

Easy2Hard: From Partially to Fully Unmatched Modalities as Negative Samples in Contrastive Learning

Zhicheng Yang¹ Yichen Liu² Chang Ge² Xiaopeng Jiang¹

¹Southern Illinois University ²University of Minnesota

zhicheng.yang@siu.edu liu03489@umn.edu cge@umn.edu xiaopeng.jiang@siu.edu

Abstract

Contrastive learning is widely used for generating multimodal data representations by aligning embeddings of different modalities of the same data samples. This alignment is achieved through a loss function that treats matched and unmatched modality pairs as positive and negative samples within a data batch. However, when extending contrastive learning to scenarios involving more than two modalities, existing approaches either rely solely on fully unmatched modalities as negative samples, or fail to distinguish between partially and fully unmatched modalities, thereby overlooking the fine-grained contrasting relationships. To address this limitation, we propose Easy2Hard, a novel framework that explicitly separates partially and fully unmatched modalities. To learn from negative samples at improved granularity, Easy2Hard further introduces a sigmoid weighting curriculum that smoothly transitions the learning process from easy (partially unmatched) to hard (fully unmatched) contrasts. Comprehensive evaluations on five multimodal datasets across diverse domains demonstrate the superiority of our approach.

1. Introduction

Multiple complementary modalities are often available for the same data sample in many real-world applications: news stories include text, photos and audio; smartphone applications leverage data from multiple sensors [9, 10, 17]; e-commerce listings combine images, descriptions and attribute tables. Learning a unified representation that aligns these modalities enables flexible cross-modal search, many-to-one retrieval, and open-vocabulary recognition without task-specific heads [8, 14].

State-of-the-art (SOTA) cross-modal models commonly adopt contrastive learning as their primary training framework: given a mini-batch, matched tuples (positives) are encouraged to have higher similarity than mismatched tuples (negatives) via a temperature-scaled softmax (InfoNCE)

over in-batch candidates [3, 13]. In the bi-modal case (e.g., image-text), common practice typically follows one of two patterns. One line of work mines the *hardest* negative per query (i.e., the non-matching item with the largest similarity under the chosen scorer) in supervised image-text retrieval, as in VSE++ [6]. Another includes *all* non-matching pairs in the softmax, as in CLIP [14], ALIGN [8], and LiT [24]. For more than two modalities ($M > 2$), extensions often sum pairwise losses over modality pairs or bind every modality to a single pivot (frequently the image), as in ImageBind [7] and TriCoLo [15]; these designs scale well but still optimize only pairwise interactions and provide no explicit structure over negative difficulty.

Unlike the bimodal case, where under a fixed anchor a negative pair has only one form of mismatch, when $M > 2$, in-batch negatives naturally stratify by *overlap with the positive*: some negatives still share one or more non-anchor modalities with the positive (partially unmatched / easier), whereas others share none (fully unmatched / harder). Intuitively, treating all negatives uniformly can be suboptimal. The model learns more effectively by starting with easier, partially unmatched negatives and gradually introducing fully unmatched negatives as harder contrastive samples.

We propose Easy2Hard, a multimodal contrastive learning framework that explicitly separates partially unmatched and fully unmatched tuples as negative samples for contrastive learning. Our method is formulated to handle an arbitrary number of modalities. For each anchor modality, Easy2Hard splits in-batch negatives into *easy sets* (partially unmatched tuples according to the number of unmatched modalities) and a *hard set* (fully unmatched tuples), computes the loss of each contrastive sample, and gradually shifts training weight from the easy sets to the hard set over time with a sigmoid-based learning curriculum. In this process, the encoders and base objective can be the same as other SOTA contrastive learning techniques. This negative partition can be constructed in $O(B)$ time, where B is the batch size, via axis-wise shuffling of non-anchor indices.

Our experiments show that Easy2Hard improves zero-shot cross-modal retrieval over strong baselines on four tri-

modal datasets and remains effective in a 5-modal HoloAssist feasibility study. Our ablation results show that a naive linear curriculum learning strategy does not perform well. Instead, our performance gains arise from the combination of structured negative construction and the proposed smooth learning curriculum. Our contributions are: (i) a fine-grained negative sampling for M -modal contrastive learning that separates *partially* vs. *fully* unmatched tuples per anchor; (ii) a lightweight, monotone curriculum learning schedule that shifts emphasis from easy to hard without changing encoders or adding task-specific heads, designed in $O(B)$; (iii) an empirical study with ablations and head-to-head comparisons against total-correlation/multilinear inner-product (TC/MIP) and pairwise baselines.

2. Related Work

Pairwise contrastive pretraining and multimodal extensions. Large-scale image-text pretraining such as CLIP [14] and ALIGN [8] adopts dual-encoder architectures with pairwise InfoNCE/normalized-softmax over in-batch negatives, achieving strong open-vocabulary recognition and retrieval at scale. Building on this recipe, subsequent work improves the objective and scaling behavior in different ways. SigLIP [25] replaces the normalized-softmax with a sigmoid multi-label loss to reduce batch-size sensitivity; OpenCLIP [4] systematically studies data, model, and optimizer choices under a unified implementation; and EVA-CLIP [19] strengthens CLIP-style pretraining with more powerful visual backbones. Related multimodal pretraining frameworks such as LiT [24] and Florence [23] further explore alternative data mixtures, distillation, and supervision sources.

A straightforward route to $M > 2$ is to extend pairwise contrastive learning by summing or symmetrizing losses across modality pairs, or by binding all modalities to a pivot modality. Early multi-view contrastive objectives such as Contrastive Multiview Coding (CMC) [20] already follow this decomposition into view-pair interactions, which later multimodal extensions inherit. FILIP [22] and ALBEF [12] enrich pairwise supervision through token-level alignment and cross-attention, while ImageBind [7] aligns diverse modalities in an image-anchored space without requiring full co-observation. MXM-CLR [21] addresses the challenge of multiple positive observations per instance across modalities. Despite these advances, the objective in such methods still fundamentally factorizes into pairwise terms, which can under-specify negative difficulty and miss joint dependencies among three or more modalities.

Beyond pairwise: TC/MIP and structured multimodal negatives. Mutual-information and total-correlation (TC) objectives have long been explored in multi-view and multimodal representation learning [1, 13]. Symile [16] is partic-

ularly relevant to our setting because it introduces a model-agnostic objective based on TC, instantiated through multilinear inner-product (MIP) scoring, to capture dependencies beyond pairwise interactions with an $O(B)$ in-batch construction. However, Symile treats all in-batch negatives equally. Under a fixed anchor, it does not distinguish between *partially* unmatched and *fully* unmatched tuples, nor does it vary their relative emphasis over training. Our method is complementary to this line of work: we retain the same TC/MIP scoring and encoders, but impose additional structure on the negative set by partitioning tuples according to cross-modal overlap and then reweighting them through an easy-to-hard schedule. Our method outperforms the SOTA baselines, and our ablation results show that our performance gains arise from the combination of structured negative construction and the proposed smooth curriculum.

Hard negatives. Independently of architecture and objective, leveraging harder negatives or scheduling difficulty has often been shown to improve discrimination and optimization. VSE++ [6], for example, mines the hardest negative per query in supervised image-text retrieval. Choi et al. [5] propose an adaptive hard-negative sampling strategy for multimodal Human Activity Recognition (HAR), using an adjustable concentration parameter to emphasize negatives that are closer to the anchor in the latent space. Their method therefore focuses on modulating negative hardness through similarity-based sampling within a multimodal contrastive framework. However, such methods do not explicitly model the structured negative hierarchy induced by cross-modal overlap in $M > 2$ settings, where negatives can differ in how many non-anchor modalities remain matched with the positive tuple.

Curriculum learning. Curriculum learning [2] and self-paced learning [11] formalize the idea of ordering training examples by difficulty, and have been applied across vision and language tasks [18]. While some of the aforementioned works incorporate new sampling strategies, they primarily modify the loss function to integrate them. In contrast, we use curriculum scheduling to modulate the relative emphasis of structured negative subsets, while keeping the underlying contrastive objective and encoders fixed.

3. Preliminaries

Notation. We consider M modalities with modality-specific encoders that produce embeddings $z_i^{(m)} \in \mathbb{R}^d$ for the i -th sample and modality m . For a mini-batch of size B , indices $i, j \in \{1, \dots, B\}$ refer to batch elements. We use `logit_scale` to parameterize the temperature as $\tau = \exp(\text{logit_scale})$.

Multilinear inner-product (MIP) scoring. Following Symile [16], we use a TC-style multilinear inner product. Given an index tuple $\mathbf{j} = (j_1, \dots, j_M)$, where each j_m specifies which batch element is selected for modality m , define

$$s(\mathbf{j}) = \sum_{r=1}^d \prod_{m=1}^M z_{j_m, r}^{(m)}. \quad (1)$$

For an anchor modality $a \in \{1, \dots, M\}$ and a pair of batch elements (i, j) , we construct a mixed index tuple where the anchor comes from i and all non-anchors from j :

$$\mathbf{j}^{(a)}(i, j) = (j_1, \dots, j_M), \quad j_a = i, \quad j_m = j \text{ for } m \neq a, \quad (2)$$

and define the corresponding temperature-scaled score

$$s_{i,j}^{(a)} = \tau s(\mathbf{j}^{(a)}(i, j)). \quad (3)$$

Per-anchor contrastive loss. For a batch $\mathcal{B}$ and anchor modality a , we collect scores $\{s_{i,j}^{(a)}\}_{i,j=1}^B$ into a matrix $L^{(a)} \in \mathbb{R}^{B \times B}$. For each anchor sample i , the entry with $j = i$ corresponds to the matched (positive) tuple, while entries with $j \neq i$ correspond to in-batch negatives.

$$\ell^{(a)}(\mathcal{B}; \theta) = -\frac{1}{B} \sum_{i=1}^B \log \frac{\exp(L_{i,i}^{(a)})}{\sum_{j=1}^B \exp(L_{i,j}^{(a)})}. \quad (4)$$

4. Multimodal Representation Learning via Fine-grained Contrasting

4.1. Problem Definition

We define a matched M -tuple as views originating from the same underlying instance (same index i). Our goal is to learn modality-specific encoders that align matched tuples and separate mismatched tuples in a shared space. These encoders can be used for different down-stream tasks, or zero-shot cross-modal retrieval (Sec. 5) without task-specific heads.

The base TC/MIP objective averages the per-anchor contrastive losses over anchors and mini-batches:

$$L_{\text{TC/MIP}}(\theta) = \mathbb{E}_{\mathcal{B} \sim \mathcal{D}} \left[\frac{1}{M} \sum_{a=1}^M \ell^{(a)}(\mathcal{B}; \theta) \right]. \quad (5)$$

Training then amounts to solving

$$\theta^* = \arg \min_{\theta} L_{\text{TC/MIP}}(\theta). \quad (6)$$

On top of this base objective, Easy2Hard partitions in-batch negatives into partially and fully unmatched sets and reweights their contributions over training time, while keeping the encoders and MIP scoring unchanged.

4.2. Motivating Application Scenarios

Many real-world data naturally provide multiple complementary views per item (e.g., a news article with text and images, a product listing with photos and descriptions, or a short video with frames, audio, and subtitles). Our objective is to forge a unified embedding that robustly aligns these heterogeneous modalities ($M > 2$), enabling flexible, open-vocabulary retrieval and classification tasks that extend beyond closed training sets.

Taking cross-modal retrieval and search as an example, given one or multiple query views, the model ranks candidates from another view in a shared space. In particular, two-to-one retrieval (e.g., $\text{Txt} \otimes \text{Aud} \rightarrow \text{Img}$) is naturally handled by the MIP used in TC-style training (see Sec. 3), enabling zero-shot retrieval without task-specific heads. This matches our evaluation protocol in Sec. 5.

4.3. Overview

Easy2Hard augments TC-based contrastive training with a structured, time-varying negative set. Figure 1 contrasts the two regimes: in panel (i), standard TC/MIP training treats all in-batch negatives equally (the positives are occluded along the diagonal line in yellow), ignoring their difficulty structure, while panel (ii) shows how *Easy2Hard* partitions the negative set into partially unmatched (easy) and fully unmatched (hard) sets. For clarity, Fig. 2 zooms into a single anchor slice and illustrates representative partially and fully unmatched tuples. For each anchor modality, we (i) build these distinct easy and hard sets via axis permutations, (ii) compute cross-entropy on each set, and (iii) convex-combine the resulting losses via an *external* schedule $w(t) \in [0, 1]$, followed by averaging across anchors. The remaining components of the pipeline (encoders, MIP scoring, optimizer) remain unchanged as in conventional contrastive learning.

4.4. Fine-grained Negative Sample Construction

Levels by overlap (definition). With an anchor modality m fixed and the other views given by $\mathcal{V} = \{1, \dots, M\} \setminus \{m\}$, negatives are grouped by the *overlap count* $c \in \{0, \dots, M-2\}$ —the number of non-anchor modalities that coincide with the positive item. Since a negative tuple cannot match all $M-1$ non-anchor modalities simultaneously, we define the difficulty level as $\ell = (M-1) - c \in \{1, 2, \dots, M-1\}$; larger ℓ means harder (smaller overlap). Let $\mathcal{N}_{\ell}$ be the set of negatives at level ℓ and let $\mathcal{L}_{\ell}$ denote the CE restricted to $\mathcal{N}_{\ell}$ (no weighting here).

Trimodal example. For $M=3$ with (Img, Txt, Aud) and the image as anchor, Figure 2 illustrates one mini-batch with size $B=4$. Each sub-cube in the panel labelled “ $I = i$ ” corresponds to a tuple $\langle I_i, T_q, A_p \rangle$, where the row index p selects the audio item and the column index q selects the text item. In each slice $I = i$, the yellow cell $\langle I_i, T_i, A_i \rangle$

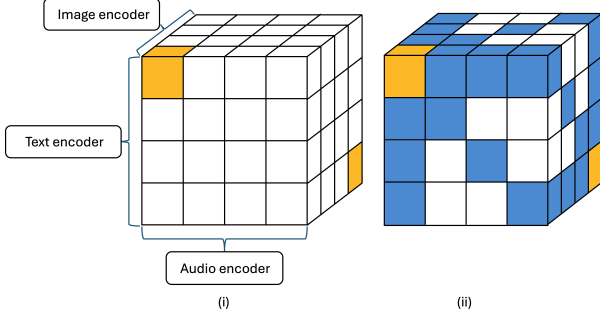


Figure 1. **Negative hierarchy.** Yellow denotes the positive tuple in both panels. In panel (i), Symile treats all in-batch negatives (white) uniformly and does not distinguish partially unmatched from fully unmatched tuples. In panel (ii), Easy2Hard partitions negatives into partially unmatched (blue) and fully unmatched (white) sets. Figure 2 provides an unoccluded zoom-in view of panel (ii) for a fixed anchor slice.

Algorithm 1 BUILDNEGSETS(a, B): linear-time construction via axis permutations

Require: Anchor a , batch size B

- 1: **for** each non-anchor modality $m \neq a$ **do**
 - 2: Sample a random permutation π_m of $\{1, \dots, B\} \triangleright$ in-batch, axis-wise
 - 3: **end for**
 - 4: **Easy set** $\mathcal{N}_{\text{part}}^{(a)}$: for each i , uniformly sample one non-anchor axis $m' \neq a$; set $j_a = i$, $j_{m'} = \pi_{m'}(i)$, and $j_m = i$ for all other $m \neq a, m'$.
 - 5: **Hard set** $\mathcal{N}_{\text{full}}^{(a)}$: for each i , set $j_a = i$ and $j_m = \pi_m(i)$ for all $m \neq a$ (allow $j_u = j_v \neq i$)
 - 6: **return** $(\mathcal{N}_{\text{part}}^{(a)}, \mathcal{N}_{\text{full}}^{(a)})$
-

is the unique *positive* triple for anchor image I_i . The blue cells are *partially unmatched* negatives that share exactly one non-anchor modality with the positive: cells on the vertical line $\langle I_i, T_i, A_p \rangle$ share the same text but different audio ($p \neq i$), and cells on the horizontal line $\langle I_i, T_q, A_i \rangle$ share the same audio but different text ($q \neq i$). The remaining white cells $\langle I_i, T_q, A_p \rangle$ with $p \neq i$ and $q \neq i$ are *fully unmatched* negatives, sharing neither non-anchor with the positive. Across the four panels ($I=1, \dots, 4$), the structure repeats for each anchor image; in practice, anchors are cycled over all modalities (image, text, tabular), and the corresponding per-anchor losses are averaged.

Linear-time $O(B)$ construction. Algorithm 1 gives the linear-time construction of the partially unmatched and fully unmatched negative sets for a fixed anchor a within one mini-batch.

Algorithm 1 first samples an independent random permutation π_m of the batch indices for each non-anchor modality $m \neq a$ (lines 1–3), which specifies how we shuffle

along each non-anchor axis while keeping the anchor axis fixed. Using these permutations, the “easy” set $\mathcal{N}_{\text{part}}^{(a)}$ is constructed in line 4 by, for each anchor index i , choosing a single non-anchor modality m' and replacing only that axis by its permuted index $\pi_{m'}(i)$, while all other non-anchor modalities remain aligned with i . This produces partially unmatched tuples that differ from the positive in exactly one non-anchor modality. The “hard” set $\mathcal{N}_{\text{full}}^{(a)}$ is then constructed in line 5 by applying the permutation π_m to every non-anchor modality for each i , yielding fully unmatched tuples that share only the anchor modality. Allowing different non-anchor axes to map to the same index ($j_u = j_v \neq i$) simplifies the implementation without changing the semantics of “fully unmatched”. Line 6 returns both sets $(\mathcal{N}_{\text{part}}^{(a)}, \mathcal{N}_{\text{full}}^{(a)})$, which are then scored with the TC/MIP objective and combined via the curriculum described in Sec. 4.5.

In the rare case that a permutation accidentally recreates the positive tuple $(i, \dots, i)$ in the negative set, we simply resample that row; this affects only a negligible fraction of entries and keeps the overall $O(B)$ construction unchanged.

4.5. Curriculum Scheduling

Given the level-wise losses $\{\mathcal{L}_\ell\}_{\ell=1}^{M-1}$ from Sec. 4.4, we combine them with a smooth, time-varying schedule that mixes level-wise contributions:

$$\mathcal{L}(t) = \sum_{\ell=1}^{M-1} w_\ell(t) \mathcal{L}_\ell, \quad \sum_{\ell=1}^{M-1} w_\ell(t) = 1, \quad w_\ell(t) \geq 0.$$

We instantiate the weights $w_\ell(t)$ via *differences of sigmoids*. Let transition midpoints $t_1 < \dots < t_{M-2}$ and (optionally) slopes $k_1, \dots, k_{M-2}$, define $S_j(t) = \sigma(k_j(t - t_j))$, and set

$$\begin{aligned} w_1(t) &= 1 - S_1(t), & w_{M-1}(t) &= S_{M-2}(t), \\ w_\ell(t) &= S_{\ell-1}(t) - S_\ell(t) \quad \text{for } \ell = 2, \dots, M-2. \end{aligned}$$

This yields a nonnegative partition of unity that *progressively shifts* mass from easier, higher-overlap negatives to harder, lower-overlap negatives as training proceeds.

For the trimodal case ($M=3$), this reduces to the binary gate

$$\begin{aligned} L^{(a)}(t) &= (1 - w(t)) L_{\text{easy}}^{(a)} + w(t) L_{\text{hard}}^{(a)}, \\ w(t) &= \sigma(k(t - t_m)). \end{aligned}$$

In evaluation, we use the same $w(t)$ when computing the *validation loss* for epoch-to-epoch comparability; retrieval metrics are computed from frozen encoders and are unaffected by the schedule.

Visualizing the schedule. Figure 3 illustrates the generalized schedule for $M=5$, yielding $M-1=4$ negative difficulty levels. Each curve $w_\ell(t)$ corresponds to one negative level: w_1 starts near 1 and dominates early training (high-overlap, easiest negatives), and the weight gradually flows

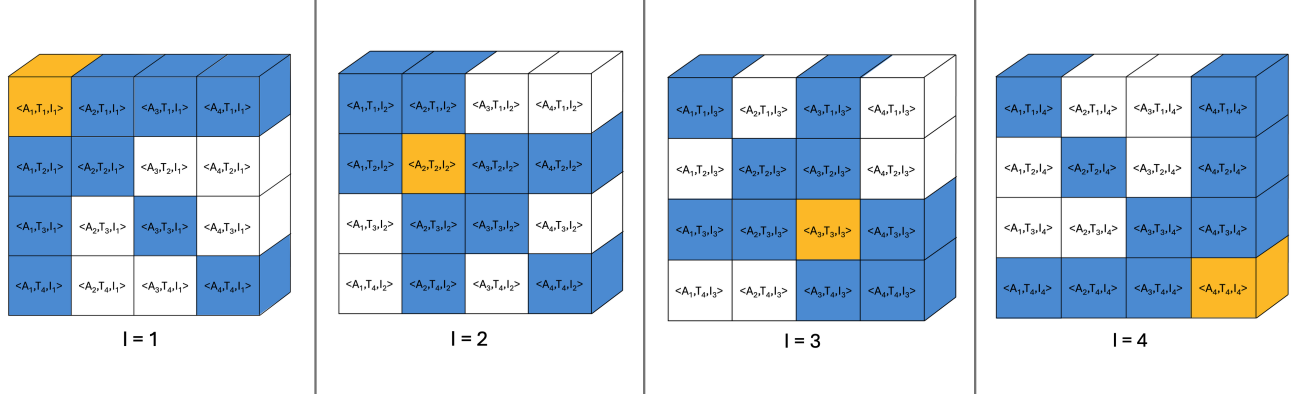


Figure 2. **Zoom-in of Easy2Hard (anchor slice $I = i$).** Let I = image, T = text, and A = audio. (a) positive $\langle I_i, T_i, A_i \rangle$ (yellow); (b) partially unmatched $\langle I_i, T_j, A_i \rangle$ (blue; shuffle T only); (c) partially unmatched $\langle I_i, T_i, A_j \rangle$ (blue; shuffle A only); (d) fully unmatched $\langle I_i, T_k, A_j \rangle$ (white; shuffle both).

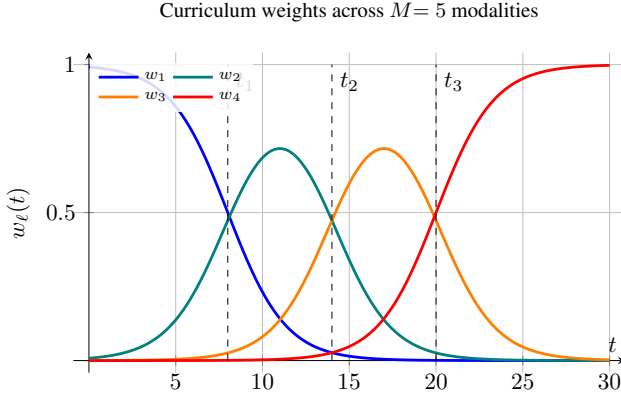


Figure 3. **Multi-stage curriculum via differences of sigmoids.** Illustrative scheduled weights $w_\ell(t)$ for the $M=5$ case. As t increases, the schedule progressively shifts mass across the ordered components.

to higher- ℓ levels as t increases. The vertical dashed lines mark the transition midpoints $t_1, \dots, t_{M-2}$ where the corresponding sigmoids $S_j(t)$ change most rapidly. For the trimodal case used in our experiments ($M=3$), only the *partially unmatched* and *fully unmatched* levels are non-empty, and the curves collapse to two branches corresponding to the easy and hard negatives.

Why Easy→Hard? Partially unmatched negatives, by sharing one non-anchor with the positive, can be viewed as yielding lower-variance gradients, which may facilitate initial alignment. Fully unmatched negatives, being more confounding, are better suited to enhance discrimination in later stages. The model and objective are held consistent; only the relative weights between $\mathcal{N}_{\text{part}}^{(a)}$ and $\mathcal{N}_{\text{full}}^{(a)}$ are varied over the training duration.

Gradient view. Without loss of generality, we use the trimodal case as an example. Let $\ell_e^{(a)}(\theta) = L_{\text{easy}}^{(a)}$ and $\ell_h^{(a)}(\theta) = L_{\text{hard}}^{(a)}$. The overall gradient is:

$$\nabla_{\theta} L(t, \theta) = \frac{1}{M} \sum_a \left[(1 - w(t)) \nabla_{\theta} \ell_e^{(a)} + w(t) \nabla_{\theta} \ell_h^{(a)} \right], \quad (7)$$

$$\partial_t L(t, \theta)|_{\theta} = w'(t) \frac{1}{M} \sum_a (\ell_h^{(a)} - \ell_e^{(a)}), \quad (8)$$

$$w'(t) = k w(t) (1 - w(t)) > 0. \quad (9)$$

Since $\ell_h \geq \ell_e$ is typical in early/mid training, the hard branch’s contribution grows smoothly and monotonically. Furthermore, $\max_t w'(t) = k/4$, so the gate is Lipschitz in t , providing controlled stability.

Complexity. By design, Easy2Hard matches the dominant cost of standard in-batch contrastive learning. Here, “linear” $O(B)$ mentioned before qualifies candidate construction only: for each anchor we draw one permutation per non-anchor axis (overall $O((M-1)B)$) and build a $B \times B$ score matrix. Computing logits and the row-wise softmax/CE over this matrix is $O(B^2)$ per anchor (the M -way product only changes constant factors). We process anchors sequentially and reuse buffers, so the per-iteration time is $O(MB^2)$ while peak memory remains $O(B^2)$. Encoders, batch composition, and inference-time scoring remain unchanged.

5. Evaluation

Our evaluation has four goals: 1) evaluate Easy2Hard on zero-shot retrieval and compare **Acc@1** against SOTA baselines; 2) quantify the benefit of splitting negatives by cross-modal overlap (partial vs. full) relative to treating all in-

batch negatives uniformly; 3) study how the schedule parameters (slope k , midpoint t_m) affect performance, and contrast the sigmoid schedule with a linear alternative; and 4) assess both transfer across trimodal datasets (EH-MIMIC, Symile-MIMIC, Channel, MM-IMDb) and scalability beyond three modalities on HoloAssist.

5.1. Datasets

EH-MIMIC (Img, Txt, Tab). We construct a trimodal clinical dataset by aligning frontal CXR images and radiology reports from MIMIC-CXR¹ with admissions and laboratory measurements from MIMIC-IV². Each sample is indexed at the admission level and represented with a 50-d lab value vector plus a 50-d missingness mask. We use a subject-disjoint 80/10/10 train/val/test split.

Symile-MIMIC (Img, ECG, Tab). A public trimodal clinical dataset with 11,622 admissions and patient-disjoint train/val/test splits on PhysioNet³; we use the official splits as released.

Channel (Img, Txt, Tab). A multimodal version of the UCI Online News Popularity dataset⁴, where the tabular article statistics are paired with the corresponding article text and images retrieved from the provided URLs.

MM-IMDb (Img, Txt, Tab). A movie dataset⁵ with poster images, plot summaries, and metadata; we use the poster and plot as image/text modalities and derive the tabular modality from year, rating, and number of votes.

HoloAssist (5-modal). To evaluate scalability beyond three modalities, we use five synchronized streams from HoloAssist⁶: eye gaze, head pose, and three IMU signals (accelerometer, gyroscope, magnetometer). We adopt the official split lists and evaluate on aligned temporal windows.

Additional preprocessing, alignment, and split details for all datasets are provided in the supplementary material.

5.2. Encoders

We use modality-specific encoders to map all modalities into a shared embedding space. Our encoder choices follow two principles: they should match the structure of each modality, and they should remain fixed across compared

objectives within a dataset so that performance differences are attributable to the training loss rather than to backbone changes. Modality-specific linear projections then map each encoder representation into the common embedding space.

For the image–text–tabular datasets (EH-MIMIC, Channel, and MM-IMDb), we use a ResNet-based image encoder for image inputs, a Transformer-based text encoder for text inputs, and an MLP for tabular features. These choices follow standard practice in multimodal representation learning and are kept fixed across methods for fair comparison. For Symile-MIMIC, following Symile [16], the three modalities are chest X-rays, electrocardiograms (ECG), and blood laboratory tests. Accordingly, we use a ResNet-50 for CXR, a ResNet-18 for 10-second 12-lead ECG signals, and a three-layer neural network for labs. In all cases, modality-specific linear projections map encoder outputs into a shared embedding space. These encoder families are chosen once based on modality type and used consistently across all baselines and our approach, providing a consistent and fair comparison across methods while preserving sufficient capacity for each modality.

For HoloAssist, we use modality-specific MLP encoders for eye gaze, head pose, and the three IMU streams, followed by projection into the same shared space. We choose MLP encoders in this setting because all five modalities are low-dimensional synchronized sensor streams rather than high-resolution image or text inputs, making lightweight per-modality encoders a natural and fair choice.

Full per-dataset architecture and preprocessing details are provided in the supplementary material.

5.3. Training Settings

Hyper-parameters. For each dataset, all compared methods share the same search space over learning rate, weight decay, and logit-scale initialization. Easy2Hard additionally tunes only its curriculum-specific midpoint t_m and slope k . Each method is then evaluated using its own best validation-selected configuration, ensuring a fair comparison under matched encoders, optimizers, schedulers, and shared tuning ranges.

Baselines. We compare against (i) *CLIP-Pairwise*: a pairwise dual-encoder trained on modality pairs and evaluated by summing similarities; and (ii) *Symile*: a trimodal baseline that uses MIP scoring consistent with TC style training. All baselines share the same encoders, patient-centric splits, optimizers, and schedulers.

5.4. Evaluation Protocol

We evaluate representation quality via *zero-shot retrieval*: given a query, the model must rank the matching target among a fixed 10-way candidate pool drawn from the held-out *test* split, without task-specific heads or fine-tuning.

¹<https://physionet.org/content/mimic-cxr/2.1.0/>

²<https://physionet.org/content/mimiciv/3.1/>

³<https://physionet.org/content/symile-mimic/1.0.0/>

⁴<https://archive.ics.uci.edu/dataset/332/online+news+popularity>

⁵<https://github.com/johnarevalo/gmu-mmimdb>

⁶<https://holoassist.github.io/>

On the trimodal datasets, we use a two-to-one retrieval setup in which two modalities form the query and the remaining modality provides the retrieval target. Concretely, for EH-MIMIC, Channel, and MM-IMDb we evaluate Text $\otimes$ Table $\rightarrow$ Image, whereas for Symile-MIMIC we evaluate ECG $\otimes$ Table $\rightarrow$ Image. On HoloAssist, we use a 10-way retrieval protocol with the eye stream as anchor. Ties are broken uniformly at random.

For MIP-based trimodal methods (Symile and Easy2Hard), let $(q^{(1)}, q^{(2)})$ denote the two query modalities and let c denote a candidate from the target modality. The retrieval score is

$$s_{\text{MIP}}(q^{(1)}, q^{(2)}, c) = \tau \langle q^{(1)} \odot q^{(2)}, c \rangle, \\ \tau = \exp(\text{logit_scale}) > 0.$$

For the pairwise baseline (CLIP-style), we use

$$s_{\text{pair}}(q^{(1)}, q^{(2)}, c) = \tau (\langle q^{(1)}, c \rangle + \langle q^{(2)}, c \rangle).$$

We report Acc@1 on candidate sets of size 10, where retrieval is counted as correct if the matched target is ranked first. We estimate uncertainty with nonparametric bootstrap over queries (10 resamples), reporting the standard error (SE) and the 95% percentile confidence interval (CI). Note that the curriculum weight $w(t)$ affects only training and validation losses; retrieval is performed using the frozen encoders and is therefore independent of $w(t)$.

5.5. Main Results

We evaluated our method on the zero-shot retrieval task and compared it with baselines including Symile and CLIP-Pairwise. The numerical results are reported in Table 1, and visualized as a bar chart in Figure 4.

Across all four datasets, our proposed Easy2Hard training method consistently outperforms the other two baselines. In absolute **Acc@1**, Easy2Hard improves over Symile by 0.017 on MM-IMDb (0.421 vs. 0.404), 0.006 on Channel (0.573 vs. 0.567), 0.028 on Symile-MIMIC (0.462 vs. 0.434), and 0.026 on EH-MIMIC (0.503 vs. 0.477). The gains over CLIP-Pairwise are even larger, up to 0.033 on MM-IMDb (0.421 vs. 0.388), 0.009 on Channel (0.573 vs. 0.564), 0.067 on Symile-MIMIC (0.462 vs. 0.395), and 0.047 on EH-MIMIC (0.503 vs. 0.456), all well above the corresponding standard errors. Overall, the performance trend among the three training strategies follows the order Easy2Hard > Symile > CLIP-Pairwise. Notably, both Symile and Easy2Hard depart from conventional CLIP-like training, while Easy2Hard further combines fine-grained negative partitioning with smooth curriculum scheduling. This combination is consistent with its stronger empirical performance.

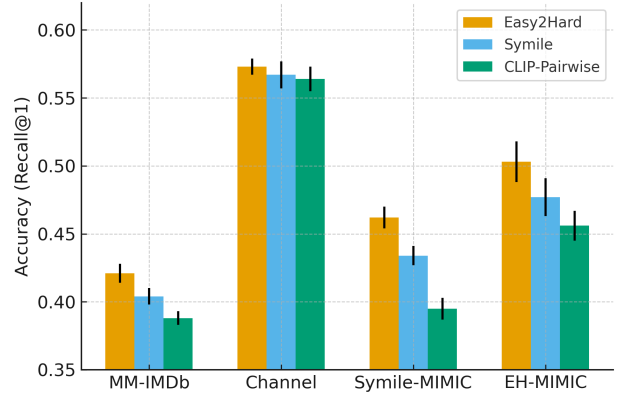


Figure 4. Zero-shot two-to-one retrieval accuracy (Acc@1) across four trimodal datasets. Easy2Hard consistently outperforms Symile and CLIP-Pairwise, with especially clear margins on the two clinical datasets.

5.6. Beyond Trimodal: 5-modal Feasibility on HoloAssist

To test scalability beyond three modalities, we conduct a 5-modal feasibility study on HoloAssist with **eye gaze** (anchor), **head pose**, and three **IMU** streams (accelerometer/gyroscope/magnetometer). We evaluate a 10-way retrieval setting (1 positive + 9 negatives) and report Acc@1. We follow the official split lists and strictly align modalities by timestamps. For comparison, we include an ImageBind-like pivot baseline that uses the eye stream as the pivot and optimizes pairwise contrastive alignment to each remaining modality. As shown in Table 2, Easy2Hard improves over pairwise and other multimodal baselines, suggesting that our framework extends naturally to $M > 3$ modalities. Compared with the ImageBind-like pivot baseline, Easy2Hard achieves substantially higher Acc@1 on HoloAssist. This trend is consistent with the hypothesis that, as the number of modalities increases, explicitly structuring negative tuples beyond pivot-based pairwise alignment becomes increasingly beneficial.

5.7. Hyperparameter Study

Slope. To examine sensitivity to the sigmoid slope, we fix the midpoint at $t_m=15$ and vary k . Table 3 reports zero-shot retrieval accuracy on Channel and EH-MIMIC. On both datasets, accuracy improves when moving from very small slopes to moderate values and then degrades again as k becomes too large. Channel peaks at $k=0.3$, whereas EH-MIMIC performs best around $k=0.5$ in the local slope sweep with fixed $t_m=15$. The drop at larger k is more pronounced on EH-MIMIC, indicating a higher sensitivity to overly aggressive transitions. In practice, these trends suggest tuning k first within a small set such as $\{0.3, 0.5, 0.7\}$.

Method	MM-IMDb	Channel	Symile-MIMIC	EH-MIMIC
Easy2Hard	0.421 ± 0.007 [0.410, 0.428]	0.573 ± 0.006 [0.567, 0.585]	0.462 ± 0.008 [0.447, 0.476]	0.503 ± 0.015 [0.480, 0.529]
Symile	0.404 ± 0.006 [0.398, 0.412]	0.567 ± 0.010 [0.548, 0.581]	0.434 ± 0.007 [0.420, 0.448]	0.477 ± 0.014 [0.457, 0.495]
CLIP-Pairwise	0.388 ± 0.005 [0.377, 0.392]	0.564 ± 0.009 [0.553, 0.577]	0.395 ± 0.008 [0.375, 0.415]	0.456 ± 0.011 [0.445, 0.479]

Table 1. Retrieval accuracy (Acc ± SE [95% CI]) across multimodal datasets for different training methods. Results use 10-way candidate pools and nonparametric bootstrap over queries (10 resamples, 95% CIs). EH-MIMIC uses the validation-selected configuration from the full tuning sweep ($k=0.9$, $t_m=17$), while Tables 3–4 show local one-factor sensitivity analyses.

Method	(Acc ± SE [95% CI])
Pairwise-CLIP	0.295 ± 0.002 [0.292, 0.297]
ImageBind-like	0.439 ± 0.001 [0.438, 0.441]
Symile	0.833 ± 0.001 [0.832, 0.836]
Easy2Hard	0.882 ± 0.001 [0.881, 0.884]

Table 2. Feasibility results on HoloAssist with $M=5$ modalities. We report Acc@1 under a 10-way retrieval protocol (1 positive + 9 negatives) with *eyes* as anchor.

	Channel	EH-MIMIC
Slope	(Acc ± SE [95% CI])	(Acc ± SE [95% CI])
0.1	0.524 ± 0.010 [0.518, 0.542]	0.458 ± 0.006 [0.453, 0.469]
0.3	0.576 ± 0.013 [0.558, 0.593]	0.478 ± 0.012 [0.462, 0.499]
0.5	0.557 ± 0.011 [0.545, 0.575]	0.488 ± 0.012 [0.473, 0.504]
0.7	0.553 ± 0.008 [0.540, 0.565]	0.446 ± 0.009 [0.436, 0.462]

Table 3. Zero-shot retrieval with varying sigmoid slopes at fixed $t_m=15$ on Channel and EH-MIMIC (10 bootstrap resamples; 95% CIs).

while keeping t_m fixed at a mid-epoch value.

Midpoint. Similarly, we fix the slope at $k=0.3$ and vary the midpoint t_m . Results on Channel and EH-MIMIC are shown in Table 4. At fixed $k=0.3$, midpoint trends are weaker and dataset-specific: Channel is relatively flat with a mild peak near $t_m \approx 15$, whereas EH-MIMIC slightly prefers earlier midpoints ($t_m \approx 11$ –13). Differences across t_m are small and CIs often overlap, suggesting that t_m has a smaller effect than the slope k ; a light sweep over $t_m \in \{11, 13, 15, 17, 19\}$ is sufficient in practice.

5.8. Ablation

We treat the linear scheduler as an ablation of the proposed smooth curriculum because it removes the midpoint and slope parameters that control the easy-to-hard transition. As shown in Table 5, the sigmoid variant performs substantially better. This comparison suggests that gains are better explained by a well-shaped transition over the structured partial/full negative split than by curriculum scheduling in an arbitrary form.

	Channel	EH-MIMIC
Midpoint	(Acc ± SE [95% CI])	(Acc ± SE [95% CI])
11	0.562 ± 0.008 [0.553, 0.575]	0.492 ± 0.013 [0.474, 0.511]
13	0.560 ± 0.010 [0.545, 0.578]	0.483 ± 0.009 [0.465, 0.492]
15	0.576 ± 0.013 [0.558, 0.593]	0.467 ± 0.013 [0.440, 0.475]
17	0.546 ± 0.008 [0.535, 0.558]	0.468 ± 0.013 [0.448, 0.490]
19	0.562 ± 0.009 [0.554, 0.582]	0.470 ± 0.014 [0.457, 0.497]

Table 4. Zero-shot retrieval with varying midpoints at fixed $k=0.3$ on Channel and EH-MIMIC (10 bootstrap resamples; 95% CIs). Numbers come from independent sweeps; minor differences may arise from randomness and bootstrap variation.

	Channel	EH-MIMIC
Scheduler	(Acc ± SE [95% CI])	(Acc ± SE [95% CI])
Linear	0.494 ± 0.009 [0.480, 0.502]	0.441 ± 0.011 [0.423, 0.456]
Sigmoid	0.573 ± 0.006 [0.567, 0.585]	0.503 ± 0.015 [0.480, 0.529]

Table 5. Linear vs. sigmoid for Easy2Hard. Channel: $k=0.3$, $t_m=15$; EH-MIMIC: $k=0.9$, $t_m=17$. (10-way; bootstrap $B=10$).

6. Conclusion

We introduced *Easy2Hard*, a TC/MIP-based multimodal contrastive learning mechanism that weights structured in-batch negatives over training time. Negatives are split by their overlap with the positive into *partially unmatched* and *fully unmatched*, and their losses are mixed by a smooth sigmoid gate without changing encoders or batch construction. The framework extends naturally beyond trimodal settings; we include a 5-modal feasibility study on HoloAssist. Under identical encoders, Easy2Hard consistently improves zero-shot retrieval on MM-IMDb, Channel, two MIMIC variants, and HoloAssist, while ablations show that the sigmoid scheduler outperforms linear scheduling and is more sensitive to slope than midpoint.

Acknowledgements

This research was supported by the National Science Foundation (NSF) under Grant No. OAC 2451611. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NSF.

References

- [1] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. Mutual information neural estimation. In *International conference on machine learning*, pages 531–540. PMLR, 2018. 2
- [2] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, page 41–48, New York, NY, USA, 2009. Association for Computing Machinery. 2
- [3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PmLR, 2020. 1
- [4] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2818–2829, 2023. 2
- [5] Hyeonju Choi, Apoorva Beedu, and Irfan Essa. Multimodal contrastive learning with hard negative sampling for human activity recognition. *arXiv preprint arXiv:2309.01262*, 2023. 2
- [6] Fartash Faghri, David J. Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: Improving visual-semantic embeddings with hard negatives, 2018. 1, 2
- [7] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15180–15190, 2023. 1, 2
- [8] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021. 1, 2
- [9] Xiaopeng Jiang, Shuai Zhao, Guy Jacobson, Rittwik Jana, Wen-Ling Hsu, Manoop Talasila, Syed Anwar Aftab, Yi Chen, and Cristian Borcea. Federated meta-location learning for fine-grained location prediction. In *2021 IEEE International Conference on Big Data (Big Data)*, pages 446–456. IEEE, 2021. 1
- [10] Xiaopeng Jiang, Han Hu, Thinh On, Phung Lai, Vijaya Datta Mayyuri, An Chen, Devu M Shila, Adriaan Larmuseau, Ruoming Jin, Cristian Borcea, et al. Flsys: Toward an open ecosystem for federated learning mobile apps. *IEEE Transactions on Mobile Computing*, 23(1):501–519, 2022. 1
- [11] M Kumar, Benjamin Packer, and Daphne Koller. Self-paced learning for latent variable models. *Advances in neural information processing systems*, 23, 2010. 2
- [12] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021. 2
- [13] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 1, 2
- [14] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. 1, 2
- [15] Yue Ruan, Han-Hung Lee, Yiming Zhang, Ke Zhang, and Angel X Chang. Tricolo: Trimodal contrastive loss for text to shape retrieval. *arXiv preprint arXiv:2201.07366*, 2022. 1
- [16] Adriel Saporta, Aahlad Manas Puli, Mark Goldstein, and Rajesh Ranganath. Contrasting with symile: Simple model-agnostic representation learning for unlimited modalities. *Advances in Neural Information Processing Systems*, 37: 56919–56957, 2024. 2, 3, 6
- [17] Pritam Sen, Xiaopeng Jiang, Qiong Wu, Manoop Talasila, Wen-Ling Hsu, and Cristian Borcea. On-device indoor place prediction using wifi-rtt and inertial sensors. *Pervasive and Mobile Computing*, page 102118, 2025. 1
- [18] Petru Soviany, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. Curriculum learning: A survey. *International Journal of Computer Vision*, 130(6):1526–1565, 2022. 2
- [19] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023. 2
- [20] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *European conference on computer vision*, pages 776–794. Springer, 2020. 2
- [21] Ye Wang, Bowei Jiang, Changqing Zou, and Rui Ma. Mxm-clr: A unified framework for contrastive learning of multifold cross-modal representations. *arXiv preprint arXiv:2303.10839*, 2023. 2
- [22] Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. Filip: Fine-grained interactive language-image pre-training. *arXiv preprint arXiv:2111.07783*, 2021. 2
- [23] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021. 2
- [24] Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18123–18133, 2022. 1, 2
- [25] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023. 2

Easy2Hard: From Partially to Fully Unmatched Modalities as Negative Samples in Contrastive Learning

Supplementary Material

7. Additional Experiment Details

7.1. Reproducibility Settings

All experiments on the MIMIC-series datasets and HoloAssist 5-modal dataset were performed on a high-performance computing cluster featuring NVIDIA L40S GPUs. Experiments on the Channel and MM-IMDb datasets were carried out on a separate workstation equipped with NVIDIA Quadro RTX 6000 GPUs.

7.2. Datasets and Splits

Symile-MIMIC. Symile-MIMIC is a clinical tri-modal dataset consisting of chest X-rays (CXR), electrocardiograms (ECG), and blood laboratory tests (labs), built from MIMIC-IV v2.2 (admissions and labs), MIMIC-IV-ECG v1.0 (ECGs), and MIMIC-CXR-JPG v2.0.0 (CXRs). Following Symile, each sample is constructed at the admission level: for a given admission, ECGs and labs are collected within the first 24 hours after admission, while CXRs are collected within a 24–72 hour window after admission; the earliest CXR, ECG, and labs in their respective windows are selected to form an aligned triplet. CXRs are restricted to PA/AP views and preprocessed by resizing the shorter side to 320 pixels, applying random square crops for training and center crops for validation/test, and normalizing with ImageNet mean and standard deviation. ECGs are 10-second, 12-lead recordings; samples containing NaNs or all-zero signals are removed, and ECG signals are normalized to $[-1, 1]$. Labs are represented using the 50 most common lab tests: values are percentile-normalized using the empirical CDF computed on the training set, concatenated with a 50-dimensional missingness indicator to form a 100-dimensional vector, and missing values are imputed with the corresponding training-set mean percentile. The dataset contains 11,622 admissions and follows the official patient-disjoint split lists released with Symile-MIMIC, with 95% of patients in the train/validation development split and 5% in the test split.

For evaluation, Symile considers zero-shot retrieval from (ECG, labs) to CXR using a 10-way candidate pool per query (one positive and nine randomly sampled negative CXRs from the test set).

EH-MIMIC. EH-MIMIC is a clinical multimodal dataset constructed from MIMIC-CXR-JPG v2.0.0 and MIMIC-IV v2.2, containing chest X-rays (CXR), radiology reports, and blood laboratory tests (labs). We first filter the

MIMIC-CXR metadata to retain studies with valid acquisition dates and standard frontal views (AP/PA). For each retained study, we load the corresponding radiology report text (one report per study) and the associated CXR image.

To associate each CXR study with a hospital admission, we align studies to admissions at the patient level using the study acquisition date. Concretely, for a given subject, we identify candidate admissions whose hospitalization interval overlaps with the CXR study date (with a ± 1 day tolerance), and assign the study to the admission whose admission time is closest to the study date; the resulting admission identifier (`hadm_id`) is used to retrieve laboratory measurements. CXR images are preprocessed by resizing the shorter side to 320 pixels, applying a center crop to 320×320 , and normalizing with ImageNet mean and standard deviation.

For labs, we consider numeric lab events (`valuenum`) from `labevents`. We select the 50 most frequent lab item IDs in the corpus and represent each admission as a 50-dimensional vector of raw lab values together with a corresponding 50-dimensional binary missingness mask, where each dimension indicates whether the lab was observed for that admission. Samples without available labs for the aligned `hadm_id` are discarded.

Finally, we perform patient-level data splitting by `subject_id` into train/validation/test sets with an 80%/10%/10% ratio to prevent patient leakage, ensuring that all studies from the same patient appear in only one split.

HoloAssist (5-modal). We use HoloAssist as a feasibility study to demonstrate applicability beyond 3 modalities. We use five modalities: eye gaze (`Eyes_sync.txt`), head pose (`Head_sync.txt`), and 3 IMU streams (`Accelerometer_sync.txt`, `Gyroscope_sync.txt`, `Magnetometer_sync.txt`).

For each recording, we align modalities by timestamps using the eye stream as the anchor: for every anchor timestamp, we select the nearest timestamp in each other modality, and keep the aligned time step only if the absolute time difference for *all* modalities is within a tolerance τ (we use $\tau = 0.02s$). Recordings with insufficient aligned points are discarded.

After strict alignment, we segment each recording into sliding windows of length 3 seconds with a stride of 1 second. Each window is kept only if it con-

tains at least 30 aligned time steps. For each modality within a window, we compute a fixed-length feature vector by concatenating per-dimension statistics: mean, standard deviation, minimum, and maximum over the aligned samples in that window. This yields one feature vector per modality per window, which we store as one JSONL entry. We follow the official HoloAssist split lists (`train-v1.2.txt`, `val-v1.2.txt`, `test-v1.2.txt`) to form train/validation/test sets, ensuring no recording appears in multiple splits.

7.3. Encoders and Input Preprocessing

Within each dataset, all compared methods share the same encoder families, initialization choices, and preprocessing so that only the loss construction and schedule differ. For the clinical trimodal datasets, modality-specific linear projections map encoder outputs to a shared space of dimension $d=8192$; for HoloAssist, we use $d=1024$.

Image Encoder. For image-based experiments (CXR), we use a ResNet-50 backbone. In the reported experiments, the image encoder is trained from scratch rather than initialized from pretrained weights. The final fully-connected layer is replaced with a linear projection to the shared embedding space of dimension d , followed by a LayerNorm on the projected embedding. CXR images are resized and cropped as described in the dataset preprocessing section and normalized using ImageNet mean and standard deviation.

Text Encoder. For text-based experiments, we use a Transformer-based text encoder following the Bio_ClinicalBERT architecture. In the main experiments, we initialize the text encoder from pretrained Bio_ClinicalBERT weights rather than training it from scratch. We tokenize the report text using the corresponding tokenizer and use the `[CLS]` representation from the last hidden layer as the sentence-level embedding (768-d). A linear layer projects this embedding to the shared d -dimensional space, followed by a LayerNorm. We adopt a pretrained text encoder because preliminary runs showed that training the text encoder from random initialization substantially degraded all compared objectives, shifting the difficulty from multimodal alignment to learning textual representations themselves.

ECG Encoder. For Symile-MIMIC experiments, we follow the encoder design used in Symile: a ResNet-18 backbone is used to encode 10-second 12-lead ECG signals, and its output is mapped to the shared d -dimensional embedding space via a linear projection. This encoder is trained from scratch in our experiments.

Tabular Encoder. For tabular features (labs), we use a three-layer MLP with GELU activations. This choice matches the low-dimensional structured nature of the laboratory inputs and avoids introducing method-specific architectural advantages. Specifically, the MLP maps the input lab vector to the shared d -dimensional embedding space with hidden sizes 256 and 1024, followed by a LayerNorm on the output embedding.

HoloAssist 5-modal Encoders. For the HoloAssist 5-modal feasibility study, each time window is represented by fixed-length vectors for five streams: `eyes`, `head`, `acc`, `gyro`, and `mag`. Because all five modalities are low-dimensional synchronized sensor streams rather than image or text inputs, we use lightweight per-modality MLP encoders in this setting.

Concretely, for each modality we compute per-dimension statistics (mean, standard deviation, minimum, and maximum) over aligned samples within a 3-second window, and concatenate them into a single vector; thus the input dimension for modality m is $4D_m$, where D_m is the number of columns in the corresponding `*_sync.txt` file after removing the timestamp column.

Each modality m is encoded by a lightweight two-layer MLP that maps the window feature vector to the shared embedding space

$$\mathbf{x}^{(m)} \in \mathbb{R}^{4D_m} \xrightarrow{\text{Linear}(4D_m \rightarrow 2d)} \text{ReLU} \xrightarrow{\text{Linear}(2d \rightarrow d)} \mathbf{z}^{(m)} \in \mathbb{R}^d.$$

We do *not* apply ℓ_2 normalization to encoder outputs in this setting, and we set the shared embedding dimension to $d=1024$.

Within the HoloAssist setting, all compared objectives (CLIP-Pairwise, Symile, Easy2Hard, and the ImageBind-like pivot baseline) share the same encoders and preprocessing; only the loss construction and/or schedule differs.

7.4. Training Recipe on HoloAssist

Optimizer and schedule. We train all HoloAssist models end-to-end with AdamW. Unless stated otherwise, we use a constant learning-rate schedule and train for 30 epochs.

Batching and compute. We use a training batch size of 2048 with gradient accumulation of 8 steps, and a validation batch size of 1024. All experiments are run on a single GPU.

Temperature scaling. We maintain a learnable scalar `logit_scale` for temperature scaling and enable exponential scaling (i.e., $\tau = \exp(\text{logit_scale})$) in training and evaluation.

Checkpointing and selection. During training, we save checkpoints corresponding to the best validation retrieval accuracy and the best validation loss (`best_val_acc.pt` and `best_val_loss.pt`). Unless otherwise noted, we select checkpoints by validation Acc@1 for downstream test evaluation.

7.5. Hyperparameter Tuning and Model Selection

For HoloAssist, model selection is performed on the validation split while keeping encoder families, preprocessing, and optimization settings matched across objectives within each dataset. The detailed hyperparameter grid for the 5-modal HoloAssist study is given below.

Search protocol (HoloAssist 5-modal). For the HoloAssist feasibility study, we perform a grid search on the validation split while keeping encoders, preprocessing, batch construction, and optimization settings fixed across objectives. We sweep learning rate, weight decay, and the initialization of `logit_scale` for all methods, and additionally sweep the Easy2Hard sigmoid schedule parameters (midpoint and slope).

Hyperparameter candidates. For all objectives (Symile, CLIP-Pairwise, and ImageBind-like), we use a constant schedule and search over $lr \in \{5 \times 10^{-5}, 10^{-4}, 5 \times 10^{-4}\}$, $weight_decay \in \{10^{-3}, 10^{-2}, 10^{-1}\}$, and $logit_scale_init \in \{-7, -5, -3, -1\}$ (with exponential temperature scaling enabled). For Easy2Hard (sigmoid schedule), we use the same grid for $(lr, weight_decay, logit_scale_init)$ and additionally search $midpoint \in \{9, 13, 15, 17, 21\}$ and $slope \in \{0.20, 0.25, 0.30, 0.35, 0.40\}$.

Evaluation and reporting. After selecting the best validation checkpoint for each configuration, we evaluate on the held-out test split under a fixed 10-way retrieval protocol (1 positive + 9 negatives per query), using the eye-gaze stream as the anchor modality. For uncertainty estimation, we report bootstrap confidence intervals with 10 bootstrap samples at $\alpha = 0.05$.

Within this grid, Easy2Hard attains the best test Acc@1 with $lr = 5 \times 10^{-5}$, $weight_decay = 10^{-3}$, $logit_scale_init = -7$, and sigmoid schedule parameters $(t_m, k) = (13, 0.2)$ (selected by validation Acc@1).

7.6. Results Beyond Three Modalities (M=5)

To evaluate scalability beyond trimodal settings, we conduct a 5-modal feasibility study on HoloAssist using `eyes`, `head`, `acc`, `gyro`, and `mag`. We use the predefined split lists released with HoloAssist (`train-v1.2.txt`,

Hyperparameter	Candidates
Learning rate	$\{5 \times 10^{-5}, 10^{-4}, 5 \times 10^{-4}\}$
Weight decay	$\{10^{-3}, 10^{-2}, 10^{-1}\}$
logit_scale init	$\{-7, -5, -3, -1\}$
Easy2Hard midpoint	$\{9, 13, 15, 17, 21\}$
Easy2Hard slope	$\{0.20, 0.25, 0.30, 0.35, 0.40\}$

Table 6. Hyperparameter grids for HoloAssist 5-modal tuning. All objectives share the same grid for optimizer and temperature parameters; Easy2Hard additionally tunes the sigmoid schedule.

`val-v1.2.txt`, `test-v1.2.txt`) to build train/val/test JSONL files.

Protocol. We evaluate 10-way zero-shot retrieval on the test split using the eye stream as the anchor modality. For each query window, we sample one matched positive and $K=9$ negatives uniformly from the test candidate pool (excluding the positive) and report Acc@1. Checkpoints are selected by validation Acc@1, and uncertainty is estimated with 10 bootstrap samples ($\alpha=0.05$).

Because HoloAssist contains many windows per recording, negatives drawn from the global test pool may be easier than within-recording confusions due to recording/session cues. Still, all methods follow the same protocol, so this experiment serves as a feasibility check of whether Easy2Hard remains beneficial when scaling to $M=5$ modalities.