EXPLANATION-GUIDED ACTIVE LEARNING BY EXPLORING DIVERSE DECISION RATIONALES

Anonymous authors
Paper under double-blind review

ABSTRACT

Skeleton-based action recognition relies on large-scale annotations, motivating active learning to identify informative samples under limited labeling budgets. Existing acquisition strategies typically quantify informativeness using predictive uncertainty, feature-space coverage, or gradient-based sensitivity, without considering the evidence underlying model decisions. Samples supported by similar spatio-temporal evidence may provide redundant supervision even when they are distant in prediction, feature, or gradient space. Conversely, samples with similar representations may exhibit distinct decision patterns and thus provide complementary supervision. We propose an explanation-guided active learning framework that selects samples based on their decision evidence. Using Layer-wise Relevance Propagation, we identify prediction-preserving spatio-temporal evidence and encode it into fixed-dimensional explanation representations. Batch acquisition then improves coverage in this explanation space, promoting diversity in decision evidence beyond predictions, features, or gradients. We further bound the generalization risk by the explanation-space covering radius, optimization error, and explanation reconstruction error, providing theoretical support for our criterion. Experiments on two skeleton-based action recognition datasets consistently outperform existing active learning baselines.

1 INTRODUCTION

Skeleton-based action recognition identifies human actions from temporal sequences of body joints (Presti & La Cascia, 2016; Chen et al., 2021; Yan et al., 2018). It has been widely applied to many applications, such as human robot interaction (Zeng et al., 2025), intelligent surveillance (Dai et al., 2024), and autonomous driving (Xin et al., 2023). However, current methods still rely on large-scale manual annotations, which are costly to obtain, especially for fine-grained actions in long motion sequences. To reduce this annotation burden, active learning (AL) (Li et al., 2023) reduces this annotation cost by selecting a small set of informative unlabeled samples for labeling, aiming to achieve competitive performance with a limited annotation budget.

A central problem in active learning is how to identify unlabeled samples that can provide complementary supervision after annotation. Existing acquisition strategies characterize sample informativeness from different perspectives. Uncertainty-based methods focus on prediction ambiguity through criteria such as confidence, margin, or entropy (Cho et al., 2024). Representation-based methods, such as CoreSet (Sener & Savarese, 2018), encourage diversity or coverage in the learned feature space. Gradient-based methods further incorporate model sensitivity: BADGE (Ash et al., 2019) constructs gradient embeddings to jointly capture uncertainty and diversity, while BAIT (Ash et al., 2021) selects samples according to expected gradients and Fisher-information geometry.

These criteria do not explicitly characterize the spatio temporal evidence that supports the current prediction. Output-level uncertainty describes how ambiguous a prediction is, feature space methods describe where a sample lies in the learned representation, and gradient- or Fisher-based criteria characterize how strongly a sample may affect the loss or model parameters. These signals therefore capture different aspects of informativeness, but they are not explicitly designed to distinguish samples according to which input evidence the model relies on when making its decision.

Fig. 1(a) illustrates this distinction using two skeleton sequences from the same throwing class. Both sequences are correctly classified as throwing and exhibit comparable predictive uncertainty.

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.

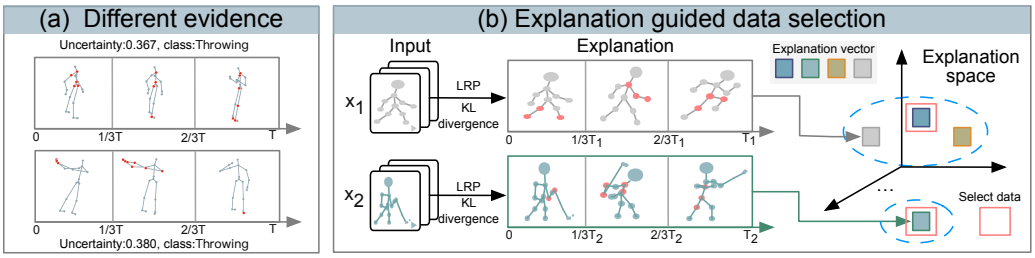

Figure 1: Overview of the motivation and our explanation-guided data selection framework. (**a**): samples with similar predictive uncertainty can rely on different spatio temporal evidence. Important joints across different temporal stages are highlighted in red. (**b**): our framework identifies important joints and frames using LRP and KL-divergence optimization and encodes the selected evidence into explanation vectors. We performs sample selection by improving coverage in the resulting explanation space.

We apply our explanation method to estimate the contribution of individual skeletal joints across frames (see Sec. 4.1 for details), and divide each sequence into three normalized temporal stages $0 - \frac{1}{3}T, \frac{1}{3}T - \frac{2}{3}T, \frac{2}{3}T - T$, where T denotes the number of frames in the sequence. The important joints identified within each stage are highlighted in red. Although the two sequences have similar predictions, the model relies on noticeably different joints at different stages of the throwing motion.

This observation motivates us to characterize unlabeled samples by their decision evidence. As illustrated in Fig. 1(b), for an unlabeled skeleton video x_i with T_i frames, we apply Layer-wise Relevance Propagation (LRP) (Bach et al., 2015) to quantify the contribution of each joint at each frame to the current prediction. By relevance conservation, these scores sum to the corresponding class logit. We then optimize a KL-divergence objective to select key joints and frames whose retained evidence preserves the original predictive distribution. To obtain a fixed-dimensional representation across videos of varying lengths, we partition the normalized temporal axis into M segments and summarize the selected joints within each segment. For example, Fig. 1(b) uses $M = 3$, corresponding to $[0, \frac{1}{3}T_i)$, $[\frac{1}{3}T_i, \frac{2}{3}T_i)$, and $[\frac{2}{3}T_i, T_i]$. Concatenating the segment-wise summaries yields an explanation vector that captures both the spatial locations and temporal evolution of prediction supporting evidence. Finally, we embed unlabeled videos into this explanation space and perform batch acquisition based on their explanation representations.

We further establish in Theorem 4.1 that generalization risk depends on the covering radius of the explanation space, together with standard optimization and explanation reconstruction terms. This provides theoretical support for our acquisition objective and motivates decision evidence coverage as a principled criterion for active learning over skeleton sequences. Extensive experiments on two datasets demonstrate that our method consistently outperforms existing active learning baselines.

2 Related Work

Active learning reduces annotation cost by selecting informative unlabeled samples for labeling. **Uncertainty-based methods** prioritize samples with uncertain predictions, using criteria such as entropy, least confidence, and margin (Shannon, 1948; Lewis, 1995; Roth & Small, 2006). A-SOiD similarly queries low confidence behavioral segments for iterative refinement (Tillmann et al., 2024). **Diversity-based methods** aim to select representative and non-redundant batches. CoreSet performs k-center selection in feature space (Sener & Savarese, 2018), while Deep Metric Learning with CoreSet applies the same principle in a task-adapted metric space (Vepa et al., 2024). **Gradient-based methods** characterize sample utility through model gradients. BADGE selects diverse samples in a pseudo-label gradient embedding space using k-means++ (Ash et al., 2019), whereas BAIT uses expected gradients and Fisher information for batch selection (Ash et al., 2021). **Perturbation-based methods** measure informativeness through model sensitivity to parameter perturbations. LDM-S prioritizes samples with unstable predictions under parameter perturbation (Cho et al., 2024), while Noise Stability constructs perturbation-response representations and performs diversity-aware selection in the resulting space (Li et al., 2024).

3 BACKGROUND

3.1 ACTIVE LEARNING

We consider a pool-based active learning setting for human pose videos, where samples are iteratively selected from an unlabeled pool for annotation. Let $\mathcal{X} = \{x_i\}_{i=1}^{N}$ denote the complete set of video samples and $\mathcal{Y}$ the label space. At acquisition round t, the labeled set $\mathcal{L}_t \subseteq \mathcal{X} \times \mathcal{Y}$ contains the currently annotated samples, while the unlabeled pool is defined as $\mathcal{U}_t = \mathcal{X} \setminus \{x_i \mid (x_i, y_i) \in \mathcal{L}_t\}$. Each sample is represented as $x_i \in \mathbb{R}^{T_i \times V \times d}$, where T_i, V, and d denote the numbers of frames, body nodes, and node feature dimensions, respectively. At round t, a model $f(\theta_t)$ is trained on $\mathcal{L}_t$. Given an acquisition budget b_t, an acquisition function $\mathcal{A}$ selects a subset of unlabeled samples,

$$\mathcal{B}_t = \mathcal{A}\left(\mathcal{U}_t, \mathcal{L}_t, f(\theta_t), b_t\right) \subseteq \mathcal{U}_t, \qquad |\mathcal{B}_t| = b_t, \qquad \sum_t b_t = b,$$

where b is the total budget. After obtaining labels for the selected samples, the pools are updated as

$$\mathcal{L}_{t+1} = \mathcal{L}_t \cup \{(x_i, y_i) \mid x_i \in \mathcal{B}_t\}, \qquad \mathcal{U}_{t+1} = \mathcal{U}_t \setminus \mathcal{B}_t,$$

The model is then retrained on the expanded labeled set, and the acquisition process is repeated until the annotation budget is exhausted.

3.2 CTR-GCN MODEL

We employ CTR-GCN Chen et al. (2021) as the backbone $f(\theta_t)$ for skeleton-based action recognition. Given a skeleton sequence $x_i \in \mathbb{R}^{T_i \times V \times d}$, we set $S_i^{(0)} = x_i$ and denote the representation at the l-th block by $S_i^{(l)} \in \mathbb{R}^{T_i^{(l)} \times V \times C_l}$, where $C_0 = d$, and $T_i^{(l)}$, V, and C_l denote the temporal length, number of joints, and feature dimension at l-th layer, respectively.

CTR-GC. We first describe the computation of a single CTR-GC unit. Given $S_i^{(l)}$, the input features are transformed as

$$\widetilde{S}_i^{(l)} = \mathcal{T}_l(S_i^{(l)}) = S_i^{(l)} W_l \in \mathbb{R}^{T_i^{(l)} \times V \times C_{l+1}}, \quad W_l \in \mathbb{R}^{C_l \times C_{l+1}},$$

Two projections $\psi_l, \phi_l : \mathbb{R}^{C_l} \to \mathbb{R}^{C_r}$ are used to infer sample-dependent joint correlations. After temporal average pooling, we obtain

$$\bar{S}_i^{\psi,(l)} = \frac{1}{T_i^{(l)}} \sum_{t=1}^{T_i^{(l)}} \psi_l(S_i^{(l)}[t]) \in \mathbb{R}^{V \times C_r}, \qquad \bar{S}_i^{\phi,(l)} = \frac{1}{T_i^{(l)}} \sum_{t=1}^{T_i^{(l)}} \phi_l(S_i^{(l)}[t]) \in \mathbb{R}^{V \times C_r},$$

For each joint pair (u, v), the channel-wise correlation is computed and aggregated as

$$q_i^{(l)}[u, v] = \xi_l \left[\tanh\left(\bar{S}_i^{\psi,(l)}[u] - \bar{S}_i^{\phi,(l)}[v]\right)\right] \in \mathbb{R}^{C_{l+1}}, Q_i^{(l)} = \left[q_i^{(l)}[u, v]\right]_{u,v}^{V} \in \mathbb{R}^{V \times V \times C_{l+1}}.$$

Here, $\xi_l : \mathbb{R}^{C_r} \to \mathbb{R}^{C_{l+1}}$ denotes a learnable channel transformation. The channel specific correlations are combined with the shared topology $A_l \in \mathbb{R}^{V \times V}$ to perform spatial feature aggregation:

$$S_{o,i}^{(l)}[\tau, u, c] = \sum_{v=1}^{V} \left(A_l[u, v] + \alpha_l Q_i^{(l)}[u, v, c]\right) \widetilde{S}_i^{(l)}[\tau, v, c], \quad S_{o,i}^{(l)} = \mathrm{CTRGC}\left(S_i^{(l)}; A_l\right). \quad (1)$$

where α_l is a learnable scalar and $S_{o,i}^{(l)} \in \mathbb{R}^{T_i^{(l)} \times V \times C_{l+1}}$.

Spatial modeling. In the CTR-GCN block, multiple CTR-GC units are applied in parallel to model spatial relations. Specifically, the outputs of K CTR-GC branches are aggregated as

$$H_i^{(l)} = \mathrm{ReLU}\left(\mathrm{BN}_l\left[\sum_{k=1}^{K} \mathrm{CTRGC}_k\left(S_i^{(l)}; A_{l,k}\right)\right] + \psi_l^{\mathrm{res}}\left(S_i^{(l)}\right)\right),$$

where $A_{l,k}$ denotes the topology associated with the k-th spatial branch, $\mathrm{BN}_l(\cdot)$ denotes batch normalization in the l-th CTR-GCN block, and $\psi_l^{\mathrm{res}}(\cdot)$ denotes the corresponding residual mapping.

Temporal modeling. The spatial representation $H_i^{(l)}$ is subsequently processed by a multi-scale temporal convolution module consisting of parallel temporal branches with different receptive fields:

$$U_i^{(l)} = \mathrm{Concat}\left(\left\{\mathcal{T}_{l,d}\left(H_i^{(l)}\right)\right\}_{d\in\{1,2\}}, \mathcal{M}_l\left(H_i^{(l)}\right), \mathcal{P}_l\left(H_i^{(l)}\right)\right) \in \mathbb{R}^{T_i^{(l+1)}\times V\times C_{l+1}},$$

where $\mathcal{T}_{l,d}(\cdot)$ represents a 5×1 temporal convolution branch with dilation $d\in\{1,2\}$, preceded by a 1×1 channel projection; $\mathcal{M}_l(\cdot)$ denotes a 3×1 temporal max-pooling branch; and $\mathcal{P}_l(\cdot)$ denotes a 1×1 projection branch. Finally, the block-level residual connection is incorporated to produce the output of the l-th CTR-GCN block:

$$S_i^{(l+1)} = \mathrm{ReLU}\left(U_i^{(l)} + \psi_l^{\mathrm{res}}\left(S_i^{(l)}\right)\right) \in \mathbb{R}^{T_i^{(l+1)}\times V\times C_{l+1}},$$

After the final CTR-GCN block, we obtain a sequence-level representation g_i by averaging over the temporal and joint dimensions, and predict the class probabilities with a linear classifier:

$$g_i = \frac{1}{T_i^{(L)}V}\sum_{\tau=1}^{T_i^{(L)}}\sum_{v=1}^{V} S_i^{(L)}[\tau, v, :], \quad z_i = W_{\mathrm{cls}}g_i + b_{\mathrm{cls}}, \quad \hat{p}_i = \mathrm{softmax}(z_i).$$

4 METHOD

Our method selects batches by covering the locations of model-attributed decision evidence. Given a model parameter θ, we first apply Layer-wise Relevance Propagation (LRP) to quantify the contributions of individual frames and their corresponding nodes in each input video, as detailed in Section 4.1. Based on the resulting contribution matrix, we then use an approximate KL divergence criterion to identify the key frames and joints that capture the most informative decision evidence in Section 4.2. We then compute distances between these explanation vectors and select batches that improve the coverage of this space, as described in Section 4.3. This coverage based acquisition strategy is further supported by our theoretical analysis in Theorem 4.1. It establishes a connection between the explanation-space covering radius and the model's generalization risk. Queried samples are subsequently added to the labeled set and used for model training.

4.1 LRP ATTRIBUTIONS IN CTR-GCN

Given an input video, we use Layer-wise Relevance Propagation (LRP) (Bach et al., 2015) to decompose the model's output logit into relevance scores for individual nodes across frames, quantifying their contributions to the model prediction. The decomposition follows the relevance conservation principle, such that the total relevance accumulated over all nodes and all frames is equal to the output logit of the video. we first introduce the original LRP methods.

Original LRP Decomposition method. LRP explains a model prediction by redistributing the output relevance from the prediction layer toward the input through successive layers of the network. Consider a neuron in layer $l+1$ expressed as $h^{(l+1)} = \rho\left([h_1^{(l)}, \ldots, h_n^{(l)}]\right)$, where ρ denotes either a linear transformation or a linear transformation followed by a nonlinear activation. Given the relevance $R^{(l+1)}$, it is attributed to the neurons in the preceding layer by:

$$R_i^{(l)} = \frac{h_i^{(l)}w_i^l}{\sum_{i'} h_{i'}^{(l)}w_{i'}^l}R^{(l+1)}, \quad \sum_i R_i^{(l)} = R^{(l+1)}. \tag{2}$$

where w_i^l represents the connection weight from neuron $h_i^{(l)}$ in layer l to the neuron in layer $l+1$.

Starting from the output logit, denoted by $R^{(L)}$, the same redistribution rule is applied recursively through all network layers until the input representation is reached. Accordingly, the relevance associated with an input neuron $h_i^{(0)}$ can be written as:

$$R_i^{(0)} = \sum j, \ldots, k \frac{h_i^{(0)}w_i^0}{\sum_{i'} h_{i'}^{(0)}w_{i'}^0}\cdots\frac{h_k^{(L-1)}w_k^{L-1}}{\sum_{k'} h_{k'}^{(L-1)}w_{k'}^{L-1}}R^{(L)}, \quad \sum_i R_i^{(0)} = R^{(L)}. \tag{3}$$

LRP Decomposition for CTR-GCN models. The original LRP rule in Eq. (2) applies directly to standard linear transformations, optionally followed by nonlinear activations. CTR-GCN, however, also involves normalization, multiplicative interactions, and graph aggregation, which require operation-specific relevance redistribution rules. We consider a general operator $Y = \mathcal{F}(X_1, \ldots, X_m)$, where $X_1, \ldots, X_m$ denotes its activation inputs and m is the number of inputs. For a single input operator, such as BatchNorm, we set $m = 1$ and simply write the input as X. For multi-input operators, such as element-wise multiplication, or concatenation, we retain $X_1, \ldots, X_m$. Given the output relevance $R(Y)$, we denote the relevance assigned to the i-th activation input X_i by $R(X_i)$, or simply $R(X)$ for single input operators.

Batch normalization. When the model parameters θ is fixed, at inference time, BatchNorm is an affine mapping. It can be written as:

$$Y = \mathrm{BN}(X) = \gamma \odot \frac{X - \mu}{\sqrt{\sigma^2 + \epsilon_{\mathrm{BN}}}} + \beta = s \odot X + b_{\mathrm{BN}}, \quad s = \frac{\gamma}{\sqrt{\sigma^2 + \epsilon_{\mathrm{BN}}}}, \quad b_{\mathrm{BN}} = \beta - s \odot \mu$$

The the relevance or input is given by:

$$R(X) = (s \odot X) \odot \frac{R(Y)}{Y - b_{\mathrm{BN}}} = R(Y). \tag{4}$$

Grahp aggregation. The distinctive operation in CTR-GCN is the channel-wise graph aggregation in Eq. (1). To make the two input dependent factors explicit, we define

$$G_i^{(l)}[u, v, c] = A_l[u, v] + \alpha_l Q_i^{(l)}[u, v, c], \quad S_{o,i}^{(l)}[\tau, u, c] = \sum_{v=1}^{V} G_i^{(l)}[u, v, c]\, \widetilde{S}_i^{(l)}[\tau, v, c] \tag{5}$$

Each summand represents a multiplicative interaction between a relation coefficient and a node feature. Since standard LRP does not directly specify relevance redistribution for such interactions, we follow AttnLRP (Achtibat et al., 2024) and treat matrix multiplication as a bilinear operation. When attributing relevance to one input, the other input is held fixed and treated as a constant, and vice versa. The resulting propagation rules are given by

$$\begin{aligned} R(G_i^{(l)})[u, v, c] &= \frac{1}{2} \sum_{\tau} \frac{G_i^{(l)}[u, v, c]\, \widetilde{S}_i^{(l)}[\tau, v, c]}{S_{o,i}^{(l)}[\tau, u, c]} R\Big(S_{o,i}^{(l)}[\tau, u, c]\Big), \\ R(\widetilde{S}_{o,i}^{(l)})[\tau, v, c] &= \frac{1}{2} \sum_{u} \frac{G_i^{(l)}[u, v, c]\, \widetilde{S}_i^{(l)}[\tau, v, c]}{S_{o,i}^{(l)}[\tau, u, c]} R\Big(S_{o,i}^{(l)}[\tau, u, c]\Big). \end{aligned} \tag{6}$$

Hence, each graph-message relevance $R(S_{o,i}^{(l)}[\tau, u, c])$ is split equally between the node-feature path $R(\widetilde{S}_{o,i}^{(l)})$ and the relation-learning path $R(G_i^{(l)})$.

For Eq.(5), If A_l is a fixed model parameter, the entire relevance is propagated through the input:

$$R(A_l[u, v]) = 0, \qquad R(Q_i^{(l)}[u, v, c]) = R(G_i^{(l)}[u, v, c]). \tag{7}$$

If A_l is activation dependent and therefore needs to be further propagated, the relevance is redistributed according to the two linear contributions:

$$\begin{aligned} R_c(A_l[u, v]) &= \frac{A_l[u, v]}{G_i^{(l)}[u, v, c]} R(G_i^{(l)}[u, v, c]), \quad R(A_l[u, v]) = \sum_c R_c(A_l[u, v]) \\ R(Q_i^{(l)}[u, v, c]) &= \frac{\alpha_l Q_i^{(l)}[u, v, c]}{G_i^{(l)}[u, v, c]} R(G_i^{(l)}[u, v, c]). \end{aligned} \tag{8}$$

Since $A_l[u, v]$ is shared across channels, its relevance sums over channels.

Relevance propagation. Given a target class k, we initialize the relevance at the output logit as $R(z_i[k]) = z_i[k]$ and propagate it backward through the network. We first propagate relevance through the linear classifier and global average pooling layer to obtain $R(S_i^{(L)})$, and then recursively through the CTR-GCN blocks from $l = L - 1$ to $l = 0$. Standard linear transformations,

including fully connected layers, channel projections, and temporal convolutions, follow Eq. (2), while BatchNorm follows Eq. (4). Within each CTR-GC unit, relevance is decomposed into the node-feature and relation-learning paths according to Eq. (6). The relation relevance is further propagated through $Q_i^{(l)}$ using Eq. (7) or Eq. (8), depending on whether A_l is a fixed parameter or activation-dependent. Recursively applying these rules yields the input-level skeleton relevance,

$$R_k(x_i) = R(S_i^{(0)}) \in \mathbb{R}^{T_i \times V \times d}, \qquad \sum_{t=1}^{T_i} \sum_{v=1}^{V} \sum_{c=1}^{d} R_k(x_i)[t, v, c] = z_i[k]. \tag{9}$$

$R_k(x_i)[\tau, v, c]$ denotes the contribution of the c-th feature of node v in the τ-th frame to $z_i[k]$.

4.2 Selection of Important Frames and Joints

While $R_k(x_i)$ provides estimate of the contribution of each joint at each frame, directly selecting the top-k entries may not faithfully preserve the model prediction. Since the mapping from logits to predictive probabilities is nonlinear, relevance scores are not linearly related to their effects on the output distribution. We therefore formulate the selection of informative frames and joints as an optimization problem. Specifically, we minimize the Kullback–Leibler (KL) divergence between the predictive distributions of the selected spatio-temporal components and the original input. The resulting key frames and joints are then encoded into an explanation vector, which serves as the basis for subsequent data selection.

For each sample x_i, we aggregate the $R_k(x_i)$ over the input feature dimension and obtain the class conditioned spatio-temporal contribution scores as

$$R_k^{\text{node}}(x_i)[\tau, v] = \sum_{c=1}^{d} R_k(x_i)[\tau, v, c], \quad R_k^{\text{node}}(x_i) \in \mathbb{R}^{T_i \times V}, \quad z_i[k] = \sum_{\tau=1}^{T_i} \sum_{v=1}^{V} R_k^{\text{node}}(x_i)[\tau, v].$$

Here, $R_k^{\text{node}}(x_i)[\tau, v]$ measures the contribution of joint v at frame τ to the k-th class logit.

For each class $k \in 1, \ldots, K$, we first obtain its node-level contribution map $R_k^{\text{node}}(x_i)$. We then concatenate the contribution maps of all K classes along the class dimension to obtain the overall node-level contribution representation:

$$R^{\text{node}}(x_i) = \text{stack}\left(R_1^{\text{node}}(x_i), R_2^{\text{node}}(x_i), \ldots, R_K^{\text{node}}(x_i)\right) \in \mathbb{R}^{T_i \times V \times K}, \tag{10}$$

where $R^{\text{node}}(x_i)[\tau, v, k]$ measures the contribution of joint v at frame τ to the k-th class logit. Summing the node-level contributions over all frames and nodes recovers the corresponding class logit $\sum_{\tau=1}^{T_i} \sum_{v=1}^{V} R^{\text{node}}(x_i)[\tau, v, k] = z_i[k]$.

To identify which nodes at which frames are most important for preserving the original prediction, we introduce a continuous selection matrix $D_i \in [0, 1]^{T_i \times V}$, where $D_i[\tau, v]$ represents the selection weight assigned to joint v at frame τ. The logits and predictive distribution reconstructed from the selected nodes are then given by

$$\tilde{z}_i(D_i)[k] = \sum_{\tau=1}^{T_i} \sum_{v=1}^{V} D_i[\tau, v]\, R^{\text{node}}(x_i)[\tau, v, k], \quad \tilde{p}_i(D_i) = \text{softmax}\left(\tilde{z}_i(D_i)\right)$$

The KL divergence between $\hat{p}_i$ and $\tilde{p}_i(D_i)$ is

$$\begin{aligned} \text{KL}\left(\hat{p}_i \parallel \tilde{p}_i(D_i)\right) &= \sum_{k=1}^{K} \hat{p}_i[k] \log \hat{p}_i[k] - \sum_{k=1}^{K} \hat{p}_i[k] \log \tilde{p}_i(D_i)[k] \\ &= \sum_{k=1}^{K} \hat{p}_i[k] \log \hat{p}_i[k] - \sum_{k=1}^{K} \hat{p}_i[k] \tilde{z}_i(D_i)[k] + \sum_{k=1}^{K} \hat{p}_i[k] \log\Big(\sum_{j=1}^{K} \exp\left(\tilde{z}_i(D_i)[j]\right)\Big). \end{aligned}$$

Since $\sum_{k=1}^{K} \hat{p}_i[k] = 1$, we obtain

$$\text{KL}\left(\hat{p}_i \parallel \tilde{p}_i(D_i)\right) = \underbrace{\sum_{k=1}^{K} \hat{p}_i[k] \log \hat{p}_i[k]}_{\text{constant w.r.t. } D_i} - \sum_{k=1}^{K} \hat{p}_i[k] \tilde{z}_i(D_i)[k] + \log\left(\sum_{j=1}^{K} \exp\left(\tilde{z}_i(D_i)[j]\right)\right).$$

The first term is independent of the selection matrix D_i and can therefore be omitted during optimization. The resulting objective is given by:

$$D_i^* = \underset{\substack{D_i \in [0,1]^{T_i \times V} \\ \|D_i\|_1 = n}}{\arg\min} \quad - \sum_{k=1}^{K} \hat{p}_i[k] \sum_{\tau=1}^{T_i} \sum_{v=1}^{V} D_i[\tau, v] R^{\text{node}}(x_i)[\tau, v, k] + \log \left[\sum_{k=1}^{K} \exp \left(\sum_{\tau=1}^{T_i} \sum_{v=1}^{V} D_i[\tau, v] R^{\text{node}}(x_i)[\tau, v, k] \right) \right]. \tag{11}$$

Where n denotes a selection budget. The optimized selection matrix D_i^* is subsequently used to derive the explanation vector, which further serves as the basis for selecting the informative data.

4.3 Explanation-Guided Sample Selection

Since videos vary in length, the optimized selection matrices $D_i^* \in \mathbb{R}^{T_i \times V}$ are not directly comparable across samples. We therefore aggregate node selections over normalized temporal stages to obtain a fixed-dimensional explanation vector, which captures both *which* nodes are important and *when* they become important. This representation embeds videos of different lengths into a common explanation space. At each sample-selection round, we favor samples that improve the coverage of this space. Our theoretical analysis further links explanation-space coverage to model generalization risk, supporting explanation diversity as a criterion for sample selection.

First, we divide the normalized temporal axis of each video into M relative temporal stages. Specifically, for sample x_i with T_i frames, the m-th temporal stage is defined as:

$$\mathcal{T}_m(T_i) = \left\{ \tau \;\middle|\; \left\lfloor \frac{(m-1)T_i}{M} \right\rfloor + 1 \le \tau \le \left\lfloor \frac{mT_i}{M} \right\rfloor \right\}, \qquad m = 1, \ldots, M.$$

Here, $\mathcal{T}_m(T_i)$ contains the frames located within the m-th relative stage of the sequence. Thus, instead of aligning videos according to their absolute frame indices, we align them according to their relative temporal positions. The hyperparameter M controls the temporal granularity of the resulting explanation: a larger M preserves finer temporal variations, whereas a smaller M provides a more coarse-grained summary. Given these temporal stages, we aggregate and normalize the selected nodes within each stage as

$$E_i[m, v] = \sum_{\tau \in \mathcal{T}_m(T_i)} D_i^*[\tau, v], \quad \widetilde{E}_i[m, v] = \frac{E_i[m, v]}{\|E_i\|_1}, \quad E_i, \widetilde{E}_i \in \mathbb{R}^{M \times V}. \tag{12}$$

The resulting matrix captures how frequently each node is selected at different relative stages of the sequence. It provides a unified explanation-space representation for subsequent sample selection.

At each sample-selection round, we perform selection in the explanation space following the k-Center formulation (Sener & Savarese, 2018). We measure the discrepancy between two samples using the Jensen–Shannon divergence (JSD). Specifically, for samples x_i and x_j,

$$d_{\text{JSD}}(x_i, x_j) = \text{JSD}\left(\tilde{E}_i, \tilde{E}_j\right) = \frac{1}{2} D_{\text{KL}}\left(\tilde{E}_i \,\middle\|\, \frac{\tilde{E}_i + \tilde{E}_j}{2}\right) + \frac{1}{2} D_{\text{KL}}\left(\tilde{E}_j \,\middle\|\, \frac{\tilde{E}_i + \tilde{E}_j}{2}\right).$$

A larger $d_{\text{JSD}}(x_i, x_j)$ indicates that two samples induce distinct spatiotemporal explanation patterns.

Let $\mathcal{L}_t$ denote the set of samples that have already been selected after round t, and let $\mathcal{U}_t$ denote the remaining candidate pool. Given a selection budget b, we select a new batch $\mathcal{B}_t \subseteq \mathcal{U}_t$ by solving the following k-Center objective:

$$\mathcal{B}_t^* = \arg \min_{\substack{\mathcal{B}_t \subseteq \mathcal{U}_t \\ |\mathcal{B}_t| \le b}} \max_{x_j \in \mathcal{U}_t} \min_{x_i \in \mathcal{L}_t \cup \mathcal{B}_t} d_{\text{JSD}}(x_i, x_j). \tag{13}$$

The k-Center criterion therefore favors samples with distinct explanation patterns. Similar explanation vectors indicate that the model relies on similar spatiotemporal evidence across samples, making their explanatory information largely redundant. By selecting samples with diverse explanation patterns, we capture model behaviors and cover a broader region of the explanation space under a limited selection budget. The overall framework is shown in Algorithm 1 in Appendix.

We therefore provide a theoretical analysis that connects our explanation-aware acquisition strategy to the model's post query generalization performance through this covering radius. For the theoretical analysis, let

$$\mathcal{J}_{t+1}(\mathcal{B}_t) = \mathbb{E}_{(X,Y)\sim p_{\mathcal{Z}}}\left[-\log \operatorname{softmax}\left(f_{\theta_{t+1}(\mathcal{B}_t)}(X)\right)[Y]\right], \tag{14}$$

where (X, Y) denotes a random pair drawn from $p_{\mathcal{Z}}$, while (x_i, y_i) denotes a specific sample. Then we can obtain the following theorem:

Theorem 4.1 (Explanation Aware Post Query Generalization Bound)**.** *Consider active-learning round t and an admissible candidate batch $\mathcal{B}_t \subseteq \mathcal{U}_t$ with $|\mathcal{B}_t| \leq b$. Let $r_t(\mathcal{B}_t)$ denote the covering radius of the post-query labeled set in the explanation space. There exists the constant α_t, $\bar{\kappa}_t$ and $C_{\mathrm{geo},t} = 4nG_t + \sqrt{2}\,nL_{R,t} + L_{Y,t}$ such that*

$$\mathcal{J}_{t+1}(\mathcal{B}_t) \leq \underbrace{\mathcal{E}_t^{\mathrm{base}}(\mathcal{B}_t)}_{\text{population and optimization terms}} + \underbrace{\frac{4}{\alpha_t\sqrt{K}}\sqrt{\bar{\kappa}_t}}_{\text{LRP reconstruction distortion}} + \underbrace{C_{\mathrm{geo},t}\, r_t(\mathcal{B}_t)}_{\text{explanation-space coverage error}}. \tag{15}$$

Theorm 4.1 separates the post query population risk into three components. The first collects population approximation, post-query training, model-update, and labeled-set variation effects. The second is controlled by the KL reconstruction objective used to construct the LRP explanations. The last term scales linearly with the covering radius in the explanation space, directly connecting the explanation k-Center acquisition objective to the population risk bound. Consequently, an acquisition strategy that improves explanation-space coverage and reduces $r_t(\mathcal{B}_t)$ yields a tighter population-risk bound. The proof is shown in Appendix C.

5 Experiments

Datasets. We evaluate our method on two skeleton-based action recognition datasets: NW-UCLA (Wang et al., 2014) and NTU RGB+D 60 (Shahroudy et al., 2016). NW-UCLA contains 1,494 sequences from 10 action classes performed by 10 subjects and captured from three Kinect views, with 20 joints per skeleton. Following the standard cross-view protocol, we train on views 1 and 2 and test on view 3, resulting in 1,018 training and 476 test sequences. NTU RGB+D 60 contains 56,880 sequences from 60 action classes performed by 40 subjects and captured with three Kinect cameras, with 25 joints per skeleton. We adopt the cross-subject (X-Sub) protocol, using 20 subjects for training and the remaining 20 for testing, resulting in 40,320 training and 16,560 test sequences. We use only the skeleton modality for both datasets. All methods use CTR-GCN (Chen et al., 2021) as the backbone. Detailed descriptions of the training protocols, backbone adaptations, and implementation settings are provided in Appendix B.

Table 1: Accuracy (%) at different labeling budgets on NW-UCLA. Best results are shown in bold, and second-best results are underlined.

Method	5%	10%	15%	20%	25%	30%	35%	40%
Rand	45.91	42.46	55.17	48.06	51.08	57.76	57.11	55.17
Entropy	35.78	51.51	35.13	46.77	56.90	61.21	56.47	57.97
Margin	43.10	51.29	56.90	54.96	61.42	54.74	57.11	55.60
CoreSet	34.70	46.12	44.40	48.71	57.76	50.86	48.06	53.45
Metric-CoreSet	35.99	46.77	50.22	56.90	58.41	63.15	50.22	61.42
BADGE	36.21	51.09	52.37	49.14	48.28	61.64	58.62	58.62
BAIT	38.79	45.04	57.33	53.45	56.68	60.99	59.48	53.66
LDM-S	37.72	50.00	51.08	52.16	51.72	57.97	56.68	57.76
NoiseStab	45.26	46.12	50.00	47.20	55.82	64.66	60.13	**66.59**
Ours	**50.65**	**55.17**	**62.28**	**64.22**	**62.93**	**67.46**	**63.15**	<u>62.07</u>

Baselines. We compare our method against a diverse set of active learning baselines, including random sampling (Rand); uncertainty-based methods, i.e., entropy sampling (Entropy) (Shannon,

1948) and margin sampling (Margin) (Roth & Small, 2006); feature-space coverage methods, including CoreSet (Sener & Savarese, 2018) and its task-adapted metric variant, Metric-CoreSet (Vepa et al., 2024); gradient-based batch acquisition methods, including BADGE (Ash et al., 2019) and BAIT (Ash et al., 2021); prediction-sensitivity-based sampling (LDM-S) (Cho et al., 2024); and representation-stability-based sampling (NoiseStab) (Li et al., 2024).

Table 2: Accuracy (%) at different labeling budgets on NTU RGB+D 60. Best results are shown in bold, and second-best results are underlined.

Method	1%	2%	5%	10%
Rand	20.19	27.13	53.00	65.42
Entropy	19.12	27.71	48.29	59.29
Margin	19.82	27.56	53.51	64.20
CoreSet	11.72	18.86	42.51	57.40
Metric-CoreSet	18.91	26.93	50.30	63.67
BADGE	19.72	27.03	51.31	64.19
BAIT	18.60	28.32	51.70	65.21
LDM-S	<u>20.03</u>	<u>29.75</u>	52.96	<u>65.56</u>
NoiseStab	11.96	18.53	41.44	56.83
Ours	**21.11**	**30.47**	**54.36**	**66.44**

Performance Table 1 and Table 2 demonstrate a clear advantage of our method under limited annotation budgets. On UCLA, our method reaches 62.28% accuracy with only 15% labeled data, outperforming the strongest baseline at the same budget, BAIT (57.33%), by 4.95 percentage points. This result highlights its improved annotation efficiency. In comparison, random and uncertainty-based strategies may select redundant or weakly informative samples, while diversity-based methods such as CoreSet and Metric-CoreSet primarily emphasize feature-space coverage and may overlook task-relevant informativeness. Although NoiseStab becomes competitive at larger budgets, it remains less effective in the low-budget regime. On NTU RGB+D 60, our method achieves the best performance across all evaluated budgets, consistently outperforming competing strategies from 1% to 10% labeled data. The consistent gains across both datasets suggest that the proposed selection strategy generalizes across different data scales and annotation regimes. Notably, the improvements are already substantial at the early stages of active learning, indicating that our method can identify informative samples when labeled supervision is scarce. Overall, these results validate the effectiveness of our method for annotation-efficient skeleton-based action recognition, particularly in the practically relevant low-budget setting.

6 Conclusions and Limitations

We introduce an explanation-guided active learning framework for skeleton-based action recognition. Unlike existing acquisition strategies that measure informativeness through predictive uncertainty, feature-space diversity, or gradient-based sensitivity, our method captures the spatiotemporal evidence underlying model predictions. Specifically, we combine Layer-wise Relevance Propagation (LRP) with a KL-divergence objective to identify prediction-preserving joints and frames, and encode their temporal evolution into a fixed-dimensional explanation representation. Batch acquisition then improves coverage in this explanation space, encouraging selected samples to provide complementary supervision with respect to the evidence used by the model. We further establish a theoretical connection between generalization risk and the covering radius of the explanation space, providing a principled justification for our acquisition criterion. Experiments on UCLA and NTU RGB+D 60 consistently demonstrate the effectiveness of our approach across annotation budgets and acquisition baselines. A limitation is that constructing explanation representations requires LRP and selection optimization over unlabeled samples, resulting in higher computational cost.

AI USE STATEMENT

We used generative AI tools solely to assist with language editing and improving the clarity and readability of the manuscript.

REFERENCES

Reduan Achtibat, Sayed Mohammad Vakilzadeh Hatefi, Maximilian Dreyer, Aakriti Jain, Thomas Wiegand, Sebastian Lapuschkin, and Wojciech Samek. Attnlrp: attention-aware layer-wise relevance propagation for transformers. *arXiv preprint arXiv:2402.05602*, 2024.

Jordan Ash, Surbhi Goel, Akshay Krishnamurthy, and Sham Kakade. Gone fishing: Neural active learning with fisher embeddings. *Advances in Neural Information Processing Systems*, 34:8927–8939, 2021.

Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*, 2019.

Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.

Yuxin Chen, Ziqi Zhang, Chunfeng Yuan, Bing Li, Ying Deng, and Weiming Hu. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13339–13348. IEEE, 2021.

Seong Jin Cho, Gwangsu Kim, Junghyun Lee, Jinwoo Shin, and Chang Yoo. Querying easily flip-flopped samples for deep active learning. In *International Conference on Learning Representations*, volume 2024, pp. 33212–33245, 2024.

Cheng Dai, Shoupeng Lu, Chuanjie Liu, and Bing Guo. A light-weight skeleton human action recognition model with knowledge distillation for edge intelligent surveillance applications. *Applied Soft Computing*, 151:111166, 2024.

David D Lewis. A sequential algorithm for training text classifiers: Corrigendum and additional data. In *Acm sigir forum*, volume 29, pp. 13–19. ACM New York, NY, USA, 1995.

Jingyuan Li, Trung Le, and Eli Shlizerman. Al-sar: Active learning for skeleton-based action recognition. *IEEE Transactions on Neural Networks and Learning Systems*, 35(11):16966–16974, 2023.

Xingjian Li, Pengkun Yang, Yangcheng Gu, Xueying Zhan, Tianyang Wang, Min Xu, and Chengzhong Xu. Deep active learning with noise stability. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 13655–13663, 2024.

Liliana Lo Presti and Marco La Cascia. 3d skeleton-based human action classification: A survey. *Pattern Recognition*, 53:130–147, 2016.

Dan Roth and Kevin Small. Margin-based active learning for structured output spaces. In *European conference on machine learning*, pp. 413–424. Springer, 2006.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=H1aIuk-RW`.

Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1010–1019, 2016.

Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

Jens F Tillmann, Alexander I Hsu, Martin K Schwarz, and Eric A Yttri. A-soid, an active-learning platform for expert-guided, data-efficient discovery of behavior. *Nature Methods*, 21(4):703–711, 2024.

Arvind M Vepa, Zukang Yang, Andrew Choi, Jungseock Joo, Fabien Scalzo, and Yizhou Sun. Integrating deep metric learning with coreset for active learning in 3d segmentation. *Advances in Neural Information Processing Systems*, 37:71643–71671, 2024.

Jiang Wang, Xiaohan Nie, Yin Xia, Ying Wu, and Song-Chun Zhu. Cross-view action modeling, learning and recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2649–2656, 2014.

Wentian Xin, Ruyi Liu, Yi Liu, Yu Chen, Wenxin Yu, and Qiguang Miao. Transformer for skeleton-based action recognition: A review of recent advances. *Neurocomputing*, 537:164–186, 2023.

Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Fan Zeng, Yusheng Wang, Masahiro Nishio, and Jun Ota. Human-robot interaction with skeleton-based action recognition and motion prediction. *Journal of Intelligent & Robotic Systems*, 111 (4):129, 2025.

A ALGORITHM

Algorithm 1 Explanation-guided active learning

Input: initial labeled set $\mathcal{L}_0$, initial unlabeled set $\mathcal{U}_0$, model f_θ, total number of epochs T, query interval q, candidate size c, query size b, selection ration n
Initialize query round $t \leftarrow 0$
for $t = 1, 2, \ldots, T$ **do**
  Train f_θ for one epoch using the current labeled set $\mathcal{L}_t$
  if $t \bmod q = 0$ **and** $|\mathcal{U}_t| > 0$ **then**
    Randomly sample a candidate set $\mathcal{C}_t \subseteq \mathcal{U}_t$, where $|\mathcal{C}_t| = \min(c, |\mathcal{U}_t|)$
    for $x_i \in \mathcal{C}_t \cup \mathcal{L}_t$ **do**
      Compute $R_k(x_i)$ according to Eq.(9), for $k = 1, \cdots, K$
      Given n and obtain D^* according to Eq.(11)
      Obtain $\widetilde{E}_i$ according to Eq.(12)
    end for
    Obtain $\mathcal{B}_t^*$ according to Eq.(13)
    Update the labeled set: $\mathcal{L}_{t+1} \leftarrow \mathcal{L}_t \cup \mathcal{B}_t^*$
    Update the unlabeled set: $\mathcal{U}_{t+1} \leftarrow \mathcal{U}_t \setminus \mathcal{B}_t^*$
    $t \leftarrow t + 1$
  end if
end for
Output: trained model f_θ and final labeled set $\mathcal{L}_t$

B EXPERIMENTAL DETAILS

B.1 DATASETS AND EVALUATION PROTOCOLS

NW-UCLA. NW-UCLA (Wang et al., 2014) contains 1,494 action sequences from 10 action classes performed by 10 subjects and recorded from three Kinect camera views. Each skeleton contains 20 body joints. We use the skeleton modality only. Camera views 1 and 2 are used for training, and camera view 3 is used for testing, giving 1,018 training sequences and 476 test sequences.

NTU RGB+D 60. NTU RGB+D 60 (Shahroudy et al., 2016) contains 56,880 action sequences from 60 action classes performed by 40 subjects and recorded by three Kinect cameras. Each skeleton contains 25 body joints. We use the skeleton modality and the cross-subject (X-Sub) protocol. Twenty subjects are used for training and the remaining 20 subjects are used for testing, giving 40,320 training sequences and 16,560 test sequences.

B.2 BACKBONE AND VARIABLE-LENGTH ADAPTATION

CTR-GCN backbone. We use CTR-GCN (Chen et al., 2021) as the backbone network for all experiments. CTR-GCN performs skeleton-based action recognition using channel-wise topology refinement graph convolutions together with temporal convolutional modules. We use the joint coordinates as the input modality.

B.3 TRAINING AND ACTIVE LEARNING PROTOCOL

We use the same backbone training protocol for all acquisition methods. Stochastic training augmentation is disabled, and the training repeat factor is set to one. This keeps the effective sample exposure determined by the current labeled pool rather than repeated or randomly transformed copies. Training sequences shorter than 11 frames are excluded. The complete training and active learning settings are summarized in Table 3.

The labeled pool is initialized by random sampling at the first acquisition round. When the per-round query size is not specified, the remaining annotation budget is distributed across the remaining acquisition rounds. Newly queried samples are added to the labeled pool, and training continues

Table 3: Training and active learning settings on NW-UCLA and NTU RGB+D 60.

Setting	NW-UCLA	NTU RGB+D 60
Input modality	Joint	Joint
Dropout	0.5	0.5
Training repeat	1	1
Training augmentation	None	None
Minimum training length	11 frames	11 frames
Batch size	16	16
Evaluation batch size	256	128
Optimizer	SGD with momentum	SGD with momentum
Base learning rate	0.01	0.01
Momentum	0.9	0.9
Nesterov	Yes	Yes
Weight decay	10^{-4}	10^{-4}
Learning-rate schedule	Constant	Constant
Training epochs	100	300
Acquisition interval	10 epochs	30 epochs
Annotation budgets	5–45% (step 5%)	1%, 2%, 5%, 10%

Table 4: Explanation-guided acquisition settings on NW-UCLA and NTU RGB+D 60.

Setting	NW-UCLA	NTU RGB+D 60
Candidate screening	Random	Random
Candidate pool ratio	30%	3%
LRP classes	All classes	Top-10 classes
Retained frame-joint ratio	10%	10%
Temporal bins	5	5
KL solver	MOSEK	MOSEK
KL formulation	Continuous	Continuous

from the current model parameters. The acquisition schedules and annotation budgets for both datasets are also listed in Table 3.

B.4 Explanation-Guided Acquisition Settings

Our method first random the unlabeled pool and then computes LRP explanations for the resulting candidate set. For each candidate, 30% of the frame-joint units are retained by the prediction-preserving KL objective. The selected units use 5 temporal bins,i.e, $M = 5$ and sample distances are computed using the square root of the Jensen–Shannon divergence (JSD). Table 4 summarizes the dataset-specific explanation and acquisition settings.

NW-UCLA is sufficiently small to use all-class LRP and the discrete KL selection formulation. For NTU RGB+D 60, we restrict LRP computation to the Top-10 predicted classes and use a continuous relaxation of the KL selection problem to reduce the computational cost. The relaxed solution is discretized by retaining the largest variables under the same frame-joint budget. We set a 300-second time limit for the KL optimization of each sample. The resulting discrete mask is used for all subsequent representation and acquisition steps.

C Theoretical Analysis

We use t to denote the beginning of a selection round. Let f_{θ_t} be the current model, $\mathcal{L}_t$ the currently labeled set, and $\mathcal{U}_t$ the remaining unlabeled pool. Define $\mathcal{P}_t := \mathcal{L}_t \cup \mathcal{U}_t$. For a candidate batch $\mathcal{B}_t \subseteq \mathcal{U}_t$, $|\mathcal{B}_t| \leq b$, let $\mathcal{L}_t^+ = \mathcal{L}_t \cup \mathcal{B}_t$ be the labeled set immediately after querying $\mathcal{B}_t$. Throughout the proof, its dependence on $\mathcal{B}_t$ is left implicit to keep the notation light. For a candidate batch $\mathcal{B}_t$, we define the loss as

$$\ell^{t+1}(x, y; \mathcal{B}_t) = -\log \operatorname{softmax}\big(f_{\theta_{t+1}(\mathcal{B}_t)}(x)\big)\,[y].$$

For brevity, we let $\ell_i^{t+1}(\mathcal{B}_t) = \ell^{t+1}(x_i, y_i; \mathcal{B}_t)$. The corresponding population risk is

$$\mathcal{J}_{t+1}(\mathcal{B}_t) = \mathbb{E}_{(X,Y)\sim p_{\mathcal{Z}}}\left[\ell^{t+1}(X, Y; \mathcal{B}_t)\right].$$

We want to obtain the ideal query batch such that

$$\mathcal{B}_t^* \in \underset{\substack{\mathcal{B}_t \subseteq \mathcal{U}_t \\ |\mathcal{B}_t| \leq b}}{\arg\min}\, \mathcal{J}_{t+1}(\mathcal{B}_t).$$

Since $\mathcal{J}_{t+1}(\mathcal{B}_t)$ is unavailable at selection time, we introduce the post-query empirical risk for $\mathcal{S} \in \{\mathcal{P}_t, \mathcal{L}_t^+\}$:

$$\widehat{\mathcal{J}}_{\mathcal{S}}^{t+1}(\mathcal{B}_t) = \frac{1}{|\mathcal{S}|} \sum_{x_i \in \mathcal{S}} \ell_i^{t+1}(\mathcal{B}_t).$$

By adding and subtracting the empirical losses over $\mathcal{P}_t$ and $\mathcal{L}_t^+$, we obtain the following equation by (Sener & Savarese, 2018):

$$\begin{aligned}
\mathcal{J}_{t+1}(\mathcal{B}_t) &= \mathbb{E}_{(X,Y)\sim p_{\mathcal{Z}}}\left[\ell^{t+1}(X,Y;\mathcal{B}_t)\right] \\
&= \left(\mathbb{E}_{(X,Y)\sim p_{\mathcal{Z}}}\left[\ell^{t+1}(X,Y;\mathcal{B}_t)\right] - \frac{1}{|\mathcal{P}_t|}\sum_{x_i\in\mathcal{P}_t}\ell_i^{t+1}(\mathcal{B}_t)\right) + \frac{1}{|\mathcal{P}_t|}\sum_{x_i\in\mathcal{P}_t}\ell_i^{t+1}(\mathcal{B}_t) \\
&\leq \underbrace{\left|\mathbb{E}_{(X,Y)\sim p_{\mathcal{Z}}}\left[\ell^{t+1}(X,Y;\mathcal{B}_t)\right] - \frac{1}{|\mathcal{P}_t|}\sum_{x_i\in\mathcal{P}_t}\ell_i^{t+1}(\mathcal{B}_t)\right|}_{\Gamma_{t+1}(\mathcal{B}_t)} + \frac{1}{|\mathcal{P}_t|}\sum_{x_i\in\mathcal{P}_t}\ell_i^{t+1}(\mathcal{B}_t) \\
&= \Gamma_{t+1}(\mathcal{B}_t) + \left(\frac{1}{|\mathcal{P}_t|}\sum_{x_i\in\mathcal{P}_t}\ell_i^{t+1}(\mathcal{B}_t) - \frac{1}{|\mathcal{L}_t^+|}\sum_{x_i\in\mathcal{L}_t^+}\ell_i^{t+1}(\mathcal{B}_t)\right) + \frac{1}{|\mathcal{L}_t^+|}\sum_{x_i\in\mathcal{L}_t^+}\ell_i^{t+1}(\mathcal{B}_t) \\
&\leq \Gamma_{t+1}(\mathcal{B}_t) + \frac{1}{|\mathcal{L}_t^+|}\sum_{x_i\in\mathcal{L}_t^+}\ell_i^{t+1}(\mathcal{B}_t) + \underbrace{\left|\frac{1}{|\mathcal{P}_t|}\sum_{x_i\in\mathcal{P}_t}\ell_i^{t+1}(\mathcal{B}_t) - \frac{1}{|\mathcal{L}_t^+|}\sum_{x_i\in\mathcal{L}_t^+}\ell_i^{t+1}(\mathcal{B}_t)\right|}_{\mathcal{C}_{t+1}(\mathcal{B}_t)}.
\end{aligned}$$

where $\Gamma_{t+1}(\mathcal{B}_t)$ denotes the population pool gap and $\mathcal{C}_{t+1}(\mathcal{B}_t)$ denotes the core-set loss. Starting from the post query core-set loss,

$$\begin{aligned}
\mathcal{C}_{t+1}(\mathcal{B}_t) &= \left|\frac{1}{|\mathcal{P}_t|}\sum_{x_i\in\mathcal{P}_t}\ell_i^{t+1}(\mathcal{B}_t) - \frac{1}{|\mathcal{L}_t^+|}\sum_{x_i\in\mathcal{L}_t^+}\ell_i^{t+1}(\mathcal{B}_t)\right| \\
&= \left|\left[\frac{1}{|\mathcal{P}_t|}\sum_{x_i\in\mathcal{P}_t}\ell_i^{t} - \frac{1}{|\mathcal{L}_t^+|}\sum_{x_i\in\mathcal{L}_t^+}\ell_i^{t}\right]\right. \\
&\qquad \left. + \left[\frac{1}{|\mathcal{P}_t|}\sum_{x_i\in\mathcal{P}_t}\left(\ell_i^{t+1}(\mathcal{B}_t) - \ell_i^t\right) - \frac{1}{|\mathcal{L}_t^+|}\sum_{x_i\in\mathcal{L}_t^+}\left(\ell_i^{t+1}(\mathcal{B}_t) - \ell_i^t\right)\right]\right| \\
&\leq \underbrace{\left|\frac{1}{|\mathcal{P}_t|}\sum_{x_i\in\mathcal{P}_t}\ell_i^{t} - \frac{1}{|\mathcal{L}_t^+|}\sum_{x_i\in\mathcal{L}_t^+}\ell_i^{t}\right|}_{\mathcal{C}_t(\mathcal{B}_t)} \\
&\qquad + \underbrace{\left|\frac{1}{|\mathcal{P}_t|}\sum_{x_i\in\mathcal{P}_t}\left(\ell_i^{t+1}(\mathcal{B}_t) - \ell_i^t\right) - \frac{1}{|\mathcal{L}_t^+|}\sum_{x_i\in\mathcal{L}_t^+}\left(\ell_i^{t+1}(\mathcal{B}_t) - \ell_i^t\right)\right|}_{\Xi_t(\mathcal{B}_t)}.
\end{aligned} \tag{16}$$

Hence,

$$\mathcal{C}_{t+1}(\mathcal{B}_t) \leq \mathcal{C}_t(\mathcal{B}_t) + \Xi_t(\mathcal{B}_t), \tag{17}$$

where $\mathcal{C}_t(\mathcal{B}_t)$ is the core-set loss evaluated using the current model, while $\Xi_t(\mathcal{B}_t)$ captures the discrepancy introduced by the unknown model update. For a candidate batch $\mathcal{B}_t$, define its covering radius in the explanation space as

$$r_t(\mathcal{B}_t) := \max_{x_j \in \mathcal{U}_t} \min_{x_i \in \mathcal{L}_t^+} \delta_{ij}.$$

Since $\delta_{ij} = \sqrt{d_{\mathrm{JSD}}(x_i, x_j)}$, minimizing $r_t(\mathcal{B}_t)$ is equivalent to minimizing the JSD k-Center objective used in the acquisition rule. For each $x_j \in \mathcal{P}_t$, let

$$\pi_{\mathcal{B}_t}(j) \in \arg\min_{x_i \in \mathcal{L}_t^+} \delta_{ij},$$

with $\pi_{\mathcal{B}_t}(j) = j$ for $x_j \in \mathcal{L}_t^+$. Hence, $\delta_{j,\pi_{\mathcal{B}_t}(j)} \le r_t(\mathcal{B}_t)$ for every unselected sample.

Starting from the selection-time core-set loss, we obtain

$$\begin{aligned}
\mathcal{C}_t(\mathcal{B}_t) &= \left| \frac{1}{|\mathcal{P}_t|} \sum_{x_j \in \mathcal{P}_t} \ell_j^t - \frac{1}{|\mathcal{L}_t^+|} \sum_{x_i \in \mathcal{L}_t^+} \ell_i^t \right| \\
&= \left| \frac{1}{|\mathcal{P}_t|} \sum_{x_j \in \mathcal{P}_t} \left(\ell_j^t - \ell_{\pi_{\mathcal{B}_t}(j)}^t \right) + \frac{1}{|\mathcal{P}_t|} \sum_{x_j \in \mathcal{P}_t} \ell_{\pi_{\mathcal{B}_t}(j)}^t - \frac{1}{|\mathcal{L}_t^+|} \sum_{x_i \in \mathcal{L}_t^+} \ell_i^t \right| \\
&\le \frac{1}{|\mathcal{P}_t|} \sum_{x_j \in \mathcal{P}_t} \left| \ell_j^t - \ell_{\pi_{\mathcal{B}_t}(j)}^t \right| + \left| \frac{1}{|\mathcal{P}_t|} \sum_{x_j \in \mathcal{P}_t} \ell_{\pi_{\mathcal{B}_t}(j)}^t - \frac{1}{|\mathcal{L}_t^+|} \sum_{x_i \in \mathcal{L}_t^+} \ell_i^t \right| \\
&\le \frac{1}{|\mathcal{P}_t|} \sum_{x_j \in \mathcal{P}_t} \left| \ell_j^t - \ell_{\pi_{\mathcal{B}_t}(j)}^t \right| + \underbrace{\left(\max_{x_i \in \mathcal{L}_t^+} \ell_i^t - \min_{x_i \in \mathcal{L}_t^+} \ell_i^t \right)}_{\mathrm{osc}_t(\mathcal{B}_t)}
\end{aligned} \tag{18}$$

We continual to derive the upper bound of $\left| \ell_j^t - \ell_{\pi_{\mathcal{B}_t}(j)}^t \right|$ and propose the following proposition.

Proposition C.1 (Pairwise loss bound)**.** *For any $x_i, x_j \in \mathcal{P}_t$ with $\delta_{ij} \le r$,*

$$\boxed{|\ell_i^t - \ell_j^t| \le 2\varepsilon_t^{\mathrm{KL}} + 4nG_t\, r + \sqrt{2}\, n\mathcal{H}_t(r) + \mathcal{Y}_t(r).} \tag{19}$$

Proof.

$$\begin{aligned}
|\ell_i^t - \ell_j^t| &= |\ell_i(z_i) - \ell_j(z_j)| \\
&\le |\ell_i(z_i) - \ell_i(\widetilde{z}_i)| + |\ell_i(\widetilde{z}_i) - \ell_i(\widetilde{z}_j)| + |\ell_i(\widetilde{z}_j) - \ell_j(\widetilde{z}_j)| + |\ell_j(\widetilde{z}_j) - \ell_j(z_j)|\,.
\end{aligned}$$

For the first and fourth terms, let the reconstruction residual be $\mathbf{e}_i = z_i - \widetilde{z}_i$. Since softmax is invariant to a common shift of all logits, only the centered component of $\mathbf{e}_i$ affects the predictive distribution. We therefore define

$$\mathbf{e}_i^c = \mathbf{e}_i - \frac{1}{K}(\mathbf{1}^\top \mathbf{e}_i)\mathbf{1}, \quad p_i(s) = \mathrm{softmax}(z_i - s\mathbf{e}_i), \qquad \alpha_i = \min_{\substack{s \in [0,1] \\ k \in \{1,\dots,K\}}} p_i(s)[k],$$

The reconstruction-induced supervised-loss discrepancy can therefore be bounded directly as

$$\begin{aligned}
|\ell_i(z_i) - \ell_i(\widetilde{z}_i)| &\le \sqrt{2}\, \|\mathbf{e}_i^c\|_2 \\
&\le \sqrt{\frac{4}{K\alpha_i^2} \int_0^1 (1-s)\, \mathbf{e}_i^\top \Big(\mathrm{Diag}(p_i(s)) - p_i(s)p_i(s)^\top \Big) \mathbf{e}_i \, ds} \\
&= \sqrt{\frac{4}{K\alpha_i^2} \left[F(z_i - \mathbf{e}_i) - F(z_i) + \nabla F(z_i)^\top \mathbf{e}_i \right]} \\
&= \sqrt{\frac{4\kappa_i}{K\alpha_i^2}} \\
&= \frac{2}{\alpha_i \sqrt{K}} \sqrt{\kappa_i}.
\end{aligned} \tag{20}$$

Where $\kappa_i = \mathrm{KL}\left(\hat{p}_i \,\|\, \widetilde{p}_i\right)$. Then, we have the following equation:

$$\varepsilon_t^{\mathrm{KL}} = \max_{x_i \in \mathcal{P}_t} \frac{2}{\alpha_i \sqrt{K}} \sqrt{\kappa_i}, \quad |\ell_i(z_i) - \ell_i(\widetilde{z}_i)| + |\ell_j(\widetilde{z}_j) - \ell_j(z_j)| \le 2\varepsilon_t^{\mathrm{KL}}.$$

Our Method defines $E_i[m, v] = \sum_{\tau \in \mathcal{T}_m(T_i)} D_i^*[\tau, v]$ and $\widetilde{E}_i = E_i / \|E_i\|_1$. Since $\|D_i^*\|_1 = n$, we have $\|E_i\|_1 = n$ and therefore

$$\widetilde{E}_i[m, v] = \frac{E_i[m, v]}{n}.$$

For $E_i[m, v] > 0$, define the selected average relevance within a temporal stage and joint by

$$\overline{\mathbf{R}}_{i,mv} = \frac{\sum_{\tau \in \mathcal{T}_m(T_i)} D_i^*[\tau, v]\, \mathbf{R}_{i,\tau v}}{E_i[m, v]},$$

For each stage–joint contribution, define its centered relevance vector and the corresponding finite-pool relevance scale as

$$\overline{\mathbf{R}}_{i,mv}^{\,c} = \overline{\mathbf{R}}_{i,mv} - \frac{1}{K}\left(\mathbf{1}^\top \overline{\mathbf{R}}_{i,mv}\right)\mathbf{1}, \quad G_t = \max_{\substack{x_i \in \mathcal{P}_t \\ 1 \le m \le M,\ 1 \le v \le V}} \left\| \overline{\mathbf{R}}_{i,mv}^{\,c} \right\|_2.$$

$$\mathcal{H}_t(r) = \max_{\substack{x_i, x_j \in \mathcal{P}_t \\ \delta_{ij} \le r}} \frac{1}{2} \sum_{m=1}^{M} \sum_{v=1}^{V} \left(\widetilde{E}_i[m, v] + \widetilde{E}_j[m, v]\right) \times \left\| \overline{\mathbf{R}}_{i,mv}^{\,c} - \overline{\mathbf{R}}_{j,mv}^{\,c} \right\|_2.$$

For any pair $x_i, x_j \in \mathcal{P}_t$ satisfying $\delta_{ij} \le r$, the loss discrepancy induced by the reconstructed logits can be bounded directly as

$$\begin{aligned}
|\ell_i(\widetilde{z}_i) - \ell_i(\widetilde{z}_j)| &\le \sqrt{2} \left\| (\widetilde{z}_i - \widetilde{z}_j) - \frac{1}{K} \mathbf{1}^\top (\widetilde{z}_i - \widetilde{z}_j) \mathbf{1} \right\|_2 \\
&= \sqrt{2}\, n \left\| \sum_{m=1}^{M} \sum_{v=1}^{V} \left(\widetilde{E}_i[m, v]\, \overline{\mathbf{R}}_{i,mv}^{\,c} - \widetilde{E}_j[m, v]\, \overline{\mathbf{R}}_{j,mv}^{\,c} \right) \right\|_2 \\
&= \sqrt{2}\, n \Bigg\| \frac{1}{2} \sum_{m=1}^{M} \sum_{v=1}^{V} \left(\widetilde{E}_i[m, v] - \widetilde{E}_j[m, v]\right) \left(\overline{\mathbf{R}}_{i,mv}^{\,c} + \overline{\mathbf{R}}_{j,mv}^{\,c}\right) \\
&\qquad + \frac{1}{2} \sum_{m=1}^{M} \sum_{v=1}^{V} \left(\widetilde{E}_i[m, v] + \widetilde{E}_j[m, v]\right) \left(\overline{\mathbf{R}}_{i,mv}^{\,c} - \overline{\mathbf{R}}_{j,mv}^{\,c}\right) \Bigg\|_2 \\
&\le \frac{\sqrt{2}\, n}{2} \sum_{m=1}^{M} \sum_{v=1}^{V} \left| \widetilde{E}_i[m, v] - \widetilde{E}_j[m, v] \right| \left\| \overline{\mathbf{R}}_{i,mv}^{\,c} + \overline{\mathbf{R}}_{j,mv}^{\,c} \right\|_2 \\
&\qquad + \frac{\sqrt{2}\, n}{2} \sum_{m=1}^{M} \sum_{v=1}^{V} \left(\widetilde{E}_i[m, v] + \widetilde{E}_j[m, v]\right) \left\| \overline{\mathbf{R}}_{i,mv}^{\,c} - \overline{\mathbf{R}}_{j,mv}^{\,c} \right\|_2 \\
&\le \sqrt{2}\, n G_t \sum_{m=1}^{M} \sum_{v=1}^{V} \left| \widetilde{E}_i[m, v] - \widetilde{E}_j[m, v] \right| + \sqrt{2}\, n \mathcal{H}_t(r) \\
&= \sqrt{2}\, n G_t \left\| \widetilde{E}_i - \widetilde{E}_j \right\|_1 + \sqrt{2}\, n \mathcal{H}_t(r) \\
&\le \sqrt{2}\, n G_t \sqrt{8\ \mathrm{JSD}\left(\widetilde{E}_i, \widetilde{E}_j\right)} + \sqrt{2}\, n \mathcal{H}_t(r) \\
&= 4 n G_t\, \delta_{ij} + \sqrt{2}\, n \mathcal{H}_t(r) \\
&\le 4 n G_t\, r + \sqrt{2}\, n \mathcal{H}_t(r).
\end{aligned} \tag{21}$$

Let

$$\mathcal{Y}_t(r) = \max_{\substack{x_i, x_j \in \mathcal{P}_t \\ \delta_{ij} \le r}} \max\Big\{ |\widetilde{z}_i[y_i] - \widetilde{z}_i[y_j]|, \quad |\widetilde{z}_j[y_i] - \widetilde{z}_j[y_j]| \Big\}. \tag{22}$$

The third term is at most $\mathcal{Y}_t(r)$ by definition. Summing the four terms proves the result. □

Thus, we obtain the following population-risk bound:

$$\begin{aligned}\mathcal{J}_{t+1}(\mathcal{B}_t) \le & \underbrace{\Gamma_{t+1}(\mathcal{B}_t) + \widehat{\mathcal{J}}^{\,t+1}_{\mathcal{L}_t^+}(\mathcal{B}_t) + \Xi_t(\mathcal{B}_t) + \mathrm{osc}_t(\mathcal{B}_t)}_{\text{population, training, and model-update terms}} \\ & + \underbrace{2\varepsilon_t^{\mathrm{KL}}}_{\text{LRP reconstruction distortion}} \\ & + \underbrace{4nG_t\, r_t(\mathcal{B}_t) + \sqrt{2}\, n\mathcal{H}_t(r_t(\mathcal{B}_t)) + \mathcal{Y}_t(r_t(\mathcal{B}_t))}_{\text{explanation-space approximation error}}.\end{aligned} \tag{23}$$

For compactness, define the first group of terms as

$$\begin{aligned}\mathcal{E}_t^{\mathrm{base}}(\mathcal{B}_t) =& \Gamma_{t+1}(\mathcal{B}_t) + \widehat{\mathcal{J}}^{\,t+1}_{\mathcal{L}_t^+}(\mathcal{B}_t) \\ & + \Xi_t(\mathcal{B}_t) + \mathrm{osc}_t(\mathcal{B}_t).\end{aligned}$$

We first simplify the KL reconstruction term. Recall that

$$\varepsilon_t^{\mathrm{KL}} = \max_{x_i \in \mathcal{P}_t} \frac{2}{\alpha_i \sqrt{K}} \sqrt{\kappa_i}.$$

Define the finite-pool quantities

$$\alpha_t = \min_{x_i \in \mathcal{P}_t} \alpha_i, \qquad \bar{\kappa}_t = \max_{x_i \in \mathcal{P}_t} \kappa_i.$$

Since $\mathcal{P}_t$ is finite and $\alpha_i > 0$ for every sample, we have $\alpha_t > 0$. Therefore,

$$2\varepsilon_t^{\mathrm{KL}} = 4 \max_{x_i \in \mathcal{P}_t} \frac{\sqrt{\kappa_i}}{\alpha_i \sqrt{K}} \le \frac{4}{\alpha_t \sqrt{K}} \sqrt{\bar{\kappa}_t}.$$

We next simplify the local relevance and label heterogeneity terms without imposing a global smoothness assumption. For each pair $x_i, x_j \in \mathcal{P}_t$, define its local relevance discrepancy as

$$h_{ij} = \frac{1}{2} \sum_{m=1}^{M} \sum_{v=1}^{V} \left(\widetilde{E}_i[m,v] + \widetilde{E}_j[m,v] \right) \times \left\| \overline{\mathbf{R}}^{\,c}_{i,mv} - \overline{\mathbf{R}}^{\,c}_{j,mv} \right\|_2, \mathcal{H}_t(r) = \max_{\substack{x_i, x_j \in \mathcal{P}_t \\ \delta_{ij} \le r}} h_{ij}$$

Similarly, define the pairwise label-induced discrepancy

$$y_{ij} = \max\left\{ |\widetilde{z}_i[y_i] - \widetilde{z}_i[y_j]|, |\widetilde{z}_j[y_i] - \widetilde{z}_j[y_j]| \right\}, \quad \mathcal{Y}_t(r) = \max_{\substack{x_i, x_j \in \mathcal{P}_t \\ \delta_{ij} \le r}} y_{ij}.$$

Because the current pool is finite, we can construct finite-pool affine envelopes directly. Define

$$L_{R,t} = \max_{\substack{x_i, x_j \in \mathcal{P}_t \\ \delta_{ij} > 0}} \frac{h_{ij}}{\delta_{ij}}, \qquad L_{Y,t} = \max_{\substack{x_i, x_j \in \mathcal{P}_t \\ \delta_{ij} > 0}} \frac{y_{ij}}{\delta_{ij}}.$$

By construction, for every pair $x_i, x_j \in \mathcal{P}_t$,

$$h_{ij} \le L_{R,t} \delta_{ij}, \qquad y_{ij} \le L_{Y,t} \delta_{ij}.$$

Hence,

$$\mathcal{H}_t(r) = \max_{\delta_{ij} \le r} h_{ij} \le \max_{\delta_{ij} \le r} L_{R,t} \delta_{ij} \le L_{R,t} r,$$

Analogously, define

$$\mathcal{Y}_t(r) = \max_{\delta_{ij} \le r} y_{ij} \le \max_{\delta_{ij} \le r} L_{Y,t} \delta_{ij} \le L_{Y,t} r.$$

Substituting these equations gives

$$\begin{aligned} & 4nG_t\, r + \sqrt{2}\, n\mathcal{H}_t(r) + \mathcal{Y}_t(r) \\ & \le 4nG_t\, r + \sqrt{2}\, nL_{R,t} r + L_{Y,t} r \\ & = \Big(4nG_t + \sqrt{2}\, nL_{R,t} + L_{Y,t}\Big)\, r \\ & = C_{\mathrm{geo},t}\, r, \end{aligned} \tag{24}$$

We finally obtain

$$\mathcal{J}_{t+1}(\mathcal{B}_t) \le \underbrace{\mathcal{E}_t^{\mathrm{base}}(\mathcal{B}_t)}_{\substack{\text{population, training,} \\ \text{and model-update error}}} + \underbrace{\frac{4}{\alpha_t\sqrt{K}}\sqrt{\bar{\kappa}_t}}_{\text{LRP reconstruction distortion}} + \underbrace{C_{\mathrm{geo},t} r_t(\mathcal{B}_t)}_{\text{explanation-space coverage error}}. \tag{25}$$