted by models trained with CE, PEAL, or AXE. Figure 4 shows the distribution of these rates. Across both prediction heads and reference sets, PEAL achieves higher median retention rates than CE, suggesting that it better preserves frequently observed adjacent action transitions.

4 Related Work

Alignment-based objectives compare predicted and target sequences (Graves et al., 2006; Sakoe & Chiba, 1978; Cuturi & Blondel, 2017; Ghazvininejad et al., 2020). These objectives allow

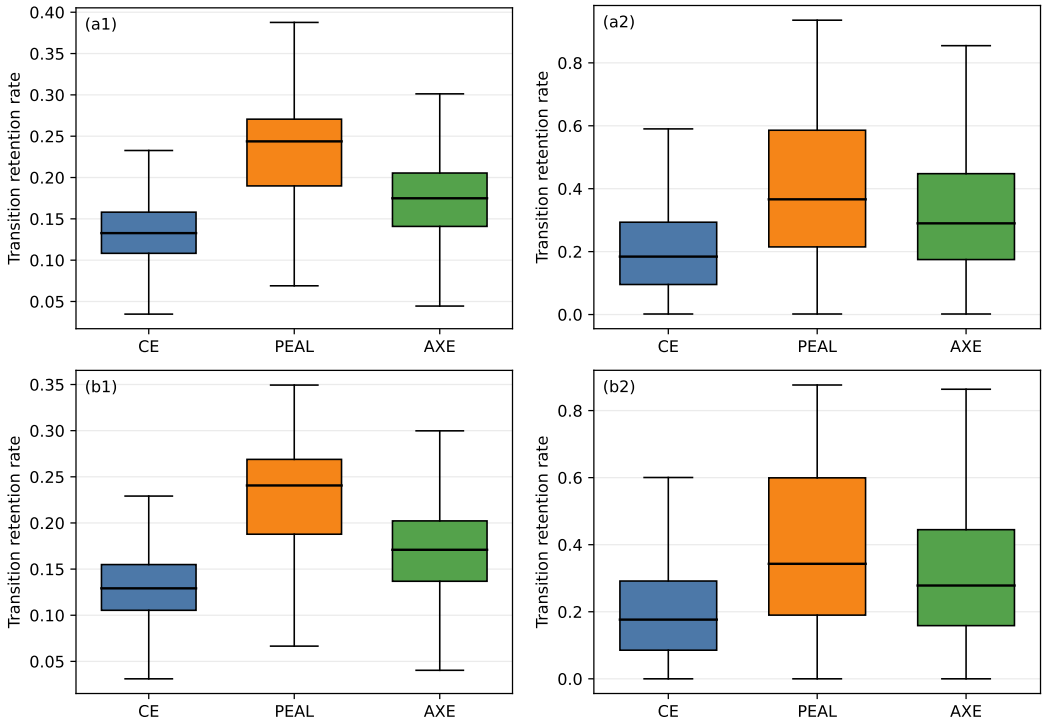

Figure 4: Retention rates of the top 80% frequent adjacent action pairs. (a1) Verb and (a2) noun training-reference annotations; (b1) verb and (b2) noun validation-reference annotations. Blue, orange, and green boxes represent CE, PEAL, and AXE, respectively. Boxes span the 25th–75th percentiles. Black lines mark the medians.

alignments between sequences, accommodating timing variation and positional shifts. However, their implications for structured prediction capability require further analysis.

Task-specific objectives incorporate supervision tailored to particular sequence prediction problems. For TAS, task-specific losses supervise how frames and segments represent actions (Lu & Elhamifar, 2024; Xu & Zheng, 2024; Tian, 2025; Yang et al., 2023). When frame-level labels are unavailable, reconstruction losses supervise skeleton recovery, whereas commitment losses align encoder outputs with motion prototypes (Spurio et al., 2025; Gökay et al., 2025). For ASR, auxiliary CTC losses apply transcript supervision to intermediate encoder outputs, encouraging each layer to produce progressively usable frame-to-token alignments (Hojo et al., 2024; Kim et al., 2024). Consistency and distillation losses instead reduce differences between posterior distributions produced by different encoder layers, thereby improving prediction stability (Yao et al., 2025; Li et al., 2025). Their robustness across tasks lacks comprehensive evaluation.

To address these two limitations, PEAL provides a common sequence-level objective guiding updates toward lower-cost sequence alignments. Under the stated assumptions, minimizing standalone PEAL guarantees model optimality within a fixed model class, while experiments across ASR, TAS, LTA, and multiple backbone architectures support its robustness.

5 Conclusion

We propose PEAL, which computes the minimum alignment cost between predicted and target sequences. It guides models toward lower-cost structural alignments, improving their structured prediction capability. Experiments on ASR, TAS, and LTA demonstrate benefits of incorporating PEAL into training objectives for discrete sequence prediction. PEAL provides a training objective that models sequence alignment and helps improve structured prediction capability in discrete sequence prediction. It offers both theoretical and empirical support for training-objective design. Future work will explore generalizing PEAL beyond discrete sequence prediction to broader sequence prediction settings, including regression tasks with continuous-valued outputs.

AI USE STATEMENT

In this work, we used generative AI tools to polish the language and improve the clarity of the manuscript. Generative AI tools were not used in developing the study rationale or in generating the results and figures reported in this work. We verified all technical content, analyses, figures, and tables. We take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the assistance of generative AI.

REFERENCES

Pranjal Awasthi, Anqi Mao, Mehryar Mohri, and Yutao Zhong. H-consistency bounds for surrogate loss minimizers. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 1117–1174. PMLR, 2022.

Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick von Platen, Yatharth Saraf, Juan Pino, Alexei Baevski, Alexis Conneau, and Michael Auli. XLS-R: self-supervised cross-lingual speech representation learning at scale. In Hanseok Ko and John H. L. Hansen (eds.), *23rd Annual Conference of the International Speech Communication Association, Interspeech 2022, Incheon, Korea, September 18-22, 2022*, pp. 2278–2282. ISCA, 2022. doi: 10.21437/INTERSPEECH.2022-143.

Emad Bahrami, Olga Zatsarynna, Gianpiero Francesca, and Juergen Gall. Towards generalizing temporal action segmentation to unseen views. *International Journal of Computer Vision*, 134(4): 180, 2026.

Martijn Bartelds, Nay San, Bradley McDonnell, Dan Jurafsky, and Martijn Wieling. Making more of little data: Improving low-resource automatic speech recognition using data augmentation. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 715–729, Toronto, Canada, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.42.

Nadine Behrmann, S. Alireza Golestaneh, Zico Kolter, Jürgen Gall, and Mehdi Noroozi. Unified fully and timestamp supervised temporal action segmentation via sequence to sequence translation. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner (eds.), *Computer Vision – ECCV 2022*, pp. 52–68, Cham, 2022. Springer Nature Switzerland. ISBN 978-3-031-19833-5.

Quentin Berthet, Mathieu Blondel, Olivier Teboul, Marco Cuturi, Jean-Philippe Vert, and Francis R. Bach. Learning with differentiable perturbed optimizers. *CoRR*, abs/2002.08676, 2020.

Qiaohui Chu, Haoyu Zhang, Meng Liu, Yisen Feng, Haoxiang Shi, and Liqiang Nie. Intention-guided cognitive reasoning for egocentric long-term action anticipation. In Sven Koenig, Chad Jenkins, and Matthew E. Taylor (eds.), *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026, Singapore, January 20-27, 2026*, pp. 17436–17444. AAAI Press, 2026.

Marco Cuturi and Mathieu Blondel. Soft-DTW: a differentiable loss function for time-series. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 894–903. PMLR, 2017.

Yazan Abu Farha and Jurgen Gall. Ms-tcn: Multi-stage temporal convolutional network for action segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

Marjan Ghazvininejad, Vladimir Karpukhin, Luke Zettlemoyer, and Omer Levy. Aligned cross entropy for non-autoregressive machine translation. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3515–3523. PMLR, 2020.

Uzay Gökay, Federico Spurio, Dominik R. Bach, and Juergen Gall. Skeleton motion words for unsupervised skeleton-based temporal action segmentation. In *IEEE/CVF International Conference on Computer Vision, ICCV 2025, Honolulu, HI, USA, October 19-25, 2025*, pp. 12101–12111. IEEE, 2025. doi: 10.1109/ICCV51701.2025.01125.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh Kumar Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Zhongcong Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Abrham Gebreselasie, Cristina González, James Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jáchym Kolář, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz, Merey Ramazanova, Leda Sari, Kiran Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Ziwei Zhao, Yunyi Zhu, Pablo Arbeláez, David Crandall, Dima Damen, Giovanni Maria Farinella, Christian Fuegen, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18995–19012, June 2022.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, pp. 369–376, 2006.

Keigo Hojo, Yukoh Wakabayashi, Kengo Ohta, Atsunori Ogawa, and Norihide Kitaoka. Boosting ctc-based ASR using inter-layer attention-based CTC loss. In Itshak Lapidot and Sharon Gannot (eds.), *25th Annual Conference of the International Speech Communication Association, Interspeech 2024, Kos, Greece, September 1-5, 2024*. ISCA, 2024. doi: 10.21437/INTERSPEECH.2024-1776.

Eungbeom Kim, Hantae Kim, and Kyogu Lee. Guiding frame-level CTC alignments using self-knowledge distillation. In Itshak Lapidot and Sharon Gannot (eds.), *25th Annual Conference of the International Speech Communication Association, Interspeech 2024, Kos, Greece, September 1-5, 2024*. ISCA, 2024. doi: 10.21437/INTERSPEECH.2024-363.

Hilde Kuehne, Ali Arslan, and Thomas Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

Vladimir I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8):707–710, 1966.

Longhao Li, Yangze Li, Hongfei Xue, Jie Liu, Shuai Fang, Kai Wang, and Lei Xie. Delayed-kd: Delayed knowledge distillation based CTC for low-latency streaming ASR. In Odette Scharenborg, Catharine Oertel, and Khiet Truong (eds.), *26th Annual Conference of the International Speech Communication Association, Interspeech 2025, Rotterdam, The Netherlands, 17-21 August 2025*. ISCA, 2025. doi: 10.21437/INTERSPEECH.2025-1691.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2980–2988, 2017.

Zijia Lu and Ehsan Elhamifar. Fact: Frame-action cross-attention temporal modeling for efficient action segmentation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18175–18185. IEEE, 2024.

Arthur Mensch and Mathieu Blondel. Differentiable dynamic programming for structured prediction and attention. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3462–3471. PMLR, 2018.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An ASR corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5206–5210. IEEE, 2015.

Mirco Ravanelli, Titouan Parcollet, Peter Plantinga, Aku Rouhe, Samuele Cornell, Loren Lugosch, Cem Subakan, Nauman Dawalatabad, Abdelwahab Heba, Jianyuan Zhong, et al. Speechbrain: A general-purpose speech toolkit. *arXiv preprint arXiv:2106.04624*, 2021.

Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(1):43–49, 1978. doi: 10.1109/TASSP.1978.1163055.

Federico Spurio, Emad Bahrami, Gianpiero Francesca, and Juergen Gall. Hierarchical vector quantization for unsupervised action segmentation. In Toby Walsh, Julie Shah, and Zico Kolter (eds.), *Thirty-Ninth AAAI Conference on Artificial Intelligence, Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence, Fifteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2025, Philadelphia, PA, USA, February 25 - March 4, 2025*, pp. 6996–7005. AAAI Press, 2025. doi: 10.1609/AAAI.V39I7.32751.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, 2016. doi: 10.1109/CVPR.2016.308.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Haitao Tian. Duoclr: Dual-surrogate contrastive learning for skeleton-based human action segmentation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13772–13782. IEEE, 2025.

Syed Talal Wasim, Hamid Suleman, Olga Zatsarynna, Muzammal Naseer, and Juergen Gall. Mixant: Observation-dependent memory propagation for stochastic dense action anticipation. In *IEEE/CVF International Conference on Computer Vision, ICCV 2025, Honolulu, HI, USA, October 19-25, 2025*, pp. 14613–14622. IEEE, 2025. doi: 10.1109/ICCV51701.2025.01356.

Angchi Xu and Wei-Shi Zheng. Efficient and effective weakly-supervised action segmentation via action-transition-aware boundary alignment. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18253–18262. IEEE, 2024.

Di Yang, Yaohui Wang, Antitza Dantcheva, Quan Kong, Lorenzo Garattoni, Gianpiero Francesca, and François Brémond. LAC - latent action composition for skeleton-based action segmentation. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pp. 13633–13644. IEEE, 2023. doi: 10.1109/ICCV51070.2023.01258.

Zengwei Yao, Wei Kang, Xiaoyu Yang, Fangjun Kuang, Liyong Guo, Han Zhu, Zengrui Jin, Zhaoqing Li, Long Lin, and Daniel Povey. CR-CTC: consistency regularization on CTC for improved speech recognition. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

Olga Zatsarynna, Emad Bahrami, Yazan Abu Farha, Gianpiero Francesca, and Juergen Gall. MANTA: diffusion mamba for efficient and effective stochastic long-term dense action anticipation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pp. 3438–3448. Computer Vision Foundation / IEEE, 2025. doi: 10.1109/CVPR52734.2025.00326.

Mingyuan Zhang and Shivani Agarwal. Bayes consistency vs. h-consistency: The interplay between surrogate loss functions and the scoring function class. In *Advances in Neural Information Processing Systems*, volume 33, pp. 16927–16936, 2020.

Qi Zhao, Shijie Wang, Ce Zhang, Changcheng Fu, Minh Quan Do, Nakul Agarwal, Kwonjoon Lee, and Chen Sun. Antgpt: Can large language models help long-term action anticipation from videos? In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 56677–56697, 2024.

A Appendix

This appendix provides supplementary statements, experimental details, theoretical derivations, and additional validation of PEAL. Appendix B provides the ethics and reproducibility statements. Experimental protocols, model configurations, training objectives, decoding procedures, and evaluation details are provided in Appendix C. Additional training-objective, hyperparameter, and prediction-horizon ablations are reported in Appendix D, while qualitative examples and further discussion of PEAL's alignment semantics are provided in Appendix E.

The theoretical development begins with the notation, risk definitions, surrogate-objective relationship, and applicability conditions established in Appendix F. The equivalence between the dynamic-programming formulation and the complete edit-path summation, together with the induced Gibbs edit-path distribution, is derived in Appendix G. Appendix H derives the gradients of PEAL with respect to prediction probabilities and logits and analyzes the sensitivity to insertion and deletion costs. Appendix I establishes the per-sample approximation bounds relating PEAL to edit distance, including the one-sided bound for arbitrary probability outputs.

The risk-level transfer and conditional empirical H-consistency theorem are proved in Appendix J, which also gives verifiable sufficient conditions for finite equally weighted samples and discusses the scope of the guarantee. Finally, Appendix K constructs a counterexample showing that CE can fail to be H-consistent with respect to edit distance, while Appendix L enumerates the candidate outputs, provides the corresponding risk calculations, and verifies the PEAL consistency certificate.

B Additional Statements

B.1 Ethics Statement

This work follows the ICLR Code of Ethics. We use only publicly released research datasets, including Ego4D for long-term action anticipation, Breakfast for temporal action segmentation, and the Gronings speech dataset for automatic speech recognition. These datasets were accessed and used in accordance with their respective licenses, terms of use, and established research guidelines. We did not collect new data, recruit human participants, or conduct experiments involving human subjects or animals. No personally identifiable information was collected or used beyond the content made available in the original datasets, and we did not attempt to identify individuals or infer sensitive attributes. We also did not perform experiments that introduced additional privacy or security risks. Throughout the study, we aim to maintain responsible data use, transparency, and research integrity.

B.2 Reproducibility Statement

We have made every effort to ensure that the results reported in this paper are reproducible. The formulation of PEAL, its dynamic-programming implementation, and the corresponding path-summation equivalence are specified in Sections 2–2.3 and appendices. The assumptions, approximation bounds, conditional H-consistency results, and complete proofs are provided in Appendices F and J.

The datasets, data splits, model architectures, optimization settings, decoding procedures, evaluation metrics, and ablation protocols used in our experiments are documented in Appendix C and Appendix D. These details are intended to support independent verification and further research based on PEAL.

C Experimental Protocols and Reproducibility Details

This appendix describes the data, models, training objectives, decoding procedures, evaluation scripts, and reproduction entry points used in the main experiments. LTA and TAS follow their respective

official data organization and evaluation implementations. Comparisons in the main tables are conducted within each task.

C.1 LTA: SlowFast Configuration

The SlowFast experiments use pre-extracted video features from Ego4D LTA. Each sample contains eight consecutive observed clips and 20 subsequent target actions, each comprising a verb and a noun. Input features are video representations extracted by the two-pathway SlowFast network. After spatial pooling, the 2048-dimensional and 256-dimensional pathway features are concatenated into a 2304-dimensional vector. The prediction model uses an encoder–decoder architecture. A six-layer, eight-head Transformer encoder aggregates input features, and the decoder outputs logits for 117 verb classes and 521 noun classes. The model predicts 20-step action sequences and generates five candidates per input.

Predicted sequences are sampled from the class softmax distributions. Following the official Ego4D LTA protocol, the evaluation score is the minimum normalized edit distance between the reference sequence and any of the five generated hypotheses. Validation samples are processed in a fixed order with a batch size of 512. All comparison checkpoints are reevaluated using identical validation samples, input features, and evaluation procedures to ensure a fair comparison.

C.2 LTA: Qwen-3.5-4B Configuration

The Qwen-3.5-4B experiment used the original videos from Ego4D LTA as the input. The base language model is fine-tuned using low-rank adaptation (LoRA), with parallel verb and noun prediction heads. LoRA uses rank $r = 32$, $\alpha = 64$, and dropout $= 0.05$; the prediction length is fixed at 20. Training uses two GPUs with distributed execution and a global effective batch size of 16. The optimizer is AdamW with weight decay 0.01.

The first 280 updates jointly optimize the LoRA parameters and prediction heads, whereas the final 20 updates fine-tune only the prediction heads as a linear probe, with the base model and LoRA parameters frozen. All objectives use the same prespecified training schedule and optimization settings; the training objective is the only intended difference.

C.3 TAS: Breakfast and UVAST-S Configuration

TAS experiments use Breakfast split 1, with 1,460 training videos, 252 test videos, 2048-dimensional pre-extracted features, and 48 action classes. UVAST-S uses a 10-layer encoder, a two-layer decoder, 64 feature-map channels, and a decoder feed-forward dimension of 2048. It has approximately 1.1 million parameters and dropout 0.1.

For a fair comparison, all objectives continue training from the same pretrained checkpoint at epoch 500 through epoch 600, using a fixed random seed. Training uses Adam, an initial learning rate of 0.0005, weight decay 0.00001, and batch size 1. Each epoch traverses all 1,460 training samples, yielding 146,000 updates in the experimental training phase. Evaluation uses the official Viterbi decoding pipeline with a frame sampling rate of 5. Average action durations are estimated from the current training split. To accelerate training, no intermediate inference is performed; only the final checkpoint is evaluated.

C.4 ASR: Gronings and XLS-R Configuration

ASR experiments use the Gronings speech data released with Bartelds et al. (2023). Using a fixed random seed of 4892, we select 217 manually transcribed utterances totaling approximately 24 minutes from the training split. We fine-tune Wav2Vec2-XLS-R-300M on 16 kHz mono audio with a 33-class character vocabulary, including blank. Training uses AdamW with a learning rate of 10^{-5}, a per-device batch size of 8, and four gradient accumulation steps, yielding an effective batch size of 32. Evaluation uses greedy decoding with consecutive-repeat merging and blank removal, reporting word error rate (WER) and character error rate (CER).

We compare CE, CTC, AXE, FL, LS, and PEAL. The baseline uses the native CTC objective; AXE, FL, LS, and PEAL are added as auxiliary objectives to CTC. We separately construct a frame-aligned

Table 6: Final held-out test-set results for the ten PEAL parameter configurations on Breakfast split 1. Hyperparameter selection was performed using an inner validation split of the official training data; the test set was not used for selection.

ID	τ	α	c_{ins}	c_{del}	Avg F1	Accuracy
base	0.10	2.0	1.0	1.0	67.330	63.402
tau05	0.05	2.0	1.0	1.0	67.922	65.376
tau20	0.20	2.0	1.0	1.0	67.387	65.423
alpha1	0.10	1.0	1.0	1.0	65.556	63.566
alpha3	0.10	3.0	1.0	1.0	68.004	63.918
ci12	0.10	2.0	1.2	1.0	**69.439**	**67.190**
cd12	0.10	2.0	1.0	1.2	68.669	65.551
ci08	0.10	2.0	0.8	1.0	67.156	64.368
cd08	0.10	2.0	1.0	0.8	65.963	63.446
tau15	0.15	2.0	1.0	1.0	67.983	64.789

CE baseline because acoustic output frames and reference characters have different sequence lengths. A trained CTC teacher, taken from the supervised baseline at step 12,000, generates the highest-probability CTC path consistent with each training transcript through forced alignment. These paths provide fixed character and blank labels for every valid acoustic frame. The student is trained exclusively with ordinary cross-entropy averaged over valid frames, excluding padding. The CE student is initialized from the original self-supervised XLS-R weights with a randomly initialized character classification head. The convolutional feature encoder is frozen, while the Transformer encoder and classification head are updated.

D Training-Objective and Hyperparameter Ablations

D.1 TAS Parameter Combinations

Table 6 reports the final held-out test-set results for all ten PEAL configurations on Breakfast split 1. Hyperparameter selection was performed exclusively on an inner validation split of the official training data. Specifically, we reserved one-fifth of the official training videos as an inner validation set using a video-level split, preventing frames or segments from the same video from appearing in both subsets. The remaining four-fifths were used for the initial training of all candidates. The epoch-500 UVAST-S checkpoint used as the common warm start was trained without any of the videos assigned to this inner validation split. Therefore, none of the validation videos had been observed by the warm-start checkpoint or by the candidate models before validation-based selection. The official test set was kept untouched throughout this selection stage.

All ten candidates shared the same UVAST-S warm start, batch size, training budget, random seed, and official Viterbi evaluation pipeline. We selected the configuration with the highest validation Avg F1, using Accuracy as a secondary criterion. After this selection, each of the ten configurations was retrained from the same warm-start checkpoint on the complete official training split and evaluated on the held-out test set under identical conditions. The values reported in Table 6 are the resulting test-set results from these complete-training runs. The test-set results were not used to select the configuration. The selected configuration was $\tau = 0.10$, $\alpha = 2.0$, $c_{\text{ins}} = 1.2$, and $c_{\text{del}} = 1.0$.

D.2 Prediction-horizon Analysis

We evaluate SlowFast output prefixes of length $K \in \{10, 15, 20\}$. ED@K compares the first K predicted elements with the corresponding target prefix, separately for verbs and nouns. Table 7 reports these prefix-based reevaluation results separately from the main results.

As K increases from 10 to 20, PEAL's Verb ED@K decreases, whereas its Noun ED@K increases less than those of the comparison methods. PEAL achieves the lowest Verb ED@K at all three horizons and the lowest Noun ED@K at $K = 15$ and $K = 20$. Its advantage over the strongest baseline widens for verbs, while its noun performance improves from below the best baseline to

Table 7: Prefix ED@K on SlowFast (Ego4D LTA) for $K \in \{10, 15, 20\}$. Lower is better; the best result in each column is bold.

	Verb ED@$K \downarrow$			Noun ED@$K \downarrow$		
Objective	K=10	K=15	K=20	K=10	K=15	K=20
CE	0.7425	0.7496	0.7546	0.7719	0.7830	0.7909
LS-Joint	0.7184	0.7245	0.7279	**0.7518**	0.7623	0.7694
FL-Joint	0.7137	0.7208	0.7242	0.7541	0.7643	0.7713
AXE-Joint	0.7063	0.7108	0.7140	0.7525	0.7615	0.7682
PEAL-Joint	**0.6962**	**0.6920**	**0.6875**	0.7574	**0.7603**	**0.7628**

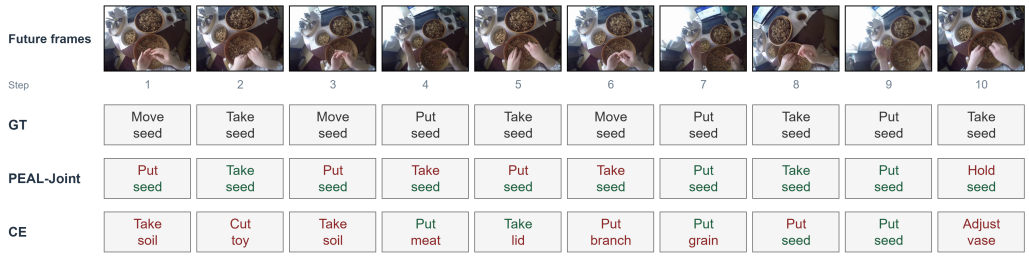

Figure 5: Qualitative example on Ego4D LTA.

above it. These results support PEAL's effectiveness over longer prediction horizons, consistent with its ability to align complete sequences while accommodating positional shifts.

E Qualitative Examples and Discussion

E.1 Qualitative Examples

Figure 5 compares ten future actions predicted by the SlowFast-based model trained with CE or PEAL-Joint on the LTA task. Green and red indicate correct and incorrect predictions, respectively. At positions 7–9, PEAL-Joint preserves the complete subsequence "Put seed $\rightarrow$ Take seed $\rightarrow$ Put seed", whereas CE predicts "Put grain $\rightarrow$ Put seed $\rightarrow$ Put seed". This example visually demonstrates that PEAL-Joint can mitigate CE's myopia and improve the model's structured prediction capability.

E.2 Discussion

Dynamic time warping (DTW) aligns sequences by allowing one element to match multiple elements in the other sequence while preserving their order, thereby accommodating differences in timing and duration (Sakoe & Chiba, 1978). Soft-DTW retains this matching rule but replaces the minimum alignment cost with a smooth approximation that incorporates all valid alignments, enabling gradient-based training (Cuturi & Blondel, 2017). Figure 6 contrasts the matching rule of Soft-DTW with PEAL for prediction (A, A, B, C) and target (A, B, C, C). Panel (a) shows a valid Soft-DTW alignment: both predicted A elements match the single target A, and the single predicted C matches both target C elements. Panel (b) shows how PEAL explicitly accounts for the extra A through deletion and the missing C through insertion, while retaining one-to-one, ordered matches for A, B, and C.

F Notation, Risks, and Applicability Conditions

F.1 Sequences, Edit Paths, and Risks

Notations. Let $\mathcal{Y} = \{1, \ldots, C\}$ denote the label space. Given input x, let $z_{i,c}$ denote the logit for class c at prediction position i, where $i \in \{1, \ldots, n\}$ and $c \in \mathcal{Y}$. The corresponding class

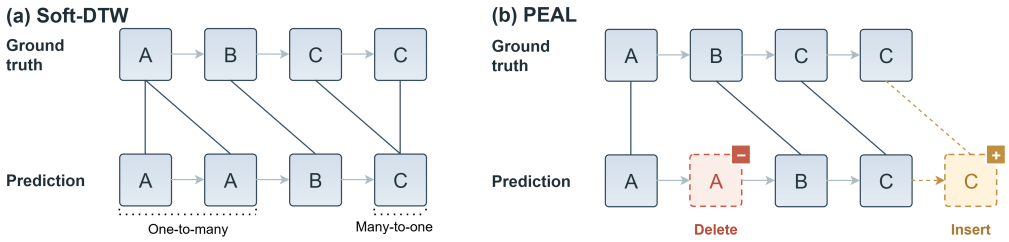

Figure 6: Comparison of Soft-DTW and PEAL alignments.

probabilities and decoded labels are

$$p_{i,c} = \frac{e^{z_{i,c}}}{\sum_{r=1}^{C} e^{z_{i,r}}}, \qquad \widehat{y}_i = \arg\max_{c \in \mathcal{Y}} p_{i,c}, \tag{8}$$

where ties are broken by a fixed, prespecified rule. We denote the class-probability matrix by $Z = h_\theta(x) = (p_{i,c}) \in [0,1]^{n \times C}$. Denote the complete decoded sequence by $\widehat{y}_h(x) = (\widehat{y}_1, \ldots, \widehat{y}_n) \in \mathcal{Y}^n$. After removing padding symbols and invalid tokens, denote the target sequence by $y = (y_1, \ldots, y_m) \in \mathcal{Y}^m$, with $j \in \{1, \ldots, m\}$ indexing target positions. Thus, $\widehat{y}_h(x)$ is the model's decoded prediction for input x, and y is the corresponding target sequence. Their lengths, n and m, need not be equal. Edit paths perform deletion, insertion, and substitution steps between the two sequences.

Let $\Pi_{n,m}$ be the set of all edit paths from $(0,0)$ to (n,m). Each path consists of deletion steps $D = (1,0)$, insertion steps $I = (0,1)$, and substitution steps $S = (1,1)$. A substitution step consumes one predicted label and one target label. When $\widehat{y}_i = y_j$, its discrete cost is zero, implicitly representing a match; otherwise, it incurs one substitution cost. Let $N_D(\pi)$ and $N_I(\pi)$ denote the numbers of deletion and insertion steps in path π, respectively. Let $S(\pi)$ denote the set of position pairs at which the path takes substitution steps. The discrete edit cost with unit operation costs is

$$C_{\mathrm{ED}}(\pi; \widehat{y}, y) = N_D(\pi) + N_I(\pi) + \sum_{(i,j) \in S(\pi)} \mathbf{1}\{\widehat{y}_i \neq y_j\}. \tag{9}$$

We define the Levenshtein edit distance used in this paper as

$$d_{\mathrm{ED}}(\widehat{y}, y) = \min_{\pi \in \Pi_{n,m}} C_{\mathrm{ED}}(\pi; \widehat{y}, y). \tag{10}$$

Thus, d_{ED} is the minimum path cost allowing only insertions, deletions, and substitutions, each with cost 1. Adjacent transpositions are excluded. PEAL assigns continuous probabilistic costs to the same path space. Let

$$q_{i,j} = (1 - p_{i,y_j})^\alpha, \tag{11}$$

where $\alpha > 0$ is the local cost exponent. Given insertion and deletion costs $c_{\mathrm{ins}}, c_{\mathrm{del}} > 0$, the PEAL cost of path π is

$$C_p(\pi; Z, y) = N_D(\pi) c_{\mathrm{del}} + N_I(\pi) c_{\mathrm{ins}} + \sum_{(i,j) \in S(\pi)} q_{i,j}. \tag{12}$$

These path definitions allow general n and m. For risk analysis, we additionally fix both prediction and target lengths to n, and set $c_{\mathrm{ins}} = c_{\mathrm{del}} = 1$. Define the normalized target loss as

$$\ell_{\mathrm{E}}(h; x, y) = \frac{1}{n} d_{\mathrm{ED}}(\widehat{y}_h(x), y) \in [0,1]. \tag{13}$$

Let $\mathcal{D}$ be a fixed distribution over input–target sequence pairs (X, Y), and let H be a fixed, nonempty probabilistic hypothesis class. For any loss ℓ, define the risk under $\mathcal{D}$, the minimum risk within H, and the excess risk as

$$R_\ell^{\mathcal{D}}(h) = \mathbb{E}_{(X,Y)\sim\mathcal{D}}[\ell(h; X, Y)], \tag{14}$$

$$R_{\ell,H}^{\mathcal{D},*} = \inf_{h\in H} R_\ell^{\mathcal{D}}(h), \tag{15}$$

$$\mathcal{E}_{\ell,H}^{\mathcal{D}}(h) = R_\ell^{\mathcal{D}}(h) - R_{\ell,H}^{\mathcal{D},*}. \tag{16}$$

All comparisons take infima over the same nonempty hypothesis class H. The results therefore directly characterize optimal decisions within a given model class rather than assuming an unrestricted Bayes class. This formulation follows the hypothesis-class-relative risk perspective of H-consistency bounds in Awasthi et al. (2022).

Fix a distribution $\mathcal{D}$, hypothesis class H, surrogate loss ℓ_{S}, and target loss ℓ_{E}. Suppose there exists a nondecreasing function $f : [0,\infty) \to [0,\infty)$, with $f(0) = 0$ and $\lim_{t\downarrow 0} f(t) = 0$, such that every $h \in H$ satisfies

$$\mathcal{E}_{\mathrm{E},H}^{\mathcal{D}}(h) \le f\left(\mathcal{E}_{\mathrm{S},H}^{\mathcal{D}}(h)\right), \tag{17}$$

Then ℓ_{S} is conditionally H-consistent with respect to ℓ_{E} on $(\mathcal{D}, H)$. This means that target risk approaches its within-class optimum whenever surrogate risk approaches its optimum within the same class. We establish a conditional result for fixed $(\mathcal{D}, H)$. We do not claim that it holds for arbitrary distributions, arbitrary hypothesis classes, or unrestricted function classes.

Specifically, let T denote a fixed model architecture and output protocol, and let H_T be the induced probabilistic hypothesis class. Let $\mathcal{D}_X$ denote the input marginal of $\mathcal{D}$. For $0 \le \xi < 1/2$, define the uniformly high-confidence class as

$$\mathcal{H}^\star(T,\xi) = \left\{ h \in H_T : \max_{c\in\mathcal{Y}} p_{h,i,c}(x) \ge 1-\xi,\ \forall i,\ \mathcal{D}_X\text{-almost every } x \right\}. \tag{18}$$

The conditional H-consistency results in Appendices I–J hold for any fixed nonempty subclass $\varnothing \ne H \subseteq \mathcal{H}^\star(T,\xi)$.

F.2 Relationship between the Training Objective and the Theoretical Surrogate

This section relates the practical PEAL training objective to the surrogate used in the theoretical analysis. Both prediction and target lengths are fixed to n, so the scaling factor and path count below are independent of the model.

Let $K_{n,n} = |\Pi_{n,n}|$ denote the total number of valid edit paths. The hard-path objective, per-sample PEAL at temperature $\tau > 0$, and its path-count-corrected form are, respectively,

$$L_0(Z, y) = \min_{\pi\in\Pi_{n,n}} C_p(\pi; Z, y), \tag{19}$$

$$L_\tau(Z, y) = -\tau \log \sum_{\pi\in\Pi_{n,n}} \exp\left[-\frac{C_p(\pi; Z, y)}{\tau}\right], \qquad \overline{L}_\tau(Z, y) = L_\tau(Z, y) + \tau \log K_{n,n}. \tag{20}$$

When $Z = h(x)$, these three quantities are also written as $L_0(h; x, y)$, $L_\tau(h; x, y)$, and $\overline{L}_\tau(h; x, y)$. To simplify subsequent derivations and place PEAL on the same scale as normalized edit distance, define the theoretical surrogate

$$\ell_{\mathrm{S},\tau}(Z, y) = \frac{\overline{L}_\tau(Z, y)}{n} = \frac{L_\tau(Z, y) + \tau \log K_{n,n}}{n}. \tag{21}$$

Let $R^{\mathcal{D}}_{L_\tau}(h)$ denote the population risk of the practical per-sample PEAL loss L_τ. Then

$$R^{\mathcal{D}}_{\mathrm{S},\tau}(h) = \frac{1}{n} R^{\mathcal{D}}_{L_\tau}(h) + \frac{\tau \log K_{n,n}}{n}. \tag{22}$$

Since n and $K_{n,n}$ are fixed in this setting, neither scaling nor the additive correction changes risk-optimal models. The corresponding excess risks satisfy

$$\mathcal{E}^{\mathcal{D}}_{\mathrm{S},\tau,H}(h) = \frac{1}{n} \mathcal{E}^{\mathcal{D}}_{L_\tau,H}(h). \tag{23}$$

Thus, zero theoretical surrogate excess risk is equivalent to zero practical PEAL excess risk. Any sequence approaching optimal practical PEAL risk also approaches optimal theoretical surrogate risk, and vice versa. The path-count correction compensates for the soft minimum's worst-case path-count offset. Fixed scaling facilitates comparison with normalized edit distance. Neither transformation changes the optimal decisions addressed by the subsequent H-consistency results.

F.3 Path Counts

The path count $K_{n,m}$ is a Delannoy number satisfying

$$K_{0,0} = 1, \qquad K_{i,0} = K_{0,j} = 1, \tag{24}$$

$$K_{i,j} = K_{i-1,j} + K_{i,j-1} + K_{i-1,j-1}. \tag{25}$$

An equivalent closed form is

$$K_{n,m} = \sum_{k=0}^{\min(n,m)} \binom{n}{k} \binom{m+k}{k}. \tag{26}$$

G Equivalence of Dynamic Programming and Path Summation

G.1 Path Partition Function Formulation

We express the sum of exponential weights over complete edit paths using a path partition function, and relate it to the soft Bellman states.

Define the path partition function from $(0, 0)$ to (i, j) as

$$G_{i,j} = \sum_{\pi \in \Pi_{i,j}} \exp\left(-\frac{C_p(\pi)}{\tau}\right). \tag{27}$$

Then

$$R_{i,j} = -\tau \log G_{i,j} \tag{28}$$

satisfies the smooth Bellman recurrence derived below, and

$$R_{n,m} = L_\tau. \tag{29}$$

G.2 Recurrence Proof by Final-Step Decomposition

We partition paths by their final step to derive a three-way soft Bellman recurrence for computing PEAL.

Any path reaching (i, j) must end with a deletion, insertion, or substitution. Partitioning by this final step gives

$$\begin{aligned} G_{i,j} = e^{-c_{\text{del}}/\tau} G_{i-1,j} + e^{-c_{\text{ins}}/\tau} G_{i,j-1} \\ + e^{-q_{i,j}/\tau} G_{i-1,j-1}. \end{aligned} \tag{30}$$

Using $R_{a,b} = -\tau \log G_{a,b}$, rewrite this as

$$G_{i,j} = e^{-(R_{i-1,j}+c_{\text{del}})/\tau} + e^{-(R_{i,j-1}+c_{\text{ins}})/\tau} + e^{-(R_{i-1,j-1}+q_{i,j})/\tau}. \tag{31}$$

Applying $-\tau \log$ to both sides yields

$$R_{i,j} = \text{softmin}_\tau \left(R_{i-1,j} + c_{\text{del}}, R_{i,j-1} + c_{\text{ins}}, R_{i-1,j-1} + q_{i,j} \right). \tag{32}$$

Each boundary admits only one path, consisting entirely of deletions or insertions. The path partition function therefore satisfies

$$G_{0,0} = 1, \qquad G_{i,0} = e^{-ic_{\text{del}}/\tau}, \qquad G_{0,j} = e^{-jc_{\text{ins}}/\tau}. \tag{33}$$

The corresponding soft Bellman boundary conditions are

$$R_{0,0} = 0, \qquad R_{i,0} = ic_{\text{del}}, \qquad R_{0,j} = jc_{\text{ins}}. \tag{34}$$

Recursing to (n, m) yields the soft minimum over the complete path set. The soft Bellman recurrence therefore implements temperature-controlled soft minimization over complete edit paths. This completes the proof.

G.3 Origin of the Gibbs Edit-Path Distribution

We normalize exponential path weights to obtain the Gibbs edit-path distribution and express PEAL gradients in a unified expectation form.

Temperature-controlled soft minimization of path costs yields a loss value and naturally induces a probability distribution over complete edit paths. For each $\pi \in \Pi_{n,m}$, define its unnormalized Boltzmann weight as

$$w_\tau(\pi; Z, y) = \exp\left[-\frac{C_p(\pi; Z, y)}{\tau} \right]. \tag{35}$$

By the path partition function in Section G.1, the sum of all path weights is

$$\mathcal{Z}_\tau(Z, y) = \sum_{\pi \in \Pi_{n,m}} w_\tau(\pi; Z, y) = G_{n,m}, \qquad L_\tau(Z, y) = -\tau \log \mathcal{Z}_\tau(Z, y). \tag{36}$$

Dividing each path weight by this normalizing constant yields the Gibbs edit-path distribution

$$\boxed{P_\tau(\pi \mid Z, y) = \frac{w_\tau(\pi; Z, y)}{\mathcal{Z}_\tau(Z, y)} = \frac{\exp[-C_p(\pi; Z, y)/\tau]}{\sum_{\pi' \in \Pi_{n,m}} \exp[-C_p(\pi'; Z, y)/\tau]}.} \tag{37}$$

This distribution satisfies $\sum_\pi P_\tau(\pi \mid Z, y) = 1$. Lower-cost paths receive greater weights. Lower temperatures concentrate probability mass on low-cost paths, whereas higher temperatures allow

more near-optimal paths to influence the soft minimum. Consider any scalar parameter u entering the objective only through path cost C_p. With the path set and temperature fixed, differentiating the log-partition form directly gives

$$\frac{\partial L_\tau}{\partial u} = \sum_{\pi \in \Pi_{n,m}} P_\tau(\pi \mid Z, y) \frac{\partial C_p(\pi; Z, y)}{\partial u}. \tag{38}$$

Thus, PEAL gradients are expectations of the corresponding path statistics under the Gibbs path distribution. Appendix H applies this identity to local substitution costs, position-level class logits, and insertion and deletion costs.

H Gradient Derivations for PEAL

H.1 Gradients with Respect to Prediction Probabilities

We derive the gradient of PEAL with respect to prediction probabilities and explain how path costs determine each path's contribution. Throughout this derivation, the path set, temperature $\tau > 0$, and insertion and deletion costs are fixed. The probabilities satisfy $0 < p_{i,c} < 1$, as produced by softmax with finite logits.

The partition function and loss are

$$\mathcal{Z}_\tau(Z, y) = \sum_{\pi \in \Pi_{n,m}} \exp[-C_p(\pi; Z, y)/\tau], \qquad L_\tau(Z, y) = -\tau \log \mathcal{Z}_\tau(Z, y). \tag{39}$$

We differentiate this expression directly with respect to $p_{i,c}$:

$$\begin{aligned} \frac{\partial L_\tau}{\partial p_{i,c}} &= -\frac{\tau}{\mathcal{Z}_\tau} \sum_{\pi \in \Pi_{n,m}} \exp[-C_p(\pi; Z, y)/\tau] \left(-\frac{1}{\tau}\right) \frac{\partial C_p(\pi; Z, y)}{\partial p_{i,c}} \\ &= \sum_{\pi \in \Pi_{n,m}} P_\tau(\pi \mid Z, y) \frac{\partial C_p(\pi; Z, y)}{\partial p_{i,c}}, \end{aligned} \tag{40}$$

where the gradient weight of path π is

$$P_\tau(\pi \mid Z, y) = \frac{\exp[-C_p(\pi; Z, y)/\tau]}{\sum_{\pi' \in \Pi_{n,m}} \exp[-C_p(\pi'; Z, y)/\tau]}. \tag{41}$$

These weights sum to one and follow a Gibbs distribution. For any two paths π and π',

$$\frac{P_\tau(\pi \mid Z, y)}{P_\tau(\pi' \mid Z, y)} = \exp\left[-\frac{C_p(\pi; Z, y) - C_p(\pi'; Z, y)}{\tau}\right]. \tag{42}$$

Thus, a lower-cost path receives a greater gradient weight.

We define the indicator of a diagonal step as

$$I_{i,j}(\pi) = \begin{cases} 1, & \pi \text{ contains the step } (i-1, j-1) \to (i, j), \\ 0, & \text{otherwise.} \end{cases} \tag{43}$$

This step includes both matches and mismatches. The path cost can then be written as

$$C_p(\pi; Z, y) = N_D(\pi) c_{\text{del}} + N_I(\pi) c_{\text{ins}} + \sum_{i=1}^{n} \sum_{j=1}^{m} I_{i,j}(\pi)(1 - p_{i,y_j})^\alpha. \tag{44}$$

The insertion and deletion terms have zero derivatives with respect to prediction probabilities because their costs and operation counts are fixed for each path. The substitution terms give

$$\frac{\partial C_p(\pi; Z, y)}{\partial p_{i,c}} = -\alpha(1 - p_{i,c})^{\alpha-1} \sum_{j: y_j = c} I_{i,j}(\pi). \tag{45}$$

Consequently, the probability gradient is

$$\boxed{\frac{\partial L_\tau}{\partial p_{i,c}} = -\alpha(1-p_{i,c})^{\alpha-1} \sum_{\pi\in\Pi_{n,m}} P_\tau(\pi \mid Z, y) \sum_{j:y_j=c} I_{i,j}(\pi).} \tag{46}$$

For a fixed prediction probability, this derivative becomes more negative as the weighted support for aligning position i with target label c increases. It therefore favors increasing probabilities that reduce substitution costs along highly weighted paths. Insertion and deletion costs influence this gradient indirectly by changing path costs and their associated weights.

The preference for minimum-cost paths becomes explicit in the low-temperature limit. We define

$$\Pi_* = \arg\min_{\pi\in\Pi_{n,m}} C_p(\pi; Z, y). \tag{47}$$

For fixed prediction probabilities,

$$\lim_{\tau\to 0^+} P_\tau(\pi \mid Z, y) = \begin{cases} 1/|\Pi_*|, & \pi \in \Pi_*, \\ 0, & \pi \notin \Pi_*. \end{cases} \tag{48}$$

The gradient therefore concentrates on minimum-cost paths as the temperature decreases, with equal limiting weights when several paths share the minimum cost.

H.2 From Prediction Probabilities to Logits

We propagate the probability gradient through softmax to obtain the gradient with respect to the model's logits. The prediction probabilities satisfy

$$p_{i,c} = \frac{\exp(z_{i,c})}{\sum_{r\in\mathcal{Y}} \exp(z_{i,r})}, \tag{49}$$

and the softmax Jacobian is

$$\frac{\partial p_{i,c}}{\partial z_{i,r}} = p_{i,c}\left(\mathbf{1}\{c=r\} - p_{i,r}\right). \tag{50}$$

For compact notation, we define the weighted contribution of diagonal step (i, j) as

$$W_{i,j} = \sum_{\pi\in\Pi_{n,m}} P_\tau(\pi \mid Z, y) I_{i,j}(\pi). \tag{51}$$

The probability gradient becomes

$$\frac{\partial L_\tau}{\partial p_{i,c}} = -\alpha(1-p_{i,c})^{\alpha-1} \sum_{j:y_j=c} W_{i,j}. \tag{52}$$

The chain rule then yields

$$\begin{aligned} \frac{\partial L_\tau}{\partial z_{i,r}} &= \sum_{c\in\mathcal{Y}} \frac{\partial L_\tau}{\partial p_{i,c}} \frac{\partial p_{i,c}}{\partial z_{i,r}} \\ &= -\sum_{j=1}^{m} W_{i,j}\, \alpha(1-p_{i,y_j})^{\alpha-1} p_{i,y_j} \left(\mathbf{1}\{r=y_j\} - p_{i,r}\right). \end{aligned} \tag{53}$$

When multiple target positions share a label, their weighted contributions jointly influence the corresponding prediction probability. The softmax Jacobian couples the class probabilities at each position, so their updates also depend on competing labels. The gradient establishes a preference for lower-cost alignments, rather than guaranteeing that every supported probability increases after each parameter update.

H.3 Sensitivity to Insertion and Deletion Costs

We distinguish derivatives with respect to prediction probabilities from sensitivity to the insertion and deletion costs themselves. Although these costs contribute no direct probability derivatives, varying either cost changes the loss.

For each fixed path,

$$\frac{\partial C_p(\pi; Z, y)}{\partial c_{\text{del}}} = N_D(\pi), \qquad \frac{\partial C_p(\pi; Z, y)}{\partial c_{\text{ins}}} = N_I(\pi). \tag{54}$$

The same weighted-gradient identity therefore gives

$$\frac{\partial L_\tau}{\partial c_{\text{del}}} = \sum_{\pi \in \Pi_{n,m}} P_\tau(\pi \mid Z, y) N_D(\pi), \qquad \frac{\partial L_\tau}{\partial c_{\text{ins}}} = \sum_{\pi \in \Pi_{n,m}} P_\tau(\pi \mid Z, y) N_I(\pi). \tag{55}$$

These derivatives describe sensitivity to the cost hyperparameters through weighted operation counts. Because the costs remain fixed during training, these expressions do not represent trainable parameter updates.

I Per-Sample Approximation of Edit Distance by PEAL

I.1 Bounds Relating Soft and Hard Objectives

For any Z, y and $\tau > 0$,

$$\boxed{L_0 - \tau \log K_{n,m} \le L_\tau \le L_0.} \tag{56}$$

Equivalently,

$$\boxed{L_0 \le \overline{L}_\tau \le L_0 + \tau \log K_{n,m}.} \tag{57}$$

Proof. Let $C_* = \min_\pi C_p(\pi) = L_0$. The partition function contains at least one optimal path, so

$$\sum_\pi e^{-C_p(\pi)/\tau} \ge e^{-C_*/\tau}, \tag{58}$$

which implies $L_\tau \le C_*$. Conversely, every path has cost at least C_*, so

$$\sum_\pi e^{-C_p(\pi)/\tau} \le K_{n,m} e^{-C_*/\tau}, \tag{59}$$

which implies $L_\tau \ge C_* - \tau \log K_{n,m}$. This completes the proof.

I.2 Local Substitution Costs under Uniform High Confidence

Fix any nonempty subclass

$$\varnothing \ne H \subseteq \mathcal{H}^\star(T, \xi). \tag{60}$$

For every $h \in H$:

1. If $\widehat{y}_{h,i}(x) = y_j$, then

$$q_{i,j} \le \xi^\alpha. \tag{61}$$

2. If $\widehat{y}_{h,i}(x) \neq y_j$, then

$$q_{i,j} \geq (1-\xi)^\alpha. \tag{62}$$

Proof. If the decoded label equals the target label, the uniform high-confidence condition gives probability at least $1-\xi$, and hence $q_{i,j} = (1-p_{i,y_j})^\alpha \leq \xi^\alpha$. If the labels differ, the decoded label itself has probability at least $1-\xi$. The total probability of all other labels is at most ξ, so the target label has probability at most ξ. Therefore, $q_{i,j} = (1-p_{i,y_j})^\alpha \geq (1-\xi)^\alpha$. This completes the proof.

I.3 Two-Sided Bounds Relating Hard PEAL and Edit Distance

With unit insertion and deletion costs,

$$\boxed{(1-\xi)^\alpha d_{\mathrm{ED}}(\widehat{y}_h(x), y) \leq L_0(h; x, y) \leq d_{\mathrm{ED}}(\widehat{y}_h(x), y) + n\xi^\alpha.} \tag{63}$$

Lower-bound proof. For any path, the PEAL costs of insertion and deletion are both 1. A substitution between mismatched labels costs at least $(1-\xi)^\alpha$, whereas a matching step has discrete edit cost 0. Since $(1-\xi)^\alpha \leq 1$, the probabilistic path cost is at least $(1-\xi)^\alpha$ times the discrete edit cost of the same path. Taking the minimum over paths proves the lower bound.

Upper-bound proof. Consider a path attaining the minimum discrete edit distance. Its insertion and deletion costs remain unchanged. Substitutions between mismatched labels satisfy $q_{i,j} \leq 1$, and those between matching labels satisfy $q_{i,j} \leq \xi^\alpha$. A path from $(0,0)$ to (n,n) contains at most n substitution steps, so its probabilistic cost is at most

$$d_{\mathrm{ED}}(\widehat{y}_h(x), y) + n\xi^\alpha. \tag{64}$$

Since L_0 cannot exceed the cost of any particular path, the upper bound follows. This completes the proof.

I.4 Per-Sample Edit-Distance Approximation

Define

$$r_- = 1 - (1-\xi)^\alpha, \tag{65}$$

$$r_+ = \xi^\alpha + \frac{\tau \log K_{n,n}}{n}. \tag{66}$$

Then, for every $h \in H$ and $\mathcal{D}$-almost every (x, y),

$$\boxed{-r_- \leq \ell_{\mathrm{S},\tau}(h; x, y) - \ell_{\mathrm{E}}(h; x, y) \leq r_+.} \tag{67}$$

Proof. By Appendices I.1 and I.3,

$$\ell_{\mathrm{S},\tau} \geq \frac{L_0}{n} \geq (1-\xi)^\alpha \ell_{\mathrm{E}}. \tag{68}$$

Since $0 \leq \ell_{\mathrm{E}} \leq 1$,

$$\ell_{\mathrm{S},\tau} - \ell_{\mathrm{E}} \geq -\left[1 - (1-\xi)^\alpha\right] = -r_-. \tag{69}$$

Conversely,

$$\ell_{\mathrm{S},\tau} \leq \frac{L_0}{n} + \frac{\tau \log K_{n,n}}{n} \leq \ell_{\mathrm{E}} + \xi^\alpha + \frac{\tau \log K_{n,n}}{n}, \tag{70}$$

so the upper bound is r_+. This completes the proof.

I.5 Supplementary Result: One-Sided Control for Arbitrary Probability Outputs

The preceding two-sided approximation requires uniform high confidence. As a supplementary result, arbitrary probability outputs satisfy a one-sided bound without this condition. Let

$$\eta = \min\left\{2^{-\alpha}, c_{\text{ins}}, c_{\text{del}}\right\}. \tag{71}$$

Then

$$L_0(Z, y) \geq \eta\, d_{\text{ED}}(\widehat{y}, y), \tag{72}$$

and consequently,

$$\boxed{d_{\text{ED}}(\widehat{y}, y) \leq \max\left\{2^{\alpha}, c_{\text{ins}}^{-1}, c_{\text{del}}^{-1}\right\} \overline{L}_\tau(Z, y).} \tag{73}$$

Proof. Consider any substitution step with $\widehat{y}_i \neq y_j$. Since $\widehat{y}_i$ is the argmax under a fixed tie-breaking rule, $p_{i,y_j} \leq 1/2$. Thus,

$$q_{i,j} = (1 - p_{i,y_j})^{\alpha} \geq 2^{-\alpha}. \tag{74}$$

Insertion and deletion steps have probabilistic path costs at least c_{ins} and c_{del}, respectively. Thus, for any path π, its probabilistic cost is at least η times its discrete edit cost. Minimizing over all paths yields

$$L_0 \geq \eta d_{\text{ED}}(\widehat{y}, y). \tag{75}$$

The result follows from $\overline{L}_\tau \geq L_0$. This completes the proof.

J From Per-Sample Approximation to Conditional H-Consistency

We lift the per-sample approximation in Appendix I to the risk level and prove the conditional H-consistency theorem in Section 2.4. An additive excess-risk bound provides an intermediate step. A consistency bound satisfying $f(0) = 0$ additionally requires exploiting a positive gap between edit-distance risk levels.

J.1 Additive Excess-Risk Bound

Let

$$\beta_{\xi,\tau} = r_- + r_+. \tag{76}$$

Under the conditions of Appendices F and I, every $h \in H$ satisfies

$$\boxed{\left|\mathcal{E}^{\mathcal{D}}_{\text{S},\tau,H}(h) - \mathcal{E}^{\mathcal{D}}_{\text{E},H}(h)\right| \leq \beta_{\xi,\tau}.} \tag{77}$$

Proof. Write

$$D(h) = R^{\mathcal{D}}_{\text{S},\tau}(h) - R^{\mathcal{D}}_{\text{E}}(h). \tag{78}$$

Taking expectations in the per-sample approximation of Appendix I.4 gives

$$D(h) \in [-r_-, r_+]. \tag{79}$$

Taking infima over $h \in H$ likewise gives

$$b := R^{\mathcal{D},*}_{\mathrm{S},\tau,H} - R^{\mathcal{D},*}_{\mathrm{E},H} \in [-r_-, r_+]. \tag{80}$$

By the definition of excess risk,

$$\mathcal{E}^{\mathcal{D}}_{\mathrm{S},\tau,H}(h) - \mathcal{E}^{\mathcal{D}}_{\mathrm{E},H}(h) = D(h) - b. \tag{81}$$

The absolute difference between any two points in this interval is at most $r_- + r_+$, which proves the result.

This additive bound immediately yields

$$\mathcal{E}^{\mathcal{D}}_{\mathrm{E},H}(h) \leq \mathcal{E}^{\mathcal{D}}_{\mathrm{S},\tau,H}(h) + \beta_{\xi,\tau}. \tag{82}$$

It applies to any fixed distribution and does not require empirical risks to lie on a discrete grid.

J.2 Full Proof of the Conditional H-Consistency Theorem

Define

$$g = g_{\mathrm{E},H,\mathcal{D}} = \inf\left\{\mathcal{E}^{\mathcal{D}}_{\mathrm{E},H}(h) : h \in H,\ \mathcal{E}^{\mathcal{D}}_{\mathrm{E},H}(h) > 0\right\}. \tag{83}$$

If the set of positive target excess risks is nonempty and

$$\beta_{\xi,\tau} < g, \tag{84}$$

then, for every $h \in H$,

$$\boxed{\mathcal{E}^{\mathcal{D}}_{\mathrm{E},H}(h) \leq \frac{1}{1 - \beta_{\xi,\tau}/g}\mathcal{E}^{\mathcal{D}}_{\mathrm{S},\tau,H}(h).} \tag{85}$$

If the set of positive target excess risks is empty, every hypothesis in H is target-optimal, and the result holds trivially.

Proof. Write

$$e = \mathcal{E}^{\mathcal{D}}_{\mathrm{E},H}(h), \qquad s = \mathcal{E}^{\mathcal{D}}_{\mathrm{S},\tau,H}(h), \qquad \beta = \beta_{\xi,\tau}. \tag{86}$$

The additive bound in Appendix J.1 gives $s \geq e - \beta$. The result holds when $e = 0$. When $e > 0$, the risk-gap definition gives $e \geq g$, so

$$\beta \leq \frac{\beta}{g} e. \tag{87}$$

Hence,

$$s \geq e - \beta \geq \left(1 - \frac{\beta}{g}\right) e. \tag{88}$$

Since $\beta < g$, rearranging proves the result.

Let

$$f(t) = \frac{t}{1 - \beta/g}, \qquad f(0) = 0, \qquad \lim_{t \downarrow 0} f(t) = 0. \tag{89}$$

The preceding inequality is precisely the H-consistency bound defined in Appendix F. Under the stated conditions, PEAL is therefore conditionally H-consistent with respect to edit distance on $(\mathcal{D}, H)$. If PEAL's minimum within-class risk is attained, every within-class PEAL minimizer is also an edit-distance minimizer. Even if the minimum is not attained, any sequence approaching optimal PEAL risk also approaches optimal edit-distance risk.

J.3 Verifiable Sufficient Conditions for Finite Equally Weighted Samples

Consider N equally weighted samples, fixed length n, and unit edit costs. Each sample has integer-valued edit distance, so the empirical target risk can only take values in

$$\left\{0, \frac{1}{Nn}, \frac{2}{Nn}, \ldots, 1\right\} \tag{90}$$

If positive target excess risk exists, then

$$g \geq \frac{1}{Nn}. \tag{91}$$

Therefore,

$$\boxed{\beta_{\xi,\tau} < \frac{1}{Nn}} \tag{92}$$

is a sufficient condition for H-consistency that can be verified without enumerating the risk levels of H. It yields

$$\boxed{\widehat{\mathcal{E}}_{\mathrm{E},H}(h) \leq \frac{1}{1 - Nn\beta_{\xi,\tau}} \widehat{\mathcal{E}}_{\mathrm{S},\tau,H}(h).} \tag{93}$$

When $\alpha = 2$,

$$\begin{aligned} \beta_{\xi,\tau} &= 1 - (1-\xi)^2 + \xi^2 + \frac{\tau \log K_{n,n}}{n} \\ &= 2\xi + \frac{\tau \log K_{n,n}}{n}. \end{aligned} \tag{94}$$

the parameter conditions can therefore be written directly as

$$\xi < \frac{1}{2Nn}, \tag{95}$$

$$0 < \tau < \frac{\frac{1}{N} - 2n\xi}{\log K_{n,n}}. \tag{96}$$

Interpretation of the sufficient condition. The result establishes optimality transfer on a fixed empirical distribution by combining uniform approximation with a positive target excess-risk gap. The condition $\beta_{\xi,\tau} < 1/(Nn)$ uses a conservative lower bound on that gap. The more general result in Appendix J.2 requires only $\beta_{\xi,\tau} < g$, where g is the smallest positive edit-distance excess risk within H. A larger actual gap therefore permits weaker approximation requirements, provided that a valid lower bound on g is available. The dependence on Nn reflects the finer empirical risk grid as the sample size or sequence length increases; it does not establish statistical consistency as N grows.

Meaning of the confidence assumption. Uniform high confidence concerns the model's predicted class, rather than the ground-truth class, and therefore does not imply accurate predictions. The CE counterexample illustrates this distinction: highly confident models can retain positive edit-distance excess risk. The assumption applies to every hypothesis in H at every empirical input and prediction position. Confidence measurements from individual checkpoints can describe those models, but cannot verify the assumption over the entire hypothesis class.

Relation to the experimental configurations. For the default LTA setting $n = 20$ and $\tau = 0.1$, the temperature term alone is

$$\frac{\tau \log K_{20,20}}{20} \approx 0.16597, \tag{97}$$

which exceeds $1/(20N)$ for every $N \geq 1$. Thus, this configuration does not satisfy the grid-based sufficient condition, even as the confidence error approaches zero. Failure of this sufficient condition does not establish inconsistency, but the present certificate cannot verify consistency for that configuration. The equal-length theorem also does not directly cover variable-length ASR or TAS sequences, joint training objectives, non-unit insertion or deletion costs, or decoding protocols other than position-wise argmax. Their practical effectiveness is evaluated empirically.

Broader bounds and statistical scope. The additive excess-risk bound in Appendix J.1 applies to any fixed distribution under the stated high-confidence assumptions, without requiring a positive risk gap. It controls target excess risk up to the approximation error $\beta_{\xi,\tau}$. Appendix I.5 additionally provides a one-sided edit-distance bound for arbitrary probability outputs. These results extend the relationship between PEAL and edit distance beyond the strict empirical consistency condition, but do not establish general population calibration. Generalization from empirical PEAL minimization to unseen data requires a separate analysis.

K Constructing a Counterexample to CE H-Consistency

This appendix provides a verifiable construction of the CE counterexample. It shows that within-class optimality under CE does not guarantee within-class edit-distance optimality. We use binary labels and five equally weighted samples with a one-to-one input–target correspondence. For a four-bit codeword b, let

$$\phi(b) = b_1^4 b_2^4 b_3^4 b_4^4. \tag{98}$$

Set

$$\begin{aligned}(x_1, x_2, x_3, x_4, x_5) &= (0000, 0001, 0010, 0011, 0100),\\ (t_1, t_2, t_3, t_4, t_5) &= (0000, 0100, 0101, 1010, 1011),\end{aligned} \tag{99}$$

and define $y_i = \phi(t_i)\|x_i$. The final four bits explicitly copy the input, while the first 16 bits carry the shared sequence structure.

The corresponding equally weighted empirical distribution is

$$\widehat{\mathcal{D}} = \frac{1}{5}\sum_{i=1}^{5} \delta_{(x_i, y_i)}. \tag{100}$$

Fix $0 < \xi < 1/2$. For any $s \in \{0,1\}^4$, define the decoded sequence of h_s on input x_i as $\phi(s)\|x_i$. Assign probability $1 - \xi$ to the decoded label and ξ to the other label. This yields a 16-element high-confidence codebook

$$H_\xi^{\text{code}} = \{h_s : s \in \{0,1\}^4\} \subseteq \mathcal{H}^\star(T, \xi). \tag{101}$$

The models explicitly read the input, so the counterexample does not rely on label ambiguity or input-independent models.

Compute the unit-cost Levenshtein risk for each of the 16 shared cores. Complete enumeration identifies $h_{0100}, h_{0101}, h_{1010}$ as the edit-distance-optimal models, with optimal risk 0.28 and next-best risk 0.32. Hence,

$$\widehat{g}_{\mathrm{E}, H_\xi^{\text{code}}} = 0.04. \tag{102}$$

The complete risk enumeration appears in Appendix L.

The final four bits match the target for every model. Each repeated four-bit block in the first 16 positions favors 0 by a 3:2 ratio across the five targets. Thus, the unique minimizer of ordinary CE is h_{0000}. Direct calculation gives

$$\widehat{\mathcal{E}}_{\mathrm{CE},H_\xi^{\mathrm{code}}}(h_{0000}) = 0, \qquad \widehat{\mathcal{E}}_{\mathrm{E},H_\xi^{\mathrm{code}}}(h_{0000}) = 0.32 - 0.28 = 0.04 = \frac{1}{25}. \tag{103}$$

CE excess risk is zero, but edit-distance excess risk is strictly positive. Thus, no function satisfying $f(0) = 0$ can transfer CE excess risk to edit-distance excess risk on this $(\widehat{\mathcal{D}}, H_\xi^{\mathrm{code}})$. CE is therefore not H-consistent with respect to edit distance for this fixed distribution and hypothesis class.

L Risk Enumeration for 16 Candidate Outputs and a PEAL Consistency Certificate

For each of the 16 shared cores $s \in \{0,1\}^4$ in Appendix K, compute unit-cost Levenshtein distances over the five paired samples and average them. The complete risk enumeration is

Table 8: Empirical edit-distance risks for the 16 candidate outputs.

Empirical edit-distance risk	Output count	Representative outputs
0.28	3	$0100, 0101, 1010$
0.32	2	$0000, 0010$
0.36	5	$0001, 0110, 1000, 1001, 1011$
0.40	3	$0011, 1100, 1101$
0.44	2	$0111, 1110$
0.48	1	1111

The edit-distance-optimal model set is therefore $\{h_{0100}, h_{0101}, h_{1010}\}$. The unique CE minimizer, h_{0000}, lies at risk level 0.32, and the positive risk gap is 0.04. This enumeration agrees with the CE counterexample and the analytical results in Appendix K.

To verify the conditions of the conditional H-consistency theorem in Section 2.4 for this construction, set $\alpha = 2$, $\xi = 10^{-3}$, and $\tau = 10^{-3}$. Since $N = 5$ and $n = 20$, the empirical risk grid gives $g \geq 1/(Nn) = 0.01$, whereas

$$\beta_{\xi,\tau} = 2\xi + \frac{\tau \log K_{n,n}}{n} \approx 0.0036596896077 < 0.01. \tag{104}$$

The conditional H-consistency theorem in Section 2.4 therefore yields

$$\widehat{\mathcal{E}}_{\mathrm{E},H_\xi^{\mathrm{code}}}(h) \leq 1.57721\, \widehat{\mathcal{E}}_{\mathrm{S},\tau,H_\xi^{\mathrm{code}}}(h). \tag{105}$$

This inequality gives an explicit consistency mapping satisfying $f(0) = 0$, verifying that PEAL meets the conditions of the main theorem in Section 2.4 for this construction. In contrast, CE retains strictly positive edit-distance excess risk even when its own excess risk is zero.