NeurIPS'22 Cross-Domain MetaDL Challenge: Results and lessons learned

Dustin Carrión-Ojeda[1,2][*] DUSTIN.CARRION@VISINF.TU-DARMSTADT.DE
Mahbubul Alam[3] MAHBUBUL.ALAM@HAL.HITACHI.COM
Sergio Escalera[4,5,6] SESCALERA@UB.EDU
Ahmed Farahat[3] AHMED.FARAHAT@HAL.HITACHI.COM
Dipanjan Ghosh[3] DIPANJAN.GHOSH@HAL.HITACHI.COM
Teresa Gonzalez Diaz[3] TERESA.GONZALEZDIAZ@HAL.HITACHI.COM
Chetan Gupta[3] CHETAN.GUPTA@HAL.HITACHI.COM
Isabelle Guyon[4,7,8] GUYON@CHALEARN.ORG
Joël Roman Ky[9] JOEL.KY@UNIV-LORRAINE.FR
Xian Yeow Lee[3] XIAN.LEE@HAL.HITACHI.COM
Xin Liu[13] ATTCB63442@MAIL.USTC.EDU.CN
Felix Mohr[10] FELIX.MOHR@UNISABANA.EDU.CO
Manh Hung Nguyen[4] HUNGNM.VNU@GMAIL.COM
Emmanuel Pintelas[11] E.PINTELAS@UPATRAS.GR
Stefan Roth[1,2] STEFAN.ROTH@VISINF.TU-DARMSTADT.DE
Simone Schaub-Meyer[1,2] SIMONE.SCHAUB@VISINF.TU-DARMSTADT.DE
Haozhe Sun[7] HAOZHE.SUN@UNIVERSITE-PARIS-SACLAY.FR
Ihsan Ullah[7] IHSAN2131.@GMAIL.COM
Joaquin Vanschoren[12] J.VANSCHOREN@TUE.NL
Lasitha Vidyaratne[3] LASITHA.VIDYARATNE@HAL.HITACHI.COM
Jiamin Wu[13] JIAMINWU@MAIL.USTC.EDU.CN
Xiaotian Yin[13] XIAOTIANYIN@MAIL.USTC.EDU.CN

[1] *Department of Computer Science, Technical University of Darmstadt, Germany*
[2] *hessian.AI, Germany*
[3] *Industrial AI Lab, Hitachi America, Ltd. R&D, USA*
[4] *ChaLearn, USA*
[5] *Universitat de Barcelona, Barcelona, Spain*
[6] *Computer Vision Center, Bellaterra, Barcelona, Spain*
[7] *LISN/INRIA/CNRS, Université Paris-Saclay, France*
[8] *Google Brain, USA*
[9] *RESIST/INRIA/CNRS, Université de Lorraine, France*
[10] *Universidad de La Sabana, Chía, Colombia*
[11] *Department of Mathematics, University of Patras, Greece*
[12] *Eindhoven University of Technology, The Netherlands*
[13] *University of Science and Technology of China, China*

Editors: Marco Ciccone, Gustavo Stolovitzky, Jacob Albrecht

[*] The first author is the principal challenge organizer; other authors are listed in alphabetical order.

Abstract

Deep neural networks have demonstrated the ability to outperform humans in multiple tasks, but they often require substantial amounts of data and computational resources. These resources may be limited in certain fields. Meta-learning seeks to overcome these challenges by utilizing past task experiences to efficiently solve new tasks, achieving better performance with limited training data and modest computational resources. To further advance the ChaLearn MetaDL competition series, we organized the Cross-Domain MetaDL Challenge for NeurIPS'22. This challenge aimed to solve "any-way" and "any-shot" tasks from 10 domains through cross-domain meta-learning. In this paper, authored collaboratively by the competition organizers, top-ranked participants, and external collaborators, we describe the technical aspects of the competition, baseline methods, and top-ranked approaches that have been open-sourced. Additionally, we provide a detailed analysis of the competition results. Lessons learned from this competition include the critical role of pre-trained backbones, the necessity of preventing overfitting, and the significance of using data augmentation or domain adaptation techniques in conjunction with extra optimizations to improve performance.

Keywords: Image Classification, Competition, Few-Shot Learning, Cross-Domain Meta-Learning

1. Introduction

Computer vision challenges have significantly contributed to advance the state of the art in several tasks such as image classification (Russakovsky et al., 2015), object detection (Everingham et al., 2006), and image segmentation (Lin et al., 2014). In recent years, meta-learning has emerged as a promising approach to enable neural networks to quickly adapt to new tasks by leveraging knowledge acquired from previous ones (Liu et al., 2021; Brazdil et al., 2022). Thus, in 2020, ChaLearn[1] started the *Challenge series in meta-learning (MetaDL)*,[2] focusing on image classification and few-shot learning.

We consider image classification tasks in the "few-shot learning setting", for which few labeled examples (usually 1 to 20) are available for training. The first two challenges of the MetaDL series (El Baz et al., 2022) focused on the "conventional" setting (Vinyals et al., 2016), in which tasks or "episodes" include a number of classes or "ways" that are *identical for all tasks*, as well as a number of labeled examples per class or "shots" *also identical for all tasks*. Moreover, the first two challenges only tackled the within-domain scenario where the training and testing data come from the same domain. However, real-world applications often involve multiple domains with different distributions, posing a significant challenge for meta-learning algorithms (Phoo and Hariharan, 2021).

Considering the limitations of the previous competitions, the contributions of the **Cross-Domain MetaDL Challenge** are the following. (1) We introduce a more realistic evaluation setting to motivate the development of methods that generalize across domains (cross-domain scenario) in different regimes of ways and shots (any-way any-shot learning). (2) We use the newly created Meta-Album[3] meta-dataset (Ullah et al., 2022) to ensure that the testing datasets are not already known to the meta-learning community. (3) We create two separate competition leagues to analyze if using pre-trained models leads to significantly

[1] http://www.chalearn.org

[2] https://metalearning.chalearn.org

[3] https://meta-album.github.io

better results than "de novo" training, *i.e.,* the models' weights are randomly initialized. (4) All the winning solutions are open-source. (5) We conduct post-challenge analyses on the baselines and winning solutions. (6) The top-ranked method, MetaBeyond, surpasses the winning solution of our previous challenge, MetaDelta++ (Chen et al., 2021), by more than 10% points of normalized accuracy.

2. Competition Setup

This section briefly describes the problem tackled by this challenge, the used datasets, and the competition protocol. Our companion paper about the competition design and baseline results (Carrión-Ojeda et al., 2022) provides a detailed description of the competition setup.

2.1. Problem Definition

The few-shot learning problems are commonly known as N-way k-shot problems. In these problems, each task $\mathcal{T}$ is composed of a limited training set $\mathcal{D}_{\mathcal{T}}^{train}$ and a (usually relatively small) test set drawn from the same distribution $\mathcal{D}_{\mathcal{T}}^{test}$, which are also called *support* and *query* sets, respectively (Vinyals et al., 2016). The number of ways N represents the number of classes in a task; the same N classes are present in both support and query sets. The number of shots k denotes the number of examples per class in the support set. We are interested in meta-learning. Thus, we assume there is a "large" supply of similar tasks from which we can draw a meta-training and a meta-testing set. Meta-training tasks come with labeled data for the support and query sets, while meta-test tasks include labels only for the support sets, and the goal is to predict the labels of the query sets.

In this competition, the number of classes in the meta-test tasks ranges from 2 to 20 ($N \in \{2, \dots, 20\}$), the support set contains 1 to 20 labeled examples per class ($k \in \{1, \dots, 20\}$), and the query set contains 20 unlabeled examples per class, *i.e.,* $|\mathcal{D}_{\mathcal{T}}^{train}| = N \times k$, and $|\mathcal{D}_{\mathcal{T}}^{test}| = N \times 20$. Furthermore, since this challenge focuses on cross-domain meta-learning, all the tasks are carved out from a meta-dataset that contains multiple datasets from ten domains. In our cross-domain scenario, each task $\mathcal{T}$ is generated from a specific dataset associated to one of the domains explained in the next section.

2.2. Datasets

This challenge uses the Meta-Album meta-dataset, prepared in conjunction with this competition (Ullah et al., 2022). Meta-Album comprises 40 image datasets, either newly created or re-purposed from existing ones. The datasets belong to 10 domains: small and large animals, plants and plant diseases, vehicles, human actions, microscopic data, satellite images, industrial textures, and printed characters (OCR). This competition only utilized 30 datasets from Meta-Album mini, which is one of the four versions of Meta-Album (original, extended, mini, micro) containing 40 images per class, grouped into three sets (Set-0, Set-1, and Set-2) of ten datasets each, one from each domain. The final test datasets were new to the meta-learning community and had not been previously included in any past meta-learning benchmarks. Sets 0–2 are already released on OpenML (Vanschoren et al., 2014), and all the information on how to download them can be found on the Meta-Album website.[3] The ten unused datasets have been reserved to organize an upcoming competition.

2.3. Competition Protocol

The competition was hosted on CodaLab[4] (Pavao et al., 2022) and was composed of 3 phases (Public, Feedback, and Final). During the *Public phase* (June 15–30, 2022), no submissions could be made; instead, the participants familiarized themselves with the competition problem described in Section 2.1 by using the provided starting kit[5] and Set-0 (see Section 2.2). Then, during the *Feedback phase* (July 1 – August 31, 2022), participants could make 2 submissions per day and a maximum of 100 submissions during the whole phase. Each submission was meta-trained on Set-0 and evaluated on 1 000 any-way any-shot tasks carved out from Set-1 (100 tasks per dataset). After the evaluation, the participants received feedback about their overall performance and the performance per dataset on the competition leaderboard. Lastly, during the *Final phase* (September 1–30, 2022), the last submission of each participant on the Feedback phase, whose performance is above the baseline performance (see Section 4.1), was automatically forwarded to the Final phase (participants could not make any further changes) to be meta-trained on Sets 0–1 and evaluated on 6 000 any-way any-shot tasks carved out from Set-2 (600 tasks per dataset). Figure 1 illustrates the competition workflow used during the Feedback and Final phases.

This competition had five different leagues, which can be found on the competition site,[4] to encourage diverse participants and types of submissions. Still, two of those leagues are the main ones since they either allow the usage of pre-trained models (Free-style league) or encourage “de novo” training (Meta-learning league). The remaining three leagues (New-in-ML, Women, and Participant of a rarely represented country) depend on the results of the previously mentioned leagues and specific information from the participants. Appendix B provides a detailed description of the differences between each league.

3. Approaches Used by Top-ranked Teams

All top-ranked teams filled out our pre-defined fact sheets, which can be found on our website.[2] They were invited to co-author this paper, and each top-ranked team summarized its solution below.

3.1. Team MetaBeyond: Lightweight Task Adaptation Network for Cross-Domain Few-Shot Learning

The team MetaBeyond proposed a solution composed of two meta-learners, each equipped with lightweight task adaptation modules. By quickly updating the task-specific parameters during meta-testing, their method can address the large domain gap by learning task-adaptive and generalizable features even for unseen tasks. For the meta-learners, they used two pre-trained backbones. One is the frequently used ResNet-50 (He et al., 2016) with semi-weakly supervised training on ImageNet-1K, and the other is the PoolFormer-S24 (Yu et al., 2022), a variant of the Vision Transformer (Dosovitskiy et al., 2021) with ImageNet-1K pre-trained weights. Following the practice of the baseline method MetaDelta++ (Chen et al., 2021), they add an MLP-based classification layer after the backbones, freeze the parameters of the shallower layers, and fine-tune the remaining network during meta-training. After

[4] https://codalab.lisn.upsaclay.fr/competitions/3627

[5] https://github.com/DustinCarrion/cd-metadl

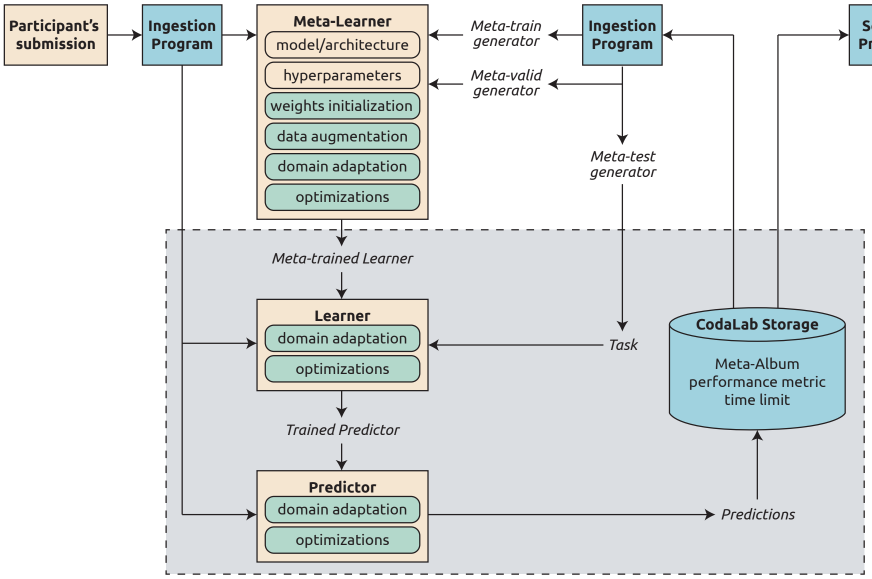

Figure 1: Competition workflow. In this setup, the blue rectangles are defined by the organizers and remain consistent across all submissions. On the other hand, the yellow rectangles are defined by the participants and can vary between submissions. The green blocks represent the techniques shared by the top-ranked teams, which are studied in Section 4.3. Additionally, the process inside the grey shaded area is repeated for each meta-test task.

obtaining the fine-tuned backbones, they remove the MLP layer and replace it with a prototype-based classification head during meta-testing. Then, to deal with the domain gap during meta-testing, they attach learnable task-adaptive parameters to the backbones and learn them from scratch using the support samples of the current meta-test task. For the ResNet-based meta-learner, the task adaptation modules are parameterized as 1×1 convolutional kernels and are inserted after the convolution layer in a residual manner to convolve with the original input features. On the other hand, for the PoolFormer-based meta-learner, the task-adaptation module is formulated as a linear adapter (Li et al., 2022) and added after the final head layer. When the task-adaptation modules are learned for both meta-learners, they are ensembled by a simple voting strategy to obtain the final prediction. The code is available on GitHub.[6]

3.2. Team EmmanuelPintelas: Improving Meta-training with Circular Augmentations and Step-back Validation Optimization

The solution of the team EmmanuelPintelas relies on the usage of "Circular Augmentations" and a novel step-back validation optimization pipeline to improve the meta-training performance of any CNN-based model. Moreover, to tackle the problem of any-way any-shot learning, they introduce an ensemble of distance- and linear-based classifiers. The proposed circular augmentation approach applies a different subset of transformation functions in a looping way (circular) for every epoch, allowing the model to emphasize only a

[6] https://github.com/Jamine-W/cdml22-ltan

few specific transformations per epoch in a distributed manner. The step-back validation optimization applied during meta-training works as follows. If the model does not obtain a higher validation score than the previous best score, the model's parameters are switched back to those used to obtain the last highest score. By using this validation scheme, the model will be optimized more smoothly, avoiding local optima. In the meta-testing phase, they use an ensemble of distance-based (Gaussian model; Rasmussen and Williams, 2006) and linear-based (passive aggressive (Shalev-Shwartz et al., 2003) and logistic regression) models. These models were selected based on exhaustive experimentation during the Feedback phase. Furthermore, based on the conducted experiments, the final ensemble is done as follows. For a low number of ways N and shots k (*i.e.,* $N < 8$, $k < 8$), only the Gaussian model is used. For a high number of ways and shots (*i.e.,* $N \geq 8$, $k \geq 8$), the linear-based models modified in a "one-versus-all" fashion are used (Rifkin and Klautau, 2004). The most-confident prediction across all models is selected for any other case. The code is available on GitHub.[7]

3.3. Team CDML: Enhancing MetaDelta++ with Contrastive Loss and Self-Optimal Transport

The team CDML made the following modifications to the MetaDelta++ (Chen et al., 2021) baseline: (1) tuning the backbone architecture, loss function, and hyperparameters during meta-training, (2) leveraging an ensemble of models, and (3) post-processing the features extracted from the backbone models. During the Feedback phase, the team found that a SEResNext101 (Hu et al., 2018), initialized with weights from ImageNet pre-training and with anti-aliasing filters (Zhang, 2019), empirically performs well on a majority of the competition domains. In contrast, SEResNext50 (Hu et al., 2018) with pre-trained weights from ImageNet performs well in the domains where the previously mentioned model struggles the most (human actions and OCR). Thus, both models were meta-trained using a weighted combination of contrastive loss (triplet margin loss; Balntas et al., 2016) and a conventional supervised loss, including standard image augmentation techniques. All these design choices lead to more generalizable feature extractors, which are less likely to overfit to a single domain. During meta-training, team CDML also performed snapshot ensembling (Huang et al., 2017) by saving the models' weights (SEResNext101 and SEResNext50) with the top two validation performances. In the meta-testing phase, the two snapshots of the SEResNext101 and a single snapshot of the SEResNext50 generate the feature representation for the support and query images. These feature representations are then concatenated to form single feature vectors for each image. The rationale behind this ensembling and concatenation is to reduce the effect of a single model generating features that might be overfitted to a particular domain. Subsequently, the concatenated feature vectors are post-processed using the Self-Optimal-Transport feature transform (Shalam and Korman, 2022), which maps the feature vectors into more separable clusters. Finally, the same iterative soft k-means decoder used by MetaDelta++ is used to classify the transformed feature vectors and produce the final predictions. The code is available on GitHub.[8]

[7] https://github.com/EmmanuelPintelas/EmmanuelPintelas-Few-Shot-Meta-Learning-Second-Place-Solution-for-the-NeurIPS-2022-Competition-Track

[8] https://github.com/lasitha-vidyaratne/cdml

3.4. Team metaCD^2: Meta Cross-Domain Contrastive Distillation

The team metaCD^2 modified the MetaDelta++ (Chen et al., 2021) baseline by including (1) a spatial-contrastive distillation loss (Ouali et al., 2021) and (2) regularized knowledge distillation using a student-teacher approach (Islam et al., 2021). Both, student and teacher models used the SWSL-ResNet50 as backbone. During the meta-training phase, the student and the teacher models are respectively fed with strongly-augmented and weakly-augmented versions of the meta-training instances. Following the MetaDelta++ approach, a MLP-based classification layer is added after the backbones, the parameters of the shallower layers are frozen, and only the remaining network of the student model is fine-tuned. For this purpose, the attention-based spatial contrastive loss is computed using each model's spatial outputs (the model outputs prior to the MLP-classification) to meta-learn visual representations transferable to unseen tasks while promoting local discriminative patterns. Then, the knowledge distillation loss is computed from both outputs. The rationale behind this loss is to match both predictions and to allow better generalization by reducing the over-clustering of the features of the same class. This distillation loss also imposes a regularization by using the predictions coming from the weakly (resp. strongly) augmented versions of the teacher (resp. student) model. The two mentioned losses are combined with the cross-entropy loss between the student model outputs and the labels to obtain the final loss function used to update the weights of the student model. During the meta-testing phase, the student model is used to generate the predictions using the same approach as MetaDelta++. The code is available on Github.[9]

4. Results

This section presents the results of the Final phase and the post-challenge analyses, including ablation studies of the top-ranked methods. Appendix D provides ablation studies of the baseline methods.

4.1. Evaluation Protocol

Performance metric. Since the meta-test tasks are any-way any-shot, this competition uses the balanced accuracy (bac), also known as macro-averaging recall, normalized with respect to the number of ways N, as the evaluation metric.[10] This metric is defined as

$$\text{Normalized Accuracy} = \frac{bac - bac_{RG}}{1 - bac_{RG}}, \tag{1}$$

where bac is defined as

$$bac = \frac{1}{N}\sum_{i=1}^{N} \frac{\text{correctly classified examples of class } i}{\text{total examples of class } i}, \tag{2}$$

and bac_{RG} is the accuracy of random guessing, *i.e.,* $1/N$. The error bars in the figures and tables below correspond to 95% confidence intervals (CI) of the mean normalized accuracy at the task level.

[9] https://github.com/joelromanky/metacd2-neurips22-competition

[10] This metric is closely related to the Cohen's Kappa coefficient (Cohen, 1960).

Evaluation setting. As mentioned in Section 2.3, the last valid submission from each participant in the Feedback phase was evaluated in the Final phase only if it surpassed the baseline performance. The baseline performance for the Free-style and Meta-learning leagues was 58.68% and 36.14%, respectively. The former is achieved by MetaDelta++ and the latter by Prototypical Networks, explained in more detail in our companion paper (Carrión-Ojeda et al., 2022). In the Final phase, each submission was evaluated on 6 000 any-way any-shot tasks carved out from Set-2 (600 tasks per dataset). Then, the average normalized classification accuracy over all meta-test tasks is computed. This process was repeated with three random seeds, and the run with the lowest performance was selected for the final ranking. Additionally, each submission was executed for a maximum of 9 hours on a worker with 4 CPU cores, 1 NVIDIA Tesla T4 GPU, 16GB RAM, and 120GB storage.

Baseline methods. This competition used the following six baseline methods: (1) *Train-from-scratch*, which does not perform any meta-training; instead, it directly learns each meta-testing task using only its support set; (2) *Fine-tuning*, which consists of pre-training a backbone network with batches of data from the concatenated meta-training datasets and then only fine-tuning the last layer at meta-test time; (3–5) three popular meta-learning methods – *Matching Networks* (Vinyals et al., 2016), *Prototypical Networks* (Snell et al., 2017), and *FO-MAML* (Finn et al., 2017); (6) *MetaDelta++* (Chen et al., 2021), which corresponds to the winning solution of our previous MetaDL challenge. All baseline methods use a ResNet-18 backbone, except for MetaDelta++, which uses a ResNet-50. The hyperparameters for each baseline are documented in our GitHub repository.[5]

4.2. Final Phase Results

During the Feedback phase, 47 teams made more than 1 000 submissions. However, only the four teams described in Section 3 surpassed the baseline performance (see Section 4.1) and thus were evaluated in the Final phase. Figure 2 displays the average normalized accuracy achieved by the worst out of three runs with different random seeds for the baselines and top-ranked methods. These results compare the average normalized accuracy in the Meta-learning and Free-style leagues. Teams EmmanuelPintelas and MetaBeyond only have results in the Free-style league, as it was not a requirement to participate in both leagues. The detailed results for all competition leagues and each meta-testing dataset are presented in Appendix B and C, respectively.

The results presented in Figure 2 confirm our baseline results already published (Carrión-Ojeda et al., 2022): there is a clear advantage of using pre-trained backbones instead of "de novo" training in the setting of the competition described in Section 4.1. Three hypotheses can be made to explain the failure of "de novo" training in relation to the use of pre-trained weights with ImageNet: (1) the insufficient time budget for training during the competition; (2) the insufficient quality and/or diversity of Meta-Album training data; (3) the limited quantity of Meta-Album data available for meta-training during the Final phase (112 280 images ≈ 11 times less data than ImageNet).

Due to time and resource constraints, we only analyzed hypothesis (1) by allowing the methods initialized with random weights (Meta-learning league) to run for 18 hours (*i.e.,* twice the competition time limit). The results of this analysis are shown in Appendix A and indicate that "de novo" training barely benefits from increasing the training time,

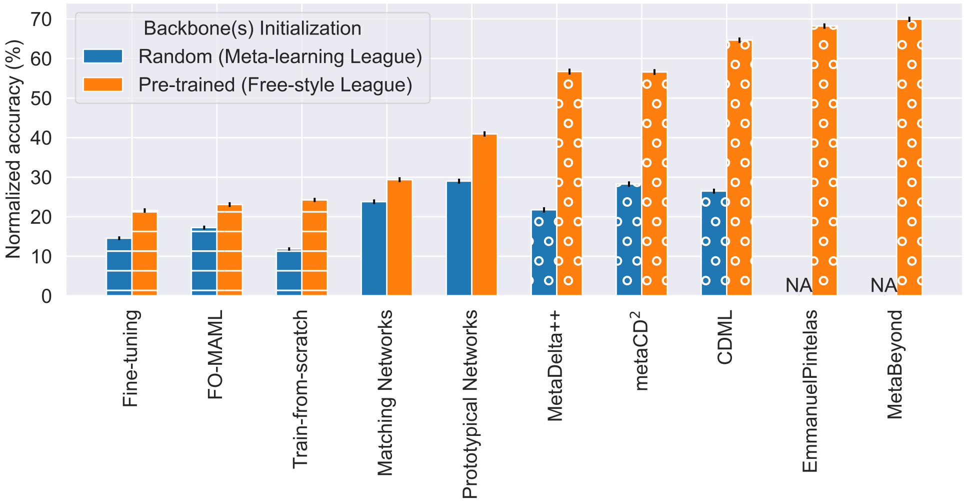

Figure 2: Comparison of Final phase results for the Meta-learning (no pre-trained backbone was allowed) and Free-style (capitalizing on pre-trained weights) leagues. Each bar shows the average normalized accuracy over 6 000 any-way any-shot meta-test tasks (600 tasks per dataset). The texture on the bars indicates the following: *horizontal lines* – baselines with a linear classifier, *no texture* – baselines with a nearest centroid classifier, and *circles* – based on MetaDelta++. The results show the superiority of the solutions based on MetaDelta++ in the Free-style league, while Prototypical Networks are the best method in the Meta-learning league.

meaning that the advantage of pre-trained backbones over randomly initialized ones could be explained mainly by hypotheses (2) and (3).

4.3. Ablation Studies of Top-ranked Methods

After analyzing the solutions of the top-ranked teams, we found four factors shared by all of them: usage of pre-trained backbones, data augmentations (*e.g.,* random cropping, Gaussian blur, horizontal flipping), domain adaptation techniques (*e.g.,* adapters, ensemble of backbones, knowledge distillation), and optimizations (*e.g.,* automatic mixed precision, self-optimal transport). For the ablation studies, we evaluated the incremental inclusion of each factor in the same order as mentioned earlier. The factors are evaluated incrementally and not isolated because, for some methods, there is a dependency between them, *e.g.,* the optimizations depend on the domain adaptations. Therefore, the order for the increments was also determined considering such dependencies and trying to find the most consistent pipeline used by all methods. Moreover, since, as detailed in Section 3, each solution utilized different backbones, data augmentation techniques, domain adaptation techniques, and optimizations, this section aims to provide a high-level overview of the impact of shared components rather than a detailed analysis of individual methods.

Table 1 presents the results of the ablation studies conducted for each top-ranked solution. Thanks to the isolated analysis of the effect of using pre-trained backbones, we verified that this factor provides an absolute improvement of all methods of more than 40% points. Furthermore, as explained in Section 3, all the top-ranked teams only fine-tune the

Table 1: Ablation study of the shared components among all top-ranked teams. The normalized accuracy is an average over 18 000 any-way any-shot meta-test tasks (3 runs of 6 000 tasks). Bold values indicate the best performance for each method.

Team	Pre-training	Data Aug.	Domain Adaptation	Optimizations	Normalized Accuracy (%)
MetaBeyond					17.53 ± 0.34
	✓				66.24 ± 0.39
	✓	✓			66.07 ± 0.39
	✓	✓	✓		(Timeout)
	✓	✓	✓	✓	$\mathbf{69.97 \pm 0.39}$
EmmanuelPintelas					20.54 ± 0.38
	✓				63.54 ± 0.42
	✓	✓			64.45 ± 0.42
	✓	✓	✓		66.04 ± 0.41
	✓	✓	✓	✓	$\mathbf{68.55 \pm 0.39}$
CDML					18.20 ± 0.36
	✓				61.16 ± 0.40
	✓	✓			62.91 ± 0.39
	✓	✓	✓		62.55 ± 0.39
	✓	✓	✓	✓	$\mathbf{64.60 \pm 0.39}$
metaCD^2					16.06 ± 0.33
	✓				56.65 ± 0.41
	✓	✓			56.19 ± 0.41
	✓	✓	✓		56.56 ± 0.41
	✓	✓	✓	✓	$\mathbf{56.76 \pm 0.42}$

last layers of the backbones, which, as shown in Appendix D, is a crucial factor to prevent overfitting of the meta-training set and allow leveraging it. On top of that, although data augmentation is a widely used technique, we found that it only helps improve the performance if applied smartly, as in the case of team EmmanuelPintelas, who introduced a novel circular augmentation strategy (see Section 3.2). Similarly, the inclusion of domain adaptation techniques is not always beneficial. However, when combined with additional optimizations, it helps to improve the performance of all methods.

5. Lessons Learned

We can draw a few lessons from the analysis in the previous section. First, the use of pre-trained backbones proved to be essential. Meta-training the pre-trained backbones with data with similar distribution to the meta-testing data can improve performance, but only if overfitting is avoided at meta-train time. As shown in Appendix D, to avoid overfitting, a simple yet effective approach is fine-tuning only the last blocks of the backbone. We also found that even when giving “de novo” training twice the time, results are worse than when using pre-trained backbones because they converged to a local minima (see Appendix A). Thus, an intelligent validation pipeline such as the one introduced by team

EmmanuelPintelas should be considered. Furthermore, although using data augmentation is a common practice during meta-training, our analysis showed that if it is not performed correctly, it could end up hurting the performance instead of improving it, as observed in the solutions of teams MetaBeyond and metaCD^2.

Our ablation studies of the baselines presented in Appendix D underline the importance of episodic learning: in almost all cases, it led to a higher average normalized accuracy. Similarly, using a nearest centroid classifier instead of a linear one proved to be beneficial for all baselines. Therefore, the highest baseline results were obtained by fine-tuning only the last block of the backbone (ResNet-18) with episodic learning using the FO-MAML algorithm but with a NCC as the classifier. It is worth mentioning that this only applies to the baseline methods. The top-ranked methods obtained higher results by using batch learning, different backbones, and including an ensemble of distance- and linear-based classifiers.

Finally, the domain adaptation techniques used to overcome the domain shift between meta-training and meta-testing sets played a crucial role in the top-ranked solutions. Nevertheless, their benefit was maximized when extra optimizations were included. For instance, the best method, team MetaBeyond, exceeded the time limit when they included their task adapters. Although these adapters are lightweight, they still increased the required time to process each task at meta-testing time. To address this issue, they employed automatic mixed precision at meta-test time to reduce the required time per task and to stay within the time limit. This example emphasizes the importance of not only using the right techniques but also employing appropriate optimization strategies to obtain the best results.

6. Conclusion and Future Work

We organized the Cross-Domain MetaDL Challenge, which is the third competition of the ChaLearn MetaDL series. This competition benchmarked techniques on cross-domain any-way any-shot learning. Overall, fine-tuning pre-trained backbones with the provided meta-training data was crucial for all the evaluated methods to address the domain gap problem. Team MetaBeyond won on 8 out of the 10 testing domains, using two robust pre-trained backbones (ResNet-50 and PoolFormer-S24) equipped with task adaptation modules. In addition, it is essential to mitigate possible overfitting to the meta-training data, which was done by fine-tuning only the last few backbone layers. Although the winners, building on top of a previous solution from a past challenge, favored batch learning over episodic learning, our post-challenge analyses on baseline methods indicate that episodic learning could provide an additional advantage, in combination with monitoring the depth of fine-tuning and using a last layer based on prototypical methods. We hope that this will encourage more work in this direction. Further studies could also include: (1) multiple experts – analyzing whether using one expert per domain would improve performance; (2) ensembles – evaluating, within the same time budget, whether method ensembles can outperform single methods; (3) NAS – evaluating whether neural architectures can adapt to the task at hand.

All the top-ranked methods, described in this paper, are open-sourced. A new edition of the competition is in preparation. To push the development of meta-learning methods harder, this new competition will address the more challenging, but more realistic "domain-independent" scenario, for which the domains seen at meta-training time are distinct from those at meta-testing time.

Acknowledgments

This paper is authored collaboratively by the competition organizers, top-ranked participants, and external collaborators. We acknowledge support from ChaLearn, ANR AI chair HUMANIA ANR-19-CHIA-0022, TAILOR, an ICT48 network funded by EU Horizon 2020 program GA 952215, and the help of Mike Huisman to create the baselines code, except MetaDelta++, which was created by Yudong Chen, Chaoyu Guan, Zhikun Wei, Xin Wang, and Wenwu Zhu. We also thank Adrien Pavao and Adrian El Baz for helpful discussions and technical support during the competition. Additionally, we recognize the effort of Romain Mussard and Gabriel Lauzzana for beta-testing the competition.

References

Vassileios Balntas, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. Learning local feature descriptors with triplets and shallow convolutional neural networks. In R. C. Wilson, E. R. Hancock, and W. A. P. Smith, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 119.1–119.11. BMVA Press, 2016.

Pavel Brazdil, Jan N. van Rijn, Carlos Soares, and Joaquin Vanschoren. *Metalearning: Applications to Automated Machine Learning and Data Mining.* Springer, 2nd edition, 2022.

Dustin Carrión-Ojeda, Hong Chen, Adrian El Baz, Sergio Escalera, Chaoyu Guan, Isabelle Guyon, Ihsan Ullah, Xin Wang, and Wenwu Zhu. NeurIPS'22 Cross-Domain MetaDL competition: Design and baseline results. In P. Brazdil, J. N. van Rijn, H. Gouk, and F. Mohr, editors, *Proceedings of the ECML/PKDD Workshop on Meta-Knowledge Transfer*, volume 191 of *Proceedings of Machine Learning Research*, pages 24–37. PMLR, 2022.

Yudong Chen, Chaoyu Guan, Zhikun Wei, Xin Wang, and Wenwu Zhu. MetaDelta: A meta-learning system for few-shot image classification. In I. Guyon, J. N. van Rijn, S. Treguer, and J. Vanschoren, editors, *Proceedings of the AAAI Workshop on Meta-Learning and MetaDL Challenge*, volume 140 of *Proceedings of Machine Learning Research*, pages 17–28. PMLR, 2021.

Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of the 9th International Conference on Learning Representations (ICLR)*, 2021.

Adrian El Baz, Ihsan Ullah, Edesio Alcobaça, André C. P. L. F. Carvalho, Hong Chen, Fabio Ferreira, Henry Gouk, Chaoyu Guan, Isabelle Guyon, Timothy Hospedales, Shell Hu, Mike Huisman, Frank Hutter, Zhengying Liu, Felix Mohr, Ekrem Öztürk, Jan N. van Rijn, Haozhe Sun, Xin Wang, and Wenwu Zhu. Lessons learned from the NeurIPS 2021 MetaDL challenge: Backbone fine-tuning without episodic meta-learning dominates for few-shot learning image classification. In D. Kiela, M. Ciccone, and B. Caputo, editors, *Proceedings of the NeurIPS 2021 Competitions and Demonstrations Track*, volume 176 of *Proceedings of Machine Learning Research*, pages 80–96. PMLR, 2022.

Mark Everingham, Andrew Zisserman, Christopher K. I. Williams, Luc Van Gool, Moray Allan, Christopher M. Bishop, Olivier Chapelle, Navneet Dalal, Thomas Deselaers, Gyuri Dorkó, Stefan Duffner, Jan Eichhorn, Jason D. R. Farquhar, Mario Fritz, Christophe Garcia, Tom Griffiths, Frederic Jurie, Daniel Keysers, Markus Koskela, Jorma Laaksonen, Diane Larlus, Bastian Leibe, Hongying Meng, Hermann Ney, Bernt Schiele, Cordelia Schmid, Edgar Seemann, John Shawe-Taylor, Amos Storkey, Sandor Szedmak, Bill Triggs, Ilkay Ulusoy, Ville Viitaniemi, and Jianguo Zhang. The 2005 PASCAL Visual Object Classes Challenge. In J. Quiñonero-Candela, I. Dagan, B. Magnini, and F. d'Alché Buc, editors, *Proceedings of the First Pascal Machine Learning Challenges Workshop (MLCW)*, Evaluating Predictive Uncertainty, Visual Object Classification, and Recognizing Textual Entailment, pages 117–176. Springer, 2006.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In D. Precup and Y. W. Teh, editors, *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126—1135. PMLR, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7132–7141, 2018.

Gao Huang, Yixuan Li, Geoff Pleiss, Zhuang Liu, John E. Hopcroft, and Kilian Q. Weinberger. Snapshot ensembles: Train 1, get m for free. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

Ashraful Islam, Chun-Fu Chen, Rameswar Panda, Leonid Karlinsky, Rogerio Feris, and Richard J. Radke. Dynamic distillation network for cross-domain few-shot recognition with unlabeled data. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 3584–3595. Curran Associates, Inc., 2021.

Wei-Hong Li, Xialei Liu, and Hakan Bilen. Cross-domain few-shot learning with task-specific adapters. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7161–7170, 2022.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, *Proceedings of the European Conference on Computer Vision (ECCV)*, volume 8693 of *Lecture Notes in Computer Science*, pages 740–755. Springer, 2014.

Zhengying Liu, Adrien Pavao, Zhen Xu, Sergio Escalera, Fabio Ferreira, Isabelle Guyon, Sirui Hong, Frank Hutter, Rongrong Ji, Julio C. S. Jacques Junior, Ge Li, Marius Lindauer, Zhipeng Luo, Meysam Madadi, Thomas Nierhoff, Kangning Niu, Chunguang Pan, Danny Stoll, Sebastien Treguer, Jin Wang, Peng Wang, Chenglin Wu, Youcheng Xiong, Arbër Zela, and Yang Zhang. Winning solutions and post-challenge analyses of the ChaLearn AutoDL Challenge 2019. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(9):3108–3125, 2021.

Yassine Ouali, Céline Hudelot, and Myriam Tami. Spatial contrastive learning for few-shot classification. In N. Oliver, F. Pérez-Cruz, S. Kramer, J. Read, and J.A. Lozano, editors, *Proceedings of the Research Track of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD)*, volume 12975 of *Lecture Notes in Computer Science*, pages 671–686. Springer, 2021.

Adrien Pavao, Isabelle Guyon, Anne-Catherine Letournel, Xavier Baró, Hugo Escalante, Sergio Escalera, Tyler Thomas, and Zhen Xu. CodaLab Competitions: An open source platform to organize scientific challenges. Technical report, 2022. URL https://hal.inria.fr/hal-03629462v1.

Cheng Perng Phoo and Bharath Hariharan. Self-training for few-shot transfer across extreme task differences. In *Proceedings of the 9th International Conference on Learning Representations (ICLR)*, 2021.

Carl Edward Rasmussen and Christopher KI Williams. *Gaussian processes for machine learning*. MIT Press, 2006.

Ryan Rifkin and Aldebaro Klautau. A defense of one-vs-all classification. *Journal of Machine Learning Research*, 5:101–141, 2004.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115:211–252, 2015.

Daniel Shalam and Simon Korman. The self-optimal-transport feature transform. *arXiv preprint: 2204.03065*, 2022.

Shai Shalev-Shwartz, Koby Crammer, Ofer Dekel, and Yoram Singer. Online passive-aggressive algorithms. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems*, volume 16, pages 1–8. MIT Press, 2003.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, 2017.

Ihsan Ullah, Dustin Carrión-Ojeda, Sergio Escalera, Isabelle Guyon, Mike Huisman, Felix Mohr, Jan N. van Rijn, Haozhe Sun, Joaquin Vanschoren, and Phan Anh Vu. Meta-Album: Multi-domain meta-dataset for few-shot image classification. In *Proceedings of* *the 36th Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. URL https://meta-album.github.io/.

Joaquin Vanschoren, Jan N. van Rijn, Bernd Bischl, and Luis Torgo. OpenML: Networked science in machine learning. *ACM SIGKDD Explorations Newsletter*, 15(2):49–60, 2014.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, koray kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29, 2016.

Weihao Yu, Mi Luo, Pan Zhou, Chenyang Si, Yichen Zhou, Xinchao Wang, Jiashi Feng, and Shuicheng Yan. MetaFormer is actually what you need for vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10819–10829, 2022.

Richard Zhang. Making convolutional networks shift-invariant again. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 7324–7334. PMLR, 2019.

Appendix A. Analysis of "De Novo" Training with Extra Time

Since Figure 2 exhibits a significant difference between the average normalized accuracy obtained by the models using "de novo" training (Meta-learning league) and the ones using pre-trained weights on ImageNet (Free-style league), we allowed the former models to run for twice the competition time limit (*i.e.,* 18 hours) to explore our hypothesis that this difference in performance could be caused by the difference in the training time of our competition and the training time used to obtain the pre-trained ImageNet weights. The results of this experiment are presented in Figure 3. These results show that even with twice the time, the models with "de novo" training only improved marginally at most and could not achieve the same average normalized accuracy obtained by the pre-trained models. Moreover, the average normalized accuracy of Prototypical Networks is the same when using twice the original time. After a careful analysis of the results of this method, we found that the reason for this behavior is the convergence to a local optimum after approximately 3 hours of meta-training.

The results of this analysis suggest that the time limit of this competition does not cause the difference in average normalized accuracy between the Free-style and Meta-learning leagues. Therefore, as explained in Section 4.2, other possible causes include: (1) the quality and diversity of the Meta-Album training data, which may not be as comprehensive as the ImageNet training data; (2) the limited quantity of Meta-Album data available for meta-training during the Final phase, which is approximately 11 times less than the amount of data used for pre-training the models on ImageNet.

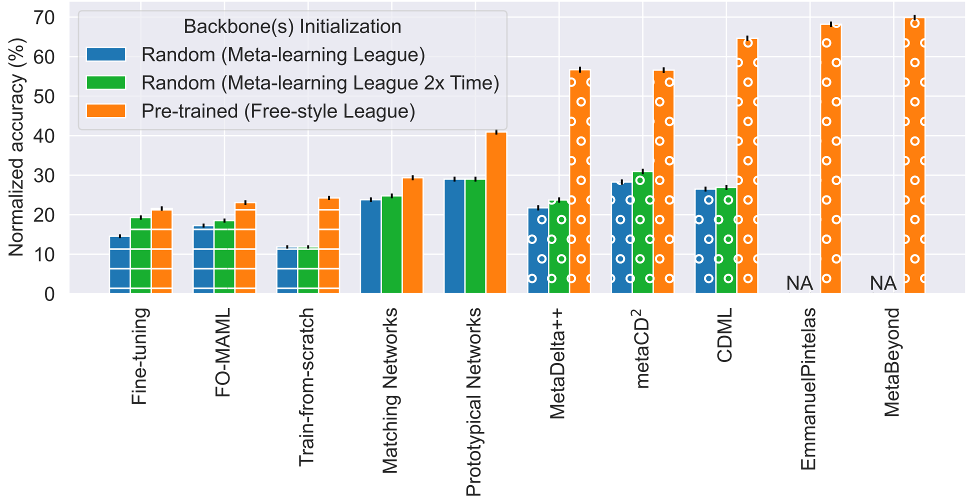

Figure 3: Analysis of the impact of doubling the time limit for the Meta-learning league (18 hours). Each bar shows the average normalized accuracy over 6 000 any-way any-shot meta-test tasks (600 tasks per dataset). The texture on the bars indicates the following: *horizontal lines* – baselines with a linear classifier, *no texture* – baselines with a nearest centroid classifier, and *circles* – based on MetaDelta++.

Appendix B. Competition Results per League

This competition had the following five leagues:

- **Free-style league:** Submit any solution (pre-trained models allowed).
- **Meta-learning league:** Submit a solution that uses "de novo" training (no pre-trained models allowed).
- **New-in-ML league:** Be a participant with less than 10 ML publications, none of which were ever accepted to the main track of a major conference.
- **Women league:** Special league to encourage women since they rarely enter challenges.
- **Participant of a rarely represented country league:** Be a participant from a country that is not in the top 10 most represented countries of Kaggle challenge participants.[11]

It is important to clarify that the teams could only make submissions to the Free-style and Meta-learning leagues, but they were not obligated to participate in both. Then, all the teams who managed to enter the Final phase of the competition in either the Free-style or the Meta-learning league were automatically selected for the remaining three leagues based on compliance with the specific requirements of each league by all team members. Table 2 shows the rank (top 3) of the teams described in Section 3 in each league.

Table 2: Detailed results per league. In the League column, F, M, N, W, and P stand for Free-style, Meta-learning, New-in-ML, Women, and Participant of a rarely represented country, respectively. For each league, the teams are ranked based on their worst average normalized accuracy (bold number) among three runs with different random seeds. The ranks with * indicate that not all team members fulfill the league requirements and, therefore, have less priority for prizes.

League	Rank	Team	Normalized Accuracy (%) seed = 1	Normalized Accuracy (%) seed = 2	Normalized Accuracy (%) seed = 3
F	1	MetaBeyond	69.96 ± 0.67	70.02 ± 0.67	**69.89 ± 0.67**
	2	EmmanuelPintelas	**68.20 ± 0.68**	68.66 ± 0.68	68.58 ± 0.68
	3	CDML	**64.63 ± 0.68**	64.72 ± 0.68	65.01 ± 0.67
M	1	metaCD^2	29.08 ± 0.69	28.63 ± 0.69	**28.25 ± 0.68**
	2	CDML	**26.47 ± 0.62**	27.45 ± 0.64	26.70 ± 0.63
N	1	metaCD^2	**56.58 ± 0.72**	56.82 ± 0.72	56.87 ± 0.72
	2*	MetaBeyond	69.96 ± 0.67	70.02 ± 0.67	**69.89 ± 0.67**
W	1*	MetaBeyond	69.96 ± 0.67	70.02 ± 0.67	**69.89 ± 0.67**
P	1	EmmanuelPintelas	**68.20 ± 0.68**	68.66 ± 0.68	68.58 ± 0.68
	2	metaCD^2	**56.58 ± 0.72**	56.82 ± 0.72	56.87 ± 0.72
	3*	CDML	**64.63 ± 0.68**	64.72 ± 0.68	65.01 ± 0.67

[11] https://towardsdatascience.com/kaggle-around-the-world-ccea741b2de2

Appendix C. Competition Results per Dataset

Table 3 presents the average normalized accuracy of the top-ranked methods and baselines on each meta-testing dataset. Since team MetaBeyond was the best overall, Figure 4 presents a detailed analysis of the improvement provided by this team on each meta-testing dataset with respect to MetaDelta++, which is the winning solution of our previous competition. These results illustrate how the solution of team MetaBeyond improved the normalized accuracy in almost all domains except for OCR, where this metric was decreased by 6 percentage points. Additionally, this figure provides an idea of the *intrinsic difficulty* of each dataset, which corresponds to the difference between the maximum theoretical normalized accuracy (100%) and the one achieved by the best method. The intrinsic difficulty indicates how much room for improvement is still available on each dataset. Therefore, proposing new methods that improve the normalized accuracy on datasets with low intrinsic difficulty, like *TEXT_ALOT* and *AWA*, is usually more challenging than focusing on datasets with high intrinsic difficulty, like *PRT* and *BTS*. However, another difficulty measure that should be considered is the *modeling difficulty*, which can be computed by subtracting the top of the blue bar (worst method) from the top of the green bar (best method). The modeling difficulty reveals the improvement in normalized accuracy caused by the proposed method. Thus, by analyzing this difficulty, we can see how the model of team MetaBeyond manages to improve significantly (> 10% points) the normalized accuracy in all datasets, including the most challenging one (*PRT*).

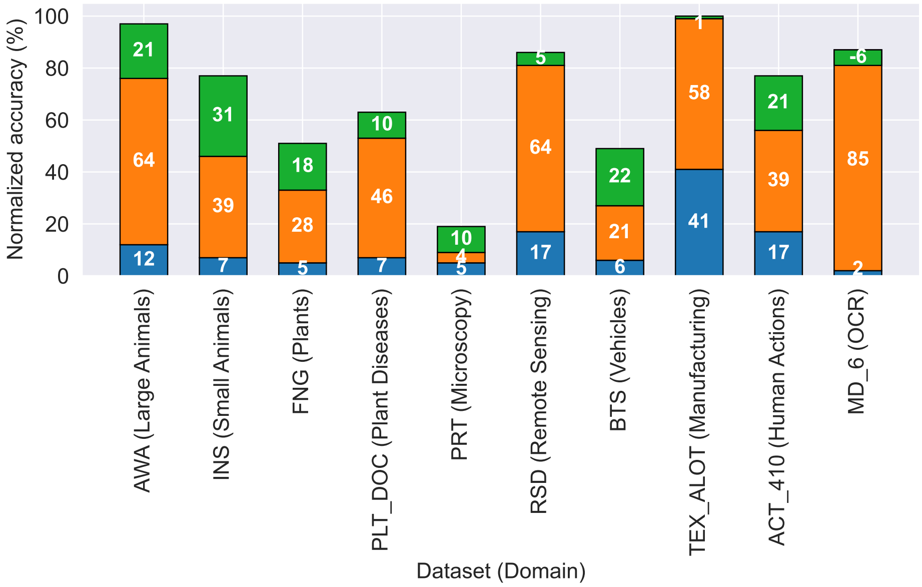

Figure 4: Improvement of team MetaBeyond on each meta-testing dataset. Each bar shows the average normalized accuracy computed over 600 any-way any-shot meta-test tasks. The numbers denote: blue bar – performance of the worst baseline (Train-from-scratch without pre-training), orange bar – improvement of the best baseline (MetaDelta++ with pre-training) over the worst baseline, green bar – improvement of the best method (team MetaBeyond) over the best baseline. The negative numbers indicate a deterioration instead of a performance improvement.

Table 3: Average normalized accuracy (%) for each meta-testing dataset (recorded on the worst run among three runs). The meta-testing datasets and their corresponding domain are: (1) *AWA* (Large Animals), (2) *INS* (Small Animals), (3) *FNG* (Plants), (4) *PLT_DOC* (Plant Diseases), (5) *PRT* (Microscopy), (6) *RSD* (Remote Sensing), (7) *BTS* (Vehicles), (8) *TEX_ALOT* (Manufacturing), (9) *ACT_410* (Human Actions), (10) *MD_6* (OCR). Bold values indicate the best method of each dataset for each league.

League	Team / Baseline	Dataset									
		1	2	3	4	5	6	7	8	9	10
Free-style	MetaBeyond	**96.57**	**76.55**	**51.13**	**62.86**	18.62	**85.78**	**48.82**	**99.89**	**77.42**	81.23
	EmmanuelPintelas	95.75	75.01	47.31	57.78	**18.89**	81.13	45.99	99.74	76.27	84.08
	CDML	94.08	66.46	41.61	58.19	18.66	80.27	41.92	99.71	70.45	74.95
	MetaDelta++	75.68	45.87	32.77	53.49	9.09	80.88	26.99	99.34	56.23	**86.52**
	metaCD^2	91.61	62.12	33.95	49.84	12.92	72.94	36.80	98.67	63.02	43.95
	Prototypical Networks	42.83	26.85	21.66	37.57	14.13	60.78	19.34	94.84	35.10	56.16
	Matching Networks	25.85	17.73	13.57	23.90	8.94	46.79	9.22	80.68	14.63	52.23
	Train-from-scratch	22.87	17.39	12.71	17.97	9.55	34.20	15.63	64.71	31.85	15.47
	FO-MAML	18.89	11.18	8.17	19.90	6.94	39.12	7.05	73.78	10.27	35.40
	Fine-tuning	20.49	13.72	10.63	14.38	2.08	34.53	9.29	65.40	18.33	26.85
Meta-learning	Prototypical Networks	**30.68**	**22.83**	**17.43**	22.75	9.74	46.50	**15.63**	84.44	32.58	7.39
	metaCD^2	28.66	16.73	14.54	**23.17**	6.59	**46.57**	13.09	**92.62**	32.14	**8.42**
	CDML	26.85	19.48	15.16	17.40	**11.37**	42.19	13.14	83.76	**33.24**	2.15
	Matching Networks	25.72	15.90	12.51	15.58	10.00	38.09	11.57	78.90	26.01	3.63
	MetaDelta++	21.13	10.93	9.70	14.34	7.15	36.09	6.06	87.14	23.91	0.94
	FO-MAML	18.88	10.77	9.32	10.15	10.10	29.11	8.18	58.65	15.54	1.77
	Fine-tuning	14.45	8.72	8.93	8.56	2.36	28.02	5.57	52.11	14.73	2.07
	Train-from-scratch	11.78	6.52	4.91	7.04	5.13	17.36	6.27	40.76	16.59	2.02

Appendix D. Ablation Studies of Baseline Methods

We conducted ablation studies on the meta-training strategy and classifier type used by the baselines except for MetaDelta++. This is due to the winning solution of our previous competition, MetaDelta++ being too complex to modify in the same manner as the other baselines to have a fair comparison. Similarly, we excluded Matching Networks because of their similarity with Prototypical Networks, and as shown in Figure 2, the latter method achieved better results. For the meta-training strategy, we considered three cases: None (*i.e.,* the meta-training phase is skipped), batch learning (*i.e.,* the model is meta-trained with batches), or episodic learning (*i.e.,* the model is meta-trained with N-way k-shot tasks). Further, we compared linear and nearest centroid classifiers (NCC) for the classifier type as they are part of the baselines. Moreover, inspired by the top-ranked methods, we analyzed the impact of freezing different blocks of the backbone model. Due to time constraints, we only used Prototypical Networks for this analysis since it is the best among the studied baselines. Figure 5 shows that freezing 8 of the 9 blocks of ResNet-18 (the backbone used by all analyzed methods) led to the highest normalized accuracy. Therefore, we conducted all the previously mentioned ablation studies for two scenarios: (1) using an unfrozen ResNet-18 and (2) using a ResNet-18 with 8 of its 9 blocks frozen, referred to as “Frozen Backbone”.

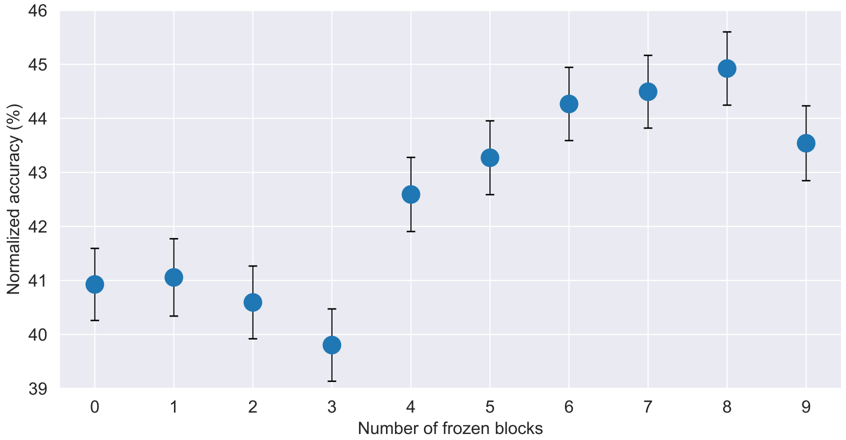

Figure 5: Analysis of the impact of freezing a varying number of ResNet-18 blocks (9 blocks in total) using Prototypical Networks. Each dot shows the average normalized accuracy over 6 000 any-way any-shot meta-test tasks (600 tasks per dataset) and the bars correspond to 95% confidence intervals.

Table 4 shows the results for all the experiments (see Appendix E for marginalized comparisons). Note that the results of *Train-from-scratch* + *Batch/Episodic learning* + *Linear classifier/NCC* are not included because, by the definition in Section 4.1, this method does not perform any meta-training. Similarly, *Prototypical Networks* must use a NCC. By analyzing the results, we found that when the unfrozen backbone was used, the best results (**43.54** ± **0.40** %) were achieved by *Train-from-scratch* + *No meta-training* + *NCC*, *Prototypical Networks* + *No meta-training* + *NCC*, and *FO-MAML* + *No meta-training* + *NCC*. It is important to highlight that these three combinations achieved the same results

Table 4: Ablation study of the meta-training strategy and classifier type of the baselines using a pre-trained ResNet-18 backbone, either unfrozen or 8 out of 9 blocks frozen. The normalized accuracy is averaged over 18 000 any-way any-shot meta-test tasks (3 runs of 6 000 tasks). Bold values indicate the best results in each case.

	Meta-training Strategy			Classifier		Normalized Accuracy (%)	
Method	None	Batch	Episodic	Lin.	NCC	Unfrozen Backbone	Frozen Backbone
Train-from-scratch	✓			✓		24.35 ± 0.31	45.93 ± 0.39
	✓				✓	$\mathbf{43.54 \pm 0.40}$	43.54 ± 0.40
Prototypical Networks	✓				✓	$\mathbf{43.54 \pm 0.40}$	43.54 ± 0.40
		✓			✓	34.13 ± 0.37	40.87 ± 0.38
			✓		✓	40.93 ± 0.39	44.92 ± 0.39
Fine-tuning	✓			✓		38.56 ± 0.41	38.56 ± 0.41
	✓				✓	41.44 ± 0.39	41.44 ± 0.39
		✓		✓		23.24 ± 0.33	42.91 ± 0.41
		✓			✓	36.99 ± 0.38	44.77 ± 0.40
			✓	✓		3.03 ± 0.12	20.75 ± 0.34
			✓		✓	37.12 ± 0.37	45.30 ± 0.40
FO-MAML	✓			✓		20.32 ± 0.35	19.08 ± 0.34
	✓				✓	$\mathbf{43.54 \pm 0.40}$	43.54 ± 0.40
		✓		✓		18.72 ± 0.34	16.33 ± 0.30
		✓			✓	22.14 ± 0.34	39.85 ± 0.40
			✓	✓		23.33 ± 0.34	27.03 ± 0.36
			✓		✓	43.00 ± 0.38	$\mathbf{47.01 \pm 0.40}$

because they are equivalent when the meta-training phase is omitted. They are equivalent because the NCC is not trained during the meta-testing phase; instead, it only generates the centroids of each class using the support set and assigns the labels to the query set based on the closest centroid using the Euclidean distance. However, when the meta-training phase is included, the strategy applied during meta-training varies for each method.

These results indicate that even if a meta-dataset is available for meta-training and contains the same domains as the meta-testing set, it is tricky to gain much benefit from it because the pre-trained backbones are already good and trained on a lot of data as discussed in Section 4.2. Therefore, when starting with pre-trained weights, there is a risk of overfitting the meta-training set. However, this problem can be alleviated by freezing some blocks of the backbone. Thus, when only the last block of the pre-trained ResNet-18 is updated, *FO-MAML + Episodic learning + NCC* improved the previously mentioned best performance by 3.47% points. Overall, using the frozen backbone boosted the performance of almost all cases, highlighting the importance of regularization during meta-training.

Additionally, the results presented in Table 4 indicate that episodic learning significantly improved performance across almost all tested ablation combinations. Also, the best result among the baseline configurations was achieved using episodic learning, as previously mentioned. It is worth noting, however, that the solutions of the top-ranked teams do not rely on episodic learning, which is consistent with the previous MetaDL competition (El Baz et al., 2022).

Finally, our ablations are consistent with the findings of Li et al. (2022) since, in both studies, using a NCC instead of a linear classifier tends to produce better results in cross-domain few-shot learning. However, it is important to note that the superiority of NCCs over linear classifiers may depend on the specific scenario. Therefore, further research is needed to fully explore the potential benefits and limitations of NCCs in few-shot learning. Nevertheless, our findings suggest that NCCs should be seriously considered as a viable alternative to linear classifiers in cross-domain few-shot learning tasks.

Appendix E. Marginalized Comparisons for Baseline Ablations

In this section, we adopt a marginalized approach to compare the individual components of the baseline ablation studies in Appendix D rather than analyzing them jointly. This marginalized analysis could offer insights into the design of novel algorithms. Figure 6, Figure 7, and Figure 8 are based on Table 4. Moreover, in all these figures, "frozen" means that 8 out of the 9 blocks of the ResNet-18 backbone are frozen, whereas "unfrozen" denotes that the whole ResNet-18 backbone is updated.

E.1. During meta-training, freezing part of the backbone or not?

Figure 6 compares the overall effect of freezing 8 of the 9 blocks of the ResNet-18 backbone. This result confirms the clear advantage of only fine-tuning the last block over updating the whole backbone.

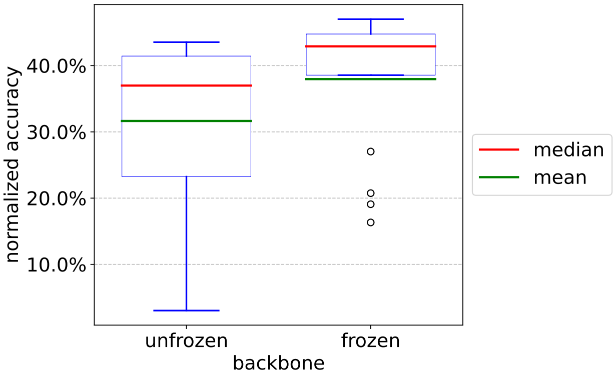

Figure 6: Boxplot for the impact of freezing part of the ResNet-18 backbone.

E.2. What meta-training strategy to use?

Figure 7 compares the effect of different meta-training strategies. The left part of this figure shows that episodic learning outperforms batch learning with both mean and median normalized accuracy when using an unfrozen backbone. However, skipping the meta-training phase is better than batch and episodic learning, meaning that in this case, using the provided meta-training dataset harms the performance despite being more similar to the meta-testing data than ImageNet (dataset used to obtain the pre-trained weights of the backbone). This result highlights the importance of using regularization techniques during meta-training, *e.g.,* fine-tuning only the last few layers of the backbones, adding weight decay, and dropout.

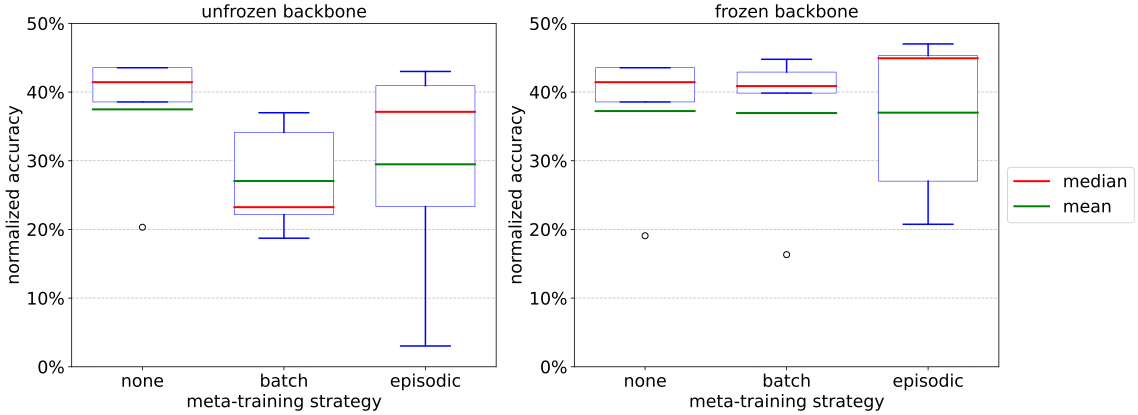

Figure 7: Boxplot for the effect of different meta-training strategies. The meta-training strategies are: "none" – the meta-training phase is skipped, "batch" – the backbone is meta-trained with batches, "episodic" – the backbone is meta-trained with episodes.

On the other hand, the right part of Figure 7 shows that with a frozen backbone, the mean normalized accuracy is almost the same for the three evaluated meta-training strategies. However, episodic learning exceeds the other meta-training strategies in median normalized accuracy. Furthermore, although episodic learning has the highest variance, it is still the best meta-learning strategy when including the outliers. In summary, episodic learning should be considered when designing novel few-shot learning algorithms, as it has the potential to improve performance if applied judiciously.

E.3. What classifier to use?

Figure 8 compares the effect of using different classifiers (linear and NCC). The NCC achieves better mean, median, best-case, and worst-case normalized accuracy with frozen and unfrozen backbones. Thus, this classifier should be considered a viable alternative to linear classifiers for few-shot learning.

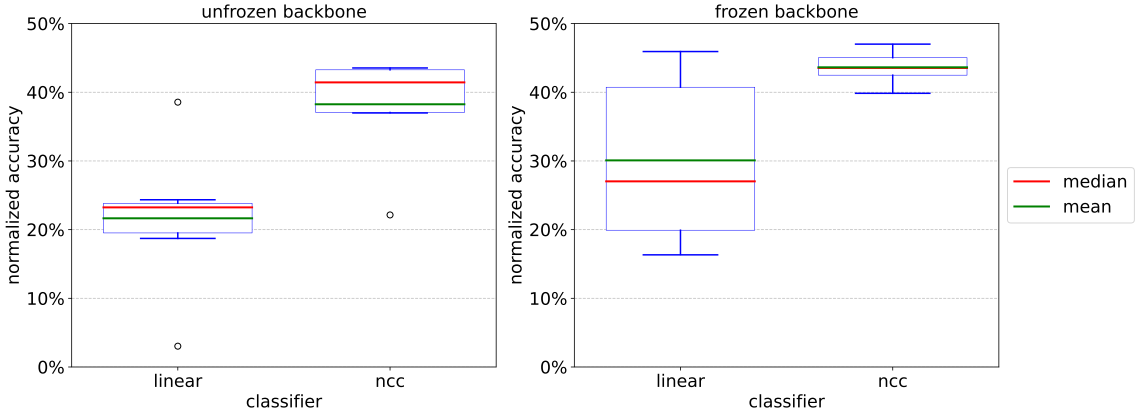

Figure 8: Boxplot for the effect of using a linear classifier or a NCC.