UNDIMIX: HARD NEGATIVE SAMPLING STRATEGIES FOR CONTRASTIVE REPRESENTATION LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

One of the challenges in contrastive learning is the selection of appropriate *hard negative* examples, in the absence of label information. Random sampling or importance sampling methods based on feature similarity often lead to sub-optimal performance. In this work, we introduce UnDiMix, a hard negative sampling strategy that takes into account anchor similarity, model uncertainty and diversity. Experimental results on several benchmarks show that UnDiMix improves negative sample selection, and subsequently downstream performance when compared to state-of-the-art contrastive learning methods. Code is available at *anon. link*.

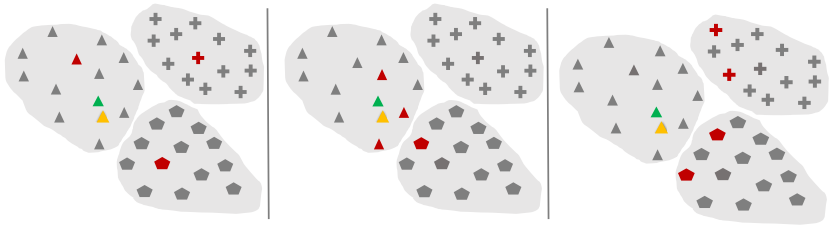
1 INTRODUCTION

Due to the potential of alleviating data annotation costs and the substantial effort requirements in encoding domain-specific knowledge, self-supervised representation learning methods have attracted research effort, with recent contrastive learning frameworks even surpassing the downstream performance of supervised learning (Chen et al., 2020a; He et al., 2020). Typically, contrastive methods aim at minimizing distances in feature space for similar example pairs (positive examples) and maximizing distances of dissimilar example pairs (negative examples) (Chopra et al., 2005). There has been a growing interest in contrastive learning research, in particular for obtaining better data representations (He et al., 2020; Robinson et al., 2021; Zhu et al., 2020; Chuang et al., 2020).

Several recent studies investigate the impact of negative sampling strategies, with recent work showing that increasing the number of negative samples results in learning better representations. However, a few of the hardest negative samples tend to have the same label as the anchor, hampering the learning process (He et al., 2020; Cai et al., 2020). Hence, selecting appropriate informative hard negative examples is a crucial step for the success of contrastive learning.

Various negative selection mechanisms have been proposed, that mainly aim to select discriminative negative examples based on the current learned feature representations, often used in conjunction with importance sampling or mixup interpolation (Li et al., 2021; Chen et al., 2020b; Shen et al., 2022; Robinson et al., 2021; Xiong et al., 2021). Most contrastive methods either uniformly sample negatives, *i.e.*, assuming that all negative examples are equally important (Chen et al., 2020a; He et al., 2020; Kalantidis et al., 2020), compute importance scores based on feature similarity (Robinson et al., 2021; Huynh et al., 2022) or uncertainty (Ma et al., 2021), or employ mixing of features in the feature/image space (Lee et al., 2021; Shen et al., 2022; Kim et al., 2020; Ge et al., 2021; Kalantidis et al., 2020; Zhu et al., 2021). As such, there is no clear notion of "informativeness" incorporated in the negative selection process. Particularly, prior works rarely consider the distance from the model decision boundary, *e.g.*, by incorporating model uncertainty, or diversity of the selected negative examples, *e.g.*, whether selected negatives indeed represent the diverse distribution of negatives.

Hard negative example selection in contrastive learning poses three challenges: (1) there is no label information available, hence there is a requirement for unsupervised strategies for instance selection, (2) an efficient sampling method should avoid false "hardest" negative samples, *i.e.*, samples that are most similar and originate from the same class as the anchor, and (3) an ideal set of negative examples should represent the whole population Robinson et al. (2021); Cai et al. (2020). A selection mechanism should ideally capture all three properties: anchor similarity, model confidence and diversity. Methods that only consider similarity with the anchor when sampling negative examples, *i.e.*, assuming that higher similarity aligns with higher importance, tend to select same-class negatives (Figure 2(b)), which can be detrimental to the representation learning (Cai



(a) Random (b) Anchor Similarity (c) UnDiMix (this work)

Figure 2: Illustration of negative samples (**red** shapes) for an anchor (**yellow** triangle) and its positive pair (**green** triangle), selected by three negative sampling techniques. Gray areas represent three clusters with different semantic labels. (a) Random sampling results in easy negatives being selected (1 **triangle**, 2 **plus**, 1 **pentagon**). (b) Methods that only consider anchor similarity may sample negatives that lie close to the anchor but also belong to the same semantic class (3 **triangles**, 1 **pentagon**). In contrast, (c) UnDiMix samples negatives that lie close to the decision boundary and are far from each other, *i.e.*, better representing the data population (2 **plus**, 2 **pentagons**).

et al., 2020). Besides, negative examples that lie closer to the decision boundary are naturally the hardest examples, encapsulating rich information about different class categories. On the other hand, diverse negative examples help to learn global representations of the data distribution. Consequently, we argue that considering diversity, along with uncertainty and anchor similarity, is helpful when selecting negative examples.

For demonstrating the importance of selecting *diverse* negative examples, we train a SimCLR Chen et al. (2020a) model on `CIFAR-100` with all default settings but imposing a modified sampling process. More specifically, we sample negative examples of class c; $0 \leq c \leq k-1$, where k is the hyper-parameter for controlling diversity of negative examples and is set to $k = \{10, 20, 30, 40, 50\}$. As k increases, *i.e.*, the diversity of selected negative examples increases, accuracy also increases (Figure 1). Hence, the diversity of the negative examples contributes to learning better representations.

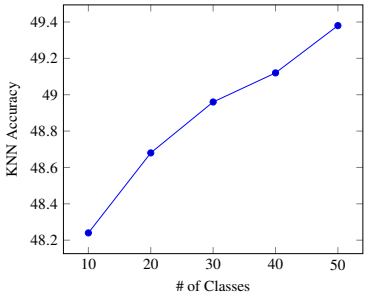


Figure 1: SimCLR top-5% KNN accuracy on `CIFAR-100`. Model trained for 100 epochs with negative examples sampled from $k = \{10, 20, 30, 40, 50\}$ classes.

To address the aforementioned limitations, this paper introduces **Un**certainty and **Di**versity **Mix**ing (**UnDiMix**) for contrastive training, a method that combines importance scores that capture model uncertainty, diversity, and anchor similarity. Specifically, as illustrated in Figure 2(c), UnDiMix utilizes uncertainty to penalize false hard negatives and pairwise distance among negatives to select diverse examples in a computationally inexpensive way. We verify our method on several visual, text and graph benchmark datasets and perform comparisons over strong contrastive baselines. Experimental and qualitative results demonstrate the effectiveness of UnDiMix.

Contributions: **(1)** We delve into an empirical analysis of the efficacy of hard negative sampling strategies and feature-based importance sampling methods, observing that incorporating diversity improves downstream performance. **(2)** Based on our observations, we introduce an efficient method to calculate the diversity score of negative examples based on pairwise similarity and propose UnDiMix, a flexible and efficient hard negative sampling method for contrastive learning that selects hard informative negatives based on anchor similarity, model confidence and diversity. **(3)** We verify the effectiveness of the proposed method and show that UnDiMix improves downstream task performance and negative selection quality on several benchmarks in 3 domains (image, text and graph data). Qualitative analysis shows that the proposed method encourages sampling a diverse set of negatives, resulting in better performance.

2 Related Work

Contrastive Learning: Recent work has largely contributed in developing contrastive self-supervised learning methods , such as SimCLR (Chen et al., 2020a), MoCo (Chen et al., 2020b),

Debiased (Chuang et al., 2020), BYOL (Grill et al., 2020), SimSiam (Chen & He, 2021), i-Mix (Lee et al., 2021), UnMix (Shen et al., 2022), FNC Huynh et al. (2022), *etc.* that have produced promising results in a variety of domains, from learning unsupervised cross-modal representations and video representations (Ma et al., 2021; Qian et al., 2021; Pan et al., 2021), to natural language processing (Aberdam et al., 2021) and graph learning (You et al., 2020), often achieving comparable results to supervised counterparts (Chen et al., 2020a; He et al., 2020). Research efforts can be largely divided into improvements over the contrastive loss calculation (Chuang et al., 2020; Xie et al., 2021; Thota & Leontidis, 2021; Oord et al., 2018; Zhu et al., 2020; Azabou et al., 2021; Ko et al., 2021; Chen et al., 2021) and improvements over the hard positive/negative selection mechanisms (Shen et al., 2020; Robinson et al., 2021; Kalantidis et al., 2020; Chen et al., 2020a). Our proposed UnDiMix falls under the latter category and improves over feature-based contrastive importance sampling methods. We compare UnDiMix with state-of-the-art hard negative sampling techniques, briefly described below.

Hard Negative Sampling: Selection strategies for mining high-quality negative instances in contrastive learning have attracted substantial research interest, resulting in a wide range of contrastive methods proposed, *e.g.*, i-Mix (Lee et al., 2021), HCL (Robinson et al., 2021), Mochi (Kalantidis et al., 2020), AdCo (Hu et al., 2021), *etc.* In particular, AdCo (Hu et al., 2021) maintains a separate global set for negative examples that is updated actively using the contrastive loss gradients w.r.t. each negative example. However, the set of negative examples remains the same for all the anchors. MMCL (Shah et al., 2021) formulated the contrastive loss function as an SVM objective and utilized the support vectors as hard negatives, resorting to approximations to solve a computationally expensive quadratic equation for each anchor. HCL (Robinson et al., 2021), Mochi (Kalantidis et al., 2020) and FNC (Huynh et al., 2022) relied on feature similarity w.r.t. the anchor while selecting negative examples and achieved improvements over MoCo-V2 and SimCLR. However, considering only feature similarity results in assigning more importance to the same-class negatives, *i.e.*, most likely false negatives which are detrimental to the representation learning process (Cai et al., 2020). Wu et al. (2020) used two hyperparameters to create a ring of negatives around the anchor. Motivated by Mixup (Zhang et al., 2018), a few methods create synthetic examples either by interpolating instances at an image/pixel or latent representation level (Kim et al., 2020; Shen et al., 2022; Zhu et al., 2021), or by interpolating virtual labels (Lee et al., 2021). Other methods use either texture-based and patch-based non-semantic augmentation techniques (Ge et al., 2021) or a asynchronously-updated approximate nearest neighbor index of corpus (Xiong et al., 2021).

On the other hand, Ma et al. (2021) selects negative examples with high model uncertainty. In active learning, Ash et al. (2020) utilized the gradients of the loss function w.r.t. the model's most confident prediction as an approximation of uncertainty. Ma et al. (2021) utilized this measure to actively sample uncertain negatives when composing a memory bank. Our approach differs in that we leverage the gradients of the last layer as a model-based uncertainty measure in order to assign more importance to the samples closer to the decision boundary while sampling, but also incorporate both anchor similarity and diversity to capture other equally useful properties.

Overall, to the best of our knowledge, none of the prior contrastive learning works jointly considers model confidence, anchor similarity and diversity, let alone analyze the importance of interpolation among such components. To this end, we propose UnDiMix, a simple and efficient method that benefits from all aforementioned components when computing importance weights for negative examples. Our experimental analysis shows that UnDiMix improves downstream performance and diversifies the negative example set.

3 METHOD

Problem Formulation: Given an unlabeled dataset $\mathcal{X}$, we wish to learn an encoding function $f: \mathcal{X} \rightarrow \mathbb{R}^d$ that maps a data point $x_i \in \mathcal{X}$ to a d-dimensional embedding space, such that embeddings of similar instances $(x_i, x_i^{'})$ lie closer to each other, and vice versa. For a random subset (batch) of N positive pairs $\mathcal{X}_N = \{(\bar{x}_i, \tilde{x}_i)\}_{i=1}^{N}$, where $\bar{x}_i, \tilde{x}_i$ are two augmented views of example x_i, the contrastive loss for learning the encoder f is defined as

$$\mathcal{L}_{x_i} = -\log \frac{\exp\left(s(\bar{x}_i, \tilde{x}_i)/\tau\right)}{\exp\left(s(\bar{x}_i, \tilde{x}_i)/\tau\right) + \sum\limits_{\tilde{x}_{j\neq i} \in \mathcal{X}_N} \exp\left(s(\bar{x}_i, \tilde{x}_j)/\tau\right)}, \tag{1}$$

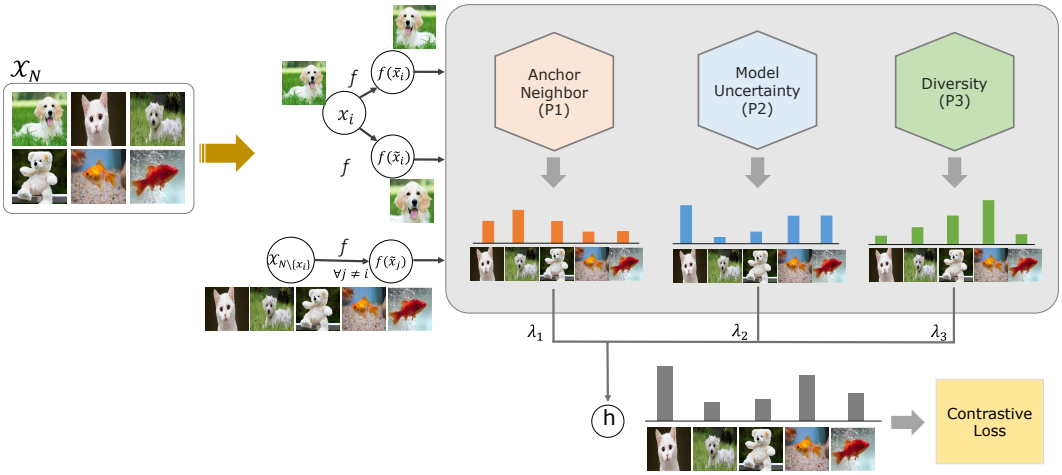


Figure 3: Overview of the UnDiMix hard negative sampling. Given an anchor x_i and a set of negative samples $\mathcal{X}_N \setminus x_i$, UnDiMix computes an importance score for each negative sample, by linearly interpolating between gradient-based uncertainty, anchor similarity and representativeness indicators, capturing desirable negative sample properties, *i.e.*, samples that are in close vicinity to the anchor (P1), lie close to the decision boundary (P2) and are represent a diverse sample population (P3).

where $s(x_i, x_j) = f(x_i)^\top f(x_j)/\|f(x_i)\|\|f(x_j)\|$ is the inner product of the normalized latent representations, and τ is a temperature scaling hyperparameter. Here, $\tilde{x}_i$ is referred as the positive sample for $\bar{x}_i$ and $(\bar{x}_j, \tilde{x}_j)_{j\neq i} \in \mathcal{X}_N$ are the remaining instances, that are considered negative samples.

The set of negative examples is typically selected by random sampling (Chen et al., 2020a;b). Recent works have individually proposed various "hard" negative mining or generation techniques, *e.g.*, based on perturbations in the input space (Lee et al., 2021; Shen et al., 2022), feature-based importance weights (Robinson et al., 2021) or uncertainty-based sampling (Ma et al., 2021). Yet, these methods consider only one selection indicator and hence achieve sub-optimal performance in learning contrastive representations. In this work, we propose a sampling technique, termed UnDiMix, that jointly considers both model-based uncertainty and diversity to select negative examples. Below, we describe how UnDiMix captures the necessary properties for mining informative negative samples.

UnDiMix Description: We wish to select high-quality informative hard negative examples that exhibit the following properties:

P1: Hard negative examples resemble the anchor example, *i.e.*, the feature representations of the hardest negative examples lie close to the anchor in the embedding space. We refer to P1 as ANCHOR VICINITY property.

P2: The selected negative example should also be close to decision boundary. P1 may sample false negatives. To alleviate this, examples that are similar to the anchor but lie further away from the decision boundary should have lower weights than examples that are close to the decision boundary and thus more informative. We refer to P2 as the DECISION BOUNDARY VICINITY property.

P3: Informative negative examples are also diverse. In other words, semantically similar but not identical diverse negative examples should be sufficient for contrastive training (Cai et al., 2020). We refer to P3 as the DIVERSITY property.

In summary, UnDiMix selects hard negative samples based on calculated importance scores. The higher the score is, the more informative the sample is assumed to be. The importance scores consist of three components:

(1) a *feature-based component* that leverages the feature space geometry via instance similarity to select informative negative samples, that satisfy **P1**,

(2) a *model-based component* that utilizes the loss gradients w.r.t. each negative sample as a measure of uncertainty and approximates **P2** by assigning more weight to the negative examples that lie closer to the decision boundary, and

(3) a *density-based component* that assigns more weight to negatives examples that are more distant on average from other negative examples in the batch, and satisfies **P3**.

To incorporate **P1** (ANCHOR VICINITY), we utilize instance similarity in the embedding space. Here, we use the inner product of the normalized vector representations as a similarity score for example x_j with respect to anchor x_i, *i.e.*, $s(\tilde{x}_j, \bar{x}_i) = f(\bar{x}_i)^\top f(\tilde{x}_j)/\|f(\bar{x}_i)\|\|f(\tilde{x}_j)\|$. This means that the more similar x_j is to anchor x_i, the higher the importance of x_j is (Robinson et al., 2021).

The lack of access to ground-truth label information makes it impossible to maintain **P2** (DECISION BOUNDARY VICINITY) completely. The challenge, therefore, lies in measuring the informativeness of negative samples without label information. Model uncertainty measures the degree of confidence of a model in its prediction *i.e.*, high model uncertainty corresponds to lower model confidence, and neural models typically assign higher uncertainty to examples closer to the decision boundary Liu et al. (2020). We use this property to assign higher importance to negative examples, such that negative examples that are closer to the anchor but far from the decision boundary will have lower importance than negatives lying closer to the decision boundary.

Inspired by the use of similar information-theoretic metrics in metric learning (Dutta et al., 2020), out-of-distribution detection (Mundt et al., 2019), and reinforcement learning (Zhao et al., 2019), we consider a gradient-based uncertainty metric. In particular, pseudo-labeling (as an implicit method for entropy minimization) and gradient-based uncertainty (where a smaller gradient norm corresponds to higher model confidence) are established in semi-supervised and active learning Lee et al. (2013); Ash et al. (2020). In addition, gradient-based uncertainty comes with theoretical justifications for our chosen type of pseudo-labels Ash et al. (2020). More formally, we first define a pseudo-label space induced by the data distribution. We denote the most confident prediction for a negative example $(\bar{x}_j, \tilde{x}_j)_{j\neq i} \in \mathcal{X}_N$ as its pseudo-label $\hat{y}_j$, and utilize the gradient of the last encoder layer w.r.t. this pseudo-label. Specifically, we calculate the pseudo-posterior of the negative example x_j via

$$p\left(y_j \,\middle|\, \tilde{x}_j, \mathcal{X}_N\right) = \frac{\exp\left(s(\bar{x}_i, \tilde{x}_j)\right)}{\sum_{\bar{x}_{i'\neq j} \in \mathcal{X}_N} \exp\left(s(\bar{x}_{i'}, \tilde{x}_j)\right)}, i \in [1, 2, \ldots, N] \tag{2}$$

where $s(x_i, x_j) = f(x_i)^\top f(x_j)/\|f(x_i)\|\|f(x_j)\|$ is the inner product of the normalized representations of x_i and x_j, respectively. Equation (2) calculates the posterior as the similarity of a negative x_j and all other examples $x_i \in \mathcal{X}_N$, considering them as individual anchors.

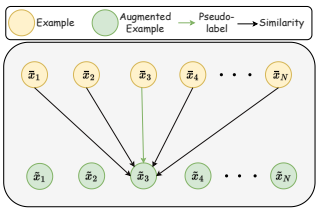


Figure 4: Pseudo space for augmented example pairs. The pseudo-labeling task is defined as matching an augmented example with its corresponding augmented view.

Due to the absence of class information, we design an auxiliary pseudo-labeling task of predicting the corresponding augmented example $\bar{x}_j$ given its pair $\tilde{x}_j$. We denote the most confident prediction as $\hat{y}_j = = \arg\max_k \left[p\left(y_j = \bar{x}_k \,\middle|\, \tilde{x}_j, \mathcal{X}_N\right)\right]_k$, where $[\cdot]_k$ corresponds to the index of example x_i in the batch and $1 \leq k \leq N$. That is, given the augmented view $\tilde{x}_i$ of example x_i, the goal is to locate the matching augmented example $\bar{x}_i$ from the set of augmented views $\{\bar{x}_k\}_{k=1}^N$ for all examples in the batch. Figure 4 pictorially illustrates the pseudo-labeling task. We calculate the gradient of the cross-entropy loss via

$$g_{x_j} = \frac{\partial}{\partial \theta_{last}} \ell_{CE}\left(p\left(y_j \,\middle|\, \tilde{x}_j, \mathcal{X}_N\right), \hat{y}_j\right)\Big|_{\theta=\theta_f}, \tag{3}$$

where ℓ_{CE} is the cross-entropy loss function and θ_{last} is the parameter vector of the last layer of encoder f. Intuitively, the gradient g_{x_j} measures the model change caused by the negative example x_j. The more uncertain the model is about its prediction for a particular sample, the higher the update of the model parameters. Similarly, we compute g_{x_i} for a specific anchor x_i. We adopt the concept of gradient similarity, Dhaliwal & Shintre (2018) to compute the influence of x_j on the loss w.r.t x_i. The uncertainty score of an example x_j with respect to anchor x_i is defined as $u(\tilde{x}_j, \bar{x}_i) = g_{x_i}^\top g_{x_j}$. This uncertainty metric assigns higher scores for the negative examples that are more influential for the corresponding anchor.

Even so, incorporating only uncertainty and anchor similarity does not allow for a negative sample set that is diverse. The addition of an appropriate diversity score aids in selecting informative hard diverse negatives, maintaining **P3** (DIVERSITY). Recent works utilize clustering of feature representations for selecting prototypical examples (Li et al., 2021). However, clustering after each

Algorithm 1 Pseudocode for UnDiMix

Input: Dataset $\mathcal{X}$, batch size N, encoder f
for batch $\mathcal{X}_N = \{(\bar{x}_i, \tilde{x}_i)\}_{i=1}^{N} \sim \mathcal{X}$ **do**
  `loss` := 0
  for $i, j = 1, \ldots, N$ and $j \neq i$ **do**
    # Acquire embeddings for positives and negatives, and calculate pairwise similarity scores
    $s(\tilde{x}_j, \bar{x}_i) = f(\bar{x}_i)^\top f(\tilde{x}_j) / \|f(\bar{x}_i)\| \|f(\tilde{x}_j)\|$
    $p\left(y_j \mid \tilde{x}_j, \mathcal{X}_N\right) = \frac{\exp\left(s(\bar{x}_i, \tilde{x}_j)\right)}{\sum_{\bar{x}_{i' \neq j} \in \mathcal{X}_N} \exp\left(s(\bar{x}_{i'}, \tilde{x}_j)\right)}$ # Calculate pseudo-labels using Eq. (2)
    $g_{x_j} = \nabla_{\theta_{last}} \ell_{CE}\left(p\left(y_j \mid \tilde{x}_j, \mathcal{X}_N\right), \hat{y}_j\right)\big|_{\theta=\theta_f}$ # Calculate gradients using Eq. (3)
    $u(\tilde{x}_j, \bar{x}_i) = g_{x_i}^\top g_{x_j}$ # Calculate uncertainty score
    $r(\tilde{x}_j, \bar{x}_i) = \frac{1}{N-2} \sum_{\substack{j'=1 \\ j' \notin \{i,j\}}}^{N} \left(1 - s(\tilde{x}_j, \tilde{x}_{j'})\right)$ # Calculate diversity score using Eq. (4)
    $w(\tilde{x}_j, \bar{x}_i) = \lambda_1 u(\tilde{x}_j, \bar{x}_i) + \lambda_2 s(\tilde{x}_j, \bar{x}_i) + \lambda_3 r(\tilde{x}_j, \bar{x}_i)$ # Compute importance ($\lambda_1, \lambda_2, \lambda_3$ learnable)
    # Compute contrastive loss with importance weights for negatives
    $\texttt{loss} \mathrel{+}= -\log \frac{\exp\left(s(\bar{x}_i, \tilde{x}_i)/\tau\right)}{\exp\left(s(\bar{x}_i, \tilde{x}_i)/\tau\right) + \sum_{\tilde{x}_{j \neq i} \in \mathcal{X}_N} w(\tilde{x}_j, \bar{x}_i) \exp\left(s(\bar{x}_i, \tilde{x}_j)/\tau\right)}$
  end for
  update the model to minimize the loss
end for

update is computationally challenging and requires hyperparameters (number of clusters). To simplify the calculation, we instead compute its average distance from all other negative examples in the embedding space. The diversity score of an example x_j given anchor x_i is

$$r(\tilde{x}_j, \bar{x}_i) = \frac{1}{N-2} \sum_{\substack{j'=1 \\ j' \notin \{i,j\}}}^{N} \left(1 - s\left(\tilde{x}_j, \tilde{x}_{j'}\right)\right). \tag{4}$$

Intuitively, the negative examples which are farther away from other negative examples will have higher scores and the negative examples which are closer to each other will have lower scores. In our experiments, we observe that the proposed diversity score performs well and encourages sampling a diverse set of negatives but adds little to no computational overhead. Finally, we define the importance score of a negative example as follows:

$$w(\tilde{x}_j, \bar{x}_i) = h\Big(u(\tilde{x}_j, \bar{x}_i), s(\tilde{x}_j, \bar{x}_i), r(\tilde{x}_j, \bar{x}_i)\Big). \tag{5}$$

Here h is an aggregation function, *e.g.*, linear interpolation or attention weights. In our experiments, we use the latter and model the weight of each component as a learned hyper-parameter. Moreover, we present ablation studies for a variation with fixed equal weights. Figure 3 presents an overview of UnDiMix and Algorithm 1 provides the pseudocode for the calculation of importance scores.

4 Experiments

We evaluate UnDiMix on several benchmarks from three (3) different domains (visual, text and graph), totaling fifteen (15) evaluation tasks (six text, three image and six graph benchmarks) and comparing against state-of-the-art contrastive learning methods.

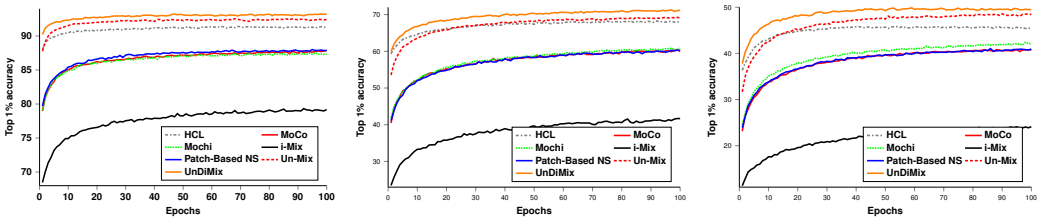
4.1 Image Representations

Baselines: The baseline set is the most representative w.r.t. hard negative selection or generation with competitive results over related works:

MoCo (Chen et al., 2020b), more specifically MoCo-V2, a general dictionary-based contrastive learning method, for which negative examples are randomly sampled and stored in the dictionary.

Table 1: Top 1% accuracy comparison over baselines. Mean and standard deviation reported over 10 trials. Green arrows indicate relative gains over the next best method.

Method	CIFAR-10	CIFAR-100	TINY-IMAGENET
SwAV (Caron et al., 2020)	76.90±0.02	43.60±0.01	29.00±0.10
i-Mix (Lee et al., 2021)	79.26±0.18	41.58 ±0.25	24.10±0.02
MoCo (Chen et al., 2020b)	87.88±0.18	59.96±0.20	40.76±0.40
Patch-Based NS (Ge et al., 2021)	87.86±0.06	60.24±0.11	40.91±0.06
Mochi (Kalantidis et al., 2020)	87.33±0.12	60.83±0.06	42.11±0.18
HCL (Robinson et al., 2021)	91.19±0.03	67.87±0.09	45.62±0.07
Un-Mix (Shen et al., 2022)	92.42±0.06	69.15±0.06	48.44±0.06
UnDiMix	**93.40±0.06** ↑0.98	**71.60±0.13** ↑2.45	**49.87±0.10** ↑1.43



(a) **CIFAR-10** (b) **CIFAR-100** (c) **Tiny-Imagenet**

Figure 5: Learning curves, UnDiMix (**orange** lines) surpasses baselines across all datasets.

Mochi (Kalantidis et al., 2020), built on top of MoCo, this method generates synthetic hard negatives by mixing negatives stored in the dictionary.

SwAV (Caron et al., 2020), an online contrastive learning algorithm that uses a swapped prediction mechanism for clustering assignments.

i-Mix (Lee et al., 2021), a regularization strategy that mixes data in both input and prediction level.

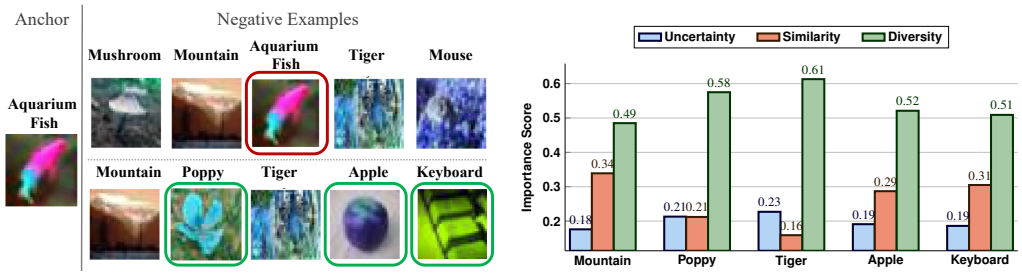
Patch-based NS (Ge et al., 2021), a negative sample generation technique built upon MoCo-V2 that generates negative examples from the anchor using patch-based techniques.

HCL (Robinson et al., 2021), a hard negative selection strategy that improves negative selection upon SimCLR (Chen et al., 2020a) by computing importance scores based on feature representations.

Un-Mix (Shen et al., 2022), a self-mixture strategy in the image space using Mixup (Zhang et al., 2018) and Cutmix (Yun et al., 2019).

Linear Evaluation: We follow prior contrastive learning works and train a linear classifier on frozen feature representations acquired from pre-trained contrastive models, and evaluate performance on the CIFAR-10, CIFAR-100, and TINY-IMAGENET datasets (Krizhevsky et al., 2009; Le & Yang, 2015). Figure 5 depicts the consistent improvement of UnDiMix over all other baseline methods. Table 1 presents the top-1% accuracy after fine-tuning the linear classifier for 100 epochs. Results are averaged over multiple trials, *i.e.*, we report the mean and standard deviation over 10 independent trials, and green arrows indicate relative gains over the next best method. UnDiMix outperforms the best baseline (Un-Mix) by 1.06% on CIFAR-10, 3.54% on CIFAR-100 and 2.95% on TINY-IMAGENET. As far as the rest of the baselines, UnDiMix obtains on average an improvement of 2.4% on CIFAR-10, 5.49% on CIFAR-100 and 9.3% on TINY-IMAGENET.

Qualitative Analysis: We compare the sampled negatives of both HCL (Robinson et al., 2021) and UnDiMix. Figure 6(a) depicts the five most important negative examples sampled by HCL (top row) and UnDiMix (bottom row) for the same anchor example of class "*Acquarium Fish*". As discussed earlier, UnDiMix samples more diverse negative examples and avoids false negatives. In contrast, HCL has sampled the anchor as negative (indicated with a red bounding box). UnDiMix components (uncertainty, feature similarity, and diversity) are aggregated with learned hyper-parameters as weights. Figure 6(b) depicts the contribution of each component in calculating the overall importxance score



(a) Qualitative evaluation on CIFAR-100 (b) UnDiMix component contributions

Figure 6: **(a)** Qualitative evaluation on CIFAR-100. Best viewed in color. HCL (Robinson et al., 2021) (top row) has sampled the anchor as a negative example (**red** bounding box). UnDiMix (bottom row) samples more diverse true negative examples (**green** bounding boxes) and avoids false negative examples. **(b)** Individual contribution of each of the three importance score components, for the top-5 important negative examples selected by our method (bottom row of left figure).

Table 2: Top-1% accuracy of different UnDiMix variations.

Components of w	**CIFAR-10**	**CIFAR-100**	**TINY-IMAGENET**
Feature similarity	92.7 ± 0.07	70.1 ± 0.10	47.7 ± 0.10
Uncertainty	92.9 ± 0.08	70.5 ± 0.15	48.6 ± 0.07
Diversity	93.0 ± 0.05	70.2 ± 0.12	49.0 ± 0.10
Feature Similarity + Uncertainty	93.1 ± 0.08	70.6 ± 0.14	49.5 ± 0.05
Feature Similarity + Diversity	93.1 ± 0.07	70.4 ± 0.02	49.5 ± 0.10
Uncertainty + Diversity	93.0 ± 0.06	70.4 ± 0.10	49.6 ± 0.08
All (UnDiMix)	**93.40±0.06**	**71.60±0.13**	**49.87±0.10**

of the top five important negative examples selected by UnDiMix. We can observe that the diversity component contributes the most. Additional qualitative examples can be found in Appendix G.

Importance score components: We perform an ablation analysis for each of the score components, *i.e.*, uncertainty (u), similarity (s), and diversity (r). We train different variations of UnDiMix, with one and two components at a time. Table 2 presents the top-1% linear evaluation accuracy on the CIFAR-10, CIFAR-100 and TINY-IMAGENET datasets, respectively. Note that including only feature similarity is essentially similar to HCL (Robinson et al., 2021). We notice that including uncertainty and diversity outperforms HCL, and variants that combine two components outperform the one component versions. Moreover, adding all components (UnDiMix) results in performance improvements across all datasets.

Aggregation Function: UnDiMix combines uncertainty, similarity and diversity in a linearly interpolated importance score, computed for each example via attention weights that are learned during training. Table 3 presents the accuracy of linear evaluation for UnDiMix and a variation with fixed equal weights. We observe that attention weights perform comparatively well.

4.2 Sentence Representations

We evaluate UnDiMix on learning sentence representations using the Quick-Thought (QT) vectors (Logeswaran & Lee, 2018), following the same experimental setting as Logeswaran & Lee (2018). Specifically, we train sentence embeddings using the BookCorpus dataset (Kiros et al., 2015) and evaluate the learned embeddings on six downstream tasks: semantic relatedness (SICK), product reviews (CR), subjectivity classification (SUBJ), opinion polarity (MPQA), question type classification (TREC), and paraphrase identification (MSRP). We reimplement the HCL baseline (Robinson et al., 2021) for this experiment[1]. Results are reported in Table 4, with UnDiMix outperforming baselines in all tasks.

[1]The HCL code for the text representation experiment is not provided, hence we implemented the method based on the descriptions provided in the paper. All code will be made publicly available.

Table 3: Top-1% accuracy comparison over UnDiMix variations with learned or fixed equal weights for each component.

Method	CIFAR-10	CIFAR-100	TINY-IMAGENET
UnDiMix $_{Fixed}$	93.16 ± 0.08	71.07 ± 0.09	49.45 ± 0.12
UnDiMix $_{Learned}$	**93.40 ± 0.06**	**71.60± 0.13**	**49.87 ± 0.10**

Table 4: Classification accuracy on CR, SUBJ, MPQA, TREC, MSRP downstream tasks and test Pearson Correlation for Semantic-Relatedness (SICK) task. Sentence representations are learned using Quick-Thought (QT) vectors on the BookCorpus dataset and evaluated on six classification tasks. Evaluation with 10-fold cross-validation for binary classification tasks (CR, SUBJ, MPQA) and over multiple trials for the remaining tasks (TREC, MSRP).

Method	SICK	CR	SUBJ	MPQA	TREC	MSRP	
						(Acc)	(F1)
QT	67.7	67.5	79.9	**80.3**	66.0	68.0	80.1
HCL	60.6	62.7	74.1	79.3	58.6	68.3	79.8
UnDiMix	**74.7**$^{\uparrow 6.97}$	**78.0**$^{\uparrow 10.5}$	**86.8**$^{\uparrow 6.9}$	78.7	**82.8**$^{\uparrow 16.8}$	**70.7**$^{\uparrow 2.7}$	**80.9**$^{\uparrow 0.76}$

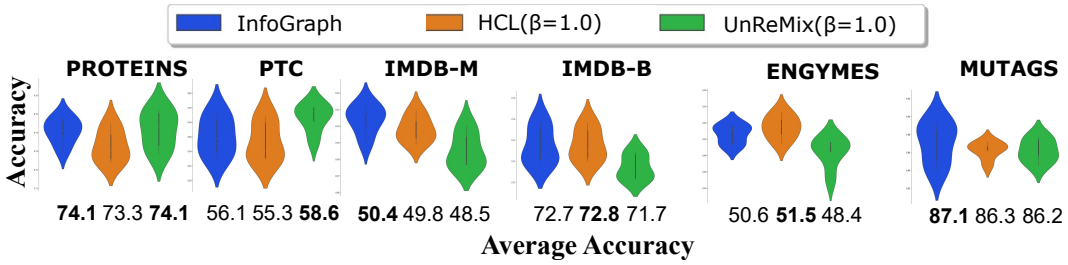


Figure 7: Classification accuracy for six benchmark datasets. Results reported are averaged over 5 independent runs, each with 10-fold cross-validation.Best performance highlighted with bold.

4.3 Graph Representations

We also evaluate UnDiMix on a graph representation task. Using the experimental settings of HCL (Robinson et al., 2021), we utilize the InfoGraph method (Sun et al., 2019) as the baseline and fine-tune an SVM readout function on the learned representations using $\beta = 1$ (HCL hyperparameter) for six graph datasets. Classification accuracy is presented in Figure 7. Overall InfoGraph performs better in 3 out of the 6 benchmarks, and HCL has the best performance 2 out of 6 times. Our method works better or is comparable to InfoGraph and HCL in 2 benchmark datasets. We hypothesize that there is less variability in some of the graph datasets and thus diversity is not utilized completely.

5 Conclusion

In this work, we present UnDiMix, a hard negative selection strategy that samples informative negative examples for contrastive training. We define the notion of "informativeness" by utilizing feature representations, model uncertainty and diversity. As a measure of uncertainty, we extract the gradients of the loss function w.r.t. a computed pseudo-posterior for the negative examples. In addition, we utilize the average distance of each negative example from all other examples as a measure of diversity. Feature representations from the last encoder layer are utilized in computing anchor similarity. Our approach interpolates these three indicators to determine the importance of negative samples. Through experimental analysis on a variety of visual, sentence and graph downstream benchmarks, we showcase that our proposed approach, UnDiMix, outperforms previous state-of-the-art contrastive learning methods, that either rely on random sampling for selecting negative samples, or on importance sampling calculated solely via feature similarity. In the future, we hope to evaluate our method in multi-modal large-scale benchmark datasets and extend UnDiMix to prototypical and graph contrastive learning.

6 REPRODUCIBILITY

All experiments and results reported in this paper are based on publicly available datasets, with links and referrences included in the main tetx and appendices. Experimental setup information that is necessary for reproducing our results, such as hyperparameters, training and evaluation details, are documented in the main paper and in Appendix A. All code will be made publicly available with appropriate documentation.

REFERENCES

Aviad Aberdam, Ron Litman, Shahar Tsiper, Oron Anschel, Ron Slossberg, Shai Mazor, R. Manmatha, and Pietro Perona. Sequence-to-sequence contrastive learning for text recognition. In *CVPR*, 2021.

Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. In *ICLR*, 2020.

Mehdi Azabou, Mohammad Gheshlaghi Azar, Ran Liu, Chi-Heng Lin, Erik C. Johnson, Kiran Bhaskaran-Nair, Max Dabagia, Bernardo Avila-Pires, Lindsey Kitchell, Keith B. Hengen, William Gray-Roncal, Michal Valko, and Eva L. Dyer. Mine your own view: Self-supervised learning through across-sample prediction. *arXiv preprint*, 2021.

Tiffany Tianhui Cai, Jonathan Frankle, David J Schwab, and Ari S Morcos. Are all negatives created equal in contrastive instance discrimination? *arXiv preprint*, 2020.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.

Shuo Chen, Gang Niu, Chen Gong, Jun Li, Jian Yang, and Masashi Sugiyama. Large-margin contrastive learning with distance polarization regularizer. In *International Conference on Machine Learning*, pp. 1673–1683. PMLR, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020a.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *CVPR*, 2021.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint*, 2020b.

Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, 2005.

Ching-Yao Chuang, Joshua Robinson, Lin Yen-Chen, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. In *NeurIPS*, 2020.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

Jasjeet Dhaliwal and Saurabh Shintre. Gradient similarity: An explainable approach to detect adversarial attacks against deep learning. *arXiv preprint arXiv:1806.10707*, 2018.

Ujjal Kr Dutta, Mehrtash Harandi, and C Chandra Sekhar. Unsupervised metric learning with synthetic examples. In *AAAI*, 2020.

Songwei Ge, Shlok Mishra, Chun-Liang Li, Haohan Wang, and David Jacobs. Robust contrastive learning using negative samples with diminished semantics. In *NeurIPS*, 2021.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

Qianjiang Hu, Xiao Wang, Wei Hu, and Guo-Jun Qi. Adco: Adversarial contrast for efficient learning of unsupervised representations from self-trained negative adversaries. In *CVPR*, 2021.

Tri Huynh, Simon Kornblith, Matthew R. Walter, Michael Maire, and Maryam Khademi. Boosting contrastive self-supervised learning with false negative cancellation. In *WACV*, 2022.

Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. In *NeurIPS*, 2020.

Sungnyun Kim, Gihun Lee, Sangmin Bae, and Se-Young Yun. Mixco: Mix-up contrastive learning for visual representation. *arXiv preprint*, 2020.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Ryan Kiros, Yukun Zhu, Russ R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Skip-thought vectors. In *NeurIPS*, 2015.

Ching-Yun Ko, Jeet Mohapatra, Sijia Liu, Pin-Yu Chen, Luca Daniel, and Lily Weng. Revisiting contrastive learning through the lens of neighborhood component analysis: an integrated framework. *arXiv preprint*, 2021.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10/cifar100 (canadian institute for advanced research). 2009.

Ya Le and X. Yang. Tiny imagenet visual recognition challenge. 2015.

Guillaume Leclerc, Andrew Ilyas, Logan Engstrom, Sung Min Park, Hadi Salman, and Aleksander Madry. FFCV: an optimized data pipeline for accelerating ML training. `https://github.com/libffcv/ffcv/`, 2022.

Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning at ICML*, 2013.

Kibok Lee, Yian Zhu, Kihyuk Sohn, Chun-Liang Li, Jinwoo Shin, and Honglak Lee. i-Mix: A domain-agnostic strategy for contrastive representation learning. In *ICLR*, 2021.

Junnan Li, Pan Zhou, Caiming Xiong, Richard Socher, and Steven CH Hoi. Prototypical contrastive learning of unsupervised representations. In *ICLR*, 2021.

Jeremiah Liu, Zi Lin, Shreyas Padhy, Dustin Tran, Tania Bedrax Weiss, and Balaji Lakshminarayanan. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. *Advances in Neural Information Processing Systems*, 33:7498–7512, 2020.

Lajanugen Logeswaran and Honglak Lee. An efficient framework for learning sentence representations. In *ICLR*, 2018.

Shuang Ma, Zhaoyang Zeng, Daniel McDuff, and Yale Song. Active contrastive learning of audio-visual video representations. In *ICLR*, 2021.

Martin Mundt, Iuliia Pliushch, Sagnik Majumder, and Visvanathan Ramesh. Open set recognition through deep neural network uncertainty: Does out-of-distribution detection require generative classifiers? In *ICCV Workshops*, 2019.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint*, 2018.

Tian Pan, Yibing Song, Tianyu Yang, Wenhao Jiang, and Wei Liu. Videomoco: Contrastive video representation learning with temporally adversarial examples. In *CVPR*, 2021.

Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. In *CVPR*, 2021.

Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. In *ICLR*, 2021.

Anshul Shah, Suvrit Sra, Rama Chellappa, and Anoop Cherian. Max-margin contrastive learning. *arXiv preprint*, 2021.

Zhiqiang Shen, Zechun Liu, Zhuang Liu, Marios Savvides, Trevor Darrell, and Eric Xing. Un-mix: Rethinking image mixtures for unsupervised visual representation learning. In *AAAI*, 2020.

Zhiqiang Shen, Zechun Liu, Zhuang Liu, Marios Savvides, Trevor Darrell, and Eric Xing. Un-mix: Rethinking image mixtures for unsupervised visual representation learning. In *AAAI*, 2022.

Fan-Yun Sun, Jordan Hoffman, Vikas Verma, and Jian Tang. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *ICLR*, 2019.

Mamatha Thota and Georgios Leontidis. Contrastive domain adaptation. In *CVPR*, 2021.

Mike Wu, Milan Mosse, Chengxu Zhuang, Daniel Yamins, and Noah Goodman. Conditional negative sampling for contrastive learning of visual representations. In *International Conference on Learning Representations*, 2020.

Zhenda Xie, Yutong Lin, Zheng Zhang, Yue Cao, Stephen Lin, and Han Hu. Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning. In *CVPR*, 2021.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *ICLR*, 2021.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. In *NeurIPS*, 2020.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019.

Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4l: Self-supervised semi-supervised learning. In *CVPR*, 2019.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018.

Xujiang Zhao, Shu Hu, Jin-Hee Cho, and Feng Chen. Uncertainty-based decision making using deep reinforcement learning. In *2019 22th International Conference on Information Fusion (FUSION)*, 2019.

Benjin Zhu, Junqiang Huang, Zeming Li, Xiangyu Zhang, and Jian Sun. Eqco: Equivalent rules for self-supervised contrastive learning. *arXiv preprint*, 2020.

Rui Zhu, Bingchen Zhao, Jingen Liu, Zhenglong Sun, and Chang Wen Chen. Improving contrastive learning by visualizing feature transformation. In *ICCV*, 2021.

A TRAINING AND HYPER-PARAMETER DETAILS

Image representation: For all models, we adopt the training setup of HCL (Robinson et al., 2021). More specifically, we use Resnet-50 (He et al., 2016) as the base encoder, followed by a projection head or reducing the dimensionality of the feature representations from 2048 to 128. We train with Adam (Kingma & Ba, 2015), 10^{-3} learning rate and 10^{-6} weight decay. While our method is implemented on top of Un-Mix, the underlying importance score computations are fairly general and can be easily incorporated into any contrastive learning method. We pre-train all models for 400 epochs with a batch size of 256. All experiments are performed on an NVIDIA T4 GPU with 16GB memory.

Sentence representation: Similar to Robinson et al. (2021), we built upon the official experimental settings of quick-thoughts vectors (`https://github.com/lajanugen/S2V`). For each anchor sentence, the previous and next k sentences (hyper-parameter) are considered positive examples, and all other examples are considered negative examples. Subsequently, Equation 1 is used as loss function to learn the model with Adam optimizer (Kingma & Ba, 2015), batch size of 400, learning rate of 5×10^{-4}, and sequence length of 30. We use the default values for all other hyper-parameters and implement our importance calculation in `s2v-model.py`. Since the official BookCorpus dataset Kiros et al. (2015) is not available, we use an unofficial version obtained from `https://github.com/soskek/bookcorpus`, and following instructions from Robinson et al. (2021).

Graph representation: We adopt the code of Robinson et al. (2021) (`https://github.com/joshr17/HCL/tree/main/graph`) and incorporate our importance score calculation mechanism in `gan_losses.py`. We utilize all datasets downloaded from `www.graphlearning.io`. For a fair comparison to the original `InfoGraph` method and HCL Robinson et al. (2021), we train all the models using the same hyper-parameters values. Following Robinson et al. (2021) we use the GIN architecture Xu et al. (2018) with $K = 3$ layers and embedding dimension $d = 32$, trained for 200 epochs with 128 batch size, Adam optimizer, 10^{-3} learning rate, and 10^{-6} weight decay. Experiments in Figure 7 are reported over 10 experimental trials.

B EVALUATION ON IMAGENET-1K

We evaluate UnDiMix on IMAGENET-1K Deng et al. (2009) and compare with the baselines MoCo-V2 Chen et al. (2020b) and UnMix Shen et al. (2022). For this experiment, we make use of the recently released FFCV data loader Leclerc et al. (2022). We directly use the MoCo-V2 and UnMix official implementation. Table 5 presents the top-1% and top-5% accuracy after training with contrastive loss for 1000 epochs, and then fine-tuning for 100 epochs, with UnDiMix outperforming the baselines.

Table 5: Comparison of performance on IMAGENET-1K

Method	Accuracy	
	Top-1%	Top-5%
MoCo-V2 (Chen et al., 2020b)	48.74	72.7
UnMix (Shen et al., 2022)	50.8	74.9
UnDiMix	**52.2**$^{\uparrow 1.4}$	**76.1**$^{\uparrow 2.2}$

C TRANSFER LEARNING

We also evaluate the performance of UnDiMix in transfer learning. After pre-training models with IMAGENET-1K we fine-tune on CIFAR-10 and CIFAR-100 datasets. Table 6 presents a top-1% accuracy comparison of UnDiMix with the state-of-the-art baseline UnMix Shen et al. (2022). We observe that UnDiMix outperforms UnMix by 0.6 and 1.1 in both CIFAR-10 and CIFAR-100 datasets respectively.

Table 6: Transfer learning performance on CIFAR-10 and CIFAR-100 with models pre-tained on IMAGENET-1K

Method	Top-1% Accuracy	
	CIFAR-10	CIFAR-100
UnMix (Shen et al., 2022)	88.8	60.3
UnDiMix	**89.4**$^{\uparrow 0.6}$	**61.4**$^{\uparrow 1.1}$

D COMPARISON OF COMPUTATIONAL OVERHEAD

Figure 8 presents comparison of the pre-training time in minutes of our method UnDiMix with two types of models: 1) a feature-based model, HCL (Robinson et al., 2021) 2) a clustering-based model, SWAV (Caron et al., 2020). We observe that UnDiMix takes very little time to complete one epoch in comparison with the clustering-based model SWAV and takes similar time as HCL. Hence, UnDiMix improves negative sample selection without adding significant computational overhead.

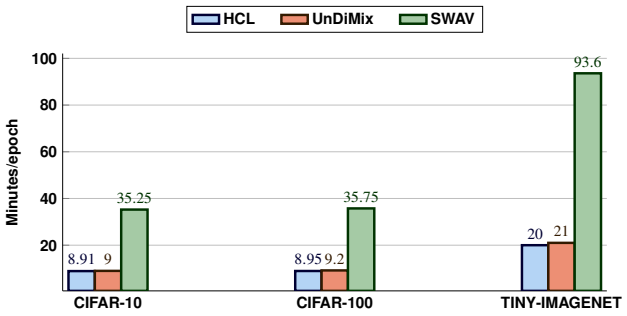


Figure 8: Minutes required per epoch for HCL (Robinson et al., 2021) (feature-based), UnDiMix and SWAV (Caron et al., 2020) (clustering-based).

E ABLATION STUDIES OF IMAGE REPRESENTATIONS

Table 7: Top-1% accuracy of UnDiMix variations with gradients computed with (a) Cross-Entropy loss (CE) and (b) NT-Xent loss.

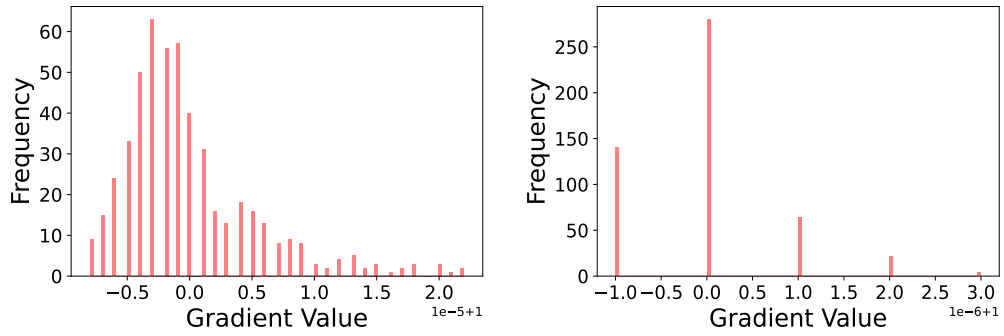
Method	CIFAR-10	CIFAR-100	TINY-IMAGENET
UnDiMix $_{CE}$	**91.99**	**69.43**	**48.12**
UnDiMix $_{NT-Xent}$	91.95	69.27	47.29

E.1 LOSS FUNCTION FOR GRADIENT CALCULATION

In this experiment, we incorporate UnReMix to HCL to study the effect of the loss without the influence of image interporaltion. We experiment with two types of loss functions for calculating gradients w.r.t. each negative example in Eq. (3): 1) CROSS-ENTROPY (CE) loss and 2) NT-XENT, *i.e.*, the Normalized Temperature-scaled Cross-Entropy loss in Eq. (1). Table 7 presents top-1% linear evaluation accuracy for UnDiMix trained with both variants and Figure 9 presents the distribution of gradient values for negative examples. We notice that CE gradients exhibit more variance than NT-XENT gradients (Figure 9), and that the gradient component based on CE results in higher performance than the NT-XENT variation (Table 7). This might allude to the usefulness of pseudo-labeling strategies for tasks with limited labels Lee et al. (2013); Zhai et al. (2019).

F LIMITATIONS

Gradient-based uncertainty assigns higher importance to examples that lie closer to decision boundary and vice versa. By doing so, it reduces the importance of "easy false negative" examples *i.e.*, those examples who are of the same class as the anchor and have low uncertainty, but it fails to reduce the importance of the "hard false negatives" *i.e.*, examples that are of the same class as the anchor and also have high uncertainty (closer to the decision boundary). However, in our experimental analysis



(a) CROSS-ENTROPY (b) NT-XENT

Figure 9: Distribution of gradients of negative examples using (a) Cross-Entropy Loss and (b) NT-Xent Loss.

we verify that the probability of such examples in a given batch decreases as the number of classes increases, *e.g.*, in a batch of 256 examples, the number of negative examples of the same label as the anchor is 26 and 2 for `CIFAR-10` and `CIFAR-100` respectively. Intuitively, the number will be smaller for `Tiny-Imagenet`. Hence, the occurrence of such a situation is so infrequent that it barely affects model training.

G QUALITATIVE EVALUATION

We present additional qualitative results for UnDiMix and HCL Robinson et al. (2021) for `CIFAR-10`, `CIFAR-100` and `TINY-IMAGENET`. Figures 10 and 11 depict the five most important negative examples sampled by HCL (top row) and UnDiMix (bottom row), for eight anchor examples from the `CIFAR-10` and `CIFAR-100` datasets, respectively. Similarly, Figures 12 and 13 depict the seven most important negative examples sampled by HCL (top row) and UnDiMix (bottom row), for each of the six anchor examples from the `TINY-IMAGENET` dataset, observing in total consistent qualitative improvements for more than 22 anchor examples across all datasets.

`CIFAR-10`: In Figure 10 (a) we can observe that HCL assigns most importance to two pairs of negative examples from the same class "Airplane" and "Truck" (**orange** bounding boxes (bboxes)) whereas UnDiMix selects one example from those classes and additionally selects examples from diverse classes including "Deer" and "Automobile" (**green** bboxes). Similar scenarios are visible in Figure 10 (b), (d), (e), (f), (g), (h). Moreover, Figure 10 (c), (e), (f), (g) depicts that HCL assigns more importance to the negative example with the same class as the anchor, *i.e.*, "Bird" (Figure 10 (c), (e)), "Truck" (Figure 10 (f)), "Frog" (Figure 10 (g)) (**red** bboxes). On the other hand, UnDiMix avoids negative examples of the same class as the anchor by including gradient component in the importance calculation of the negative examples. However, UnDiMix sometimes redundant negative examples can also appear in the most five important negatives selected by UnDiMix because of the fewer number of ground-truth classes in `CIFAR-10` dataset (Figure 10 (c), (d), (h)).

`CIFAR-100`: In Figure 11 (a), (f), (h), we notice that the top five most important negative examples for HCL contain examples of the same class as the anchor *i.e.*, "Ray", "Girl", "Possum" (**red** bboxes) whereas UnDiMix assigns most importance to diverse negative examples (**green**). Besides, in Figure 11 (b), (c), (d), (e), (g), we can see that HCL assigns most importance to redundant examples of the same image with different augmentations ((b), (c), (e), (g)) or examples of the same class ((d)) (**orange** bboxes). On the other hand, UnDiMix avoids assigning high importance to different augmentations of the same example or examples of the same class.

`TINY-IMAGENET`: The qualitative results also follow similar trends as in `CIFAR-10` and `CIFAR-100`. For example, Figure 12 depicts that HCL selects redundant negative examples in the top seven most important negatives, *i.e.*, different augmentations of the same image (**orange** bboxes). On the other hand, UnDiMix selects only one of those images (*e.g.*, class 186) and additionally selects examples from diverse classes (**green** bboxes). Similar observations can be made in Figure 12 (c), Figure 13. Moreover, in Figure 12 (b), it is noticeable that HCL assigns most importance to a negative

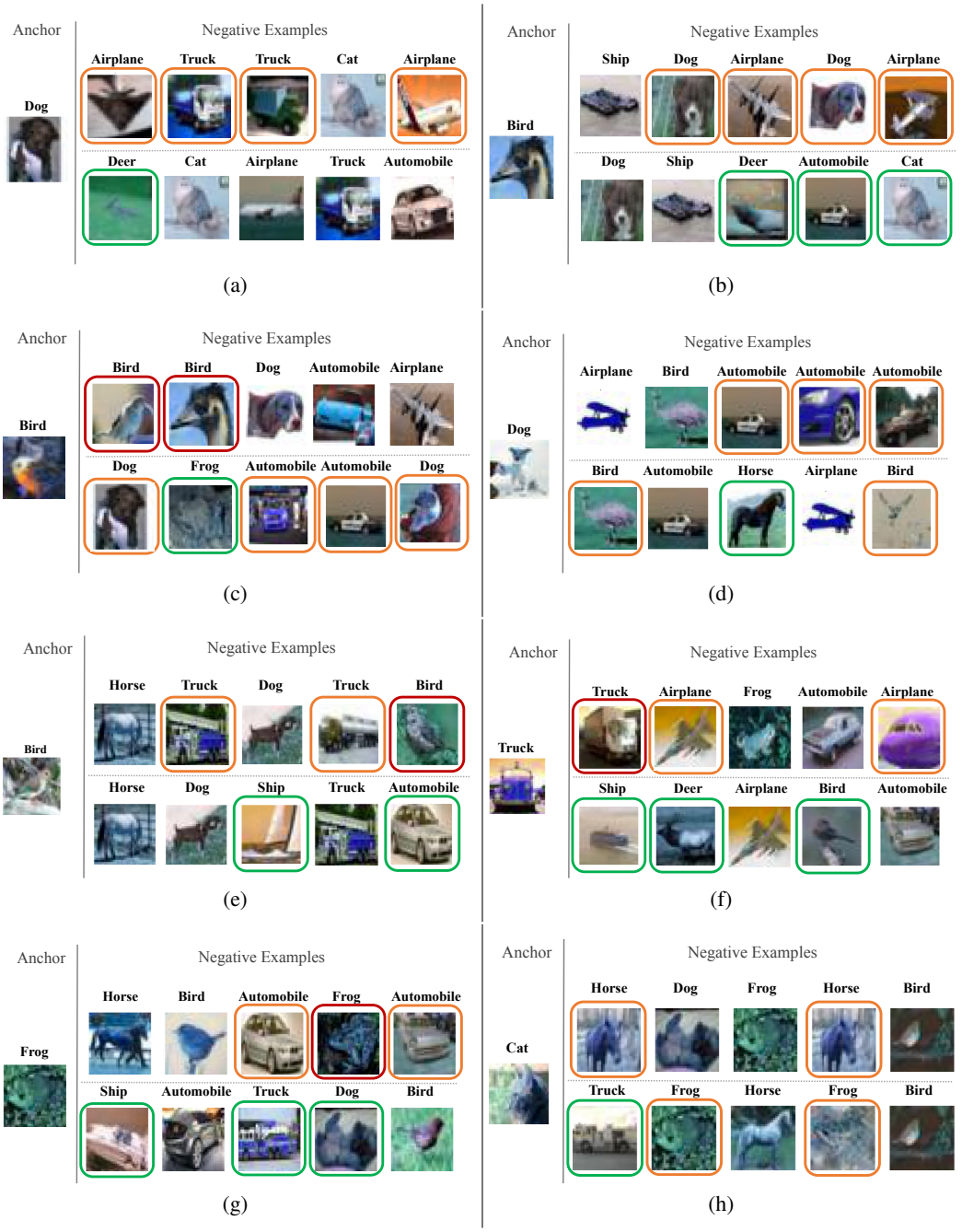


Figure 10: Qualitative evaluations on `CIFAR-10`, best viewed in color. Top Row: HCL (Robinson et al., 2021) has sampled negative examples of the same class as the anchor as a negative example (**red** bbox) or selected examples of the same class multiple times (**orange** bboxes). Bottom Row: UnDiMix samples more diverse true negative examples and avoids false negative examples (**green** bboxes).

example of the same label as the anchor (**red**). On the other hand, UnDiMix does not assign much importance to that negative example and instead selects a diverse example set as top seven negatives.

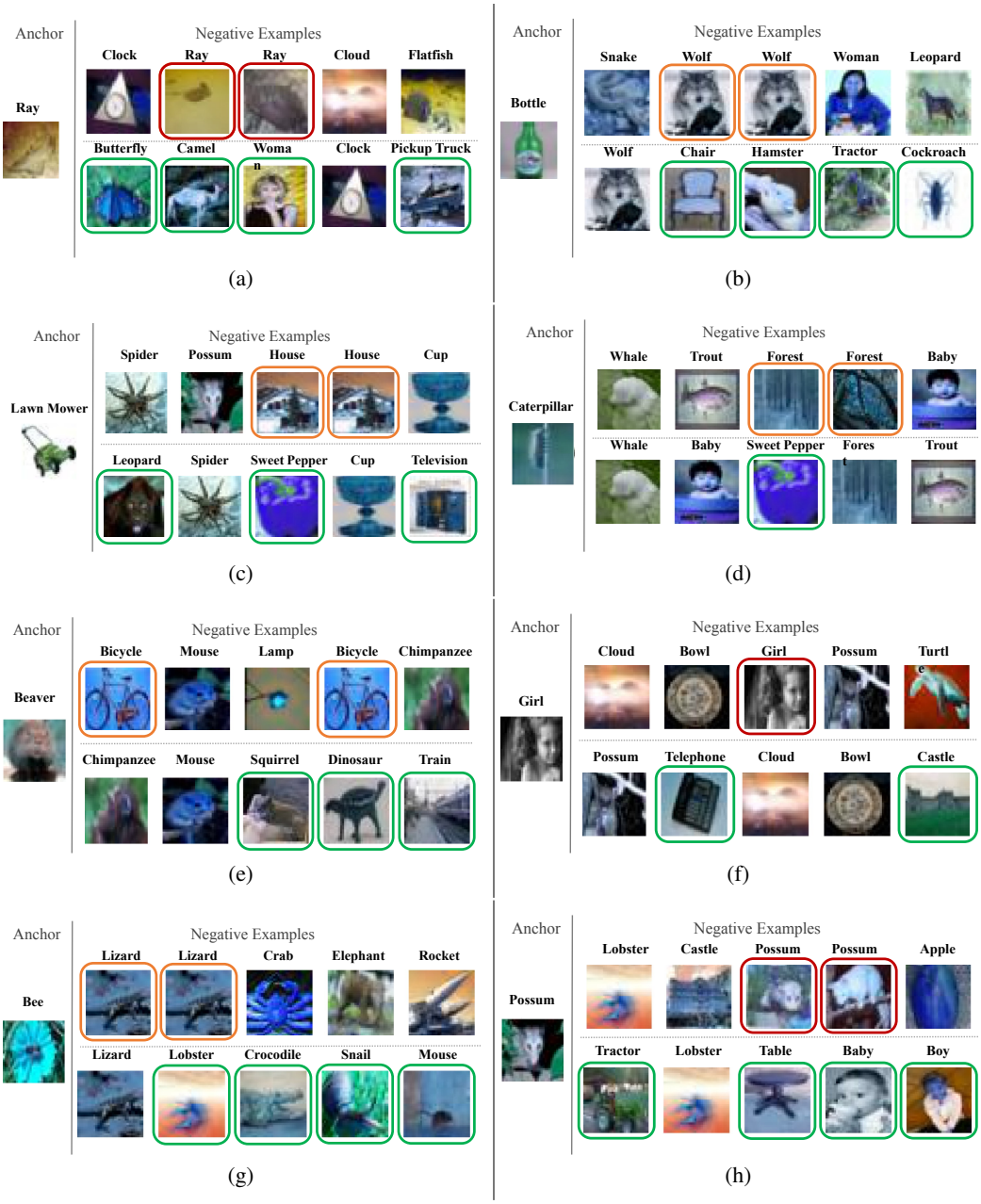
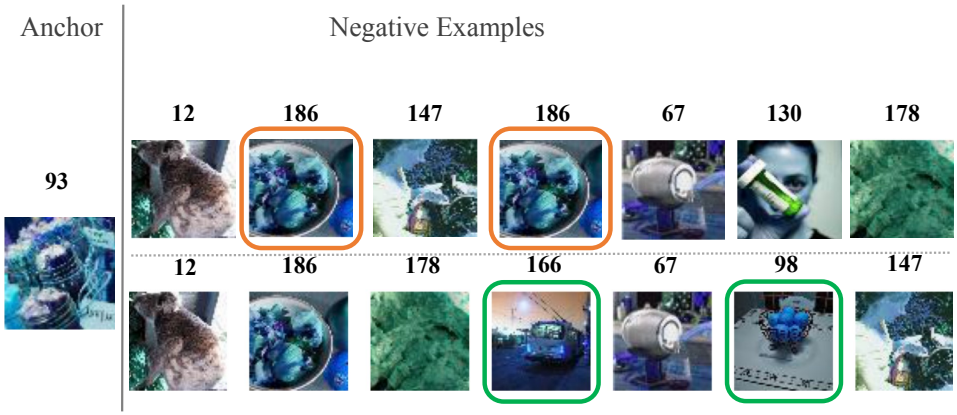
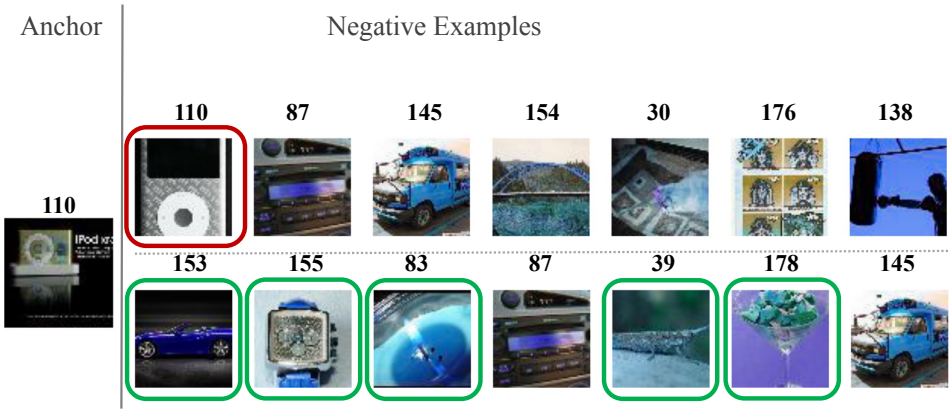


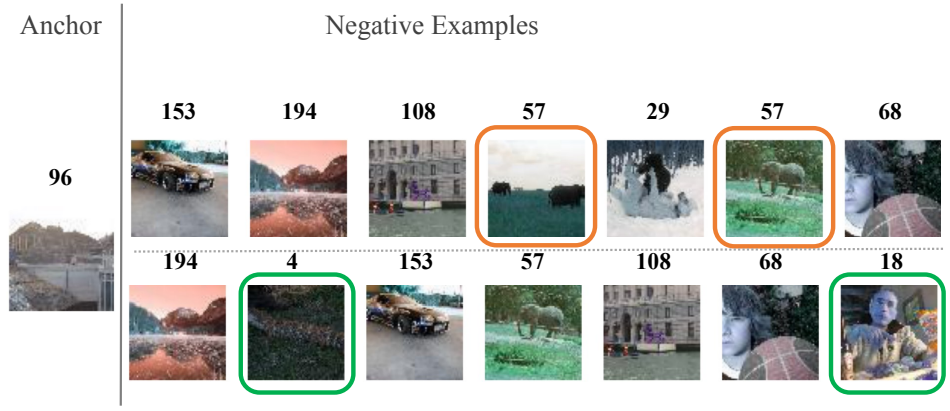
Figure 11: Qualitative evaluations on `CIFAR-100`, best viewed in color. Top Row: HCL (Robinson et al., 2021) has sampled negative examples of the same class as the anchor as a negative example (**red** bbox) or selected examples of the same class multiple times (**orange** bboxes). Bottom Row: UnDiMix samples more diverse true negative examples and avoids false negative examples (**green** bboxes).



(a)

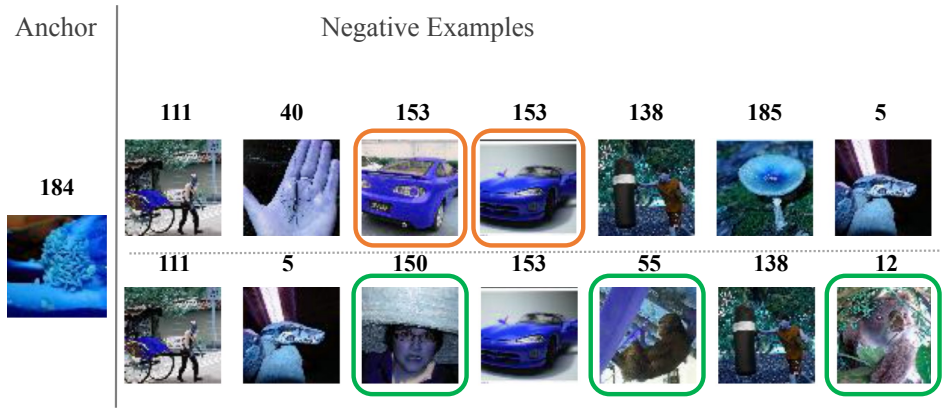


(b)

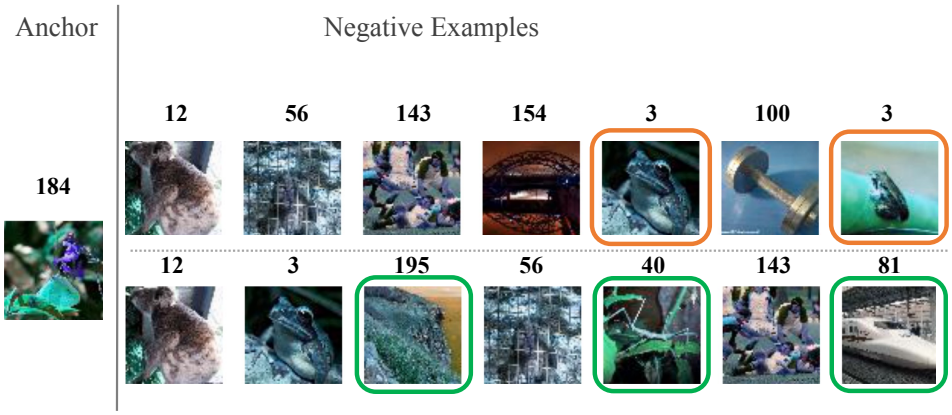


(c)

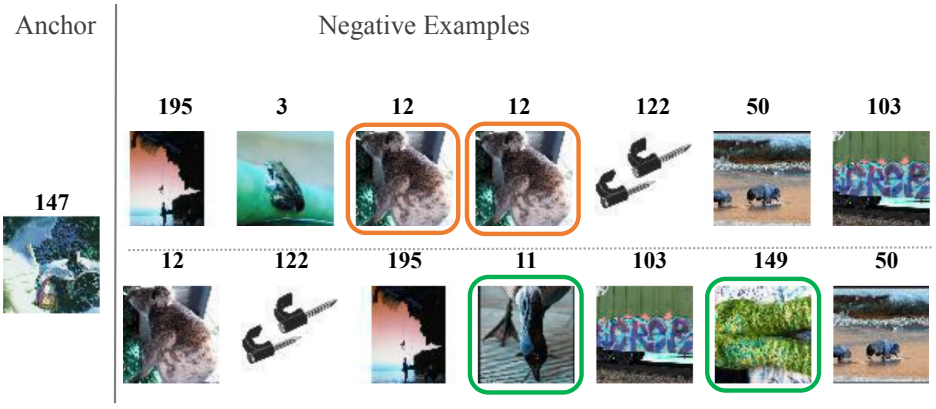
Figure 12: Qualitative evaluations on TINY-IMAGENET, best viewed in color. Top Row: HCL (Robinson et al., 2021) has sampled negative examples of the same class as the anchor as a negative example (**red** bbox) or selected examples of the same class multiple times (**orange** bboxes). Bottom Row: UnDiMix samples more diverse true negative examples and avoids false negative examples (**green** bboxes).



(a)



(b)



(c)

Figure 13: Qualitative evaluations on TINY-IMAGENET, best viewed in color. Top Row: HCL (Robinson et al., 2021) has sampled negative examples of the same class as the anchor as a negative example (**red** bbox) or selected examples of the same class multiple times (**orange** bboxes). Bottom Row: UnDiMix samples more diverse true negative examples and avoids false negative examples (**green** bboxes).