CROSS-DOMAIN FEW-SHOT LEARNING BY REPRESENTATION FUSION

Anonymous authors
Paper under double-blind review

ABSTRACT

In order to quickly adapt to new data, few-shot learning aims at learning from few examples, often by using already acquired knowledge. The new data often differs from the previously seen data due to a domain shift, that is, a change of the input-target distribution. While several methods perform well on small domain shifts like new target classes with similar inputs, larger domain shifts are still challenging. Large domain shifts may result in abstract concepts that are not shared between the original and the new domain. However, low-level concepts like edges in images might still be shared and useful. For cross-domain few-shot learning, we suggest representation fusion to unify different abstraction levels of a deep neural network into one representation. We propose Cross-domain Hebbian Ensemble Few-shot learning (CHEF), which achieves representation fusion by an ensemble of Hebbian learners acting on different layers of a deep neural network that was trained on the original domain. On the few-shot datasets *mini*Imagenet and *tiered*Imagenet, where the domain shift is small, CHEF is competitive with state-of-the-art methods. On cross-domain few-shot benchmark challenges with larger domain shifts, CHEF establishes novel state-of-the-art results in all categories. We further apply CHEF on a real-world cross-domain application in drug discovery. We consider a domain shift from bioactive molecules to environmental chemicals and drugs with twelve associated toxicity prediction tasks. On these tasks, that are highly relevant for computational drug discovery, CHEF significantly outperforms all its competitors.

1 INTRODUCTION

Currently, deep learning is criticized because it is data hungry, has limited capacity for transfer, insufficiently integrates prior knowledge, and presumes a largely stable world (Marcus, 2018). In particular, these problems appear after a domain shift, that is, a change of the input-target distribution. A domain shift forces deep learning models to adapt. The goal is to exploit models that were trained on the typically rich original data for solving tasks from the new domain with much less data. Examples for domain shifts are new users or customers, new products and product lines, new diseases (e.g. adapting from SARS to COVID19), new images from another field (e.g. from cats to dogs or from cats to bicycles), new social behaviors after societal change (e.g. introduction of cell phones, pandemic), self-driving cars in new cities or countries (e.g. from European countries to Arabic countries), and robot manipulation of new objects.

Domain shifts are often tackled by meta-learning (Schmidhuber, 1987; Bengio et al., 1990; Hochreiter et al., 2001), since it exploits already acquired knowledge to adapt to new data. One prominent application of meta-learning dealing with domain shifts is few-shot learning, since, typically, from the new domain much less data is available than from the original domain. Meta-learning methods perform well on small domain shifts like new target classes with similar inputs. However, larger domain shifts are still challenging for current approaches. Large domain shifts lead to inputs, which are considerably different from the original inputs and possess different abstract concepts. Nonetheless, low level concepts are often still shared between the inputs of the original domain and the inputs of the new domain. For images, such shared low level concepts can be edges, textures, small shapes, etc. One way of obtaining low level concepts is to train a new deep learning model from scratch, where the new data is merged with the original data. However, although models of the original domain are often available, the original data, which the models were trained on, often are not. This might have several reasons, e.g. the data owner does no longer grant access to the data, General

Data Protection Regulation (GDPR) does no longer allow access to the data, IP restrictions prevent access to the data, sensitive data items must not be touched anymore (e.g. phase III drug candidates), or data is difficult to extract again. We therefore suggest to effectively exploit original data models directly by accessing not only high level but also low level abstractions. In this context, we propose a cross-domain few-shot learning method extracting information from different levels of abstraction in a deep neural network.

Representation fusion. Deep Learning constructs neural network models that represent the data at multiple levels of abstraction (LeCun et al., 2015). We introduce *representation fusion*, which is the concept of unifying and merging information from different levels of abstraction. Representation fusion uses a fast and adaptive system for detecting relevant information at different abstraction levels of a deep neural network, which we will show allows solving versatile and complex cross-domain tasks.

CHEF. We propose cross-domain ensemble few-shot learning (CHEF) to tackle the problem of domain shifts. CHEF consists of representation fusion by an ensemble of Hebbian learners that are built upon a trained network. CHEF naturally addresses the demands of real-world applications such as drug discovery (Sturm et al., 2020; Mayr et al., 2018). Furthermore, since CHEF only builds on representation fusion, it can adapt to new characteristics of tasks like unbalanced data sets, classes with few examples, change of the measurement method, new measurements in unseen ranges, new kind of labeling errors, and more. The usage of simple Hebbian learners allows the application of CHEF without needing to backpropagate information through the backbone network.

The main contributions of this paper are:

- We introduce representation fusion as the concept of unifying and merging information from different layers of abstraction.
- We introduce CHEF[1] as our new cross-domain few-shot learning method that builds on representation fusion. We show that using different layers of abstraction allows one to successfully tackle various few-shot learning tasks across a wide range of different domains. CHEF does not need to backpropagate information through the backbone network.
- We apply CHEF to various cross-domain few-shot tasks and obtain several state-of-the-art results. We further apply CHEF to cross-domain real-world applications from drug discovery, where we outperform all competitors.

Related work. Representation fusion builds on learning a meaningful representation (Bengio et al., 2013; Girshick et al., 2014) at multiple levels of abstraction (LeCun et al., 2015; Schmidhuber, 2015). The concept of using representations from different layers of abstraction has been used in CNN architectures (LeCun et al., 1998) such as Huang et al. (2017); Rumetshofer et al. (2018); Hofmarcher et al. (2019), in CNNs for semantic segmentation in the form of multi-scale context pooling (Yu & Koltun, 2015; Chen et al., 2018), and in the form of context capturing and symmetric upsampling (Ronneberger et al., 2015). Work on domain shifts discusses the problem that new inputs are considerably different from the original inputs (Kouw & Loog, 2019; Wouter, 2018; Webb et al., 2018; Gama et al., 2014; Widmer & Kubat, 1996). Domain adaptation (Pan & Yang, 2009; Ben-David et al., 2010) overcomes this problem by e.g. reweighting the original samples (Jiayuan et al., 2007) or learning a classifier in the new domain. Domain adaptation where only few data is available in the new domain (Ben-David et al., 2010; Lu et al., 2020) is called cross-domain few-shot learning (Guo et al., 2019; Lu et al., 2020; Tseng et al., 2020), which is an instance of the general few-shot learning setting (Fei-Fei et al., 2006). Few-shot learning can be roughly divided into three approaches (Lu et al., 2020; Hospedales et al., 2020): (i) augmentation, (ii) metric learning, and (iii) meta-learning. For (i), where the idea is to learn an augmentation to produce more than the few samples available, supervised (Dixit et al., 2017; Kwitt et al., 2016) and unsupervised (Hariharan & Girshick, 2017; Pahde et al., 2019; Gao et al., 2018) methods are considered. For (ii), approaches aim to learn a pairwise similarity metric under which similar samples obtain high similarity scores (Koch et al., 2015; Ye & Guo, 2018; Hertz et al., 2006). For (iii), methods comprise *embedding and nearest-neighbor* approaches (Snell et al., 2017b; Sung et al., 2018; Vinyals et al., 2016), *finetuning* approaches (Finn et al., 2017; Rajeswaran et al., 2019; Ravi & Larochelle, 2017; Andrychowicz et al., 2016), and *parametrized* approaches (Gidaris & Komodakis, 2018; Ye et al., 2020; Lee et al., 2019;

[1] Our implementation is available at github.com/tomte812/chef.

Yoon et al., 2019; Mishra et al., 2018; Hou et al., 2019; Rusu et al., 2018). Few-shot classification under domain shifts for metric-based methods has been discussed in Tseng et al. (2020). Ensemble methods for few-shot learning have been applied in Dvornik et al. (2019), where an ensemble of distance-based classifiers is designed from different networks. In contrast, our method builds an ensemble of different layers from the same network. Hebbian learning as part of a few-shot learning method has been implemented in Munkhdalai & Trischler (2018), where fast weights that are used for binding labels to representations are generated by a Hebbian learning rule.

2 Cross-domain few-shot learning

Domain shifts. We assume to have data $(\boldsymbol{x}, \boldsymbol{y})$, where $\boldsymbol{x} \in \boldsymbol{X}$ is the input data and $\boldsymbol{y} \in \boldsymbol{Y}$ is the target data. A domain is a distribution p over $\boldsymbol{X} \times \boldsymbol{Y}$ assigning each pair $(\boldsymbol{x}, \boldsymbol{y})$ a probability $p(\boldsymbol{x}, \boldsymbol{y})$. A domain shift is a change from $p(\boldsymbol{x}, \boldsymbol{y})$ to $\tilde{p}(\boldsymbol{x}, \boldsymbol{y})$. We measure the magnitude of the domain shift by a distance $d(p, \tilde{p})$ between the distributions p and $\tilde{p}$. We consider four types of domain shifts (Kouw & Loog, 2019; Wouter, 2018; Webb et al., 2018; Gama et al., 2014; Widmer & Kubat, 1996):

- **Prior shift (small domain shift)**: $p(\boldsymbol{y})$ is changed to $\tilde{p}(\boldsymbol{y})$, while $p(\boldsymbol{x} \mid \boldsymbol{y})$ stays the same. For example, when new classes are considered (typical case in few-shot learning): $p(\boldsymbol{x}, \boldsymbol{y}) = p(\boldsymbol{y})p(\boldsymbol{x} \mid \boldsymbol{y})$ and $\tilde{p}(\boldsymbol{x}, \boldsymbol{y}) = \tilde{p}(\boldsymbol{y})p(\boldsymbol{x} \mid \boldsymbol{y})$.
- **Covariate shift (large domain shift)**: $p(\boldsymbol{x})$ is changed to $\tilde{p}(\boldsymbol{x})$, while $p(\boldsymbol{y} \mid \boldsymbol{x})$ stays the same. For example, when new inputs are considered, which occurs when going from color to grayscale images, using a new measurement device, or looking at traffic data from different continents: $p(\boldsymbol{x}, \boldsymbol{y}) = p(\boldsymbol{x})p(\boldsymbol{y} \mid \boldsymbol{x})$ and $\tilde{p}(\boldsymbol{x}, \boldsymbol{y}) = \tilde{p}(\boldsymbol{x})p(\boldsymbol{y} \mid \boldsymbol{x})$.
- **Concept shift**: $p(\boldsymbol{y} \mid \boldsymbol{x})$ is changed to $\tilde{p}(\boldsymbol{y} \mid \boldsymbol{x})$, while $p(\boldsymbol{x})$ stays the same. For example, when including new aspects changes the decision boundaries: $p(\boldsymbol{x}, \boldsymbol{y}) = p(\boldsymbol{x})p(\boldsymbol{y} \mid \boldsymbol{x})$ and $\tilde{p}(\boldsymbol{x}, \boldsymbol{y}) = p(\boldsymbol{x})\tilde{p}(\boldsymbol{y} \mid \boldsymbol{x})$.
- **General domain shift**: domain shift between $p(\boldsymbol{x}, \boldsymbol{y})$ to $\tilde{p}(\boldsymbol{x}, \boldsymbol{y})$. For example, going from Imagenet data to grayscale X-ray images (typical case in cross-domain datasets).

Domain shift for images. We consider the special case that the input $\boldsymbol{x}$ is an image. In general, domain shifts can be measured on the raw image distributions e.g. by using the $\mathcal{H}$-divergence (Ben-David et al., 2010). However, distances between raw image distributions were shown to be less meaningful in computer vision tasks than abstract representations of deep neural networks (Heusel et al., 2017; Salimans et al., 2016). We approximate the distance between the joint distributions $d(p(\boldsymbol{x}, \boldsymbol{y}), \tilde{p}(\boldsymbol{x}, \boldsymbol{y}))$ by the distance between the marginals $d(p(\boldsymbol{x}), \tilde{p}(\boldsymbol{x}))$, which is exact in the case of the covariate shift for certain choices of $d(\cdot, \cdot)$, like e.g. the Jensen-Shannon divergence. For the distance between the marginals $d(p(\boldsymbol{x}), \tilde{p}(\boldsymbol{x}))$ we use the Fréchet Inception Distance (FID; Heusel et al., 2017), which has proven reliable for measuring performance of Generative Adversarial Networks (Goodfellow et al., 2014).

Cross-domain few-shot learning. Large domain shifts lead to inputs, which are considerably different from the original inputs. As a result, the model trained on the original domain will not work anymore on the new domain. To overcome this problem, domain adaptation techniques are applied (Pan & Yang, 2009; Ben-David et al., 2010). Domain adaption can be achieved in several ways, e.g. by reweighting the original samples (Jiayuan et al., 2007). Another possibility is to learn a classifier in the new domain. Domain adaptation where in the new domain only few data is available (Ben-David et al., 2010) which can be used for learning is called cross-domain few-shot learning (Guo et al., 2019; Lu et al., 2020; Tseng et al., 2020). In an N-shot K-way few-shot learning setting, the training set (in meta learning also called one episode) consists of N samples for each of the K classes.

3 Cross-domain Hebbian Ensemble Few-shot learning (CHEF)

We propose a new cross-domain few-shot learning method, CHEF, that consists of an ensemble of Hebbian learners built on representation fusion. Figure 1 sketches our CHEF approach. In principle, any learning algorithm can be used for representation fusion. We choose a Hebbian learning rule because it is simple and fast while being robust and reliable.

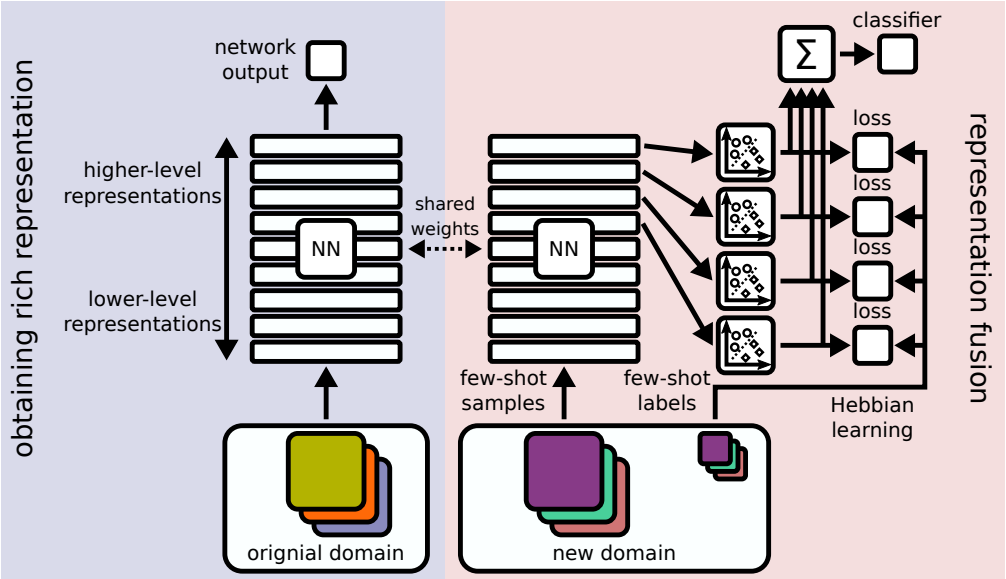

Figure 1: Working principle of CHEF. An ensemble of Hebbian learners is applied to the upper layers of a trained neural network. Distilling information from different layers of abstraction is called representation fusion. Each Hebbian learner is iteratively optimized and the results are combined.

Hebbian few-shot learning built on representation fusion. CHEF builds its ensemble of Hebbian learners using representation fusion. Deep learning models (LeCun et al., 2015) provide hierarchical representations that allow to fuse information from different layers of abstraction. In contrast to many other methods, CHEF does not require backpropagation of error signals through the entire backbone network. Only the parameters of the Hebbian learners that are obtained for the uppermost layers need adjustment. This makes CHEF extremely fast and versatile.

Obtaining one Hebbian Learner. We consider an N-shot K-way few-shot learning setting. Let $\boldsymbol{z}_i \in \mathbb{R}^D$ be a feature vector obtained from activating a pre-trained backbone network with a sample $\boldsymbol{x}_i$ up to a certain layer, where D is the number of units in that layer. We combine the NK feature vectors into a matrix $\boldsymbol{Z} \in \mathbb{R}^{NK \times D}$ and initialize a weight matrix $\boldsymbol{W} \in \mathbb{R}^{K \times D}$. In accordance with Hebb (2005); Frégnac (2002), we use the Hebbian learning rule

$$\boldsymbol{W} \leftarrow \boldsymbol{W} - \alpha \boldsymbol{V}^\top \boldsymbol{Z} \tag{1}$$

for a given number of steps, where α is a Hebbian learning rate and $\boldsymbol{V} \in \mathbb{R}^{NK \times K}$ is the matrix of postsynaptic responses $\boldsymbol{v}_i$. We design the number of steps for which to run equation 1 as a hyperparameter of our method. Given a loss function $\mathcal{L}(\cdot, \cdot)$ and few-shot labels $\boldsymbol{y}_i$, we choose the postsynaptic response

$$\boldsymbol{v}_i = \nabla_{\hat{\boldsymbol{y}}} \mathcal{L}(\boldsymbol{y}_i, \hat{\boldsymbol{y}})|_{\hat{\boldsymbol{y}} = \boldsymbol{W}\boldsymbol{z}_i}. \tag{2}$$

We initialize the weight matrix $\boldsymbol{W}$ with zeros. In principle, any other initialization is possible. However, the learning rule is designed for rather strong updates and, therefore, the initialization scheme is of minor importance.

Combining several Hebbian Learners. The closer a layer is to the network output the more specific are its features. Conversely, the closer a layer is to the input of the network, the more general are the features. In cross-domain few-shot learning, it is not a priori clear how specific or general the features should be because this depends on how close the target domain is to the training domain. Therefore, we design our few-shot learning algorithm such that it can flexibly choose the specificity of the features depending on the current episode. We achieve this by representation fusion, where the Hebbian learning rule is applied to several layers at different levels of the backbone network in parallel. This yields a separate prediction for each level of abstraction. The final classification result is then obtained from the sum of logits arising from the respective Hebbian learners. A schematic view of CHEF is shown in Alg. 1.

Algorithm 1 CHEF algorithm. The data matrix $\boldsymbol{X}$ consists of input vectors $\boldsymbol{x}_i$ and the label matrix $\boldsymbol{Y}$ consists of the corresponding label vectors $\boldsymbol{y}_i$. The function BB activates the backbone network up to a certain layer specified by an index l. L is the set of indices specifying the layers used in the ensemble, $\mathcal{L}$ is the loss function of the few-shot learning task at hand. The function HEBBRULE yields a weight matrix $\boldsymbol{W}$ that maps the feature vectors in $\boldsymbol{Z}$ to vectors of length K, which are then used for K-fold classification. α is the Hebbian learning rate.

Require: $\alpha, I, \text{SOFTMAX}(\cdot), \mathcal{L}(\cdot,\cdot), \text{BB}(\cdot,\cdot)$
  function HEBBRULE($\boldsymbol{X}, \boldsymbol{Y}, l$)
    $\boldsymbol{W} \leftarrow \boldsymbol{0}, \boldsymbol{Z} \leftarrow \text{BB}(\boldsymbol{X}, l)$
    for $i \in \{1, \ldots, I\}$ **do**
      $\boldsymbol{V} \leftarrow \nabla_{(\boldsymbol{Z}\boldsymbol{W}^\top)} \mathcal{L}(\boldsymbol{Y}, \text{SOFTMAX}(\boldsymbol{Z}\boldsymbol{W}^\top))$
      $\boldsymbol{W} \leftarrow \boldsymbol{W} - \alpha \boldsymbol{V}^\top \boldsymbol{Z}$
    end for
    return $\boldsymbol{W}$
  end function
  function ENSEMBLE($\boldsymbol{X}, \boldsymbol{Y}, L$)
    $\boldsymbol{U} = \sum_{l \in L} \text{BB}(\boldsymbol{X}, l)\, \text{HEBBRULE}(\boldsymbol{X}, \boldsymbol{Y}, l)^\top$
    return SOFTMAX($\boldsymbol{U}$)
  end function

4 EXPERIMENTS

We apply CHEF to four cross-domain few-shot challenges, where we obtain state-of-the-art results in all categories. The four cross-domain few-shot challenges are characterized by domain shifts of different size, which we measure using the Fréchet-Inception-Distance (FID). We conduct ablation studies showing the influence of the different layer representations on the results. Further, we test CHEF on two standardized image-based few-shot classification benchmark datasets established in the field, which are characterized by a prior domain shift: *mini*Imagenet (Vinyals et al., 2016) and *tiered*Imagenet (Ren et al., 2018). Finally, we illustrate the impact of our CHEF approach on two real-world applications in the field of drug discovery, which are characterized first by a small domain shift and second by a large domain shift.

4.1 CROSS-DOMAIN FEW-SHOT LEARNING

Dataset and evaluation. The cross-domain few-shot learning challenge (Guo et al., 2019) uses *mini*Imagenet as training domain and then evaluates the trained models on four different test domains with increasing distance to the training domain: 1) CropDisease (Mohanty et al., 2016) consisting of plant disease images, 2) EuroSAT (Helber et al., 2019), a collection of satellite images, 3) ISIC2018 (Tschandl et al., 2018; Codella et al., 2019) containing dermoscopic images of skin lesions, and 4) ChestX (Wang et al., 2017) containing a set of X-ray images. For evaluation, we measure the accuracy drawing 800 tasks (five test instances per class) from the cross-domain test set. Following prior work, we focus on 5-way/5-shot, 5-way/20-shot, and 5-way/50-shot tasks. We report the average accuracy and a 95 % confidence interval across all test images and tasks.

Measuring the domain shifts via FID. In Guo et al. (2019), the four datasets of the new domain are characterized by their distance to the original domain using three criteria: whether images contain perspective distortion, the semantic content of images, and color depth. In Table 1, we provide measurements of the domain shift of these four datasets with respect to the original *mini*Imagenet dataset using the FID. The FID measurements confirm the characterization in Guo et al. (2019), except that the EuroSAT dataset is closer to the original domain than the CropDisease dataset. The difference in both FID measurements is mostly driven by the mean terms. This can be explained by the fact that the FID does not measure perspective distortion and satellite images might have a higher variety of shapes and colors than plant images.

CHEF implementation. We perform pre-training on the *mini*Imagenet dataset similar but not identical to that in Ye et al. (2020). We utilize a softmax output layer with as many units as classes are contained in the meta-training and the meta-validation sets combined. We make a validation split on the combination of these two sets for supervised learning, i.e. instead of separating whole classes

Dataset	Conceptual difference to original domain (*mini*Imagenet)	FID
CropDisease	None	257.58
EuroSAT	No perspective distortion	151.64
ISIC2018	No perspective distortion, unnatural content	294.05
ChestX	No perspective distortion, unnatural content, different color depth	312.52

Table 1: Conceptual difference and domain shift between *mini*Imagenet and the four cross-domain datasets CropDisease, EuroSAT, ISIC2018, and ChestX. The domain shift is measured using the FID.

	CropDiseases 5-way			**EuroSAT 5-way**		
Method	**5-shot**	**20-shot**	**50-shot**	**5-shot**	**20-shot**	**50-shot**
MatchingNet†	66.39 ± 0.78	76.38 ± 0.67	58.53 ± 0.73	64.45 ± 0.63	77.10 ± 0.57	54.44 ± 0.67
MatchingNet+FWT†	62.74 ± 0.90	74.90 ± 0.71	75.68 ± 0.78	56.04 ± 0.65	63.38 ± 0.69	62.75 ± 0.76
MAML†	78.05 ± 0.68	89.75 ± 0.42	-	71.70 ± 0.72	81.95 ± 0.55	-
ProtoNet†	79.72 ± 0.67	88.15 ± 0.51	90.81 ± 0.43	73.29 ± 0.71	82.27 ± 0.57	80.48 ± 0.57
ProtoNet+FWT†	72.72 ± 0.70	85.82 ± 0.51	87.17 ± 0.50	67.34 ± 0.76	75.74 ± 0.70	78.64 ± 0.57
RelationNet†	68.99 ± 0.75	80.45 ± 0.64	85.08 ± 0.53	61.31 ± 0.72	74.43 ± 0.66	74.91 ± 0.58
RelationNet+FWT†	64.91 ± 0.79	78.43 ± 0.59	81.14 ± 0.56	61.16 ± 0.70	69.40 ± 0.64	73.84 ± 0.60
MetaOpt†	68.41 ± 0.73	82.89 ± 0.54	91.76 ± 0.38	64.44 ± 0.73	79.19 ± 0.62	83.62 ± 0.58
CHEF (Ours)	$\mathbf{86.87 \pm 0.27}$	$\mathbf{94.78 \pm 0.12}$	$\mathbf{96.77 \pm 0.08}$	$\mathbf{74.15 \pm 0.27}$	$\mathbf{83.31 \pm 0.14}$	$\mathbf{86.55 \pm 0.15}$
	ISIC 5-way			**ChestX 5-way**		
Method	**5-shot**	**20-shot**	**50-shot**	**5-shot**	**20-shot**	**50-shot**
MatchingNet†	36.74 ± 0.53	45.72 ± 0.53	54.58 ± 0.65	22.40 ± 0.7	23.61 ± 0.86	22.12 ± 0.88
MatchingNet+FWT†	30.40 ± 0.48	32.01 ± 0.48	33.17 ± 0.43	21.26 ± 0.31	23.23 ± 0.37	23.01 ± 0.34
MAML†	40.13 ± 0.58	52.36 ± 0.57	-	23.48 ± 0.96	27.53 ± 0.43	-
ProtoNet†	39.57 ± 0.57	49.50 ± 0.55	51.99 ± 0.52	24.05 ± 1.01	28.21 ± 1.15	29.32 ± 1.12
ProtoNet+FWT†	38.87 ± 0.52	43.78 ± 0.47	49.84 ± 0.51	23.77 ± 0.42	26.87 ± 0.43	30.12 ± 0.46
RelationNet†	39.41 ± 0.58	41.77 ± 0.49	49.32 ± 0.51	22.96 ± 0.88	26.63 ± 0.92	28.45 ± 1.20
RelationNet+FWT†	35.54 ± 0.55	43.31 ± 0.51	46.38 ± 0.53	22.74 ± 0.40	26.75 ± 0.41	27.56 ± 0.40
MetaOpt†	36.28 ± 0.50	49.42 ± 0.60	54.80 ± 0.54	22.53 ± 0.91	25.53 ± 1.02	29.35 ± 0.99
CHEF (Ours)	$\mathbf{41.26 \pm 0.34}$	$\mathbf{54.30 \pm 0.34}$	$\mathbf{60.86 \pm 0.18}$	$\mathbf{24.72 \pm 0.14}$	$\mathbf{29.71 \pm 0.27}$	$\mathbf{31.25 \pm 0.20}$

† Results reported in Guo et al. (2019)

Table 2: Comparative results of few-shot learning methods on four proposed cross-domain few-shot challenges CropDiseases, EuroSAT, ISIC, and ChestX. The average 5-way few-shot classification accuracies ($\%$, top-1) along with 95% confidence intervals are reported on the test split of each dataset.

into the validation set (vertical split) we move a randomly selected fraction of samples of each class into the validation set (horizontal split) as it is standard in supervised learning. We evaluate CHEF using the same ResNet-10 backbone architecture as in Guo et al. (2019). For better representation fusion, we place two fully connected layers after the last convolutional layer. We perform model selection during training using the cross-entropy loss function on the horizontal data split, and perform hyperparameter selection for CHEF on the vertical data split.

Results and ablation study. CHEF achieves state-of-the-art performance in all 12 categories. Results are provided in Table 2. To further study the influence and power of the representation fusion, we use a pre-trained PyTorch (Paszke et al., 2019) ResNet-18 network. 5-way 5-shot results are reported in Fig. 2 (5-way 20-shot and 5-way 50-shot results can be found in the appendix). Results are obtained by applying our Hebbian learning rule to the logits of the output layer and to the pre-activations of the blocks 4 through 8 individually and we also examine an ensemble of them. The results are considerably better than the above reported ResNet-10 results, which presumably arises from the fact that the power of representation fusion is larger since the ResNet-18 network is pretrained on the whole Imagenet dataset. This illustrates the power of CHEF considering better feature abstractions. Another interesting insight is that for the ChestX dataset, the dataset with the largest domain shift, the lower level features gain importance. In general, the farther the domain is away from the original domain the more important are features from lower layers, i.e. features that are less specific to the original domain. Since CHEF combines features of different specificity to the training domain, it is particularly powerful in cross-domain settings.

Method	Backbone	*mini*Imagenet 5-way 1-shot	*mini*Imagenet 5-way 5-shot	*tiered*Imagenet 5-way 1-shot	*tiered*Imagenet 5-way 5-shot
MatchingNet (Vinyals et al., 2016)	Conv-4	43.56 ± 0.84	55.31 ± 0.73	-	-
Meta-LSTM (Ravi & Larochelle, 2017)	Conv-4	43.44 ± 0.77	60.60 ± 0.71	-	-
MAML (Finn et al., 2017)	Conv-4	48.70 ± 1.84	63.11 ± 0.92	$51.67 \pm 1.81^{\dagger}$	$70.30 \pm 1.75^{\dagger}$
ProtoNets (Snell et al., 2017a)	Conv-4	49.42 ± 0.78	68.20 ± 0.66	$48.58 \pm 0.87^{\dagger}$	$69.57 \pm 0.75^{\dagger}$
Reptile (Nichol et al., 2018)	Conv-4	47.07 ± 0.26	62.74 ± 0.37	$48.97 \pm 0.21^{\dagger}$	$66.47 \pm 0.21^{\dagger}$
RelationNet (Sung et al., 2018)	Conv-4	50.44 ± 0.82	65.32 ± 0.70	$54.48 \pm 0.93^{\dagger}$	$71.32 \pm 0.78^{\dagger}$
IMP (Allen et al., 2019)	Conv-4	49.60 ± 0.80	68.10 ± 0.80	-	-
FEAT (Ye et al., 2020)	Conv-4	55.15 ± 0.20	71.61 ± 0.16	-	-
Dynamic FS (Gidaris & Komodakis, 2018)	Conv-4	56.20 ± 0.86	72.81 ± 0.62	-	-
CHEF (Ours)	Conv-4	$\mathbf{57.60 \pm 0.29}$	$\mathbf{73.26 \pm 0.23}$	$\mathbf{61.10 \pm 0.21}$	$\mathbf{75.83 \pm 0.25}$
SNAIL (Mishra et al., 2018)	ResNet-12	55.71 ± 0.99	68.88 ± 0.92	-	-
TADAM (Oreshkin et al., 2018)	ResNet-12	58.50 ± 0.30	76.70 ± 0.30	-	-
MTL (Sun et al., 2019)	ResNet-12	61.20 ± 1.80	75.50 ± 0.80	-	-
VariationalFSL (Zhang et al., 2019)	ResNet-12	61.23 ± 0.26	77.69 ± 0.17	-	-
TapNet (Yoon et al., 2019)	ResNet-12	61.65 ± 0.15	76.36 ± 0.10	63.08 ± 0.15	80.26 ± 0.12
MetaOptNet (Lee et al., 2019)	ResNet-12	62.64 ± 0.61	78.63 ± 0.46	65.81 ± 0.74	81.75 ± 0.53
CTM (Li et al., 2019)	ResNet-12	62.05 ± 0.55	78.63 ± 0.06	64.78 ± 0.11	81.05 ± 0.52
CAN (Hou et al., 2019)	ResNet-12	63.85 ± 0.48	79.44 ± 0.34	69.89 ± 0.51	84.23 ± 0.37
FEAT (Ye et al., 2020)	ResNet-12	$\mathbf{66.78 \pm 0.20}$	$\mathbf{82.05 \pm 0.14}$	$\mathbf{70.80 \pm 0.23}$	84.79 ± 0.16
Dynamic FS (Gidaris & Komodakis, 2018)	ResNet-12	55.45 ± 0.89	70.13 ± 0.68	-	-
CHEF (Ours)	ResNet-12	64.11 ± 0.32	79.99 ± 0.21	70.70 ± 0.35	$\mathbf{85.97 \pm 0.09}$

† Results reported in (Liu et al., 2019)

Table 3: Comparative results of few-shot learning methods on the two benchmark datasets *mini*Imagenet and *tiered*Imagenet. The average 5-way few-shot classification accuracies (%, top-1) along with 95% confidence intervals are reported on the test split of each dataset.

4.2 MINIIMAGENET AND TIEREDIMAGENET

Datasets and evaluation. The *mini*Imagenet dataset (Vinyals et al., 2016) consists of 100 randomly chosen classes from the ILSVRC-2012 dataset (Russakovsky et al., 2015). We use the commonly-used class split proposed in Ravi & Larochelle (2017). The *tiered*Imagenet dataset (Ren et al., 2018) is a subset of ILSVRC-2012 (Russakovsky et al., 2015), composed of 608 classes grouped in 34 high-level categories. For evaluation, we measure the accuracy drawing 800 tasks (five test instances per class) from the meta-test set. Following prior work, we focus on 5-way/1-shot and 5-way/5-shot tasks. We report the average accuracy and a 95 % confidence interval across all test images and tasks.

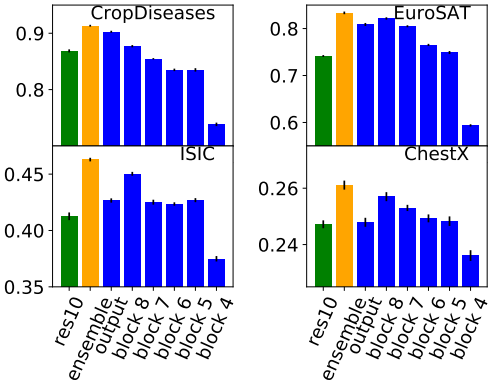

Figure 2: 5-shot top-1 accuracies (along with 95% confidence intervals) of different layers of an Imagenet-pretrained ResNet-18 and the ensemble result (orange) on the four different datasets of the cross-domain few-shot learning benchmark. For comparison, also the ResNet-10 ensemble results (green) are included.

CHEF implementation and results. We perform pre-training of the respective backbone networks on the *mini*Imagenet and the *tiered*Imagenet dataset in the same way as described in Sec. 4.1. We evaluate CHEF using two different backbone architectures: a Conv-4 and a ResNet-12 network. We use the Conv-4 network described by Vinyals et al. (2016). Following Lee et al. (2019), we configure the ResNet-12 backbone as 4 residual blocks, which contain a max-pooling and a batch-norm layer and are regularized by DropBlock (Ghiasi et al., 2018). Again, model selection and hyper-parameter tuning is performed as described in Sec. 4.1. CHEF achieves state-of-the-art performance in 5 out of 8 categories. Results are provided in Table 3. An ablation study of the *mini*Imagenet and *tiered*Imagenet results can be found in the appendix.

4.3 EXAMPLE APPLICATION: DRUG DISCOVERY

In drug discovery, it is essential to know properties of drug candidates, such as biological activities or toxicity. Since the measurements of these properties require time- and cost-intensive laboratory experiments, machine learning models are used to substitute such measurements (Hochreiter et al., 2018). However, due to the high experimental effort often only few high-quality measurements are available for training. Thus, few-shot learning is highly relevant for computational drug discovery.

Problem setting. We consider a 50-shot cross-domain few-shot learning setting in the field of toxicity prediction, utilizing the Tox21 Data Challenge dataset with twelve different toxic effects (Huang & Xia, 2017; Mayr et al., 2016). Around 50 available measurements is a typical scenario when introducing a new high-quality assay in drug design. So far, the standard approach to deal with such few data points was to use machine learning methods like Support Vector Machines (SVMs; Cortes & Vapnik, 1995) or Random Forests (RFs; Breiman, 2001). However, these methods do not exploit the rich data available, like the ChEMBL20 database (Gaulton et al., 2017). Viewing the Tox21 data as a domain shift of the ChEMBL20 database allows the application of cross-domain few-shot learning methods. In this setting, a domain shift can be observed both in the input data and in the target data. The molecules (input domain) are strongly shifted towards a specialized chemical space and the biological effects (output domain) are shifted towards toxicity. A further shift is in the distribution of the target labels, which are now much more imbalanced in comparison to ChEMBL20. In order to mirror this distribution shift correctly, the number of toxic vs. non-toxic molecules in the training sets for each of the twelve few-shot tasks (twelve different toxic effects) are sub-sampled accordingly. For example, the 50-shot few-shot scenario (50-50 toxic/non-toxic) is adjusted to a 10-90 scenario. For the twelve few-shot learning tasks, training samples are drawn from the training set and test samples are drawn from the test set of the Tox21 data, respectively. We sample individually for each of the twelve tasks.

CHEF implementation for molecules. We first train a fully-connected deep neural network (FCN) for the prediction of bioactivities from the ChEMBL20 database (original domain). The network is trained in a massive multi-task setting, where 1,830 tasks are predicted at once, such that the network is forced to learn proficient representations that can be shared for multiple tasks (Ma et al., 2015; Unterthiner et al., 2014). The total number of 892,480 features of the ChEMBL20 database was reduced by a sparseness criterion on the molecules to 1,866 features. The neurons in the input layer of the FCN represent one of 1,866 ECFP6 (Rogers & Hahn, 2010) features, which are used as a feature representation for describing the raw structure of the molecules. Each neuron of the output layer represents one of the 1,830 prediction tasks. We use the pre-trained network and apply CHEF by representation fusion of the three bottleneck layers of the network for predicting the twelve different toxic effects of the new domain of the Tox21 Data Challenge.

Method	ROC-AUC
CHEF	**0.76** $\pm$ 0.02
SVM	0.66 $\pm$ 0.03
RF	0.64$\pm$ 0.03

Table 4: ROC-AUC performance for few-shot drug discovery. CHEF is compared to conventional methods (SVM, RF) for the prediction of toxic effects. Mean and standard deviation are computed across twelve different effects and across 100 differently sampled training and test sets.

Experimental evaluation. We evaluate the performance of CHEF on the twelve tasks of the Tox21 Data Challenge and compare it to conventional methods, like SVMs and RFs, that are used in drug design when few data is available. We use SVMs with a MinMax kernel, because it also allows the use of features that are not present in the training data. For CHEF, only the 1,866 ECFP input features of ChEMBL20 pre-training network database are used. For SVMs and RFs, all available ECFP features are used. ROC-AUC values are computed across the twelve tasks of the Tox21 Data Challenge and across 100 differently sampled training and test sets. CHEF achieves significantly better ROC-AUC values than SVMs and RFs. Table 4 shows the results (p-value $< 10^{-17}$ for both SVM and RF when using a paired Wilcoxon test). Results for the twelve individual tasks and a more detailed description is given in the appendix. CHEF significantly outperforms traditional methods in drug design, which demonstrates the great potential of cross-domain few-shot learning in this field.

5 Conclusion

We have introduced CHEF as new cross-domain few-shot learning method. CHEF builds on the concept of representation fusion, which unifies information from different levels of abstraction. Representation fusion allows one to successfully tackle various few-shot learning problems with large domain shifts across a wide range of different tasks. CHEF obtains new state-of-the-art results in all categories of the broader study of cross-domain few-shot learning benchmarks. Finally, we have tested the performance of CHEF in a real-world cross-domain application in drug discovery, i.e. toxicity prediction when a domain shift appears. CHEF significantly outperforms all traditional approaches demonstrating great potential for applications in computational drug discovery.

REFERENCES

K. Allen, E. Shelhamer, H. Shin, and J. Tenenbaum. Infinite mixture prototypes for few-shot learning. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 232–241, 2019.

M. Andrychowicz, M. Denil, S. Gomez, M. W. Hoffman, D. Pfau, T. Schaul, B. Shillingford, and N. De Freitas. Learning to learn by gradient descent by gradient descent. In *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, pp. 3981–3989, 2016.

S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Vaughan. A theory of learning from different domains. *Machine Learning*, 79:151–175, 2010.

Y. Bengio, S. Bengio, and J. Cloutier. *Learning a synaptic learning rule*. Citeseer, 1990.

Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

C. Chang and C. Lin. Libsvm: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27, 2011.

L. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 801–818, 2018.

N. Codella, V. Rotemberg, P. Tschandl, M. E. Celebi, S. Dusza, D. Gutman, B. Helba, A. Kalloo, K. Liopyris, M. Marchetti, et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1902.03368*, 2019.

C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

M. Dixit, R. Kwitt, M. Niethammer, and N. Vasconcelos. Aga: Attribute-guided augmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7455–7463, 2017.

N. Dvornik, C. Schmid, and J. Mairal. Diversity with cooperation: Ensemble methods for few-shot classification. In *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, pp. 3723–3731, 2019.

L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):594–611, 2006.

C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1126–1135. JMLR. org, 2017.

Y. Frégnac. Hebbian synaptic plasticity, comparative and developmental aspects. In M. Arbib (ed.), *The Handbook of Brain Theory and Neural Networks*, pp. 515–521. MIT Press, 2002.

J. Gama, I. Zliobaite, A. Bifet, P. Mykola, and A. Bouchachia. A survey on concept drift adaptation. *ACM Computing Surveys*, 46(4), 2014. ISSN 0360-0300. doi: 10.1145/2523813.

H. Gao, Z. Shou, A. Zareian, H. Zhang, and S. Chang. Low-shot learning via covariance-preserving adversarial augmentation networks. In *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, pp. 975–985, 2018.

A. Gaulton, A. Hersey, M. Nowotka, A.P. Bento, J. Chambers, D. Mendez, P. Mutowo, F. Atkinson, L. Bellis, E. Cibrián-Uhalte, et al. The chembl database in 2017. *Nucleic acids research*, 45(D1): D945–D954, 2017.

G. Ghiasi, T.Y. Lin, and Q.V. Le. Dropblock: A regularization method for convolutional networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 31*, pp. 10727–10737. Curran Associates, Inc., 2018.

S. Gidaris and N. Komodakis. Dynamic few-shot visual learning without forgetting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4367–4375, 2018.

R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587, 2014.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, pp. 2672–2680, 2014.

Y. Guo, N. Codella, L. Karlinsky, J.R. Smith, T. Rosing, and R. Feris. A new benchmark for evaluation of cross-domain few-shot learning. *arXiv preprint arXiv:1912.07200*, 2019.

B. Hariharan and R. Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, pp. 3018–3027, 2017.

D. O. Hebb. *The organization of behavior: A neuropsychological theory*. Psychology Press, 2005.

P. Helber, B. Bischke, A. Dengel, and D. Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019.

T. Hertz, A. B. Hillel, and D. Weinshall. Learning a kernel function for classification with small training samples. In *Proc. Int. Conf. Mach. Learn. (ICML)*, pp. 401–408, 2006.

M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in neural information processing systems*, pp. 6626–6637, 2017.

S. Hochreiter, A.S. Younger, and P.R. Conwell. Learning to learn using gradient descent. In *International Conference on Artificial Neural Networks*, pp. 87–94. Springer, 2001.

S. Hochreiter, G. Klambauer, and M. Rarey. Machine learning in drug discovery, 2018.

M. Hofmarcher, E. Rumetshofer, D.A. Clevert, S. Hochreiter, and G. Klambauer. Accurate prediction of biological assays with high-throughput microscopy images and convolutional networks. *Journal of chemical information and modeling*, 59(3):1163–1171, 2019.

T. Hospedales, A. Antoniou, P. Micaelli, and A. Storkey. Meta-learning in neural networks: A survey. *arXiv*, 2020.

R. Hou, H. Chang, M.A. Bingpeng, S. Shan, and X. Chen. Cross attention network for few-shot classification. In *Advances in Neural Information Processing Systems*, pp. 4005–4016, 2019.

G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

R. Huang and M. Xia. Editorial: Tox21 challenge to build predictive models of nuclear receptor and stress response pathways as mediated by exposure to environmental toxicants and drugs. front. *Tox21 Challenge to Build Predictive Models of Nuclear Receptor and Stress Response Pathways as Mediated by Exposure to Environmental Toxicants and Drugs*, 5(3):5, 2017.

S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 448–456, Lille, France, 07–09 Jul 2015. PMLR.

H. Jiayuan, A. Gretton, K. Borgwardt, B. Schölkopf, and A.J. Smola. Correcting sample selection bias by unlabeled data. In B. Schölkopf, J. C. Platt, and T. Hoffman (eds.), *Advances in Neural Information Processing Systems 19*, pp. 601–608. MIT Press, 2007.

G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. In *Proc. Int. Conf. Mach. Learn. (ICML) Deep Learn. Workshop*, volume 2, 2015.

W. M. Kouw and M. Loog. A review of domain adaptation without target labels. *IEEE transactions on pattern analysis and machine intelligence*, October 2019. ISSN 0162-8828. doi: 10.1109/tpami.2019.2945942.

R. Kwitt, S. Hegenbart, and M. Niethammer. One-shot learning of scene locations via feature trajectory transfer. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 78–86, 2016.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

K. Lee, S. Maji, A. Ravichandran, and S. Soatto. Meta-learning with differentiable convex optimization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10657–10665, 2019.

H. Li, D. Eigen, S. Dodge, M. Zeiler, and X. Wang. Finding task-relevant features for few-shot learning by category traversal. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 1–10. Computer Vision Foundation / IEEE, 2019.

Y. Liu, J. Lee, M. Park, S. Kim, E. Yang, S.J. Hwang, and Y. Yang. Learning to propagate labels: Transductive propagation network for few-shot learning. In *ICLR*, 2019.

J. Lu, P. Gong, J. Ye, and C. Zhang. Learning from very few samples: A survey, 2020.

J. Ma, R.P. Sheridan, A. Liaw, G.E. Dahl, and V. Svetnik. Deep neural nets as a method for quantitative structure–activity relationships. *Journal of chemical information and modeling*, 55(2):263–274, 2015.

G. Marcus. Deep learning: A critical appraisal. *arXiv*, 2018.

A. Mayr, G. Klambauer, T. Unterthiner, and S. Hochreiter. Deeptox: toxicity prediction using deep learning. *Frontiers in Environmental Science*, 3:80, 2016.

A. Mayr, G. Klambauer, T. Unterthiner, M. Steijaert, J.K. Wegner, H. Ceulemans, D.A. Clevert, and S. Hochreiter. Large-scale comparison of machine learning methods for drug target prediction on chembl. *Chemical science*, 9(24):5441–5451, 2018.

N. Mishra, M. Rohaninejad, X. Chen, and P. Abbeel. A simple neural attentive meta-learner. In *International Conference on Learning Representations*, 2018.

S. P. Mohanty, D. P. Hughes, and M. Salathé. Using deep learning for image-based plant disease detection. *Frontiers in plant science*, 7:1419, 2016.

T. Munkhdalai and A. Trischler. Metalearning with hebbian fast weights. *arXiv preprint arXiv:1807.05076*, 2018.

A. Nichol, J. Achiam, and J. Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.

B. Oreshkin, P.R. López, and A. Lacoste. Tadam: Task dependent adaptive metric for improved few-shot learning. In *Advances in Neural Information Processing Systems*, pp. 721–731, 2018.

F. Pahde, M. Puscas, J. Wolff, T. Klein, N. Sebe, and M. Nabi. Low-shot learning from imaginary 3d model. In *Proc. IEEE Winter Conf. Applica. Comput. Vis. (WACV)*, pp. 978–985, 2019.

S.J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 32*, pp. 8026–8037. Curran Associates, Inc., 2019.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

A. Rajeswaran, C. Finn, S.M. Kakade, and S. Levine. Meta-learning with implicit gradients. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 32*, pp. 113–124. Curran Associates, Inc., 2019.

S. Ravi and H. Larochelle. Optimization as a model for few-shot learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

M. Ren, E. Triantafillou, S. Ravi, J. Snell, K. Swersky, J.B. Tenenbaum, H. Larochelle, and R.S. Zemel. Meta-learning for semi-supervised few-shot classification. In *International Conference on Learning Representations*, 2018.

D. Rogers and M. Hahn. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754, 2010.

O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

E. Rumetshofer, M. Hofmarcher, C. Röhrl, S. Hochreiter, and G. Klambauer. Human-level protein localization with convolutional neural networks. In *International Conference on Learning Representations*, 2018.

O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

A. Rusu, D. Rao, J. Sygnowski, O. Vinyals, R. Pascanu, S. Osindero, and R. Hadsell. Meta-learning with latent embedding optimization. In *International Conference on Learning Representations*, 2018.

T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen. Improved techniques for training gans. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 29*, pp. 2234–2242. Curran Associates, Inc., 2016.

J. Schmidhuber. Evolutionary principles in self-referential learning. on learning now to learn: The meta-meta-meta...-hook. Diploma thesis, Technische Universitat Munchen, Germany, 14 May 1987.

J. Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

J. Snell, K. Swersky, and R. Zemel. Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, pp. 4077–4087, 2017a.

J. Snell, K. Swersky, and R. Zemel. Prototypical networks for few-shot learning. In *Proc. Advances Neural Inf. Process. Syst. (NIPS)*, pp. 4077–4087, 2017b.

N. Sturm, A. Mayr, T. Le Van, V. Chupakhin, H. Ceulemans, J. Wegner, J.F. Golib-Dzib, N. Jeliazkova, Y. Vandriessche, S. Böhm, et al. Industry-scale application and evaluation of deep learning for drug target prediction. *Journal of Cheminformatics*, 12:1–13, 2020.

Q. Sun, Y. Liu, T.S. Chua, and B. Schiele. Meta-transfer learning for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 403–412, 2019.

F. Sung, Y. Yang, L. Zhang, T. Xiang, P.H.S. Torr, and T.M. Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1199–1208, 2018.

P. Tschandl, C. Rosendahl, and H. Kittler. The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific data*, 5:180161, 2018.

H.Y. Tseng, H.Y. Lee, J.B. Huang, and M.H. Yang. Cross-domain few-shot classification via learned feature-wise transformation. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020.

T. Unterthiner, A. Mayr, G. Klambauer, M. Steijaert, J.K. Wegner, H. Ceulemans, and S. Hochreiter. Multi-task deep networks for drug target prediction. In *Neural Information Processing System*, volume 2014, pp. 1–4. NeurIPS, 2014.

O. Vinyals, C. Blundell, T.P. Lillicrap, K. Kavukcuoglu, and D. Wierstra. Matching networks for one shot learning. In *NIPS*, 2016.

X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R.M. Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2097–2106, 2017.

G.I. Webb, L.K. Lee, B. Goethals, and F. Petitjean. Analyzing concept drift and shift from sample data. *Data Mining and Knowledge Discovery*, 32:1179–1199, 2018.

G. Widmer and M. Kubat. Learning in the presence of concept drift and hidden contexts. *Machine learning*, 23(1):69–101, 1996.

M. K. Wouter. An introduction to domain adaptation and transfer learning. *ArXiv*, abs/1812.11806, 2018.

H.J. Ye, H. Hu, D.C. Zhan, and F. Sha. Few-shot learning via embedding adaptation with set-to-set functions. In *Computer Vision and Pattern Recognition (CVPR)*, 2020.

M. Ye and Y. Guo. Deep triplet ranking networks for one-shot recognition. *arXiv preprint arXiv:1804.07275*, 2018.

S.W. Yoon, J. Seo, and J. Moon. Tapnet: Neural network augmented with task-adaptive projection for few-shot learning. In *ICML*, 2019.

F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

J. Zhang, C. Zhao, B. Ni, M. Xu, and X. Yang. Variational few-shot learning. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

A APPENDIX

A.1 EXPERIMENTAL SETUP

In the following, we give further details on our experimental setups.

A.1.1 CROSS-DOMAIN FEW-SHOT LEARNING

We utilize a ResNet-10 backbone architecture as proposed in Guo et al. (2019). The residual blocks have 64, 128, 256, 512, 4000, and 1000 units, where the latter two are fully connected ReLU layers. We use a learning rate of 0.1, momentum term of 0.9, L2 weight decay term of 10^{-4}, batch size of 256, dropout rate of 0.5 during pre-training. These values were tuned on the horizontal validation set of *mini*Imagenet. For few-shot learning, we choose a Hebbian learning rate of $\alpha = 0.01$ and run the Hebb rule for $I = 400$ steps. These values were tuned on the vertical validation set of *mini*Imagenet.

Figures 3, 4, 5 show the performance of the pre-trained PyTorch ResNet-18 network, where the pre-training is on the entire Imagenet dataset. Additionally, the individual performances of the ResNet-18 layers are depicted. The *mini*Imagenet pre-trained ResNet-10 is shown for comparison. The plots show the general tendency that the ensemble performance on domains which are farther away from the training domain relies more heavily on features in lower layers, i.e. features with less specificity to the original domain.

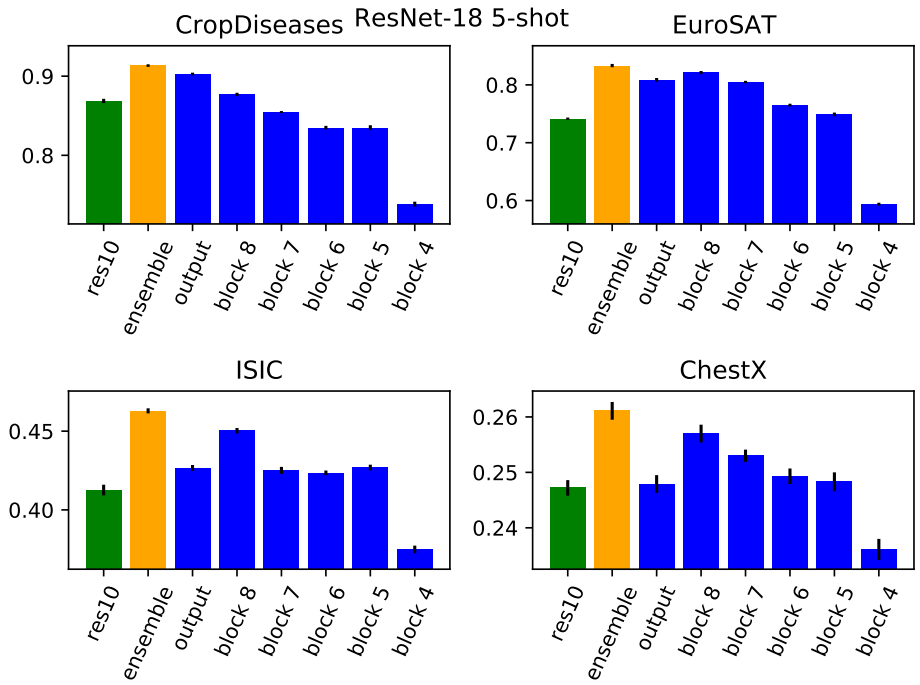

Figure 3: 5-shot top-1 accuracies (along with 95% confidence intervals) of different layers of an Imagenet-pretrained ResNet-18 and the ensemble result (orange) on the four different datasets of the cross-domain few-shot learning benchmark. For comparison, also the ResNet-10 ensemble results (green) are included.

A.1.2 MINIIMAGENET AND TIEREDIMAGENET

Backbone pre-training. For the *mini*Imagenet and *tiered*Imagenet experiments, we utilize Conv-4 and ResNet-12 architectures as backbone networks. The Conv-4 network is described in detail by Vinyals et al. (2016). It is a stack of 4 modules, each of which consists of a 3×3 convolutional layer with 64 units, a batch normalization layer (Ioffe & Szegedy, 2015), a ReLU activation and 2×2 max-pooling layer. On top, we place two fully connected ReLU layers with 400 and 100 units, respectively. The ResNet-12 is described in Lee et al. (2019). We configure the backbone as 4 residual blocks with 64, 160, 320, 640, 4000, and 1000 units, respectively, where the latter two are ReLU-activated fully connected layers. The residual blocks contain a max-pooling and a batch-norm layer and are regularized by DropBlock (Ghiasi et al., 2018) with block sizes of 1×1 for the first two blocks and 5×5 for the latter two blocks.

We pre-train these backbone models for 500 epochs with three different learning rates 0.1, 0.01, and 0.001. For this we use the PyTorch SGD module for stochastic gradient descent with a momentum term of 0.9, an L2 weight decay factor of 10^{-4}, a mini-batchsize of 256, and a dropout probability of 0.1. This pre-training is performed on the horizontal training set of the *mini*Imagenet and the *tiered*Imagenet dataset, resulting in 3 trained models per dataset. We apply early stopping by selecting the model with the lowest loss on the horizontal validation set, while evaluating the model performance after each epoch.

Few-shot learning. For few-shot learning, we perform a grid search to determine the best hyper-parameter setting for each of the datasets and each of the 1-shot and 5-shot settings, using the loss on the vertical validation set. We treat the 3 backbone models that were pre-trained with different learning rates, as described in the previous paragraph, as hyper-parameters. The hyper-parameters used for this grid-search are listed in table 5.

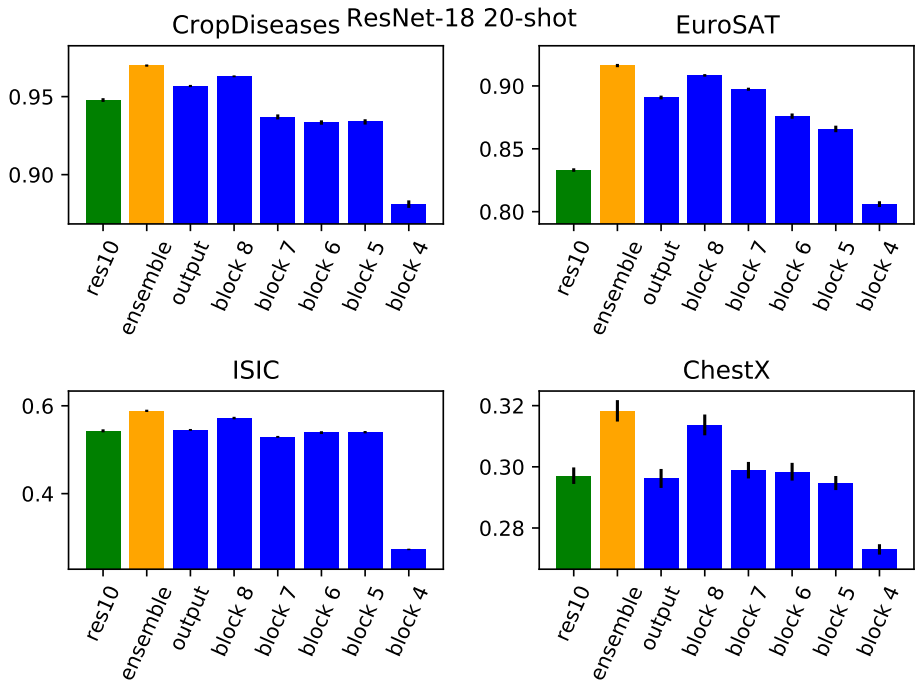

Figure 4: 20-shot top-1 accuracies (along with 95% confidence intervals) of different layers of an Imagenet-pretrained ResNet-18 and the ensemble result (orange) on the four different datasets of the cross-domain few-shot learning benchmark. For comparison, also the ResNet-10 ensemble results (green) are included.

parameter	**values**
learning rate of pre-trained model	$\{0.1, 0.01, 0.001\}$
dropout probability	0.5
Hebbian learning rate	$\{0.1, 0.01, 0.001\}$
number of Hebb rule steps	$\{1, 2, 5, 7, 10, 25, 50, 75, 100, 250, 500, 750\}$

Table 5: Hyper-parameter search space for 1-shot and 5-shot learning on *mini*Imagenet and *tiered*Imagenet using Conv-4 and ResNet-12 backbone models. Best hyper-parameters were evaluated using a grid-search and the loss on the vertical validation set of *mini*Imagenet or *tiered*Imagenet.

After determining the best hyper-parameter setting following this procedure, we perform 1-shot and 5-shot learning on the vertical test sets of *mini*Imagenet and *tiered*Imagenet using 10 different random seeds, respectively. The results are listed in table 3.

Ensemble learning and performance of individual layers. To evaluate the performance of the Hebbian learning using only individual layers versus using the ensemble of layers, we additionally perform the few-shot learning on the vertical test sets using only individual layers as input to the Hebbian learning. As shown in figures 6 and 7, the performance using only the individual layers varies strongly throughout 1-shot and 5-shot learning and the *mini*Imagenet and *tiered*Imagenet dataset. This indicates that the usefulness of the representations provided by the individual layers strongly depends on the data and task setting. In contrast to this, the ensemble of layers reliably achieves either best or second best performance throughout all settings.

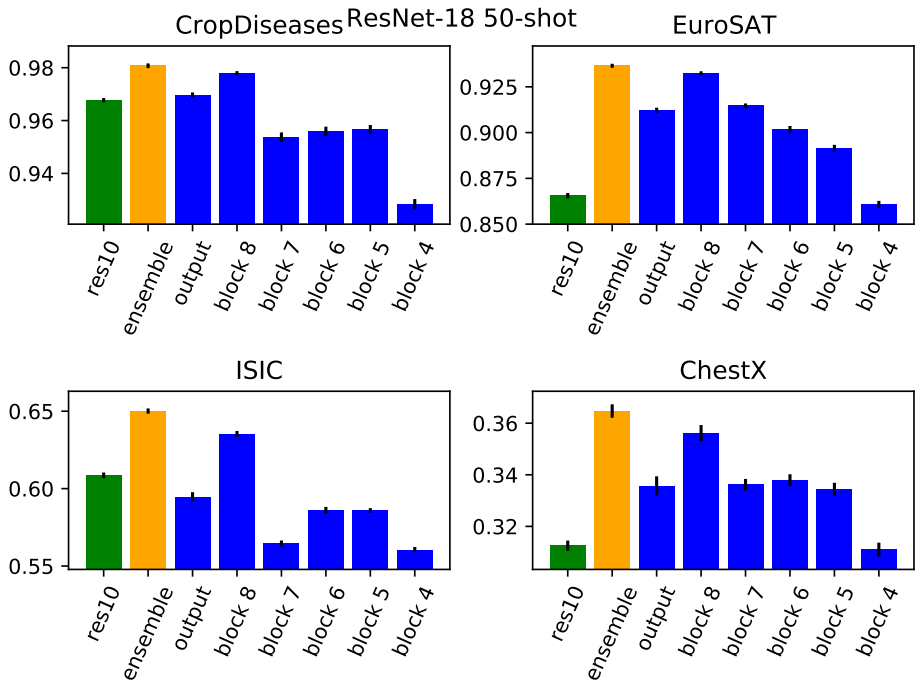

Figure 5: 50-shot top-1 accuracies (along with 95% confidence intervals) of different layers of an Imagenet-pretrained ResNet-18 and the ensemble result (orange) on the four different datasets of the cross-domain few-shot learning benchmark. For comparison, also the ResNet-10 results (green) are included.

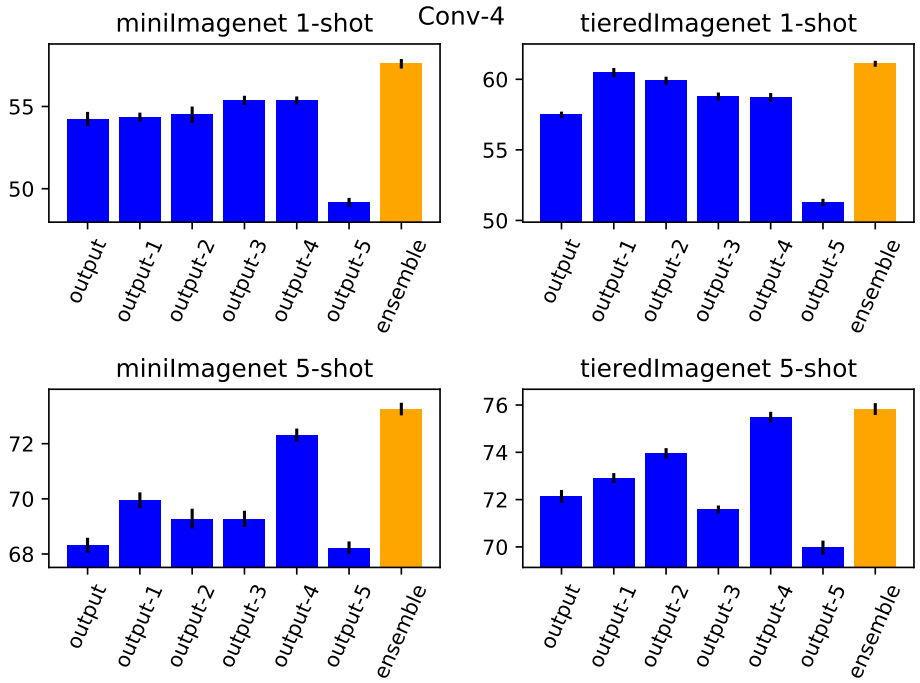

Figure 6: Ablation study of the Conv-4 architecture on the *mini*Imagenet and *tiered*Imagenet datasets for 1-shot and 5-shot. The plots show the individual performances of Hebbian learners acting on single layers and their ensemble performance along with 95% confidence intervals. The labels on the x-axis indicate how far the respective layer is from the output layer.

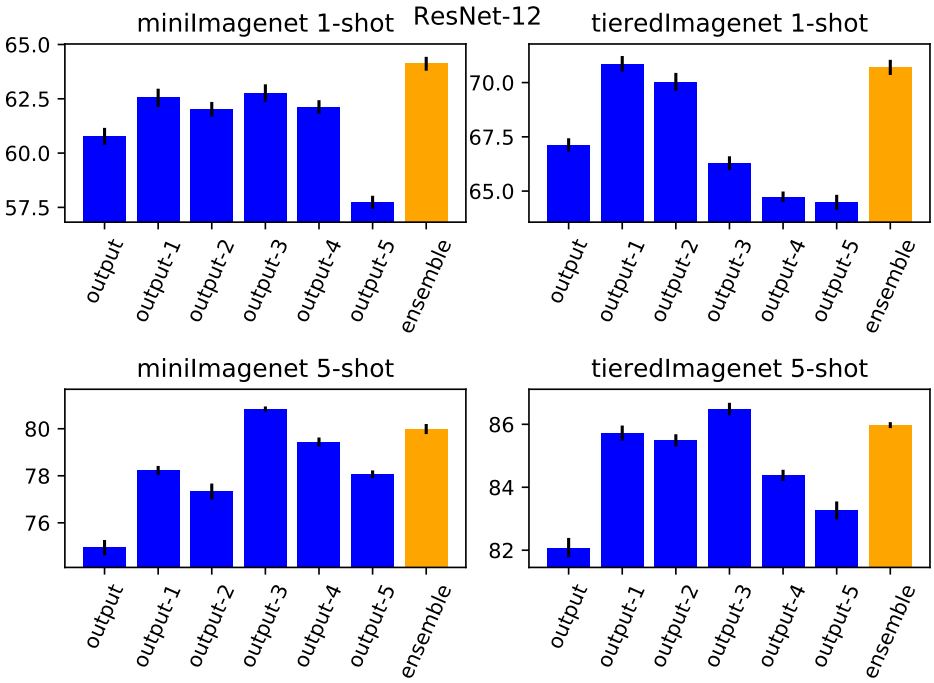

Figure 7: Ablation study of the ResNet-12 architecture on the *mini*Imagenet and *tiered*Imagenet datasets for 1-shot and 5-shot. The plots show the individual performances of Hebbian learners acting on single layers and their ensemble performance along with 95% confidence intervals. The labels on the x-axis indicate how far the respective layer is from the output layer.

A.1.3 EXAMPLE APPLICATION: DRUG DISCOVERY

Details on pre-training on the ChEMBL20 database For training a fully-connected deep neural network (FCN) on the ChEMBL20 database, the number of 892,480 features is reduced by a sparseness criterion on the molecules to 1,866 features. The FCN is trained on 1.1 million molecules for 1,000 epochs minimizing binary cross-entropy and masking out missing values by using an objective, as described in (Mayr et al., 2018).

Details on the compared methods We use ECFP6 features for a raw molecule representation. Note that the number of possible distinct ECFP6 features is not predefined, since a new molecule may be structurally different from all previously seen ones, and it might therefore consist of new unseen ECFP6 features. For SVMs, a MinMax kernel (Mayr et al., 2016) is used, which operates directly on counts of ECFP6 features and used LIBSVM (Chang & Lin, 2011) as provided by `scikit-learn` (Pedregosa et al., 2011). For RFs, the implementation of `scikit-learn` with 1000 trees and kept default values for the other hyperparameters is used.

Detailed results on the Tox21 dataset in a few-shot setup Table 6 lists detailed results and p-values of all twelve few-shot tasks of the Tox21 Data Challenge. For calculating p-values, a paired Wilcoxon test is used.

Dataset	CHEF	SVM	RF	p-value SVM	p-value RF
NR.AhR	**0.86** $\pm$ **0.07**	0.79 $\pm$ 0.07	0.75 $\pm$ 0.07	2.90e-12	1.19e-17
NR.AR	**0.79** $\pm$ **0.09**	0.60 $\pm$ 0.11	0.61 $\pm$ 0.11	1.20e-17	5.25e-18
NR.AR.LBD	**0.84** $\pm$ **0.05**	0.47 $\pm$ 0.11	0.52 $\pm$ 0.10	1.94e-18	1.95e-18
NR.Aromatase	**0.74** $\pm$ **0.08**	0.68 $\pm$ 0.09	0.64 $\pm$ 0.09	3.77e-09	1.12e-13
NR.ER	**0.73** $\pm$ **0.08**	0.70 $\pm$ 0.08	0.65 $\pm$ 0.09	1.39e-03	4.25e-11
NR.ER.LBD	**0.71** $\pm$ **0.08**	0.68 $\pm$ 0.09	0.65 $\pm$ 0.10	1.96e-03	2.40e-06
NR.PPAR.gamma	**0.66** $\pm$ **0.07**	0.61 $\pm$ 0.10	0.60 $\pm$ 0.11	8.04e-06	3.05e-06
SR.ARE	**0.76** $\pm$ **0.08**	0.66 $\pm$ 0.08	0.61 $\pm$ 0.09	2.43e-14	2.51e-17
SR.ATAD5	**0.68** $\pm$ **0.07**	0.62 $\pm$ 0.10	0.61 $\pm$ 0.10	2.23e-07	7.65e-10
SR.HSE	**0.74** $\pm$ **0.06**	0.62 $\pm$ 0.10	0.60 $\pm$ 0.10	3.42e-16	1.40e-16
SR.MMP	**0.89** $\pm$ **0.05**	0.81 $\pm$ 0.08	0.79 $\pm$ 0.09	7.36e-15	4.41e-16
SR.p53	**0.77** $\pm$ **0.08**	0.67 $\pm$ 0.10	0.63 $\pm$ 0.10	3.50e-13	8.86e-17

Table 6: ROC-AUC performances for the twelve individual few-shot tasks (rows) of the Tox21 Data Challenge. CHEF is compared to conventional methods (SVM, RF). Averages and standard deviations are computed across 100 differently sampled training and test sets. The last two columns show the results of paired Wilcoxon tests with the null hypotheses given that SVM and RF perform better, respectively.