Long-tail Recognition via Compositional Knowledge Transfer

Sarah Parisot   Pedro M. Esperança   Steven McDonagh   Tamas J. Madarasz

Yongxin Yang   Zhenguo Li

Huawei Noah's Ark Lab

Abstract

In this work, we introduce a novel strategy for long-tail recognition that addresses the tail classes' few-shot problem via training-free knowledge transfer. Our objective is to transfer knowledge acquired from information-rich common classes to semantically similar, and yet data-hungry, rare classes in order to obtain stronger tail class representations. We leverage the fact that class prototypes and learned cosine classifiers provide two different, complementary representations of class cluster centres in feature space, and use an attention mechanism to select and recompose learned classifier features from common classes to obtain higher quality rare class representations. Our knowledge transfer process is training free, reducing overfitting risks, and can afford continual extension of classifiers to new classes. Experiments show that our approach can achieve significant performance boosts on rare classes while maintaining robust common class performance, outperforming directly comparable state-of-the-art models.

1. Introduction

Standard classification models rely on the assumption that all classes of interest are equally represented in training datasets. This strong assumption is rarely valid in practice: most real life datasets exhibit long-tail distributions where a subset of classes comprise a large amount of training data (the so-called common or head classes) and remaining tail (rare) classes only possess a handful of training samples [17]. These distributions typically reflect what is observed in the wild, with data corresponding to tail classes being more challenging to learn due to sample rarity, acquisition and collation costs or combinations of these factors. This results in trained models that display highly imbalanced accuracies, achieving strong performance on common classes, and poor performance on rare classes.

Long-tail distributions present two main challenges for recognition models. Firstly, the imbalanced distribution of class specific data leads to models exceedingly biased to-
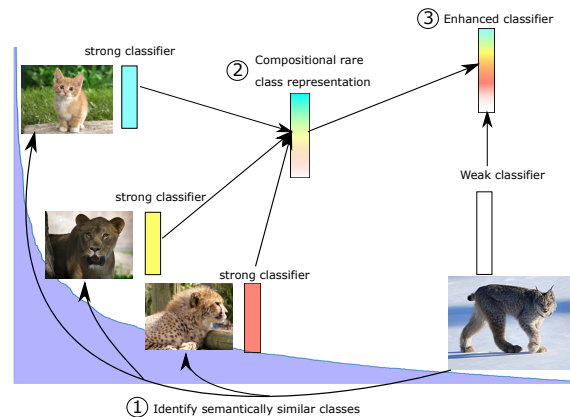

Figure 1. Our approach for long-tail recognition seeks to improve representations for rare classes via knowledge transfer from common classes. We identify semantically similar head classes using class prototypes and transfer reliable learned characteristics from these classes to obtain higher quality rare class representations.

wards common classes. Numerous strategies have been developed to address this imbalance challenge. Popular strategies aim to rebalance the training process via biased resampling [14, 17, 21, 33], rebalancing loss functions [2, 7, 12, 21, 25], or ensembling and routing [6, 29, 31].

While the class imbalance problem has been extensively studied, a second important challenge associated with long-tail distributions has received only limited attention. Rare classes on the distribution tail typically have very limited amounts of data available (*e.g.* 1 to 20 training samples per class), which can easily lead to overfitting or memorisation when attempting to learn class specific decision boundaries. This issue is exacerbated by rebalancing strategies, in particular those relying on oversampling data from the tail. However approaches dedicated to long-tail recognition very rarely include specific mechanisms to account for the limited tail data. A synthetic sample generation process was proposed in [27] to improve rare class distributions. A knowledge transfer mechanism was proposed in [17], aiming to learn better feature representations by trans-

ferring knowledge from learnable memory vectors, under the assumption that the model can automatically recognise which images require such transfer. The process, however, failed to consider a transfer mechanism for classifier learning, which is where lack of data has the largest impact [14].

In contrast, few-shot learning (FSL) has been extensively studied in settings that rely on strong assumptions, making direct application of existing solutions to the long-tail problem challenging. Indeed, FSL approaches are typically developed so as to perform well on so-called episode benchmarks [20, 26], defined as a small group of sampled classes with a fixed, consistent number of training samples per class. These assumptions typically lead to poor generalisation both with regards to common classes and when a variable number of training samples are available [17].

Class prototypes are a popular concept in FSL. They are defined as class representations computed directly from the data, which is advantageous in limited data regimes due to the difficulty of learning reliable few-shot classifiers. Instead of seeking appropriate decision boundaries, prototypical models seek to learn compact and well separated representations by minimising samples' distance with respect to their class prototypes [24]. This can be achieved by explicit prototypes computation via artificially constructed episode batches [24], or implicitly, using cosine-distance based classifiers [10, 19]. Learning distance based feature representations encourages classes with similar semantic meanings to be close in feature space. This provides an advantageous solution to leverage class similarities for our long-tail task.

In this work, we introduce a novel classification strategy for long-tail recognition that addresses the few-shot problem via a training free knowledge transfer mechanism. Our objective is to transfer knowledge, acquired from data-rich common classes, to semantically similar and yet data-hungry rare classes in order to obtain richer tail class representations. We provide an overview schematic of this idea in Figure 1. We leverage the fact that class prototypes and learned cosine classifiers provide two different representations of class cluster centres in feature spaces, and use an attention mechanism to transfer and recompose learned features from classifiers to prototypes. Importantly, information is only transferred from common to rare classes, addressing the limited reliability of classifiers trained in few shot settings and affording easy introduction of new, unseen classes in the classification process.

Our proposed approach is not tied to a specific model architecture or training strategy as it leverages a pre-trained model. As a result, we additionally provide a detailed analysis of the impact of cosine classifiers on pre-trained backbones, as well as the impact of sampling strategies on prototype quality. Based on this analysis, we propose training recommendations to optimise the feature and classifier learning stage for better knowledge transfer.

Our **contributions** can be summarised as follows:

- A novel knowledge transfer mechanism that composes semantically similar, learned, common-class representations to obtain richer and more discriminative tail-class classifiers. Performance gains of up to 5% on rare classes, while maintaining common class performance, without any additional training requirements.

- A flexible solution that affords continual adaptation to new, unseen classes; as well as the ability to construct classifiers tailored for specific class groups.

- An analysis of popular training strategies (sampling methods, classifier type) with recommendations towards optimisation of our transfer learning task.

2. Related work

2.1. Long-tail recognition.

Long-tail recognition has received increasing attention in recent years, and popular benchmarks for the task now exist [17]. Pre-existing works have mainly focused on the class imbalance problem. Solutions proposed to address the issue can be separated in three categories: a) data resampling, b) rebalancing loss functions and post processing, c) ensembling and routing strategies.

Data resampling is one of the most natural ideas when attempting to learn using imbalanced data. The idea involves oversampling less common classes [4, 14, 18] or undersampling the head classes by removing data [8], which can lead to over or underfitting issues. Multiple oversampling strategies were studied in [14], where it was shown that classifiers trained with oversampling achieve better overall performance, yet uniformly trained encoders exhibited higher robustness when applying different rebalancing techniques.

More recently, resampling strategies have been mainly leveraged for classifier training. Two-stage training methods have been proposed [13, 14, 17], where a model encoder is first trained using uniform sampling, then a classifier is retrained with class balanced sampling. A two-branch siamese model was presented in [33], with each branch receiving data samples following distinct distributions. The encoder and respective classifiers are trained using an adaptive combination of classification predictions from each branch. An adaptive sampling strategy was proposed in [21], where the notion of an optimal common-to-rare class ratio is estimated via a meta-learning procedure, while a synthetic data generation process addressing class imbalance was recently proposed in [27].

Rebalancing loss functions. Altering the training process through loss function definition constitutes a popular strategy, focusing on both representation learning and classification. Early works relied on reweighting samples in the

cross-entropy loss [2], achieving mixed results. The focal loss, which has shown great promise in object detection settings [16], also achieved limited improvement. Recent approaches involve a combination of rebalancing loss functions and post processing. A causality based strategy was introduced in [25], relying on post training adjustment of normalised classifiers to reduce the impact of biased momentum-based training. Ren *et al*. [21] show that the softmax function yields biased gradient estimations, and introduce a balanced softmax alternative. Zhang *et al*. [30] and Hong *et al*. [12] propose recalibration and postprocessing strategies to address the label distribution shift between training and test datasets. Zhong *et al*. [32] rely on label smoothing and mixup strategies, while Cui *et al*. [7] combine cross-entropy with supervised contrastive learning, achieving stronger representations at the cost of more expensive training. Finally, closest to our work, Samuel *et al*. [23] propose to rely on a prototype-based auxiliary loss, where class prototypes are computed at the beginning of every epoch during training. Introducing learnable class specific uncertainty parameters and weights yields stronger backbone representations, yet classification still relies on learned classifier weights.

Routing and ensembling has been considered in multiple recent works [3, 6, 13, 29, 31]. The key assumption is that relying on multiple experts can afford the learning of more robust classifiers. Iscen *et al*. [13] train a set of classification models, and then distill their knowledge to a new student model. Wang *et al*. [29] propose to train multiple classifiers with a diversity loss, then learn a routing mechanism, where ambiguous predictions are redirected to a different expert. Cai *et al*. [3] propose to improve on this strategy by training different experts optimised for specific class groups. Further to this, Zhang *et al*. [31] propose to learn ensembling weights in an unsupervised manner at test time via contrastive learning, in order to adapt to variable testing distributions. Ensembling strategies such as these tend to achieve superior performance, which can be partly attributed to increased capacity: each expert learns a different set of features in the last two blocks of ResNet models, consistently increasing model capacity as new experts are introduced, leading to scalability concerns. Finally, rather than learning higher capacity models, Cui *et al*. [6] separate the parameter space into blocks dedicated to recognising different subsets of class groups, which are then aggregated at inference time.

Very few approaches listed here introduce a mechanism that addresses the lack of training data at the tail and the risks of overfitting. Our work focuses on this issue, and is not tied to a rebalancing strategy. Therefore, it provides complementary benefits to approaches that focus on backbone training improvements [7, 21, 23], as discussed above.

2.2. Few-shot learning

Also related to our work, few-shot learning focuses on developing methods specifically to address data scarcity related problems. These methods are typically developed with a sole focus on the few-shot classes, in artificial settings where all classes of interest have the exact same training data distribution, and models are pre-trained on a different set of base classes on which performance is rarely evaluated [20, 26]. Popular FSL approaches typically rely on either distance based approaches [10, 19, 24], or meta-gradient based techniques [9, 15, 22] which aim to quickly adapt models using only few gradient updates.

Our work relates to distance based approaches, which commonly rely on the concept of class prototypes. Prototypes provide an alternative to classifier learning and are directly computed from the data using pre-trained encoders instead of being learned [24]. Recent works have proposed to combine prototypes with cosine classifiers [10, 19] to facilitate training on base classes. Closer to our work, [10] also sought to evaluate performance on base classes, and propagated classifier information during the rare classifier training process via an attention mechanism with rare training samples. Their approach however, still suffers from issues inherent with FSL research, as it relied on episodes to train and evaluate rare classifiers.

3. Methodology

3.1. Model pre-training

Consider a training dataset $\mathcal{D} = \{\mathcal{X}, \mathcal{Y}\}$ comprising N classes, training images $\mathcal{X}$ and corresponding class labels $\mathcal{Y}$. We assume this dataset is long-tail distributed, with n_R rare classes defined as classes with the least amount of training data available (e.g. less than 20 samples). Further to this, we consider a classification model $\mathcal{M} = \{f_\theta, W\}$ comprising a feature encoder backbone f_θ, and classifier weights $W \in \mathbb{R}^{F \times N}$, where F is the feature dimension.

Importantly, we define classifier W as a cosine classifier, characterised by the fact that classification predictions are based on distance, rather than decision boundaries. More formally, for an input image $\mathbf{x}$ with feature representation $f_\theta(\mathbf{x})$, class predictions are computed as:

$$q(y = i|\mathbf{x}, \theta, W) = \frac{\exp\{S(f_\theta(\mathbf{x}), w_i)\}}{\sum_{k \in 1...N} \exp\{S(f_\theta(\mathbf{x}), w_k)\}} \quad (1)$$

$$\text{with } S(f_\theta(\mathbf{x}), w_i) = \frac{f_\theta(\mathbf{x})}{\|f_\theta(\mathbf{x})\|} \cdot \frac{w_i^\top}{\|w_i\|} \cdot s \quad (2)$$

where w_i is the ith column of classifier W, corresponding to class i's center; and $s \in \mathbb{R}^+$ is a parameter commonly used at training time to obtain sharper softmax distributions and facilitate the training process.
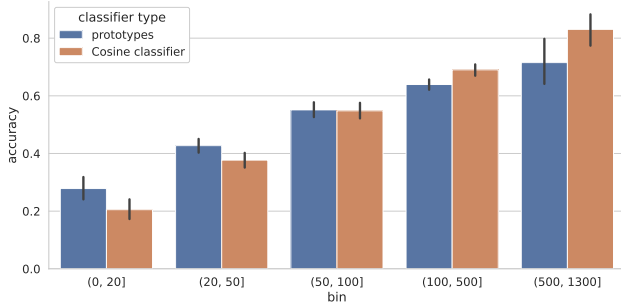
Figure 2. Comparison of class prototype and learned classifier weights accuracies with respect to the amount of class specific training data available. Validation accuracy from a model trained on the ImageNet-LT dataset.

Relying on models with the cosine similarity is crucial for our knowledge transfer process. The key advantage of training a distance based classifier is that it pushes semantically similar classes to be close to each other in feature space [10], facilitating inter-class knowledge transfer and feature comparison. Additionally, training models with fixed norm outputs has been observed to yield more stable training, and well clustered features [28].

In subsequent sections, we assume a cosine classifier model $\mathcal{M}$ has been trained on dataset $\mathcal{D}$ using a standard cross-entropy loss. In our experiments, we will further evaluate and discuss the impact of using a cosine classifier, in addition to training settings to minimise the imbalance problem during this stage. We further note that while we consider a cross-entropy loss for simplicity and ease of evaluation, more complex rebalancing loss functions could be considered as well [12, 21].

3.2. Compositional knowledge-transfer classifier

The cosine classifier provides learnable class representations that can be geometrically interpreted as class cluster centers in feature space, as they rely on cosine distance to determine classification assignments. These learned classifiers bear striking conceptual similarity to class prototypes, which are an alternative, learning-free strategy to obtain class cluster centres [24]. They are typically computed directly from the training data as the average feature representation of all training samples in a given class:

$$p_i = \frac{1}{n_i} \sum_{x_j \in \{\mathcal{X} | y_j = i\}} \frac{f_\theta(x_j)}{\|f_\theta(x_j)\|} \quad (3)$$

where n_i is the number of training samples available for class i, and y_j is the class index of sample x_j. Computed prototypes can then be used identically to learned cosine weights by replacing w_i with p_i in Equation 1.

Typically, class prototypes have higher performance on rare classes, while learned cosine classifiers achieve the strongest common class accuracy (see Figure 2). One key takeaway is that both provide two class representations with the same conceptual interpretation (feature cluster centres), favouring different class groups.

Based on this observation, we propose to leverage both prototypes and cosine classifiers to construct a higher quality classifier. Our objective is to transfer knowledge acquired via learning from data-rich common classes towards semantically similar rare classes. We take inspiration from the Neural Turing Machines' [11] reading operation, which reads and combines memory vectors according to cosine-similarity based coefficients. Using class prototypes as representation anchors due to their higher few-shot performance, we seek to identify common characteristics across classes that we recombine towards achieving high quality representation on rare classes. Using a prototype-to-classifier attention mechanism, we compute, for a given class i, a compositional class representation w_i^{kt} that recombines relevant classifier knowledge from semantically similar classes:

$$w_i^{\text{kt}}(k) = \sum_j \alpha_{ij} \frac{w_j}{\|w_j\|} \quad (4)$$

$$\alpha_{ij} = \text{softmax}\left(\frac{p_i}{\|p_i\|} \cdot \frac{w_j^\top}{\|w_j\|} \cdot \tau \right) \quad (5)$$

$$w_j \in W(k) = \{w_l | l \in 1, \dots, N, \text{ if } n_l > k\} \quad (6)$$

where τ is a temperature parameter introduced to increase distribution sharpness and k defines the minimum number of training samples available per class below which classifier knowledge is not transferred. This second parameter is crucial towards improving few-shot representations, allowing us to effectively transfer information from common to rare classes without propagating potentially unreliable few-shot features. We compute our final classifier as:

$$w_i^{\text{h}}(k) = \frac{w_i^{\text{kt}}(k)}{\|w_i^{\text{kt}}(k)\|} + \frac{p_i}{\|p_i\|} + \frac{w_i}{\|w_i\|} \quad (7)$$

As class representations can be viewed as class centroids, $w_i^{\text{h}}(k)$ can be interpreted as the barycentric position between our multiple estimated centres (learned classifier, class prototypes and knowledge transfer component). The process is illustrated in Figure 3. We highlight one more time that this stage does not involve any additional training procedure, with classifiers and model weights obtained from model $\mathcal{M}$ after standard training.

Ensembling strategy. Transferring knowledge from common to rare classes can significantly improve the discriminative power of few-shot classifiers. However, it can also reduce the positive biases inherent to trained classifier and reduce accuracy on common classes. Our knowledge

Class prototypes

$w_j \in W(k) = \{w_l | l \in 1 \dots N, \text{ if } n_l > k\}$

Learned classifiier

subsample classifiers
from common classes

$\alpha_{ij} = \text{softmax}\left(\frac{p_i}{\|P_i\|} \cdot \frac{w_j}{\|w_j\|}^T \cdot \tau\right)$

$w_i^{kt}(k) = \sum_j \alpha_{ij} \frac{w_j}{\|w_j\|}$

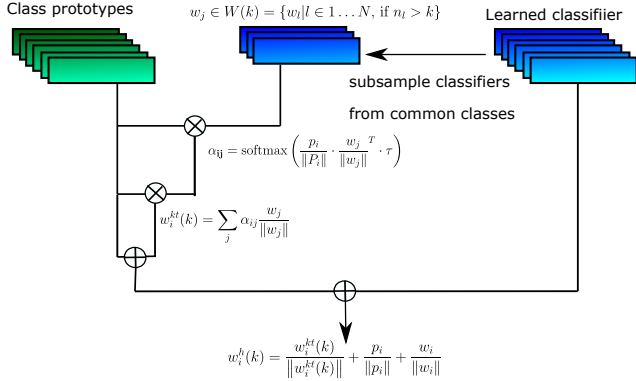$w_i^h(k) = \frac{w_i^{kt}(k)}{\|w_i^{kt}(k)\|} + \frac{p_i}{\|p_i\|} + \frac{w_i}{\|w_i\|}$

Figure 3. Illustration of the knowledge transfer classifier construction. See text for details.

transfer strategy affords flexibility on which class group classifiers are optimised for. Indeed, setting k in Equation 7 at a low value will yield classifiers dedicated to enhancing few-shot classes. Increasing this value will shift the focus to medium shot classes, as knowledge is transferred only from the most common classes. We can therefore create multiple prototype/cosine hybrid classifiers designed to achieve maximum performance on a specific class group. In light of this observation, we propose an ensembling strategy that combines predictions achieved with different classifiers optimised for different class groups:

$$q(y = i|\mathbf{x}) = \frac{1}{N_k} \sum_k q(y = i|\mathbf{x}, \theta, W^h(k)) \qquad (8)$$

Where N_k is the number of ensembled classifiers. A straightforward option is to choose the values of k so as to build one classifier per class group: for example many-, medium- and few-shot.

We highlight that we train a *single classifier*, and consequently generate a set of new classifiers $W^h(k)$ (varying k values) from the *same set* of learnt, fixed classifier weights and prototypes. We therefore gain ensemble-esque benefits, via diverse classifiers, at almost no additional cost.

Continual adaptation. Last but not least, a noticeable advantage of our strategy, which does not rely on few-shot classification weights for knowledge transfer, is the ability to extend classifiers to new, previously unseen classes seamlessly. Indeed, this can be achieved with no additional training by computing new class prototypes, and has the potential to provide a strong, practical baseline. To achieve this, we propose a continual classifier estimation that does not rely on availability of few-shot classifiers as:

$$w_i^{hc}(k) = \frac{w_i^{kt}(k)}{\|w_i^{kt}(k)\|} + \frac{p_i}{\|p_i\|} \qquad (9)$$

We will show empirically that this approach can achieve competitive performance, and allows to integrate previously unseen classes seamlessly.

4. Results

4.1. Experimental setup

We evaluate our approach on two standard long-tail recognition benchmarks: **ImageNet-LT** and **Places-LT** [17]. In this section, we provide information on the dataset and our implementation details. Common to both datasets, we set the temperature parameter for our knowledge transfer step to $\tau = 10$. Unless specified otherwise, our models are trained with square-root sampling, and with a cosine classifier with learnable scale parameter initialised at 16 (parameter value as used in [17, 25]). We provide an analysis of these two training components in our experiments. For ImageNet, we train a ResNext50 following [14]. For the Places dataset, we follow standard practice and use weights pre-trained on the full ImageNet dataset to initialise our model. We consider both supervised pre-training (standard) and unsupervised weights obtained using the SimCLR method [5], as well as ResNet101 and ResNet152 architectures. Detailed description of datasets and training parameters are provided in the supplementary material.

Following standard evaluation protocols, we report classification accuracy over the entire test datasets, as well as over class groups defined as "Many-shot" ($n \geq 100$), "Medium-shot" ($20 < n < 100$) and "Few-shot" ($n \leq 20$).

4.2. Comparison to state of the art methods

It has been shown that backbone capacity and additional training losses can substantially improve backbone quality and consequently classification performance [7, 14, 23]. As our approach is independent from the backbone training process, we train our model using the setting that matches most pre-existing approaches. We use a standard cross entropy setting and evaluate our method with respect to state of the art approaches using the same backbone architecture and training regime for fair comparison. Notably, we highlight methods leveraging normalised classifiers, as these tend to yield superior backbones (see Section 4.4).

We provide detailed results on the ImageNet dataset in Table 1, comparing individual knowledge transfer classifiers ($k = 20$, $k = 100$) with an ensemble strategy ($k = 0$, $k = 20$, $k = 100$). We can see that the ensembling strategy achieves the most robust performance, increasing or matching state of the art accuracy in all class groups. We achieve the most substantial performance gain on the few-shot classes, improving accuracy by at least 5% to 16% with respect to competing methods. As individual classifiers are tailored for a specific class group, combining them via ensembling strategy allows to maintain robust performance on

Table 1. Classification accuracies on ImageNet-LT. All methods use a ResNext50 backbone. * models trained with a normalised classifier.

Method	Classifier type	Many-shot	Medium shot	Few shot	Total
Ours*	Cosine classifier	**68.5**	45.6	20.6	50.9
	Prototypes	63.5	49.1	27.4	51.6
	Prototypes + classifier	67.0	48.7	25.2	52.4
	$w^{\mathrm{h}}(20)$	64	44.4	**48.2**	52.7
	$w^{\mathrm{h}}(100)$	50.6	**57.8**	36.8	51.9
	ensemble$\big(w^{\mathrm{h}}(0), w^{\mathrm{h}}(20), w^{\mathrm{h}}(100)\big)$	63.2	52.1	36.9	**54.2**
LWS [14]		60.2	47.2	30.3	49.9
BALMS [21]		62.2	48.8	29.8	51.4
Causal norm* [25]		62.7	48.8	31.6	51.8
RSG* [27]		63.2	48.2	32.3	51.8
LADE [12]		62.3	49.3	31.2	51.9
Disalign* [30]		62.7	52.1	31.4	53.4

all class groups while significantly improving the performance on rare classes.

We additionally provide results for the ensemble strategy ($k = 0$, $k = 20$, $k = 100$) the Places dataset in Table 3 (more detailed results are provided in the supplementary material). We provide results on ResNet101 with different pre-trained weights, as well as ResNet152 with supervised initialisation[1] for fair comparison with state of the art methods. Our first observation is that we achieve stronger results on ResNet101 despite the reduced capacity. We hypothesize that stronger overfitting issues (notably due to the square-root sampling process) can be encountered with the larger capacity backbone, and that unsupervised pre-training achieves stronger performance due to its higher compatibility with our cosine classifier. Nonetheless, we note that we achieve overall accuracy on par with state of the art models (ResNet152), and outperform all methods using our ResNet101 initialised with an unsupervised backbone. Similarly to the ImageNet dataset, we obtain the largest performance gains on few-shot classes, outperforming state of the art methods on few-shot classes by at least 2.8% in all settings.

4.3. Continual adaptation

Next, we aim to evaluate our model's ability to achieve good recognition performance in settings where some classes have never been seen before. To this end, we train a model on the ImageNet-LT dataset, removing all few-shot classes ($n \leq 20$, 146 classes) from the training set. We refer to this model as our continual model in future references. We evaluate the continual classifier w_i^{hc} as introduced in Equation 9 on both a model trained on the full dataset, and our continual model. We report results in Table 2, considering classifier accuracy, prototype accuracy. w_i^{hc} for $k = \{20, 100\}$ and an ensemble of both classifiers with class prototypes. We can see that our continual model still

achieves a robust performance despite the fact that the backbone wasn't trained to separate common from rare classes. We note that we still achieve very good recognition performance on rare classes, outperforming almost all state of the art methods on this class group (see Table 1) even though they were never seen during model training.

4.4. Backbone training

In this section, we focus on the standard learning setting using a cross-entropy loss, and aim to evaluate the impact of different training settings on backbone quality. More specifically, we evaluate the influence of data sampling strategies and type of classifier. Our main focus is to evaluate the quality of trained backbones with respect to prototype estimation and few-shot class representation.

In order to identify the preferred training strategy, we evaluate models trained using **a)** trained classifiers, **b)** prototypes computed using the training set (training prototypes), **c)** prototypes computed using the validation set (validation prototypes). We measure test accuracy for classification over all classes, as well as individual class groups only, aiming to evaluate backbones with regards to class representation quality, generalisation ability, and common class bias. Lastly, in order to evaluate whether oversampling led to important memorising, we measure accuracy on the training set, using validation prototypes as classifiers. Models are trained on the ImageNet-LT dataset using the same parameters and architectures as described in our experimental setup section unless specified otherwise.

Type of classifier. Relying on a cosine classifier is a key condition in our approach to achieve successful knowledge transfer between classifiers and prototypes. Here, we compare the performance of models trained using cosine classifiers with the popular linear dot product classifier.

We provide classifier-based accuracies in Table 4a, and prototype-based accuracies in Figure 4. We can see that usage of the cosine classifier significantly increases classification accuracy for all class groups. Prototype based accu-

---

[1]Deeper architectures such as ResNet152 are not commonly used in self-supervised learning settings.

Table 2. Classification accuracies on the ImageNet-LT dataset in the continual setting.

Method	Classifier type	Many-shot	Medium shot	Few shot	Total
Full training	Cosine classifier	**68.5**	45.6	20.6	50.9
	Prototypes	63.5	49.1	27.4	51.6
	$w^{\text{hc}}(20)$	61.7	44.7	**48.3**	51.8
	$w^{\text{hc}}(100)$	51.1	**55.3**	37.2	51.0
	ensemble$(prototypes, w^{\text{hc}}(20), w^{\text{hc}}(100))$	61.4	51.1	38.3	**53.2**
Few-shot excluded	Cosine classifier	**66.0**	40.8	N/A	N/A
	Prototypes	61.8	43.5	26.2	48.1
	$w^{\text{hc}}(20)$	57.3	37.7	**39.3**	45.6
	$w^{\text{hc}}(100)$	50	**50.2**	25.8	46.6
	ensemble$(prototypes, w^{\text{hc}}(20), w^{\text{hc}}(100))$	58.9	45.0	32.2	**48.6**

Table 3. Classification accuracies on the Places-LT dataset. * refers to models trained with a normalised classifier.

Method	Many-shot	Medium shot	Few shot	Total
ResNet101 backbone				
Ours* - supervised init	40.8	40.1	34.9	39.3
Ours* - unsupervised init	**41.6**	41.4	35.1	**40.2**
Disalign R101 [30]	39.1	**42.0**	29.1	38.5
ResNet152 backbone				
Ours* - supervised init	39.7	41.0	**34.9**	**39.2**
LWS [14]	40.6	39.1	28.6	37.6
BALMS [21]	42.0	39.3	30.5	38.6
LADE [12]	**42.8**	39	31.2	38.8
RSG* [27]	41.9	41.4	32.0	**39.3**
Disalign R152 [30]	40.4	**42.4**	30.1	**39.3**

Table 4. Classification accuracy using learned classifiers to evaluate the influence of a) classifier type, and b) sampling strategies.

(a) Classifier type

Classifier	Many-shot	Medium-shot	Few-shot	total
Dot product	65.9	37.5	7.5	44.4
Cosine	**69.2**	**43.0**	**15.4**	**49.2**

(b) Sampling method

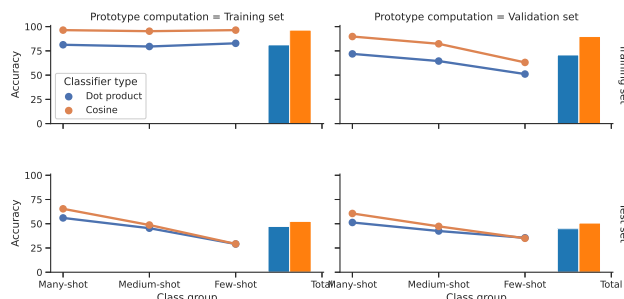| Sampling	Many-shot	Medium-shot	Few-shot	total
Uniform	**69.2**	43.0	15.4	49.2
Square root	68.5	**45.6**	**20.6**	**50.9**
Class aware	64.1	38	14.2	44.7

Figure 4. Influence of classifier type on class separability. Prototype-based prediction accuracy is computed with respect to the three different class groups on the training and test sets, with prototypes computed on training and validation sets.

racies show that we consistently achieve better class separability, with the exception of few-shot classes where performance is equivalent. This suggests that cosine classifiers facilitate model training, achieving better accuracy and separability faster. The use of a distance based classifier also appears to strongly benefit few-shot representations. Normalisation based classifiers, including cosine classifiers, were considered to be beneficial to reduce momentum induced bias in [25], and were also observed to yield consistently better performance in [30].

Sampling strategy. A simple way of rebalancing long-tail training datasets is to prevent the dominance of common classes during the training process. This can be achieved via sampling strategies when constructing training batches, with the aim to sample rare classes more often. Finding the right balance is not straightforward however, as over-sampling rare classes can lead to overfitting and memorising. Different batch sampling strategies were proposed in [14], and evaluated with respect to different long-tail approaches. It was observed that oversampling rare classes lead to stronger classifiers after standard training. However, this performance gain was lost after applying classifier rebalancing strategies, with sample uniform sampling methods achieving the best performance.

Here, we aim to further evaluate backbone encoders trained using different sampling strategies, and determine the optimal sampling strategy for our task. In addition, we aim to verify that previous observations still hold for cosine classifiers. We seek a combination of high quality encoder backbone and class prototypes. We evaluate backbones trained using cosine classifiers with identical parameters and three sampling strategies: a) **Sample uniform**: standard training setting, where images are uniformly sampled from the training database; b) **Class uniform** [4]: uni-

Table 5. Influence of data sampling strategies on class group prototype accuracy. Classification within each class group only.

Prototype source	Sampling	Few-shot	Many-shot
	Uniform	60.8	**70.6**
Training set	Square root	**61.9**	69.9
	Class aware	52.1	65
	Uniform	63.3	**69.7**
Validation set	Square root	**67.9**	68.1
	Class aware	62	63

formly sampling categories, then selecting a fixed number of samples per class (common practice is 4 samples). This results in substantially oversampling rare classes; and c) **Square root** [18]: the probability of sampling images from class j is $q_j = \frac{n_j^{1/2}}{\sum_{i=1}^{N} n_i^{1/2}}$. This increases the likelihood of sampling from rarer classes but does not dramatically alter the data distribution.

Classifier accuracy is reported in Table 4b, prototype classification accuracies are provided in Figure 5, and class group level accuracy in in Table 5. Our first observation refers to the class uniform sampling approach, which achieves the poorest performance, both with regards to classifiers and prototypes. Our experiments in Figure 5 (top left) suggest overfitting, with performance on the training set being close to 100% for both few and medium shot classes. The poorer performance compared to dot product classifiers [14] could be linked to cosine classifiers enabling faster training. Second, square root sampling achieves the highest classification accuracy, with performance slightly reduced for many-shot classes. While square-root and uniform sampling have comparable performance on training prototypes, our experiments on validation prototypes (right column in Figure 5) suggest that square root models provide better separability for medium and few-shot classes overall, which is crucial for our knowledge transfer process. This is additionally supported by Table 5, which shows, when considering classification within few-shot classes only, that square-root sampling achieves stronger accuracy.

These observations highlight that sampling strategies can have a substantial impact on model performance, and that square-root sampling can provide better representations for our objective. We note however that the optimal sampling method could differ depending on data distribution, and recommend square-root sampling for datasets exhibiting a similar distribution to the ImageNet long-tail dataset.

5. Conclusion

In this work, we introduce a novel approach for long-tail recognition focused on the tail's few-shot problem. We introduce a knowledge transfer strategy that recomposes learned classification features from data-rich common
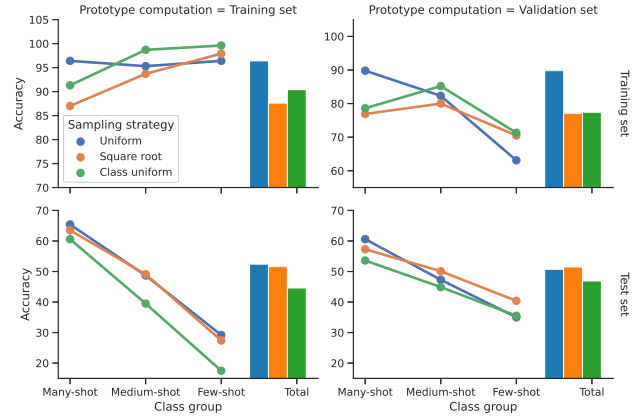

Figure 5. Influence of data sampling strategies on class separability. Prototype-based prediction accuracy is computed with respect to the three different class groups on the training and test sets, with prototypes computed on training and validation sets.

classes to improve rare class representations in a training-free manner. Our experiments on two standard benchmarks show that we achieve significant performance gains on rare classes, while maintaining strong performance on common classes. We further demonstrate our model's ability to integrate new, previously unseen classes with strong recognition accuracy, and provide an analysis of our model training strategy so as to facilitate knowledge transfer. Nonetheless, our approach is not tied to a specific backbone training approach, and further benefits could be obtained by combining our strategy with backbone training methods comprising auxiliary losses that optimise for class separability [7, 23].

Limitations. One limitation of the knowledge transfer method, described in Sec. 3, is that performance on common classes may suffer. We mitigate this effect via ensembling, but additional option may be considered, such as dual classifiers or separate memory vectors. Prototypes, as class average representations, can lack the precision that learned classifiers provide. We therefore plan in the future to explore richer prototype representations *e.g.* distribution-based or multiple prototypes per class [1]. Currently, our idea relies on a training-free strategy, and cannot fully leverage new data for backbones in e.g. continual scenarios. Training-based extensions can also be considered in future work. Finally, our approach relies on the assumption that there exists classes with similar visual properties. Our knowledge transfer approach might not be successful in settings where this is not the case. As visual similarities are currently estimated via feature comparisons, introducing additional structured, prior knowledge (*e.g.* knowledge bases) may further inform the crucial decisions of both when and if knowledge transfer should occur.

References

[1] Kelsey Allen, Evan Shelhamer, Hanul Shin, and Joshua Tenenbaum. Infinite mixture prototypes for few-shot learning. In *International Conference on Machine Learning*, pages 232–241. PMLR, 2019. 8

[2] Yuri Sousa Aurelio, Gustavo Matheus de Almeida, Cristiano Leite de Castro, and Antonio Padua Braga. Learning from imbalanced data sets with weighted cross-entropy function. *Neural processing letters*, 50(2):1937–1949, 2019. 1, 3

[3] Jiarui Cai, Yizhou Wang, and Jenq-Neng Hwang. Ace: Ally complementary experts for solving long-tailed recognition in one-shot. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 112–121, 2021. 3

[4] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority oversampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002. 2, 7

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020. 5

[6] Jiequan Cui, Shu Liu, Zhuotao Tian, Zhisheng Zhong, and Jiaya Jia. Reslt: Residual learning for long-tailed recognition. *arXiv preprint arXiv:2101.10633*, 2021. 1, 3

[7] Jiequan Cui, Zhisheng Zhong, Shu Liu, Bei Yu, and Jiaya Jia. Parametric contrastive learning, 2021. 1, 3, 5, 8

[8] Chris Drummond, Robert C Holte, et al. C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In *Workshop on learning from imbalanced datasets II*, volume 11, pages 1–8. Citeseer, 2003. 2

[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135. PMLR, 2017. 3

[10] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4367–4375, 2018. 2, 3, 4

[11] Alex Graves, Greg Wayne, and Ivo Danihelka. Neural turing machines. *arXiv preprint arXiv:1410.5401*, 2014. 4

[12] Youngkyu Hong, Seungju Han, Kwanghee Choi, Seokjun Seo, Beomsu Kim, and Buru Chang. Disentangling label distribution for long-tailed visual recognition. *arXiv preprint arXiv:2012.00321*, 2020. 1, 3, 4, 6, 7

[13] Ahmet Iscen, André Araujo, Boqing Gong, and Cordelia Schmid. Class-balanced distillation for long-tailed visual recognition. *BMVC*, 2021. 2, 3

[14] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. *arXiv preprint arXiv:1910.09217*, 2019. 1, 2, 5, 6, 7, 8

[15] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10657–10665, 2019. 3

[16] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 3

[17] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2537–2546, 2019. 1, 2, 5

[18] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013. 2, 8

[19] Hang Qi, Matthew Brown, and David G Lowe. Low-shot learning with imprinted weights. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5822–5830, 2018. 2, 3

[20] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. *ICLR*, 2017. 2, 3

[21] Jiawei Ren, Cunjun Yu, Shunan Sheng, Xiao Ma, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Balanced meta-softmax for long-tailed visual recognition. *arXiv preprint arXiv:2007.10740*, 2020. 1, 2, 3, 4, 6, 7

[22] Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. *arXiv preprint arXiv:1807.05960*, 2018. 3

[23] Dvir Samuel and Gal Chechik. Distributional robustness loss for long-tail learning. *arXiv preprint arXiv:2104.03066*, 2021. 3, 5, 8

[24] Jake Snell, Kevin Swersky, and Richard S Zemel. Prototypical networks for few-shot learning. *arXiv preprint arXiv:1703.05175*, 2017. 2, 3, 4

[25] Kaihua Tang, Jianqiang Huang, and Hanwang Zhang. Long-tailed classification by keeping the good and removing the bad momentum causal effect. In *NeurIPS*, 2020. 1, 3, 5, 6, 7

[26] Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, et al. Meta-dataset: A dataset of datasets for learning to learn from few examples. In *International Conference on Learning Representations*, 2019. 2, 3

[27] Jianfeng Wang, Thomas Lukasiewicz, Xiaolin Hu, Jianfei Cai, and Zhenghua Xu. Rsg: A simple but effective module for learning imbalanced datasets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3784–3793, 2021. 1, 2, 6, 7

[28] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR, 2020. 4

[29] Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella Yu. Long-tailed recognition by routing diverse distribution-aware experts. In *International Conference on Learning Representations*, 2021. 1, 3

[30] Songyang Zhang, Zeming Li, Shipeng Yan, Xuming He, and Jian Sun. Distribution alignment: A unified framework for long-tail visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2361–2370, 2021. 3, 6, 7

[31] Yifan Zhang, Bryan Hooi, Lanqing Hong, and Jiashi Feng. Test-agnostic long-tailed recognition by test-time aggregating diverse experts with self-supervision. *arXiv*, 2021. 1, 3

[32] Zhisheng Zhong, Jiequan Cui, Shu Liu, and Jiaya Jia. Improving calibration for long-tailed recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16489–16498, 2021. 3

[33] Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9719–9728, 2020. 1, 2