Boundary-aware Backward-Compatible Representation via Adversarial Learning in Image Retrieval

Tan Pan*, Furong Xu*†, Xudong Yang†, Sifeng He, Chen Jiang, Qingpei Guo, Feng Qian,

Xiaobo Zhang, Yuan Cheng, Lei Yang, Wei Chu

Ant Group

{pantan.pt*, booyoungxu.xfr*†, jiegang.yxd†}@antgroup.com

Abstract

Image retrieval plays an important role in the Internet world. Usually, the core parts of mainstream visual retrieval systems include an online service of the embedding model and a large-scale vector database. For traditional model upgrades, the old model will not be replaced by the new one until the embeddings of all the images in the database are re-computed by the new model, which takes days or weeks for a large amount of data. Recently, backward-compatible training (BCT) enables the new model to be immediately deployed online by making the new embeddings directly comparable to the old ones. For BCT, improving the compatibility of two models with less negative impact on retrieval performance is the key challenge. In this paper, we introduce AdvBCT, an Adversarial Backward-Compatible Training method with an elastic boundary constraint that takes both compatibility and discrimination into consideration. We first employ adversarial learning to minimize the distribution disparity between embeddings of the new model and the old model. Meanwhile, we add an elastic boundary constraint during training to improve compatibility and discrimination efficiently. Extensive experiments on GLDv2, Revisited Oxford (ROxford), and Revisited Paris (RParis) demonstrate that our method outperforms other BCT methods on both compatibility and discrimination. The implementation of AdvBCT will be publicly available at https://github.com/Ashespt/AdvBCT.

1. Introduction

Image retrieval brings great convenience to our daily life in various areas such as e-commerce search [8, 16], face recognition [17, 32], and landmark localization [13, 24]. With the rapid development of deep learning, visual re-

---

*These authors contributed equally to this research.
†Corresponding authors.

trieval systems develop towards larger models and richer databases to provide people with better services. Most modern visual retrieval systems include two core parts: (i) an online service of the embedding model which maps an input image to a high-dimensional vector, i.e., the embedding, and (ii) a large-scale vector database which stores the embeddings of the gallery set and is responsible for similarity search when a query image arrives. During the lifetime of the system, new models with better performance are trained and then deployed online to replace the old ones. Unfortunately, the embeddings of the query images extracted by the new model are not compatible with the old ones in the database in most cases. As a result, the vector database must be rebuilt by extracting embeddings for the whole gallery set with the new model. This process is called backfilling [18]. In general, practical industrial applications containing a database with millions to billions of images take days or even weeks for backfilling. During that time, the old model and database must be kept online to handle queries. This so-called cold-refresh [30] model upgrade process is shown in Fig. 1a.

Recently, to save resources and simplify the complex backfilling process, backward-compatible learning was proposed [18]. Backward-compatible learning aims to ensure the compatibility of embedding representations between models. As shown in Fig. 1b, the new model can directly replace the old one and the embeddings of images in stock are updated on-the-fly. With the hot-refresh model-update strategy, the system only needs to maintain one service and one database at a time, effectively reducing the required resource during backfilling.

Meanwhile, these hot-refresh model upgrades also pose new challenges for visual search systems. The retrieval performance during backfilling reflects the compatibility between models, which should not degrade compared to the old system. However, making the new model perfectly back-compatible with the old one leaves no room for improvement on the retrieval task. The ultimate goal of the compatible learning is to enable the compatibility between

(a) The process of the traditional upgrade. (b) The process of the compatible upgrade.
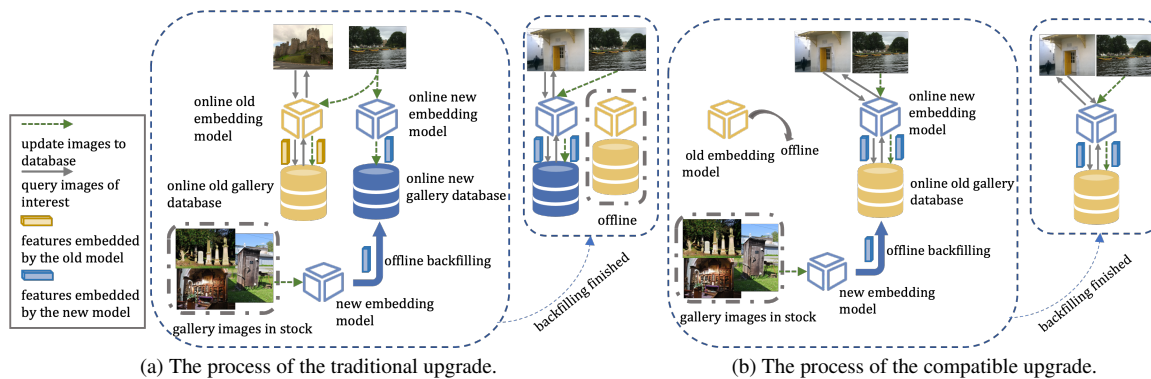
Figure 1. The processes of updating online systems on different models. **Best viewed in color.**
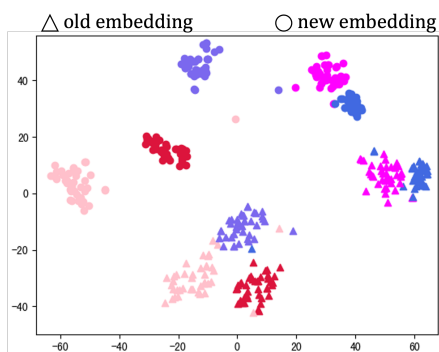

Figure 2. Distributions visualization of the old embeddings and new embeddings on RParis without compatible training by t-SNE [20]. Triangles represent old embeddings and circles represent new embeddings. The old embeddings are extracted by the model trained on 30% data of GLDv2 while the new embeddings extracted by the model trained on 100% data of GLDv2. The embeddings in the same color belong to same class.

the new model and the old model while keeping the performance gain of the back-compatible trained new model as close as possible to that of the independently trained new model. When evaluating BCT methods, both the compatibility between models and the discrimination of the new model for the retrieval task must be taken into account.

The incompatibility between models results from the discrepancy of the embedding distributions of the models, which is illustrated in Fig. 2. Most of the previous works in BCT [14, 30, 31] narrow the distribution gap by adding some regularization losses involving the old and the new embeddings. The main idea of these methods is to pull the new and old embeddings of the same class closer and to push the new and old embeddings of different classes apart from each other in a metric learning manner. Another way to minimize the distribution discrepancy is adversarial learning, which was successfully applied in domain adaptation [2,5]. We decide to combine the metric learning and the

adversarial learning as they measure and minimize the discrepancy between distributions in different ways, and this is complementary for compatible learning in our intuition.

Some works [23, 30] design the loss for the compatibility in a point-to-point manner, which is sensitive to outliers in the training data. Other works [14, 18] propose point-to-set losses to address the issues by loosely constraining the new embeddings inside the class boundary estimated by the old model. However, these estimated boundaries remain constant when training the new model, which may not be flexible for the new model to learn more discriminative embeddings. We design an elastic boundary loss in which the boundary can be dynamically adjusted during training.

In addition, existing methods are evaluated in different settings and on different datasets, which makes it difficult to fairly compare them. In this paper, we adopt a unified training and evaluation protocol to evaluate the existing backward-compatible methods and our adversarial backward-compatible training method (*AdvBCT*).

In summary, the main contributions of our work are listed as follows:

- We first propose an adversarial backward-compatible learning method to close the distribution gap between different models and employ an elastic boundary loss to improve compatibility and discrimination.

- We unify the training and evaluation protocol to assess the performance of 5 BCT works. Meanwhile, we propose a new metric named $\mathcal{P}_{\beta-score}$ to evaluate compatibility and discrimination in a comprehensive metric.

- By comprehensive experiments, we show that the proposed *AdvBCT* outperforms the related state-of-the-arts on several image retrieval datasets, *e.g.* GLDv2, ROxford and RParis in most BCT settings.
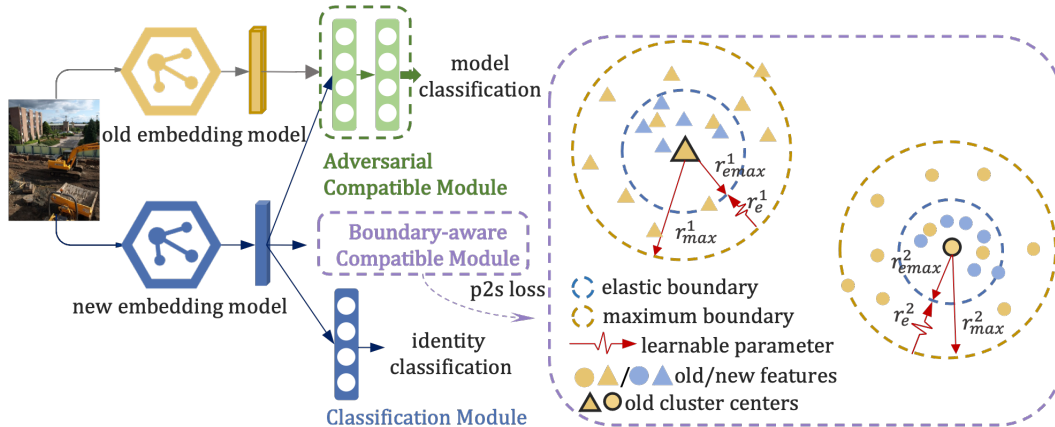
Figure 3. An overview of our *AdvBCT* framework. The adversarial and boundary-aware compatible modules minimize the discrepancy between distributions of the old and new embeddings, while the classification module improves the retrieval performance. In the compatible losses part, the blue and yellow circles refer to the boundary between the new embeddings and old class centers respectively. r_{max} minus the learnable r_e equals to r_{emax}. r_{max} is the maximum distance between old embeddings and the old class center. The solid circles and triangles represent embeddings from two different classes.

2. Related Works

Backward-Compatible Training. Aiming to deploy a new model without the operation of "backfilling", Shen et al. [18] proposed backward-compatible learning to get new embeddings compatible with old ones. Following this topic, compatible learning methods have developed into three branches: backward-compatible learning, forward-compatible learning [15, 34], and cross-model compatible learning (CMC) [1,14,19,23]. Forward-compatible learning aims to enhance the capacity of the current model to leave embedding space for the next upgrade, which is suitable for the close-set scenario. CMC supposes that there are two existing models and adopts projection heads to make models compatible. However, the upgrade of CMC models still needs two embedding models and two databases which is shown in supplemental materials. In this paper, we mainly focus on the backward-compatible training (BCT) situation, which is suitable for open-set scenarios and can simplify the process of backfilling.

On the BCT track, Wang *et al*. [14] aligned class centers between models by a transformation that can be applied to both cross-model compatibility training and compatible training. Zhang *et al*. [31] defined 4 types of model upgrading and proposed a novel structural prototype refinement algorithm to adapt to different protocols. Wu *et al*. [23] constrained the relationship between new embeddings and old embeddings inspired by contrastive learning. Those methods proved to be effective under different datasets and protocols and focused on constraints of prototypes and class centers between new and old models.

Image Retrieval. Image retrieval is a classic task in representation learning. Given a query image, the system will return the top similar images by searching a large-scale pre-encoded database. Image retrieval is related to many research topics such as landmark retrieval, face recognition, person re-identification, and so on. Researchers pay more attention to the improvement of retrieval performance [3, 13, 25, 27, 28] and there exist several comprehensive benchmarks [12, 22]. However, those works rarely discuss the model upgrade problem for visual retrieval systems which is a critical concern in real industrial applications.

Adversarial Learning. Adversarial learning is widely used in various fields such as generative adversarial networks [4, 6, 7], person re-identity [33], and domain adaptation [2,5]. Generative adversarial networks utilize a discriminator to enhance the ability of the generator to synthesize samples. Ganin and Lempitsky's work [2] learned embeddings invariant to the domain shift by a label predictor which combined adversarial learning with domain adaptation. After that, various adversarial domain adaptation methods were proposed [9, 10]. The core of adversarial learning is minimizing the discrepancy between distributions of target and source by defending against attacks.

3. Methodology

3.1. Problem Formulation

Given a training dataset $\mathcal{T} = \{X, Y\}$ of C classes, $X = \{x_1, x_2..., x_m\}$ is the image sample set, and $Y = \{y_1, y_2, ..., y_m\}$ is the one-hot label set. We can train an embedding model ϕ by using X to map image samples into embeddings $Z = \{\phi(x_1), \phi(x_2), ..., \phi(x_m)\}$.

We use ϕ_o trained on $\mathcal{T}_{old}$ and ϕ_n trained on $\mathcal{T}_{new}$ to represent the old embedding model and new embedding model

respectively. Furthermore, ϕ_* represents the model obtained with the same settings as the new embedding model but without compatibility constraints. In the inference stage, a query dataset $\mathcal{Q}$ and a gallery dataset $\mathcal{G}$ are adopted to evaluate the retrieval performance of embeddings. To measure the compatibility performance during backfilling, the embeddings of $\mathcal{G}$ are extracted by ϕ_o, and the ones of $\mathcal{Q}$ are obtained by ϕ_n. In this work, $\langle \phi(x_i), \phi(x_j) \rangle$ represents the distance between two embeddings $\phi(x_i)$ and $\phi(x_j)$ under some distance metrics. The smaller the distance is, the more similar the samples are.

3.2. Backward-Compatible Training Framework

The new model obtained by backward-compatible training needs to be compatible with the old model in embedding space, and it is also demanded to be more discriminative than the old model on retrieval performance. To meet these two requirements in model iterations, as shown in figure Fig. 3, we adopt three modules, namely the classification module, the adversarial compatible module and the boundary-aware compatible module. The classification module aims at guiding the new model to learn discriminative embeddings. The adversarial compatible module reduces the discrepancy between the distribution of the old embeddings and the distribution of the new embeddings in an adversarial manner. The boundary-aware compatible module is utilized to maintain a reasonable distance between the new embeddings and the old cluster centers to improve compatibility and discrimination efficiently. These modules are explained in detail in Sec. 3.3.

3.3. Backward-Compatible Learning

Classification Module. To ensure the discrimination of new embeddings, we adopt the commonly used method that learn the representation using close-set classification: a classifier is connected after the embedding layer of the model and cross-entropy loss is employed to minimize the classification error of the new model. In this way, the model can extract discriminative features for similarity retrieval. The cross-entropy loss is defined as:

$$\mathcal{L}_{cls} = -\frac{1}{N} \sum_{i=1}^{N} y_i \log p_i \tag{1}$$

where N is the number of instances, y_i is the label of the image x_i, and p_i is the probability that x_i belongs to the class y_i.

Adversarial Compatible Module. As mentioned in Sec. 1, reducing the dissimilarity of the embedding distribution between the new model and the old model is the key for model compatibility. The adversarial learning is applied to domain adaptation [21,26,29] to guide the embedding model to generate domain-insensitive embeddings. Inspired by this, we

adopt adversarial learning to make the embedding distribution between models as similar as possible. Specifically, we introduce a discriminator ϕ_d after the embedding layer of the new model as shown in Fig. 3. The discriminator determines whether the embedding comes from the new model or from the old model. Hence, the discrepancy of the embedding distributions is estimated by the classification loss of the discriminator, which is defined as:

$$\mathcal{L}_{adv} = E(\theta_n, \theta_d) = -\frac{1}{N} \sum_{i=1}^{N} \ell_i \log q_i \tag{2}$$

where ℓ_i is a binary label indicating by which model the embedding is generated, and q_i is the probability output by the discriminator. θ_n and θ_d are the parameters of ϕ_n and ϕ_d respectively.

At training time, the embedding model and the discriminator are optimized together in an adversarial way: the model discriminator tries to minimize $\mathcal{L}_{adv}$ while the embedding model tries to maximize it. When the training process converges, the end-to-end network comprised of the embedding model ϕ_n and the discriminator ϕ_d will look for a saddle point of $\hat{\theta}_n$ and $\hat{\theta}_d$ satisfying:

$$\hat{\theta}_d = \underset{\theta_d}{\arg\min} \, E(\hat{\theta}_n, \theta_d) \tag{3}$$

$$\hat{\theta}_n = \underset{\theta_n}{\arg\max} \, E(\theta_n, \hat{\theta}_d) \tag{4}$$

The parameters θ_d are optimized in the direction of minimizing the classification loss of retrieval as Eq. (3) shows. The parameters θ_n of the new embedding model work on maximizing model classification loss as Eq. (4) shows which is equivalent to minimizing the distribution gap.

In order to optimize the end-to-end network composed by the embedding model and the discriminator with opposite objectives using conventional optimization solver such as SGD, a gradient reversal layer (GRL) [2] is inserted between the new embedding model and discriminator. During the forward propagation process, GRL acts as an identity transform. And in the back propagation, GRL takes the gradient from the subsequent level, multiplies it by $-\beta$ and passes it to the preceding layer.

Boundary-aware Compatible Module. When the new and old embeddings are compatible, the positive pairs and negative pairs should satisfy the following conditions on distance constraints. The following formulas are based on $\forall \{i, j, k\}, y_i = y_j \neq y_k$.

$$\langle \phi_n(x_i), \phi_o(x_j) \rangle < \langle \phi_n(x_i), \phi_o(x_k) \rangle \tag{5}$$

$$\langle \phi_n(x_i), \phi_o(x_j) \rangle < \langle \phi_n(x_i), \phi_n(x_k) \rangle \tag{6}$$

Some methods followed the formulas and designed constraints of new embeddings and old embeddings, which we

call point-to-point (*p2p*) constraints. However, the *p2p* constraint is too strict because it is applied to all pairs of samples, which means outliers or corner cases will exert negative effects on training. To address these issues, we transfer this *p2p* constraint into a point-to-set (*p2s*) constraint as follows.

Here, we use Euclidean Metric as the measure of distance $\langle \phi_n(x_i), \phi_o(x_j) \rangle = \|\phi_n(x_i) - \phi_o(x_j)\|_2$. According to the triangle inequality, we can draw some conclusions on relationships of $\phi_n(x_i), \phi_o(x_j)$ and $E_o(X^c)$:

$$B_{lower} = \|\phi_n(x_i) - E_o(X^c)\|_2 - \|\phi_o(x_j) - E_o(X^c)\|_2 \tag{7}$$

$$B_{upper} = \|\phi_n(x_i) - E_o(X^c)\|_2 + \|\phi_o(x_j) - E_o(X^c)\|_2 \tag{8}$$

$$B_{lower} \le \|\phi_n(x_i) - \phi_o(x_j)\|_2 \le B_{upper} \tag{9}$$

where $E_o(X^c)$ is the expectation of $\phi_o(X^c)$ and $X^c = \{x_i\}_{i=1}^n$ is the set of instances of class c, where $\forall \{x_i, x_j\} \in X^c$. Details are given in the supplemental material Sec. 1.

Because the $\|\phi_o(x_j) - E_o(X^c)\|_2$ is a constant, and the range of $\|\phi_n(x_i) - \phi_o(x_j)\|_2$ is determined by $\|\phi_n(x_i) - \phi_o(x_j)\|_2$. Thus, by constraining the distance between $\phi_n(x_i)$ and $E_o(X^c)$, we can constrain the distance between $\phi_n(x_i)$ and $\phi_o(x_j)$. In this paper, we estimate $E_o(X^c)$ as the cluster center of the training samples for class c.

Based on the existing old model, we can easily calculate expectations $\{E_o(X^i)\}_i^C$ namely cluster centers $\mathcal{O} = \{o^1, o^2, ..., o^C\}$ of the training data. The maximum distance set of all classes is defined as $R_{max} = \{r_{max}^1, r_{max}^2, ..., r_{max}^C\}$ in which r_{max}^i represents the maximum distance between embeddings and the cluster center of class i. r_{max}^i is determined by the old embedding space of class i. A *p2s* constraint can be $\|\phi_n(x_i) - E_o(X^i)\|_2 < r_{max}^i$ in which we call r_{max}^i as the max boundary. However, corner cases and outliers will make the distribution looser which leads to a larger r_{max}^i. Although a larger r_{max}^i gives more space to improve discrimination, it will be harder to improve compatibility. We hope to constrain new embeddings and old centers by suitable boundaries. Thus, we use an elastic boundary r_e to dynamically adjust the max boundary between a threshold t and r_{max}. r_e is defined as:

$$r_e^k = w^k |r_{max}^k - t| \tag{10}$$

where w^k is a learnable parameter between 0 and 1.

We employ the elastic boundary r_e to adjust the final max boundary r_{emax}. r_{emax} is defined as Eq. (11) in which k represents class k. The equation will range r_{emax} in $[t, r_{max}]$ or $[r_{max}, t]$.

$$r_{emax}^k = \begin{cases} r_{max}^k - r_e^k & t < r_{max} \\ t - r_e^k & t > r_{max} \end{cases} \tag{11}$$
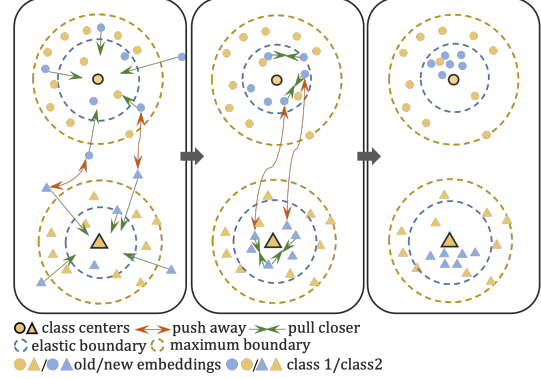

Figure 4. The distribution state evolution of embeddings during training constrained by our $\mathcal{L}_{AdvBCT}$.

Combined with Eq. (11) and Eq. (10), the final r_{emax} can be defined as follows:

$$r_{emax}^k = \begin{cases} (1 - w^k) r_{max}^k + w^k t & t < r_{max} \\ w^k r_{max}^k + (1 - w^k) t & t > r_{max} \end{cases} \tag{12}$$

Based on what mentioned above, we design an *elastic p2s loss* to dynamically minimize $\|\phi_n(x_i) - E_o(X^c)\|_2$. As shown in Fig. 3, the *p2s loss* constrains distances between new embeddings and the old cluster center in a dynamic way. The *elastic p2s loss* can be defined as the following:

$$\mathcal{D}_k = \sum_{i=1}^m max(\langle \phi_n(x_i^k), o^k \rangle - r_{emax}^k, 0) \tag{13}$$

$$\mathcal{L}_{p2s} = \sum_{k=1}^C \mathcal{D}_k \tag{14}$$

In total, our final loss on embedding model can be summed as the following:

$$\mathcal{L}_{AdvBCT} = \mathcal{L}_{cls} + \lambda \mathcal{L}_{p2s} + \gamma \mathcal{L}_{adv} \tag{15}$$

where γ and λ are factors. From the experiments, $\mathcal{L}_{adv}$ affects on the consistency of distributions efficiently in the early training while influences discrimination in the late. So, we progressively reduce γ during training. The state of distributions can be abstracted in Fig. 4.

4. Benchmarks and Metrics

Benchmarks. Training data and backbones can be considered as major factors affecting the performance of the new model when the old model is going to be upgraded. Thus, we discuss three settings as followings. (1) **Extended-data**: Extended-data supposes that the classes of the new training data set $\mathcal{T}_{new}$ remain unchanged but the data number of each class increases. In our setting, the old training data set

$\mathcal{T}_{old}$ composed of 30% images is randomly sampled from the whole data set. (2) **Extended-class**: $\mathcal{T}_{old}$ is composed of 30% classes of the whole data set. (3) **Enlarged-backbone**: The old model is trained on ResNet18 (R18) while the new model is trained on ResNet50 (R50). For Extended-data and Extended-class, the new models are trained on R18. Generally, when the backbone is enlarged, the volume of data should be enlarged too. Therefore, in Enlarged-backbone, the data will increase from 30% to 100% same as Extended-data or Extended-class.

In this article, except for the settings mentioned above, we won't take situations into account that other training schemes are changed.

Metrics. Referring to previous works, the performance metrics of compatible learning methods can be divided into two parts, the performance of compatibility during upgrading and the improvement of retrieval after updating which we call discrimination. They can be presented as followings.

$$\mathcal{P}_{comp} = sigmoid(\frac{M(\phi_n, \phi_o; \mathcal{Q}, \mathcal{G}) - M(\phi_o, \phi_o; \mathcal{Q}, \mathcal{G})}{M(\phi_*, \phi_*; \mathcal{Q}, \mathcal{G}) - M(\phi_o, \phi_o; \mathcal{Q}, \mathcal{G})})$$
(16)
$$\mathcal{P}_{up} = sigmoid(\frac{M(\phi_n, \phi_n; \mathcal{Q}, \mathcal{G}) - M(\phi_*, \phi_*; \mathcal{Q}, \mathcal{G})}{M(\phi_*, \phi_*; \mathcal{Q}, \mathcal{G})})$$
(17)

Where $M(\cdot, \cdot)$ represents the evaluation metric for image retrieval, *e.g.* mean Average Precision (mAP). $M(\phi_n, \phi_o; \mathcal{Q}, \mathcal{G})$ represents the mAP with the setting that embeddings of $\mathcal{Q}$ and embeddings of $\mathcal{G}$ are extracted by ϕ_n and ϕ_o respectively.

$\mathcal{P}_{comp}$ is the indicator of compatibility and $\mathcal{P}_{up}$ measures the performance of the new embedding model ϕ_n compared to the embedding model ϕ_* trained without compatible methods. We use $sigmoid$ function to normalize values because $M(\phi_n, \phi_o; \mathcal{Q}, \mathcal{G})$ is less than $M(\phi_o, \phi_o; \mathcal{Q}, \mathcal{G})$ and $M(\phi_n, \phi_n; \mathcal{Q}, \mathcal{G})$ is less than $M(\phi_*, \phi_*; \mathcal{Q}, \mathcal{G})$ in some cases. For those metrics, a higher value is better.

Except for the metrics mentioned above, inspired by $F - score$ [11], we propose a new metric $P - score$ to measure both the performance of compatibility and discrimination.

$$\mathcal{P}_{\beta-score} = \frac{(1 + \beta^2)\mathcal{P}_{comp} * \mathcal{P}_{up}}{\beta^2 \mathcal{P}_{comp} + \mathcal{P}_{up}}$$
(18)

$P - score$ can take $\mathcal{P}_{comp}$ and $\mathcal{P}_{up}$ into consideration together where β is the impact factor of $\mathcal{P}_{comp}$. Like widely used setting $\beta = 1$ in $F - score$, we also set $\beta = 1$ in $P - score$ as the formula Eq. (19) shows. In this case, $P - score$ is the harmonic mean of $\mathcal{P}_{comp}$ and $\mathcal{P}_{up}$.

$$\mathcal{P}_{1-score} = \frac{2\mathcal{P}_{comp} * \mathcal{P}_{up}}{\mathcal{P}_{comp} + \mathcal{P}_{up}}$$
(19)

Allocation type	Old train-set		New train-set	
	#images	#classes	#images	#classes
Extended-data	445,419	81,313	1,580,470	81,313
Extended-class	470,369	24,393	1,580,470	81,313
Extended-backbone (*class*)	445,419	81,313	1,580,470	81,313
Extended-backbone (*data*)	470,369	24,393	1,580,470	81,313

Table 1. Three different allocations for the training data set sampled from GLDv2. For experiments, the random seed is fixed to reproduce the allocation.

5. Experiments

5.1. Implementation Details

Training Data. We use Google Landmark v2 [22] (GLDv2) as the training dataset. GLDv2 is a large-scale public dataset associated with two challenges Google Landmark Recognition 2019 and Google Landmark Retrieval 2019. Following allocation types mentioned in Sec. 4, the compositions of different training settings are shown in Tab. 1.

Training Settings. All the models are trained on 4 v100 by stochastic gradient descent. For all methods, we train the transformations for 30 epochs, with the learning rate initialized as 0.1. The weight decay is set to 5e-4 and the momentum is 0.9.

Evaluation Metrics. Mean Average Precision (**mAP**) is utilized to evaluate the performance of retrieval. As mentioned in Sec. 3.1, we adopt $\mathcal{P}_{com}$, $\mathcal{P}_{up}$ and $\mathcal{P}_{1-score}$ to evaluate the performance of compatibility and discrimination. We average $\mathcal{P}_{com}$, $\mathcal{P}_{up}$ and $\mathcal{P}_{1-score}$ of every test set as the final $\mathcal{P}_{com}$, $\mathcal{P}_{up}$ and $\mathcal{P}_{1-score}$.

5.2. Ablation Study

Effectiveness of different components. As shown in Fig. 3, our *AdvBCT* has three modules. To verify the impact of each module alone on the final effect, we conduct eight split experiments on each module under the compatible settings of Extended-data and Extended-class, and the effect is evaluated on RParis and ROxford datasets. The experimental results are shown in Tab. 2.

As can be seen from the table, each component is necessary, and all components are combined to achieve the best overall effect. In the first four sets of experiments, each component is tested individually. As shown in Tab. 2, $\mathcal{L}_{cls}$ makes the new model discriminative but lacking compatibility (refer to #1 and #2), and $\mathcal{L}_{p2s}$ performs better compatibility but limited discrimination (refer to #4, #2 and #1). In addition, we also found that the effect of $\mathcal{L}_{adv}$ alone is relatively poor. It's reasonable that adversarial learning is an unsupervised constraint on old and new distributions which cannot constrain instances directly. But when $\mathcal{L}_{adv}$

#	$\mathcal{L}_{cls}$	$\mathcal{L}_{adv}$	$\mathcal{L}_{p2s}$	RParis self	RParis cross	ROxford self	ROxford cross	RParis self	RParis cross	ROxford self	ROxford cross
1(ϕ_o)	✓			75.45	-	49.15	-	74.29	-	54.34	-
2(ϕ_*)	✓			81.15	4.93	63.85	1.29	81.15	4.93	63.85	1.29
3		✓		5.8	6.69	3.25	1.66	5.14	6.8	2.44	1.76
4			✓	76.4	75.23	50.78	44.75	74.82	74.13	51.09	48.22
5	✓	✓		80.87	4.4	**63.92**	2.11	81.09	5.67	62.3	2.34
6	✓		✓	*82.12*	77.18	61.16	*51.63*	*81.66*	76.16	*63.59*	*52.92*
7		✓	✓	76.83	75.7	52.76	49.31	75.09	74.45	51.39	48.67
8	✓	✓	✓	**82.78**	**78.55**	*62.13*	**52.31**	**82.05**	**77.16**	**64.51**	**54.82**

Table 2. Comparison results of different components in Extended-data (left) and Extended-class (right) setting, where both backbones of the old and new model are R18. **Best** and *second best* are highlighted.

Allocation type	$Model_{old}$	$Model_{new}$	RParis self	RParis cross	ROxford self	ROxford cross	GLDv2-test self	GLDv2-test cross	$\mathcal{P}_{up}$	$\mathcal{P}_{comp}$	$\mathcal{P}_{1-score}$
	ϕ_o^{R18}	-	75.45	-	49.15	-	10.03	-	-	-	-
	-	ϕ_*^{R18}	81.15	4.93	63.85	1.20	16.48	0.2	-	7.19	-
Extended-data	ϕ_o^{R18}	ϕ_{BCT}^{R18}	80.58	77.37	56.34	49.66	14.61	11.30	48.02	54.71	51.13
	ϕ_o^{R18}	ϕ_{LCE}^{R18}	*81.57*	77.83	**60.85**	*51.35*	**16.48**	*12.17*	**49.65**	57.41	53.34
	ϕ_o^{R18}	ϕ_{UniBCT}^{R18}	80.93	*78.30*	57.24	50.97	16.06	**13.25**	48.90	**59.19**	**53.52**
	ϕ_o^{R18}	$\phi_{Hot-refresh}^{R18}$	79.57	76.53	58.15	50.05	13.88	10.35	47.78	52.50	50.03
	ϕ_o^{R18}	ϕ_{AdvBCT}^{R18}	**82.78**	**78.55**	*62.13*	**52.31**	15.71	11.49	*49.55*	*58.09*	*53.45*
	ϕ_o^{R18}	-	74.29	-	54.34	-	11.43	-	-	-	-
	-	ϕ_*^{R18}	81.15	4.93	63.85	1.29	16.48	0.2	-	3.38	-
Extended-class	ϕ_o^{R18}	ϕ_{BCT}^{R18}	79.45	76.13	58.94	53.43	14.79	*12.26*	48.33	52.79	50.41
	ϕ_o^{R18}	ϕ_{LCE}^{R18}	*81.26*	*76.78*	*60.49*	54.29	*16.07*	12.04	*49.37*	53.95	*51.51*
	ϕ_o^{R18}	ϕ_{UniBCT}^{R18}	76.92	74.55	59.07	**57.82**	14.80	**12.31**	48.09	*54.78*	51.17
	ϕ_o^{R18}	$\phi_{Hot-refresh}^{R18}$	78.93	75.33	60.31	51.68	14.0	10.41	48.06	47.26	47.57
	ϕ_o^{R18}	ϕ_{AdvBCT}^{R18}	**82.05**	**77.16**	**64.51**	*54.82*	**16.44**	12.05	**50.16**	**54.87**	**52.35**
	ϕ_o^{R18}	-	75.45	-	49.15	-	10.03	-	-	-	-
	-	ϕ_*^{R50}	87.66	4.81	76.56	2.26	22.12	0.2	-	15.44	-
Extended-backbone (data)	ϕ_o^{R18}	ϕ_{BCT}^{R50}	85.54	78.95	68.66	54.42	19.11	12.78	47.81	55.86	51.52
	ϕ_o^{R18}	ϕ_{LCE}^{R50}	**87.49**	**79.36**	**75.16**	57.18	**21.85**	*13.30*	**49.73**	**57.31**	**53.25**
	ϕ_o^{R18}	ϕ_{UniBCT}^{R50}	84.37	78.91	67.42	56.18	*21.57*	**13.42**	48.48	56.80	52.31
	ϕ_o^{R18}	$\phi_{Hot-refresh}^{R50}$	86.78	78.81	75.10	*57.84*	20.61	12.41	49.19	56.53	52.60
	ϕ_o^{R18}	ϕ_{AdvBCT}^{R50}	*86.79*	*79.08*	*75.71*	**59.03**	20.77	12.75	*49.31*	*57.30*	*53.01*
	ϕ_o^{R18}	-	74.29	-	54.34	-	11.43	-	-	-	-
	-	ϕ_*^{R50}	87.66	4.81	76.56	2.26	22.12	0.2	-	11.46	-
Extended-backbone (class)	ϕ_o^{R18}	ϕ_{BCT}^{R50}	84.18	77.12	68.93	57.52	18.85	13.47	47.71	54.59	50.91
	ϕ_o^{R18}	ϕ_{LCE}^{R50}	*85.77*	77.31	72.89	*58.95*	*20.84*	**14.00**	*49.04*	*55.67*	*52.15*
	ϕ_o^{R18}	ϕ_{UniBCT}^{R50}	82.48	*77.33*	66.37	58.51	18.91	*13.99*	47.29	55.51	51.06
	ϕ_o^{R18}	$\phi_{Hot-refresh}^{R50}$	85.62	77.29	**73.95**	56.83	20.47	12.44	49.01	53.63	51.21
	ϕ_o^{R18}	ϕ_{AdvBCT}^{R50}	**86.24**	**78.31**	*73.93*	**60.33**	**22.03**	13.54	**49.65**	**56.64**	**52.83**

Table 3. The compatible training benchmark testing on *BCT, LCE, Hot-refresh, UniBCT,* and *AdvBCT*. ϕ_o^{R18} represents that the backbone of the old model is ResNet18. ϕ_* is trained on the whole data set without any compatible operation. *self* represents self-test $M(\phi_o, \phi_o; \mathcal{Q}, \mathcal{G})$ or $M(\phi_n, \phi_n; \mathcal{Q}, \mathcal{G})$, and *cross* represents cross-test $M(\phi_n, \phi_o; \mathcal{Q}, \mathcal{G})$. In the allocation type of Extended-backbone, *data* represents 30% and 100% training data for the old model and the new model respectively. Similarly, *class* represents 30% class and 100% class for the old model and the new model. Our method *AdvBCT* achieves best performance in the most allocation types. **Best** and *second best* of five works are highligled.

and $\mathcal{L}_{p2s}$ are combined, better results are achieved (refer to #4 and #7). Of course, from the experimental results, the combination of the three achieves the best discrimination

and compatibility. Therefore, our proposed adversarial and boundary-aware compatible modules are all effective.

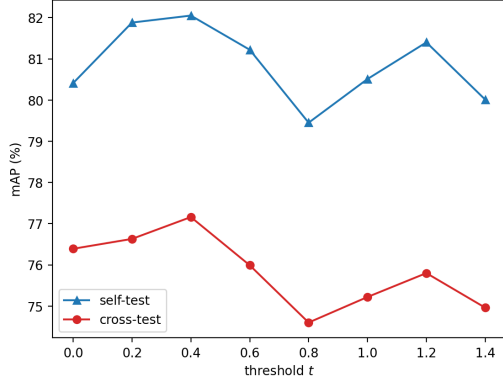**Parameter Analysis.** In order to determine an appropriate

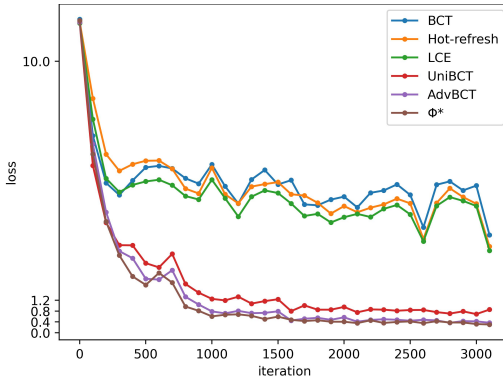Figure 5. The influence of parameter t on RParis dataset in Extended-class setting.

Figure 6. The convergence trend of *BCT* [18], *LCE* [14], *Hot-refresh* [30], *UniBCT* [31], our *AdvBCT* and ϕ_* in Extended-backbone (class) setting.

boundary to constrain the distance between the new embeddings and the old cluster centers, we set a flexible threshold t to obtain the upper or the lower of boundaries in formula Eq. (11). To observe the influence of threshold t, we conduct experiments on RParis dataset in the Extended-class setting. The retrieval performance of different t is shown in figure Fig. 5. When $t = 0.4$, we get the best mAP on both self-test and cross-test. From the figure, we can see that large thresholds and small thresholds perform worse. When the threshold is greater, the model has more flexibility to learn discriminative embeddings, but the compatibility is limited. On the contrary, when the constraint on compatibility is too strict, discrimination is affected.

5.3. Compatible Training Benchmark

Following benchmark settings mentioned in Sec. 4, we evaluate 4 previous works labeled as *BCT* [18], *LCE* [14], *Hot-refresh* [30], and *UniBCT* [31] and our work *AdvBCT*. Thanks to the open source of the Ref [30], we implemented *BCT* and *Hot-refresh* according to their work. We imple-

mented *UniBCT* without the structural prototype refinement algorithm and followed the setting in *LCE* that the transformation layer $K = 0$ while compatible learning. We mark $M(\phi_n, \phi_n; \mathcal{Q}, \mathcal{G})$ or $M(\phi_o, \phi_o; \mathcal{Q}, \mathcal{G})$ as *self-test* and $M(\phi_n, \phi_o; \mathcal{Q}, \mathcal{G})$ as *cross-test*.

The results of different allocation types are shown in Tab. 3. From the experiment results, our *AdvBCT* surpasses previous BCT works in the most cases. Especially in the scenario that classes increase, *AdvBCT* exceeds other methods more than 1% in $\mathcal{P}_{1-score}$.

It is remarkable that the self-test performance of *AdvBCT* and *LCE* outperforms the performance of ϕ_* in Extended-data and Extended-class. The results indicate that the compatibility has positive impacts on discrimination of classes, if the constraints are selected properly. Furthermore, from the results, the method *UniBCT* which minimizes the distances of the old prototypes and new embeddings directly performs not as good as the methods *LCE* and *AdvBCT* in self-test whose constraints are under boundary limits. That means boundary limits are meaningful for improvements on the self-test.

We also visualize convergence trend of several methods in Fig. 6. From the figure, we can see that our method converges fast, the loss declined smooth and the trend is close to ϕ_*. One explanation is that adversarial learning pulls two models into the same distribution which can be helpful to the distance compatible constraint. And the elastic boundary gives more freedom space to optimizing compatibility and discrimination.

6. Conclusion

In this paper, we proposed a novel backward-compatible training method in image retrieval. To better ensure compatibility, we designed the adversarial and boundary-aware compatible modules. Adversarial compatible module aims to pull the embedding distributions of the old and new models close. And boundary-aware compatible module is used to obtain a suitable boundary to constrain distance relationship between the new and old embeddings. In addition, we compare our *AdvBCT* with the existing BCT methods in uniform settings, and an eclectic metric is proposed to verify the pros and cons of all backward-compatible methods, which establishes a comprehensive benchmark for subsequent researchers to handily contribute to the field. Extensive experiments were conducted to verify the effectiveness of our *AdvBCT*. For our future work, we will explore leveraging the old embeddings to further improve discrimination while maintaining compatibility.

References

[1] Rahul Duggal, Hao Zhou, Shuo Yang, Yuanjun Xiong, Wei Xia, Zhuowen Tu, and Stefano Soatto. Compatibility-aware heterogeneous visual search. In *CVPR*, pages 10723–10732, 2021. 3

[2] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, pages 1180–1189, 2015. 2, 3, 4

[3] Yixiao Ge, Feng Zhu, Dapeng Chen, Rui Zhao, et al. Self-paced contrastive learning with hybrid memory for domain adaptive object re-id. In *NIPS*, pages 11309–11321, 2020. 3

[4] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *CACM*, 63(11):139–144, 2020. 3

[5] Lanqing Hu, Meina Kan, Shiguang Shan, and Xilin Chen. Duplex generative adversarial network for unsupervised domain adaptation. In *CVPR*, pages 1498–1507, 2018. 2, 3

[6] Yifan Jiang, Shiyu Chang, and Zhangyang Wang. Transgan: Two pure transformers can make one strong gan, and that can scale up. In *NIPS*, pages 14745–14758, 2021. 3

[7] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, pages 4401–4410, 2019. 3

[8] Beibei Li, Anindya Ghose, and Panagiotis G Ipeirotis. Towards a theory model for product search. In *WWW*, pages 327–336, 2011. 1

[9] Kai Lin, Thomas H Li, Shan Liu, and Ge Li. Real photographs denoising with noise domain adaptation and attentive generative adversarial network. In *CVPR*, pages 0–0, 2019. 3

[10] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. volume 31, 2018. 3

[11] Brian W Matthews. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *BBA-PS*, 405(2):442–451, 1975. 6

[12] Brianna Maze, Jocelyn Adams, James A Duncan, Nathan Kalka, Tim Miller, Charles Otto, Anil K Jain, W Tyler Niggel, Janet Anderson, Jordan Cheney, et al. Iarpa janus benchmark-c: Face dataset and protocol. In *ICB*, pages 158–165, 2018. 3

[13] Ke Mei, Jinchang Xu, Yanhua Cheng, Yugeng Lin, et al. 3rd place solution to" google landmark retrieval 2020". *arXiv preprint arXiv:2008.10480*, 2020. 1, 3

[14] Qiang Meng, Chixiang Zhang, Xiaoqiang Xu, and Feng Zhou. Learning compatible embeddings. In *CVPR*, pages 9939–9948, 2021. 2, 3, 8

[15] Vivek Ramanujan, Pavan Kumar Anasosalu Vasu, Ali Farhadi, Oncel Tuzel, and Hadi Pouransari. Forward compatible training for large-scale embedding retrieval systems. In *CVPR*, pages 19386–19395, 2022. 3

[16] Jennifer Rowley. Product search in e-shopping: a review and research propositions. *JCM*, 2000. 1

[17] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, pages 815–823, 2015. 1

[18] Yantao Shen, Yuanjun Xiong, Wei Xia, and Stefano Soatto. Towards backward-compatible representation learning. In *CVPR*, pages 6368–6377, 2020. 1, 2, 3, 8

[19] Megha Srivastava, Besmira Nushi, Ece Kamar, Shital Shah, and Eric Horvitz. An empirical analysis of backward compatibility in machine learning systems. In *SIGKDD*, pages 3272–3280, 2020. 3

[20] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 9(11), 2008. 2

[21] Mei Wang and Weihong Deng. Deep visual domain adaptation: A survey. *Neurocomputing*, 312:135–153, 2018. 4

[22] Tobias Weyand, Andre Araujo, Bingyi Cao, and Jack Sim. Google landmarks dataset v2-a large-scale benchmark for instance-level recognition and retrieval. In *CVPR*, pages 2575–2584, 2020. 3, 6

[23] Shengsen Wu, Liang Chen, Yihang Lou, Yan Bai, Tao Bai, Minghua Deng, and Ling-Yu Duan. Neighborhood consensus contrastive learning for backward-compatible representation. In *AAAI*, volume 36, pages 2722–2730, 2022. 2, 3

[24] Cheng Xu, Weimin Wang, Shuai Liu, Yong Wang, Yuxiang Tang, Tianling Bian, Yanyu Yan, Qi She, and Cheng Yang. 3rd place solution to google landmark recognition competition 2021. *arXiv preprint arXiv:2110.02794*, 2021. 1

[25] Furong Xu, Bingpeng Ma, Hong Chang, and Shiguang Shan. Prdp: Person reidentification with dirty and poor data. *ToC*, 2021. 3

[26] Furong Xu, Bingpeng Ma, Hong Chang, Shiguang Shan, and Xilin Chen. Style transfer with adversarial learning for cross-dataset person re-identification. In *ACCV*, pages 165–180, 2018. 4

[27] Furong Xu, Meng Wang, Wei Zhang, Yuan Cheng, and Wei Chu. Discrimination-aware mechanism for fine-grained representation learning. In *CVPR*, pages 813–822, 2021. 3

[28] Furong Xu, Wei Zhang, Yuan Cheng, and Wei Chu. Metric learning with equidistant and equidistributed triplet-based loss for product image search. In *WWW*, pages 57–65, 2020. 3

[29] Kaichao You, Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Universal domain adaptation. In *CVPR*, pages 2720–2729, 2019. 4

[30] Binjie Zhang, Yixiao Ge, Yantao Shen, Yu Li, Chun Yuan, Xuyuan Xu, Yexin Wang, and Ying Shan. Hot-refresh model upgrades with regression-free compatible training in image retrieval. In *ICLR*, 2021. 1, 2, 8

[31] Binjie Zhang, Yixiao Ge, Yantao Shen, Shupeng Su, Chun Yuan, Xuyuan Xu, Yexin Wang, and Ying Shan. Towards universal backward-compatible representation learning. 2020. 2, 3, 8

[32] Wenyi Zhao, Rama Chellappa, P Jonathon Phillips, and Azriel Rosenfeld. Face recognition: A literature survey. *CSUR*, 35(4):399–458, 2003. 1

[33] Zhedong Zheng, Xiaodong Yang, Zhiding Yu, Liang Zheng, Yi Yang, and Jan Kautz. Joint discriminative and generative learning for person re-identification. In *CVPR*, pages 2138–2147, 2019. 3

[34] Da-Wei Zhou, Fu-Yun Wang, Han-Jia Ye, Liang Ma, Shiliang Pu, and De-Chuan Zhan. Forward compatible few-

shot class-incremental learning. In *CVPR*, pages 9046–9056, 2022. 3