Advances and Challenges in Meta-Learning: A Technical Review

Anna Vettoruzzo, Mohamed-Rafik Bouguelia, Joaquin Vanschoren, Thorsteinn Rögnvaldsson, *Senior Member, IEEE*, and KC Santosh, *Senior Member, IEEE*

***Abstract*—Meta-learning empowers learning systems with the ability to acquire knowledge from multiple tasks, enabling faster adaptation and generalization to new tasks. This review provides a comprehensive technical overview of meta-learning, emphasizing its importance in real-world applications where data may be scarce or expensive to obtain. The article covers the state-of-the-art meta-learning approaches and explores the relationship between meta-learning and multi-task learning, transfer learning, domain adaptation and generalization, self-supervised learning, personalized federated learning, and continual learning. By highlighting the synergies between these topics and the field of meta-learning, the article demonstrates how advancements in one area can benefit the field as a whole, while avoiding unnecessary duplication of efforts. Additionally, the article delves into advanced meta-learning topics such as learning from complex multi-modal task distributions, unsupervised meta-learning, learning to efficiently adapt to data distribution shifts, and continual meta-learning. Lastly, the article highlights open problems and challenges for future research in the field. By synthesizing the latest research developments, this article provides a thorough understanding of meta-learning and its potential impact on various machine learning applications. We believe that this technical overview will contribute to the advancement of meta-learning and its practical implications in addressing real-world problems.**

***Index Terms*—Deep neural networks, few-shot learning, meta-learning, representation learning, transfer learning.**

I. Introduction

A. Context and motivation

DEEP representation learning has revolutionized the field of machine learning by enabling models to learn effective features from data. However, it often requires large amounts of data for solving a specific task, making it impractical in scenarios where data is scarce or costly to obtain. Most existing approaches rely on either supervised learning of a representation tailored to a single task, or unsupervised learning of a representation that captures general features that may not be well-suited to new tasks. Furthermore, learning from scratch for each task is often not feasible, especially in domains such as medicine, robotics, and rare language translation where data availability is limited.

To overcome these challenges, meta-learning has emerged as a promising approach. Meta-learning enables models to quickly adapt to new tasks, even with few examples, and generalize across them. While meta-learning shares similarities with transfer learning and multitask learning, it goes beyond these approaches by enabling a learning system to *learn how to learn*. This capability is particularly valuable in settings where data is scarce, costly to obtain, or where the environment is constantly changing. While humans can rapidly acquire new skills by leveraging prior experience and are therefore considered *generalists*, most deep learning models are still *specialists* and are limited to performing well on specific tasks. Meta-learning bridges this gap by enabling models to efficiently adapt to new tasks.

B. Contribution

This review article primarily discusses the use of meta-learning techniques in deep neural networks to learn reusable representations, with an emphasis on few-shot learning; it does not cover topics such as AutoML and Neural Architecture Search [1], which are out of scope. Similarly, even though meta-learning is often applied in the context of reinforcement learning [2], [3], it falls outside the scope of this article. Distinct from existing surveys on meta-learning, such as [4], [5], [6], [7], [8], this review article highlights several key differentiating factors:

- *Inclusion of advanced meta-learning topics:* In addition to covering fundamental aspects of meta-learning, this review article delves into advanced topics such as learning from multimodal task distributions, meta-learning without explicit task information, learning without data sharing among clients, adapting to distribution shifts, and continual learning from a stream of tasks. By including these advanced topics, our article provides a comprehensive understanding of the current state-of-the-art and highlights the challenges and opportunities in these areas.
- *Detailed exploration of relationship with other topics:* We not only examine meta-learning techniques but also

Manuscript received 10 July 2023; revised 10 November 2023; accepted 21 January 2024. Date of publication 24 January 2024; date of current version 5 June 2024. This work was supported by the "Knowledge Foundation" (KK-stiftelsen). Recommended for acceptance by M. Cho. *(Corresponding author: Anna Vettoruzzo.)*

Anna Vettoruzzo, Mohamed-Rafik Bouguelia, and Thorsteinn Rögnvaldsson are with the Center for Applied Intelligent Systems Research (CAISR), Halmstad University, 301 18 Halmstad, Sweden (e-mail: anna.vettoruzzo@hh.se; mohamed-rafik.bouguelia@hh.se; thorsteinn.rognvaldsson@hh.se).

Joaquin Vanschoren is with the Automated Machine Learning Group, Eindhoven University of Technology, 5612 AZ Eindhoven, The Netherlands (e-mail: joaquin.vanschoren@gmail.com).

KC Santosh is with the Applied AI Research Lab, Department of Computer Science, University of South Dakota, Vermillion, SD 57069 USA (e-mail: kc.santosh@usd.edu).

Digital Object Identifier 10.1109/TPAMI.2024.3357847

establish clear connections between meta-learning and related areas, including transfer learning, multitask learning, self-supervised learning, personalized federated learning, and continual learning. This exploration of the relationships and synergies between meta-learning and these important topics provides valuable insights into how meta-learning can be efficiently integrated into broader machine learning frameworks.
- *Clear and concise exposition:* Recognizing the complexity of meta-learning, this review article provides a clear and concise explanation of the concepts, techniques and applications of meta-learning. It is written with the intention of being accessible to a wide range of readers, including both researchers and practitioners. Through intuitive explanations, illustrative examples, and references to seminal works, we facilitate readers' understanding of the foundation of meta-learning and its practical implications.
- *Consolidation of key information:* As a fast-growing field, meta-learning has information scattered across various sources. This review article consolidates the most important and relevant information about meta-learning, presenting a comprehensive overview in a single resource. By synthesizing the latest research developments, this survey becomes an indispensable guide to researchers and practitioners seeking a thorough understanding of meta-learning and its potential impact on various machine learning applications.

By highlighting these contributions, this article complements existing surveys and offers unique insights into the current state and future directions of meta-learning.

C. Organization

In this article, we provide the foundations of modern deep learning methods for learning across tasks. To do so, we first define the key concepts and introduce relevant notations used throughout the article in Section II. Then, we cover the basics of multitask learning and transfer learning and their relation to meta-learning in Section III. In Section IV, we present an overview of the current state of meta-learning methods and provide a unified view that allows us to categorize them into three types: black-box meta-learning methods, optimization-based meta-learning methods, and meta-learning methods that are based on distance metric learning [9]. In Section V, we delve into advanced meta-learning topics, explaining the relationship between meta-learning and other important machine learning topics, and addressing issues such as learning from multimodal task distributions, performing meta-learning without provided tasks, learning without sharing data across clients, learning to adapt to distribution shifts, and continual learning from a stream of tasks. Finally, the article explores the application of meta-learning to real-world problems and provides an overview of the landscape of promising frontiers and yet-to-be-conquered challenges that lie ahead. Section VI focuses on these challenges, shedding light on the most pressing questions and future research opportunities.

II. Basic Notations and Definitions

In this section, we introduce some simple notations which will be used throughout the article and provide a formal definition of the term "task" within the scope of this article.

We use θ (and sometimes also ϕ) to represent the set of parameters (weights) of a deep neural network model. $\mathcal{D} = \{(x_j, y_j)\}_{j=1}^{n}$ denotes a dataset, where inputs x_j are sampled from the distribution $p(x)$ and outputs y_j are sampled from $p(y|x)$. The function $\mathcal{L}(., .)$ denotes a loss function, for example, $\mathcal{L}(\theta, \mathcal{D})$ represents the loss achieved by the model's parameters θ on the dataset $\mathcal{D}$. The symbol $\mathcal{T}$ refers to a task, which is primarily defined by the data-generating distributions $p(x)$ and $p(y|x)$ that define the problem.

In a standard supervised learning scenario, the objective is to optimize the parameters θ by minimizing the loss $\mathcal{L}(\theta, \mathcal{D})$, where the dataset $\mathcal{D}$ is derived from a single task $\mathcal{T}$, and the loss function $\mathcal{L}$ depends on that task. Formally, in this setting, a task $\mathcal{T}_i$ is a triplet $\mathcal{T}_i \triangleq \{p_i(x), p_i(y|x), \mathcal{L}_i\}$ that includes task-specific data-generating distributions $p_i(x)$ and $p_i(y|x)$, as well as a task-specific loss function $\mathcal{L}_i$. The goal is to learn a model that performs well on data sampled from task $\mathcal{T}_i$. In a more challenging setting, we consider learning from multiple tasks $\{\mathcal{T}_i\}_{i=1}^{T}$, which involves (a dataset of) multiple datasets $\{\mathcal{D}_i\}_{i=1}^{T}$. In this scenario, a set of training tasks is used to learn a model that performs well on test tasks. Depending on the specific setting, a test task can either be sampled from the training tasks or completely new, never encountered during the training phase.

In general, tasks can differ in various ways depending on the application. For example, in image recognition, different tasks can involve recognizing handwritten digits or alphabets from different languages [2], [10], while in natural language processing, tasks can include sentiment analysis [11], [12], machine translation [13], and chatbot response generation [14], [15], [16]. Tasks in robotics can involve training robots to achieve different goals [17], while in automated feedback generation, tasks can include providing feedback to students on different exams [18]. It is worth noting that tasks can share structures, even if they appear unrelated. For example, the laws of physics underlying real data, the language rules underlying text data, and the intentions of people all share common structures that enable models to transfer knowledge across seemingly unrelated tasks.

III. From Multitask and Transfer to Meta-Learning

Meta-learning, multitask learning, and transfer learning encompass different approaches aimed at learning across multiple tasks. Multitask learning aims to improve performance on a set of tasks by learning them simultaneously. Transfer learning fine-tunes a pre-trained model on a new task with limited data. In contrast, meta-learning acquires useful knowledge from past tasks and leverages it to learn new tasks more efficiently. In this section, we transition from discussing "multitask learning" and "transfer learning" to introducing the topic of "meta-learning".

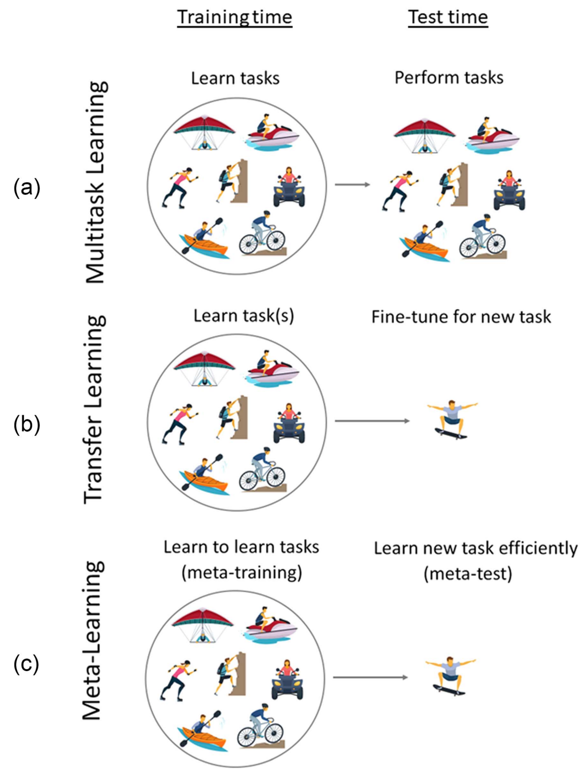


Fig. 1. Multitask learning vs transfer learning vs meta-learning.

A. *Multitask Learning Problem*

As illustrated in Fig. 1(a), multitask learning (MTL) trains a model to perform multiple related tasks simultaneously, leveraging shared structure across tasks, and improving performance compared to learning each task individually. In this setting, there is no distinction between training and test tasks, and we refer to them as $\{\mathcal{T}_i\}_{i=1}^T$.

One common approach in MTL is hard parameter sharing, where the model parameters θ are split into shared θ^{sh} and task-specific θ^i parameters. These parameters are learned simultaneously through an objective function that takes the form:

$$\min_{\theta^{\text{sh}},\theta^1,\ldots,\theta^T} \sum_{i=1}^{T} w_i \mathcal{L}_i(\{\theta^{\text{sh}}, \theta^i\}, \mathcal{D}_i),$$

where w_i can weight tasks differently. This approach is often implemented using a multi-headed neural network architecture, where a shared encoder (parameterized by θ^{sh}) is responsible for feature extraction. This shared encoder subsequently branches out into task-specific decoding heads (parameterized by θ^i) dedicated to individual tasks $\mathcal{T}_i$ [19], [20], [21].

Soft parameter sharing is another approach in MTL that encourages parameter similarity across task-specific models using regularization penalties [22], [23], [24]. In this approach, each task typically has its own model with its own set of parameters θ^i, while the shared parameters set θ^{sh} can be empty. The objective function is similar to that of hard parameter sharing, but with an additional regularization term that controls the strength of parameter sharing across tasks. The strength of regularization is determined by the hyperparameter λ. In the case of L_2 regularization, the objective function is given by:

$$\min_{\theta^{\text{sh}},\theta^1,\ldots,\theta^T} \sum_{i=1}^{T} w_i \mathcal{L}_i(\{\theta^{\text{sh}}, \theta^i\}, \mathcal{D}_i) + \lambda \sum_{i'=1}^{T} \|\theta^i - \theta^{i'}\|.$$

However, soft parameter sharing can be more memory-intensive as separate sets of parameters are stored for each task, and it requires additional design decisions and hyperparameters.

Another approach to sharing parameters is to condition a single model on a task descriptor z_i that contains task-specific information used to modulate the network's computation. The task descriptor z_i can be a simple one-hot encoding of the task index or a more complex task specification, such as language description or user attributes. When a task descriptor is provided, it is used to modulate the weights of the shared network with respect to the task at hand. Through this modulation mechanism, the significance of the shared features is determined based on the particular task, enabling the learning of both shared and task-specific features in a flexible manner. Such an approach grants fine-grained control over the adjustment of the network's representation, tailoring it to each individual task. Various methods for conditioning the model on the task descriptor are described in [25]. More complex methods are also provided in [26], [27], [28].

Choosing the appropriate approach for parameter sharing, determining the level of the network architecture at which to share parameters, and deciding on the degree of parameter sharing across tasks are all design decisions that depend on the problem at hand. Currently, these decisions rely on intuition and knowledge of the problem, making them more of an art than a science, similar to the process of tuning neural network architectures. Moreover, multitask learning presents several challenges, such as determining which tasks are complementary, particularly in scenarios with a large number of tasks, as in [29]. Interested readers can find a more comprehensive discussion of multitask learning in [30], [31].

In summary, multitask learning aims to learn a set of T tasks $\{\mathcal{T}_i\}_{i=1}^T$ at once. Even though the model can generalize to new data from these T tasks, it might not be able to handle a completely new task that it has not been trained on. This is where transfer learning and meta-learning become more relevant.

B. *Transfer Learning Via Fine-Tuning*

Transfer learning is a valuable technique that allows a model to leverage representations learned from one or more source tasks to solve a target task. As illustrated in Fig. 1(b), the main goal is to use the knowledge learned from the source task(s) $\mathcal{T}_a$ to improve the performance of the model on a new task, usually referred to as the target task $\mathcal{T}_b$, especially when the target task dataset $\mathcal{D}_b$ is limited. In practice, the source task data $\mathcal{D}_a$ is often inaccessible, either because it is too expensive to obtain or too large to store.

One common approach for transfer learning is fine-tuning, which involves starting with a model that has been pre-trained on the source task dataset $\mathcal{D}_a$. The parameters of the pre-trained model, denoted as θ, are then fine-tuned on the training data

$\mathcal{D}_b$ from the target task $\mathcal{T}_b$ using gradient descent or any other optimizer for several optimization steps. An example of the fine-tuning process for one gradient descent step is expressed as follows:

$$\phi \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}(\theta, \mathcal{D}_b),$$

where ϕ denotes the parameters fine-tuned for task $\mathcal{T}_b$, and α is the learning rate.

Models with pre-trained parameters θ are often available online, including models pre-trained on large datasets such as ImageNet for image classification [32] and language models like BERT [33], PaLM [34], LLaMA [35], and GPT-4 [36], trained on large text corpora. Models pre-trained on other large and diverse datasets or using unsupervised learning techniques, as discussed in Section V-C, can also be used as a starting point for fine-tuning.

However, as discussed in [37], it is crucial to avoid destroying initialized features when fine-tuning. Some design choices, such as using a smaller learning rate for earlier layers, freezing earlier layers and gradually unfreezing, or re-initializing the last layer, can help to prevent this issue. Recent studies such as [38] show that fine-tuning the first or middle layers can sometimes work better than fine-tuning the last layers, while others recommend a two-step process of training the last layer first and then fine-tuning the entire network [37]. More advanced approaches, such as STILTs [39], propose an intermediate step of further training the model on a labeled task with abundant data to mitigate the potential degradation of pre-trained features.

In [40], it was demonstrated that transfer learning via fine-tuning may not always be effective, particularly when the target task dataset is very small or very different from the source tasks. To investigate this, the authors fine-tuned a pre-trained universal language model on specific text corpora corresponding to new tasks using varying numbers of training examples. Their results showed that starting with a pre-trained model outperformed training from scratch on the new task. However, when the size of the new task dataset was very small, fine-tuning on such a limited number of examples led to poor generalization performance. To address this issue, meta-learning can be used to learn a model that can effectively adapt to new tasks with limited data by leveraging prior knowledge from other tasks. In fact, meta-learning is particularly useful for learning new tasks from very few examples, and we will discuss it in more detail in the remainder of this article.

C. Meta-Learning Problem

Meta-learning (or learning to learn) is a field that aims to surpass the limitations of traditional transfer learning by adopting a more sophisticated approach that explicitly optimizes for transferability. As discussed in Section III-B, traditional transfer learning involves pre-training a model on source tasks and fine-tuning it for a new task. In contrast, meta-learning trains a network to efficiently learn or adapt to new tasks with only a few examples. Fig. 1(c) illustrates this approach, where at meta-training time we *learn to learn* tasks, and at meta-test time we *learn* a new task efficiently.

During the meta-training phase, prior knowledge enabling efficient learning of new tasks is extracted from a set of training tasks $\{\mathcal{T}_i\}_{i=1}^T$. This is achieved by using a meta-dataset consisting of multiple datasets $\{\mathcal{D}_i\}_{i=1}^T$, each corresponding to a different training task. At meta-test time, a small training dataset $\mathcal{D}_{\text{new}}$ is observed from a completely new task $\mathcal{T}_{\text{new}}$ and used in conjunction with the prior knowledge to infer the most likely posterior parameters. As in transfer learning, accessing prior tasks at meta-test time is impractical. Although the datasets $\{\mathcal{D}_i\}_i$ come from different data distributions (since they come from different tasks $\{\mathcal{T}_i\}_i$), it is assumed that the tasks themselves (both for training and testing) are drawn i.i.d. from an underlying task distribution $p(\mathcal{T})$, implying some similarities in the task structure. This assumption ensures the effectiveness of meta-learning frameworks even when faced with limited labeled data. Moreover, the more tasks that are available for meta-training, the better the model can learn to adapt to new tasks, just as having more data improves performance in traditional machine learning.

In the next section, we provide a more formal definition of meta-learning and various approaches to it.

IV. Meta-Learning Methods

To gain a unified understanding of the meta-learning problem, we can draw an analogy to the standard supervised learning setting. In the latter, the goal is to learn a set of parameters ϕ for a base model h_ϕ (e.g., a neural network parametrized by ϕ), which maps input data $x \in \mathcal{X}$ to the corresponding output $y \in \mathcal{Y}$ as follows:

$$\begin{aligned} h_\phi: \ & \mathcal{X} \rightarrow \mathcal{Y} \\ & x \mapsto y = h_\phi(x). \end{aligned} \tag{1}$$

To accomplish this, a typically large training dataset $\mathcal{D} = \{(x_j, y_j)\}_{j=1}^n$ specific to a particular task $\mathcal{T}$ is used to learn ϕ.

In the meta-learning setting, the objective is to learn prior knowledge, which consists of a set of meta-parameters θ, for a procedure $\mathcal{F}_\theta(\mathcal{D}_i^{\text{tr}}, x^{\text{ts}})$. This procedure uses θ to efficiently learn from (or adapt to) a small training dataset $\mathcal{D}_i^{\text{tr}} = \{(x_k, y_k)\}_{k=1}^K$ from a task $\mathcal{T}_i$, and then make accurate predictions on unlabeled test data x^{ts} from the same task $\mathcal{T}_i$. As we will see in the following sections, $\mathcal{F}_\theta$ is typically composed of two functions: (1) a meta-learner $f_\theta(.)$ that produces task-specific parameters $\phi_i \in \Phi$ from $\mathcal{D}_i^{\text{tr}} \in \mathcal{X}^K$, and (2) a base model $h_{\phi_i}(.)$ that predicts outputs corresponding to the data in x^{ts}:

$$\begin{aligned} f_\theta: \ & \mathcal{X}^K \rightarrow \Phi \\ & \mathcal{D}_i^{\text{tr}} \mapsto \phi_i = f_\theta(\mathcal{D}_i^{\text{tr}}), \end{aligned} \qquad \begin{aligned} h_{\phi_i}: \ & \mathcal{X} \rightarrow \mathcal{Y} \\ & x \mapsto y = h_{\phi_i}(x). \end{aligned} \tag{2}$$

Note that the process of obtaining task-specific parameters $\phi_i = f_\theta(\mathcal{D}_i^{\text{tr}})$ is often referred to as "*adaptation*" in the literature, as it adapts to the task $\mathcal{T}_i$ using a small amount of data while leveraging the prior knowledge summarized in θ. The objective of meta-training is to learn the set of meta-parameters θ. This is accomplished by using a meta-dataset $\{\mathcal{D}_i\}_{i=1}^T$, which consists of a dataset of datasets, where each dataset $\mathcal{D}_i = \{(x_j, y_j)\}_{j=1}^n$ is specific to a task $\mathcal{T}_i$.

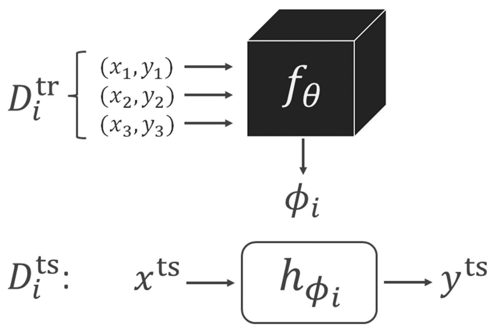


Fig. 2. Black-box meta-learning.

Algorithm 1: Black-Box Meta-Learning.

```
Randomly initialize θ
while not done do
    Sample a task T_i ~ p(T)   (or a mini-batch of tasks)
    Sample disjoint datasets D_i^tr, D_i^ts from T_i
    Compute φ_i ← f_θ(D_i^tr)
    Update θ using ∇_θ L(φ_i, D_i^ts)
end while
return θ
```

The unified view of meta-learning presented here is beneficial because it simplifies the meta-learning problem by reducing it to the design and optimization of $\mathcal{F}_\theta$. Moreover, it facilitates the categorization of the various meta-learning approaches into three categories: black-box meta-learning methods, optimization-based meta-learning methods, and distance metric-based meta-learning methods (as discussed in [9]). An overview of these categories is provided in the subsequent sections.

A. Black-Box Meta-Learning Methods

Black-box meta-learning methods, also known as model-based meta-learning [7], [9], represent f_θ as a black-box neural network that takes the entire training dataset, $\mathcal{D}_i^{\text{tr}}$, and predicts task-specific-parameters, ϕ_i. These parameters are then used to parameterize the base network, h_{ϕ_i}, and make predictions for test data-points, $y^{\text{ts}} = h_{\phi_i}(x^{\text{ts}})$. The architecture of this approach is shown in Fig. 2. The meta-parameters, θ, are optimized as shown in (3), and a general algorithm for these kinds of black-box methods is outlined in Algorithm 1.

$$\min_\theta \sum_{\mathcal{T}_i} \mathcal{L}(\underbrace{f_\theta(\mathcal{D}_i^{\text{tr}})}_{\phi_i}, \mathcal{D}_i^{\text{ts}}). \tag{3}$$

However, this approach faces a major challenge: outputting all the parameters ϕ_i of the base network h_{ϕ_i} is not scalable and is impractical for large-scale models. To overcome this issue, black-box meta-learning methods, such as MANN [41] and SNAIL [42], only output sufficient statistics instead of the complete set of parameters of the base network. These methods allow f_θ to output a low-dimensional vector z_i that encodes contextual task information, rather than a full set of parameters ϕ_i. In this case, ϕ_i consists of $\{z_i, \theta_h\}$, where θ_h denotes the trainable parameters of the network h_{ϕ_i}. The base network h_{ϕ_i} is modulated with task descriptors by using various techniques for *conditioning on task descriptors* discussed in Section III-A.

Several black-box meta-learning methods adopt different neural network architectures to represent f_θ. For instance, methods described in [41], use LSTMs or architectures with augmented memory capacities, such as Neural Turing Machines, while others, like Meta Networks [43], employ external memory mechanisms. SNAIL [42] defines meta-learner architectures that leverage temporal convolutions to aggregate information from past experience and attention mechanisms to pinpoint specific pieces of information. Alternatively, some methods, such as the one proposed in [44], use a feedforward plus averaging strategy. This latter feeds each data-point in $\mathcal{D}_i^{\text{tr}} = \{(x_j, y_j)\}_{j=1}^K$ through a neural network to produce a representation r_j for each data-point, and then averages these representations to create a task representation $z_i = \frac{1}{K}\sum_{j=1}^K r_j$. This strategy may be more effective than using a recurrent model such as LSTM, as it does not rely on the assumption of temporal relationships between data-points in $\mathcal{D}_i^{\text{tr}}$.

Recent research efforts [45], [46], [47], [48] have explored the connection between in-context learning and black-box meta-learning methods. This connection reveals that in-context learning can be viewed as a special instance of the broader meta-learning paradigm [46]. In particular, in-context learning involves training models to perform well in new tasks with minimal examples, achieved by conditioning their response on context. Kirsch et al. [47] demonstrate that general-purpose in-context learning algorithms can be trained from scratch using black-box models with minimal inductive bias (such as transformers [49]), highlighting the adaptability and potential of black-box meta-learning methods in these specialized contexts.

Black-box meta-learning methods are expressive, versatile, and easy to combine with various learning problems, including classification, regression, and reinforcement learning. However, they require complex architectures for the meta-learner f_θ, making them computationally demanding and data-inefficient. As an alternative, one can represent $\phi_i = f_\theta(\mathcal{D}_i^{\text{tr}})$ as an optimization procedure instead of a neural network. The next section explores methods that utilize this approach.

B. Optimization-Based Meta-Learning Methods

Optimization-based meta-learning offers an alternative to the black-box approach, where the meta-learner f_θ is an optimization procedure like gradient descent, rather than a black-box neural network. The goal of optimization-based meta-learning is to acquire a set of meta-parameters θ that are easy to learn via gradient descent and to fine-tune on new tasks. Most optimization-based techniques do so by defining meta-learning as a bi-level optimization problem. At the inner level, f_θ produces task-specific parameters ϕ_i using $\mathcal{D}_i^{\text{tr}}$, while at the outer level, the initial set of meta-parameters θ is updated by optimizing the performance of h_{ϕ_i} on the test set of the same task. This is shown in Fig. 3 and in Algorithm 2 in case f_θ is a gradient-based

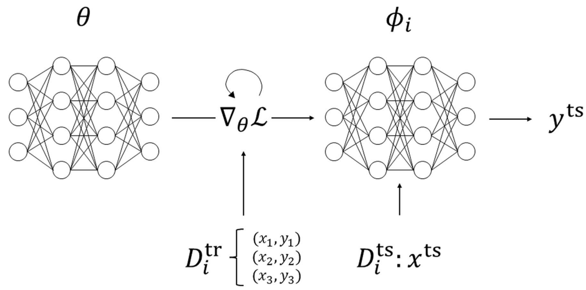


Fig. 3. Optimization-based meta-learning with gradient-based optimization.

optimization. The meta-parameters θ can represent inner optimizers [50], [51], [52], [53], neural network architectures [54], [55], other network hyperparameters [56], or the initialization of the base model $h(.)$ [2], [57]. The latter approach is similar to transfer learning via fine-tuning (cf. Section III-B), but instead of using a pre-trained θ that may not be transferable to new tasks, we learn θ to explicitly optimize for transferability.

Model-Agnostic Meta-Learning (MAML) [2] is one of the earliest and most popular optimization-based meta-learning methods. The main idea behind MAML is to learn a set of initial neural network's parameters θ that can easily be fine-tuned for any task using gradient descent with only a few steps. During the meta-training phase, MAML minimizes the objective defined as follows:

$$\min_{\theta} \sum_{\mathcal{T}_i} \mathcal{L}(\underbrace{\theta - \alpha \nabla_\theta \mathcal{L}(\theta, \mathcal{D}_i^{\text{tr}})}_{\phi_i}, \mathcal{D}_i^{\text{ts}}). \tag{4}$$

Note that in (4), the task-specific parameters ϕ_i are obtained through a single gradient descent step from θ, although in practice, a few more gradient steps are usually used for better performance.

As a result, MAML produces a model initialization θ that can be quickly adapted to new tasks with a small number of training examples. Algorithm 2 can be viewed as a simplified illustration of MAML, where θ represents the parameters of a neural network. This is similar to Algorithm 1 but with ϕ_i obtained through optimization.

During meta-test time, a small dataset $\mathcal{D}_{\text{new}}^{\text{tr}}$ is observed from a new task $\mathcal{T}_{\text{new}} \sim p(\mathcal{T})$. The goal is to use the prior knowledge encoded in θ to train a model that generalizes well to new, unseen examples from this task. To achieve this, θ is fine-tuned with a few adaptation steps using $\nabla_\theta \mathcal{L}(\theta, \mathcal{D}_{\text{new}}^{\text{tr}})$, resulting in task-specific parameters ϕ. These parameters are then used to make accurate predictions on previously unseen input data from $\mathcal{T}_{\text{new}}$.

MAML can be thought of as a computation graph (as shown in Fig. 4) with an embedded gradient operator. Interestingly, the components of this graph can be interchanged or replaced with components from the black-box approach. For instance, [50] also learned an initialization θ, but adapted θ differently by using a learned network $f_w(\theta, \mathcal{D}_i^{\text{tr}}, \nabla_\theta \mathcal{L})$ instead of the gradient $\nabla_\theta \mathcal{L}(\theta, \mathcal{D}_i^{\text{tr}})$:

$$\phi_i \leftarrow \theta - \alpha f_w(\theta, \mathcal{D}_i^{\text{tr}}, \nabla_\theta \mathcal{L})$$

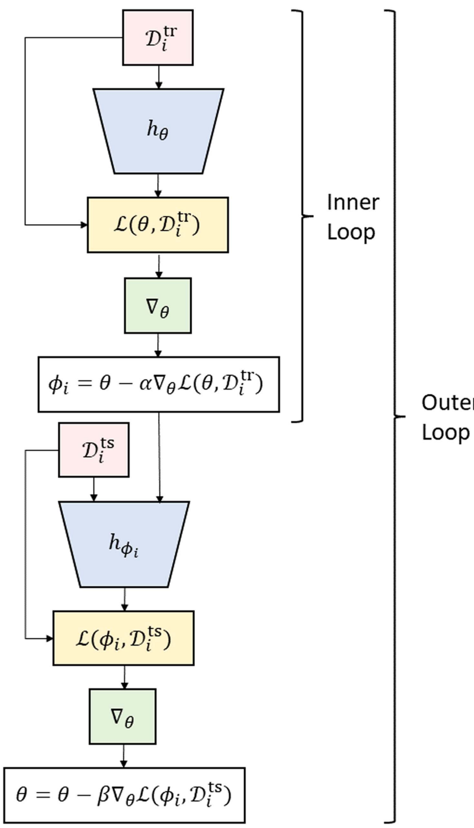


Fig. 4. Visual representation of the computation graph of MAML.

Algorithm 2: Optimization-Based Meta-Learning With Gradient-Based Optimization.

1: Randomly initialize θ
2: **while** not done **do**
3:   Sample a task $\mathcal{T}_i \sim p(\mathcal{T})$ *(or a mini-batch of tasks)*
4:   Sample disjoint datasets $\mathcal{D}_i^{\text{tr}}, \mathcal{D}_i^{\text{ts}}$ from $\mathcal{T}_i$
5:   Optimize $\phi_i \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}(\theta, \mathcal{D}_i^{\text{tr}})$
6:   Update θ using $\nabla_\theta \mathcal{L}(\phi_i, \mathcal{D}_i^{\text{ts}})$
7: **end while**
8: **return** θ

In [58], the authors investigated the effectiveness of optimization-based meta-learning in generalizing to similar but extrapolated tasks that are outside the original task distribution $p(\mathcal{T})$. The study found that, as task variability increases, black-box meta-learning methods such as SNAIL [42] and MetaNet [43] acquire less generalizable learning strategies than gradient-based meta-learning approaches like MAML.

However, despite its success, MAML faces some challenges that have motivated the development of other optimization-based meta-learning methods. One of these challenges is the instability of MAML's bi-level optimization. Fortunately, there are enhancements that can significantly improve optimization process. For instance, Meta-SGD [59] and AlphaMAML [60] learn a vector of learning rates α automatically, rather than using a manually set scalar value α. Other methods like DEML [61], ANIL [62] and BOIL [63] suggest optimizing only a subset of the parameters during adaptation. Additionally, MAML++ [64] proposes various modifications to stabilize the optimization process and further improve the generalization performance. Moreover, Bias-transformation [58] and CAVIA [65] introduce context variables for increased expressive power, while [66] enforces a well-conditioned parameter space based on the concepts of the condition number [67].

Another significant challenge in MAML is the computationally expensive process of backpropagating through multiple gradient adaptation steps. To overcome this challenge, first-order alternatives to MAML such as FOMAML and Reptile have been introduced [68]. For example, Reptile aims to find an initialization θ that is close to each task's optimal parameters. Another approach is to optimize only the parameters of the last layer. For instance, [69] and [70] perform a closed-form or convex optimization on top of meta-learned features. Another solution is iMAML [71], which computes the full meta-gradient without differentiating through the optimization path, using the implicit function theorem.

C. Meta-Learning Via Distance Metric Learning

In the context of low data regimes, such as in few-shot learning, simple non-parametric methods such as Nearest Neighbors [72] can be effective. However, black-box and optimization-based meta-learning approaches discussed so far in Sections IV-A and IV-B have focused on using parametric base models, such as neural networks. In this section we discuss meta-learning approaches that employ a non-parametric learning procedure. The key concept is to use parametric meta-learners to produce effective non-parametric learners, thus eliminating the need for second-order optimization, as required by several methods discussed in Section IV-B.

Suppose we are given a small training dataset $\mathcal{D}_i^{\text{tr}}$ that presents a 1-shot-N-way classification problem, i.e., N classes with only one labeled data-point per class, along with a test data-point x^{ts}. To classify x^{ts}, a Nearest Neighbor learner compares it with each training data-point in $\mathcal{D}_i^{\text{tr}}$. However, determining an effective space and distance metric for this comparison can be challenging. For example, using the L_2 distance in pixel space for image data may not yield satisfactory results [73]. To overcome this, a distance metric can be derived by learning how to compare instances using meta-training data. To learn an appropriate distance metric for comparing instances, a Siamese network [74] can be trained to solve a binary classification problem that predicts whether two images belong to the same class. During meta-test time, each image in $\mathcal{D}_i^{\text{tr}}$ is compared with the test image x^{ts} to determine whether they belong to the same class or not. However, there is a nuance due to the mismatch between the binary classification problem during meta-training and the N-way classification problem during meta-testing. Matching Networks, introduced in [75], address this by learning an embedding space with a network f_θ and using Nearest Neighbors in the learned space, as shown in Fig. 5. The network is trained end-to-end to ensure that meta-training is consistent with meta-testing. Algorithm 3 outlines the meta-training process used by Matching Networks. It is similar to Algorithms 1 and 2, except that the base model is non-parametric, so there is no ϕ_i (see lines 5 and 6).

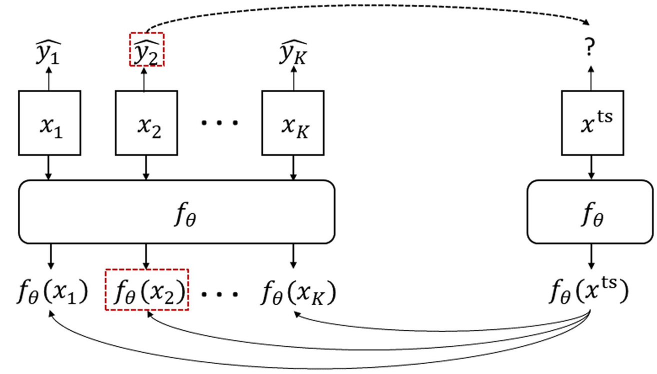


Fig. 5. Meta-learning via distance metric learning using matching network [75].

Algorithm 3: Meta-Learning via Metric Learning (Matching Networks).

1: Randomly initialize θ
2: **while** not done **do**
3: Sample a task $\mathcal{T}_i \sim p(\mathcal{T})$ *(or a mini-batch of tasks)*
4: Sample disjoint datasets $\mathcal{D}_i^{\text{tr}}, \mathcal{D}_i^{\text{ts}}$ from $\mathcal{T}_i$
5: Compute $\hat{y}^{\text{ts}} = \sum_{(x_k, y_k) \in \mathcal{D}_i^{\text{tr}}} f_\theta(x^{\text{ts}}, x_k) y_k$
6: Update θ using $\nabla_\theta \mathcal{L}(\hat{y}^{\text{ts}}, y^{\text{ts}})$
7: **end while**
8: **return** θ

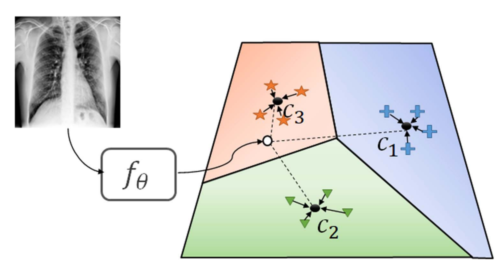


Fig. 6. Prototypical networks.

However, Matching Networks are specifically designed for 1-shot classification and cannot be directly applied to K-shot classification problems (where there are K labeled samples per class). To address this issue, other methods, such as Prototypical Networks [76], have been proposed. Prototypical Networks aggregate class information to create a prototypical embedding, as illustrated in Fig. 6. In Prototypical Networks, line 5 of Algorithm 3 is replaced with:

$$p_\theta(y = l | x) = \frac{\exp\left(-\|f_\theta(x) - c_l\|\right)}{\sum_{l'} \exp\left(-\|f_\theta(x) - c_{l'}\|\right)},$$

where c_l is the mean embedding of all the samples in the l-th class, i.e., $c_l = \frac{1}{K} \sum_{(x,y) \in \mathcal{D}_i^{\text{tr}}} \mathbb{1}(y = l) f_\theta(x)$.

TABLE I
SUMMARY COMPARISON OF META-LEARNING APPROACHES

	BlackBox	Optimization	Metric
Parametric Base Model	*yes*	*yes*	*no*
Expressive Power	*very high*	*high*	*high*
Consistency	✗	✓	✓
Versatility	✓	✓	✗
Simple Optimization	✗	✗	✓
Data Efficiency	✗	✗	✓
Positive Inductive Bias	✗	✓	✗
Resource Efficiency	✗	✗	✓
Generalizability to Tasks	✗	✓	✗

While methods such as Siamese networks, Matching Networks, and Prototypical Networks can perform few-shot classification by embedding data and applying Nearest Neighbors [74], [75], [76], they may not be sufficient to capture complex relationships between data-points. To address this, alternative approaches have been proposed. RelationNet [77] introduces a non-linear relation module that can reason about complex relationships between embeddings. Garcia et al. [78] propose to use graph neural networks to perform message passing on embeddings, allowing for the capture of more complex dependencies. Finally, Allen et al. [79] extend Prototypical Networks to learn an infinite mixture of prototypes, which improves the model's ability to represent the data distribution.

D. Comparison and Hybrid Approaches

Meta-learning approaches exhibit distinct strengths and trade-offs. Black-box, optimization-based, and distance metric-based meta-learning approaches define $\mathcal{F}_\theta(\mathcal{D}_i^{\text{tr}}, x^{\text{ts}})$ differently, and the choice of method depends on specific use-case requirements. To assist readers, we provide a unified comparison in Table I, summarizing the pros and cons of each approach based on the following criteria:

- *Parametric base model:* Indicates whether the approach is parametric (yes) or non-parametric (no), highlighting the model's flexibility.
- *Expressive power:* Evaluates the ability of $\mathcal{F}_\theta$ to represent a wide range of learning procedures.
- *Consistency:* Reflects the extent to which the learned learning procedure improves with additional data.
- *Versatility:* Compatibility with different scenarios and the ease of integration with different learning problems, such as classification, regression, and reinforcement learning.
- *Simple optimization:* Considers the complexity of optimization, including optimization challenges arising from complex models or the necessity for sophisticated second-order optimization techniques.
- *Data efficiency (in terms of training tasks):* The method's efficiency concerning the number of training tasks required for effective meta-learning.
- *Positive inductive bias at meta-learning onset:* Indicates the presence of an inherent bias favoring certain learning strategies, influencing initial meta-learning performance.
- *Resource efficiency:* Assesses computational and memory requirements, providing insights into the practical feasibility of the approach.
- *Generalizability to diverse/variable tasks:* Explores the approach's ability to acquire generalizable learning strategies, extending to similar but extrapolated tasks beyond the original task distribution.

Although black-box, optimization-based, and distance metric-based meta-learning approaches differ, they are not mutually exclusive and can be combined in various ways. For instance, in [80], gradient descent is applied while conditioning on the data, allowing the model to modulate the feature representations and capture inter-class dependencies. In [81], LEO (Latent Embedding Optimization) combines optimization-based meta-learning with a latent embedding produced by the RelationNet embedding proposed in [77]. The parameters of the model are first conditioned on the input data and then further adapted through gradient descent. In [82], the strength of both MAML and Prototypical Networks are combined to form a hybrid approach called Proto-MAML. This approach exploits the flexible adaptation of MAML, while initializing the last layer with ProtoNet to provide a simple inductive bias that is effective for very-few-shot learning. Similarly, [83] proposes a model where the meta-learner operates using an optimization-based meta-model, while the base learner exploits a metric-based approach (either Matching Network or Prototypical Network). The distance metrics used by the base learner can better adapt to different tasks thanks to the weight prediction from the meta-learner.

In summary, researchers have explored combining black-box, optimization-based, and distance metric-based meta-learning approaches to take advantage of their individual strengths. These combined approaches aim to improve performance, adaptability, and generalization in few-shot learning tasks by integrating different methodologies.

V. Advanced Meta-Learning Topics

The field of meta-learning has seen rapid development in recent years, with numerous methods proposed for learning to learn from a few examples. In this section, we delve into advanced topics in meta-learning that extend the meta-learning paradigm to more complex scenarios. We explore meta-learning from multi-modal task distributions, the challenge of out-of-distribution tasks, and unsupervised meta-learning. Additionally, we examine the relationship between meta-learning and personalized federated learning, domain adaptation/generalization, as well as the intersection between meta-learning and continual learning. By delving into these advanced topics, we can gain a deeper understanding of the potential of meta-learning and its applications in more complex real-world scenarios.

A. Meta-Learning From Multimodal Task Distributions

Meta-learning methods have traditionally focused on optimizing performance within a unimodal task distribution $p(\mathcal{T})$, assuming that all tasks are closely related and share similarities within a single application domain. However, in real-world scenarios, tasks are often diverse and sampled from a more complex task distribution with multiple unknown modes. The performance of most meta-learning approaches tends to deteriorate as the dissimilarity among tasks increases [84], [85], [86], [87], indicating that a globally shared set of meta-parameters θ may not adequately capture the heterogeneity among tasks and enable fast adaptation.

To address this challenge, MMAML [88] builds upon the standard MAML approach by estimating the mode of tasks sampled from a multimodal task distribution $p(\mathcal{T})$ and adjusting the initial model parameters accordingly. Another approach proposed in [89] involves learning a meta-regularization conditioned on additional task-specific information. However, obtaining such additional task information may not always be feasible. Alternatively, some methods propose learning multiple model initializations $\theta_1, \theta_2, \ldots, \theta_M$ and selecting the most suitable one for each task, leveraging clustering techniques applied in either the task-space or parameter-space [90], [91], [92], [93], or relying on the output of an additional network, as in MUSE [94]. CAVIA [65] partitions the initial model parameters into shared parameters across all tasks and task-specific context parameters, while LGM-Net [95] directly generates classifier weights based on an encoded task representation.

A series of related works (but outside of the meta-learning field) aim to build a "universal representation" that encompasses a robust set of features capable of achieving strong performance across multiple datasets (or modes) [44], [96], [97], [98], [99], [99], [100]. This representation is subsequently adapted to individual tasks in various ways. However, these approaches are currently limited to classification problems and do not leverage meta-learning techniques to efficiently adapt to new tasks.

A more recent line of research focuses on cross-domain meta-learning, where knowledge needs to be transferred from tasks sampled from a potentially multimodal distribution $p(\mathcal{T})$ to target tasks sampled from a different distribution. One notable study, BOIL [63], reveals that the success of meta-learning methods, such as MAML, can be attributed to large changes in the representation during task learning. The authors emphasize the importance of updating only the body (feature extractor) of the model and freezing the head (classifier) during the adaptation phase for effective cross-domain adaptation. Building on this insight, DAML [101] introduces tasks from both seen and pseudo-unseen domains during meta-training to obtain domain-agnostic initial parameters capable of adapting to novel classes in unseen domains. In [102], the authors propose a transferable meta-learning algorithm with a *meta task adaptation* to minimize the domain divergence and thus facilitate knowledge transfer across domains. To further improve the transferability of cross-domain knowledge, [103] and [104] propose to incorporate semi-supervised techniques into the meta-learning framework. Specifically, [103] combines the representation power of large pre-trained language models (e.g., BERT [33]) with the generalization capability of prototypical networks enhanced by SMLMT [105] to achieve effective generalization and adaptation to tasks from new domains. In contrast, [104] promotes the idea of task-level self-supervision by leveraging multiple views or augmentations of tasks.

B. Meta-Learning & Personalized Federated Learning

Federated learning (FL) is a distributed learning paradigm where multiple clients collaborate to train a shared model while preserving data privacy by keeping their data locally stored. FedAvg [106] is a pioneering method that combines local stochastic gradient descent on each client with model averaging on a central server. This approach performs well when local data across clients is independent and identically distributed (IID). However, in scenarios with heterogeneous (non-IID) data distributions, regularization techniques [107], [108], [109] have been proposed to improve local learning.

Personalized federated learning (PFL) is an alternative approach that aims to develop customized models for individual clients while leveraging the collaborative nature of FL. Popular PFL methods include L2GD [110], which combines local and global models, as well as multi-task learning methods like pFedMe [111], Ditto [112], and FedPAC [113]. Clustered or group-based FL approaches [114], [115], [116], [117] learn multiple group-based global models. In contrast, meta-learning-based methods interpret PFL as a meta-learning algorithm, where *personalization to a client* aligns with *adaptation to a task* [118]. Notably, various combinations of MAML-type methods with FL architectures have been explored in [118], [119], [120] to find an initial shared point that performs well after personalization to each client's local dataset. Additionally, the authors of [121] proposed ARUBA, a meta-learning algorithm inspired by online convex optimization, which enhances the performance of FedAvg.

To summarize, there is a growing focus on addressing FL challenges in non-IID data settings. The integration of meta-learning has shown promising outcomes, leading to enhanced personalization and performance in PFL methods.

C. Unsupervised Meta-Learning With Tasks Construction

In meta-training, constructing tasks typically relies on labeled data. However, real-world scenarios often involve mostly, or only, unlabeled data, requiring techniques that leverage unlabeled data to learn feature representations that can transfer to downstream tasks with limited labeled data. One alternative to address this is through "self-supervised learning" (also known as "unsupervised pre-training") [122], [123], [124]. This involves training a model on a large unlabeled dataset, as depicted in Fig. 7, to capture informative features. Contrastive learning [122], [125] is commonly used in this context, aiming to learn features by bringing similar examples closer together while pushing differing examples apart. The learned features can then be fine-tuned on a target task $\mathcal{T}_{\text{new}}$ with limited labeled data $\mathcal{D}^{\text{tr}}_{\text{new}}$, leading to improved performance compared to training from scratch. Another promising alternative is "unsupervised meta-learning," which aims to automatically construct diverse and structured training tasks from unlabeled data. These tasks can then be used with any meta-learning algorithm, such as MAML [2] and ProtoNet [76]. In this section, we explore methods for meta-training without predefined tasks and investigate strategies for automatically constructing tasks for meta-learning.

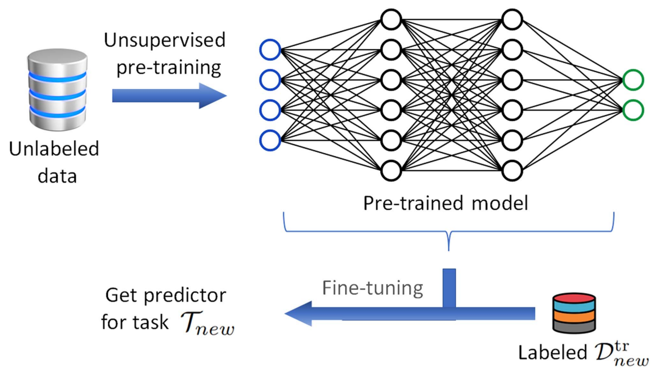


Fig. 7. Unsupervised pre-training.

The method proposed in [126] constructs tasks based on unsupervised representation learning methods such as BiGAN [127], [128] or DeepCluster [129] and clusters the data in the embedding space to assign pseudo-labels and construct tasks. Other methods such as UMTRA [130] and LASIUM [131] generate synthetic samples using image augmentations or pre-trained generative networks. In particular, the authors in [130] construct a task $\mathcal{T}_i$ for a 1-shot N-way classification problem by creating a support set $\mathcal{D}^{\mathrm{tr}}_i$ and a query set $\mathcal{D}^{\mathrm{ts}}_i$ as follows:

- Randomly sample N images and assign labels $1,\ldots,N$, storing them in $\mathcal{D}^{\mathrm{tr}}_i$.
- Augment[1] each image in $\mathcal{D}^{\mathrm{tr}}_i$, and store the resulting (augmented) images in $\mathcal{D}^{\mathrm{ts}}_i$.

Such augmentations can be based on domain knowledge or learned augmentation strategies like those proposed in [132]. In principle, task construction techniques can be applied beyond image-based augmentation. For instance, temporal aspects can be leveraged by incorporating time-contrastive learning on videos, as demonstrated in [133]. Another approach is offered by Viewmaker Networks [134], which learn augmentations that yield favorable outcomes not only for images but also for speech and sensor data. Contrary to these works focusing on generating pseudo tasks, Meta-GMVAE [135] and Meta-SVEBM [136] address the problem by using variational autoencoders [137] and energy-based models [138], respectively. However, these methods are limited to the pseudo-labeling strategies used to create tasks, they rely on the quality of generated samples and they cannot scale to large-scale datasets.

To overcome this limitation, recent approaches have investigated the possibility of using self-supervised learning techniques to improve unsupervised meta-learning methods. In particular, in [139], the relationship between contrastive learning and meta-learning is explored, demonstrating that established meta-learning methods can achieve comparable performance to contrastive learning methods, and that representations transfer similarly well to downstream tasks. Inspired by these findings, the authors in [140] integrate contrastive learning in a two-stage training paradigm consisting of sequential pre-training and meta-training stages. Another work [141] interprets a meta-learning problem as a set-level problem and maximizes the agreement between augmented sets using SimCLR [142]. Finally, PsCo [142] builds upon MoCo [123] by progressively improving pseudo-labeling and constructing diverse tasks in an online manner. These findings indicate the potential for leveraging existing advances in meta-learning to improve contrastive learning (and vice-versa).

To meta-learn with unlabeled text data, some methods use language modeling, as shown in [45] for GPT-3. Here, the support set $\mathcal{D}^{\mathrm{tr}}_i$ consists of a sequence of characters, and the query set $\mathcal{D}^{\mathrm{ts}}_i$ consists of the subsequent sequence of characters. However, this approach may not be suitable for text classification tasks, such as sentiment analysis or identifying political bias. In [105], an alternative approach (SMLMT) for self-supervised meta-learning for few-shot natural language classification tasks is proposed. SMLMT involves masking out words and classifying the masked word to construct tasks. The process involves: 1) sampling a subset of N unique words and assigning each word a unique ID as its class label, 2) sampling $K+Q$ sentences that contain each of the N words and masking out the corresponding word in each sentence, and 3) constructing the support set $\mathcal{D}^{\mathrm{tr}}_i$ and the query set $\mathcal{D}^{\mathrm{ts}}_i$ using the masked sentences and their corresponding word IDs. SMLMT (for unsupervised meta-learning) is compared to BERT [33], a method that uses standard self-supervised learning and fine-tuning. SMLMT outperforms BERT on some tasks and achieves at least equal performance on others. Furthermore, Hybrid-SMLMT (semi-supervised meta-learning, which involves meta-learning on constructed tasks and supervised tasks), is compared to MT-BERT [143] (multi-task learning on supervised tasks) and LEOPARD [144] (an optimization-based meta-learner that uses only supervised tasks). The results show that Hybrid-SMLMT significantly outperforms these other methods.

D. Meta-Learning & Domain adaptation/generalization

Domain shift is a fundamental challenge, where the distribution of the input data changes between the training and test domains. To address this problem, there is a growing interest in utilizing meta-learning techniques for more effective domain adaptation and domain generalization. These approaches aim to enable models to quickly adapt to new domains with limited data or to train robust models that achieve better generalization on domains they have not been explicitly trained on.

Effective domain adaptation via meta-learning: Domain adaptation is a form of transductive transfer learning that leverages source domain(s) $p_S(x,y)$ to achieve high performance on test data from a target domain $p_T(x,y)$. It assumes $p_S(y|x) = p_T(y|x)$ but $p_S(x) \neq p_T(x)$, treating *domains* as a particular kind of *tasks*, with a task $\mathcal{T}_i \triangleq \{p_i(x), p_i(y|x), \mathcal{L}_i\}$ and a domain $d_i \triangleq \{p_i(x), p(y|x), \mathcal{L}\}$. For example, healthcare data from different hospitals with varying imaging techniques or patient demographics can correspond to different domains. Domain adaptation is most commonly achieved via feature alignment

[1]Various augmentation techniques, like flipping, cropping, or reflecting an image, typically preserve its label. Likewise, nearby image patches or adjacent video frames share similar characteristics and are therefore assigned the same label.

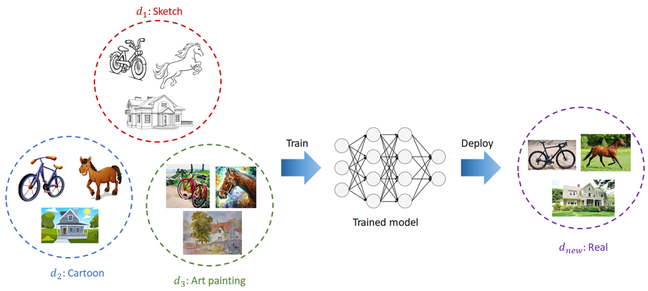


Fig. 8. Domain generalization problem.

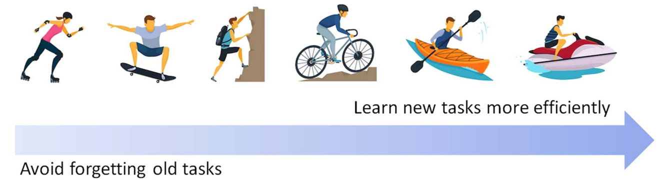


Fig. 9. Continual learning.

as in [145], [146] or via translation between domains using CycleGAN [147] as in [148], [149], [150]. Other approaches focus on aligning the feature distribution of multiple source domains with the target domain [151] or they address the multi-target domain adaptation scenario [152], [153], [154] with models capable of adapting to multiple target domains. However, these methods face limitations when dealing with insufficient labeled data in the source domain or when quick adaptation to new target domains is required. Additionally, they assume the input-output relationship (i.e., $p(y|x)$) to be the same across domains. To solve these problems, some methods [153], [155], [156], [157] combine meta-learning with domain adaptation. In particular, ARM [155] leverages contextual information extracted from batches of unlabeled data to learn a model capable of adapting to distribution shifts.

Effective domain generalization via meta-learning: Domain generalization enables models to perform well on new and unseen domains without requiring access to their data, as illustrated in Fig. 8. This is particularly useful in scenarios where access to data is restricted due to real-time deployment requirements or privacy policies. For instance, an object detection model for self-driving cars trained on three types of roads may need to be deployed to a new road without any data from that domain. In contrast to domain adaptation, which requires access to (unlabeled) data from a specific target domain during training to specialize the model, domain generalization belongs to the inductive setting. Most domain generalization methods aim to train neural networks to learn domain-invariant representations that are consistent across domains. For instance, domain adversarial training [158] trains the network to make predictions based on features that cannot be distinguished between domains. Another approach is to directly align the representations between domains using similarity metrics, such as in [159]. Data augmentation techniques are also used to enhance the diversity of the training data and improve generalization across domains [160], [161], [162]. Another way to improve generalization to various domains is to use meta-learning and applying the episodic training paradigm typical of MAML [58], as in [163], [164], [165], [166], [167], [168], [169]. For instance, MLDG [166] optimizes a model by simulating the train-test domain shift during the meta-training phase. MetaReg [167] proposes to meta-learn a regularization function that improves domain generalization. DADG [169] contains a discriminative adversarial learning component to learn a set of general features and a meta-learning-based cross-domain validation component to further enhance the robustness of the classifier.

E. Meta-Learning & Continual Learning

This section explores the application of meta-learning to continual learning, where learners continually accumulate experience over time to more rapidly acquire new knowledge or skills. Continual learning scenarios can be divided into task-incremental learning, domain-incremental learning, and class-incremental learning, depending on whether task identity is provided at test time or must be inferred by the algorithm [170]. In this section, we focus on approaches that specifically address task/class-incremental learning.

Traditionally, meta-learning has primarily focused on scenarios where a batch of training tasks is available. However, real-world situations often involve tasks presented sequentially, allowing for progressive leveraging of past experience. This is illustrated in Fig. 9, and examples include tasks that progressively increase in difficulty or build upon previous knowledge, or robots learning diverse skills in changing environments.

Standard online learning involves observing tasks in a sequential manner, without any task-specific adaptation or use of past experience to accelerate adaptation. To tackle this issue, researchers have proposed various approaches, including memory-based methods [171], [172], [173], regularization-based methods [174], [175], [176] and dynamic architectural methods [177], [178], [179]. However, each of these methods has its own limitations, such as scalability, memory inefficiency, time complexity, or the need for task-specific parameters. Meta-learning has emerged as a promising approach for addressing continual learning. In [180], the authors introduced ANML, a framework that meta-learns an activation-gating function that enables context-dependent selective activation within a deep neural network. This selective activation allows the model to focus on relevant knowledge and avoid catastrophic forgetting. Other approaches such as MER [181], OML [182], and La-MAML [183] use gradient-based meta-learning algorithms to optimize various objectives such as gradient alignment, inner representations, or task-specific learning rates and learn update rules that avoid negative transfer. These algorithms enable faster learning over time and enhanced proficiency in each new task.

VI. Open Challenges & Opportunities

Meta-learning has been a promising area of research that has shown impressive results in various machine learning domains. However, there are still open challenges that need to be addressed in order to further advance the field. In this section, we discuss some of these challenges and categorize them into three groups. Addressing these challenges can lead to significant advances in meta-learning, which could potentially lead to more generalizable and robust machine learning models.

A. Addressing Fundamental Problem Assumptions

The first category of challenges pertains to the fundamental assumptions made in meta-learning problems.

One such challenge is related to generalization to out-of-distribution tasks and long-tailed task distributions. Indeed, adaptation becomes difficult when the few-shot tasks observed at meta-test time are from a different task distribution than the ones seen during meta-training. While there have been some attempts to address this challenge, such as in [102], [184], it still remains unclear how to address it. Ideas from the domain generalization and robustness literature could provide some hints and potentially be combined with meta-learning to tackle these long-tailed task distributions and out-of-distribution tasks. For example, possible directions are to define subtle regularization techniques to prevent the meta-parameters from being very specific to the distribution of the training tasks, or use subtle task augmentation techniques to generate synthetic tasks that cover a wider range of task variations.

Another challenge in this category involves dealing with the multimodality of data. While the focus has been on meta-training over tasks from a single modality, the reality is that we may have multiple modalities of data to work with. Human beings have the advantage of being able to draw upon multiple modalities, such as visual imagery, tactile feedback, language, and social cues, to create a rich repository of knowledge and make more informed decisions. For instance, we often use language cues to aid our visual decision-making processes. Rather than developing a prior that only works for a single modality, exploring the concept of learning priors across multiple modalities of data is a fascinating area to pursue. Different modalities have different dimensionalities or units, but they can provide complementary forms of information. While some initial works in this direction have been reported, including [185], [186], [187], there is still a long way to go in terms of capturing all of this rich prior information when learning new tasks.

B. Providing Benchmarks and Real-World Problems

The second category of challenges is related to providing/improving benchmarks to better reflect real-world problems and challenges.

Meta-learning has shown promise in a diverse set of applications, including few-shot land cover classification [188], few-shot dermatological disease diagnosis [184], automatically providing feedback on student code [18], one-shot imitation learning [189], drug discovery [190], motion prediction [191], and language generation [16], to mention but a few. However, the lack of benchmark datasets that accurately reflect real-world problems with appropriate levels of difficulty and ease of use is a significant challenge for the field. Several efforts have been made towards creating useful benchmark datasets, including Meta-Dataset [82], Meta-Album Dataset [192], NEVIS'22 [193], Meta-World Benchmark [194], Visual Task Adaptation Benchmark [195], Taskonomy Dataset [196], VALUE Benchmark [197], and BIG Bench [198]. However, further work is needed to ensure that the datasets are comprehensive and representative of the diversity of real-world problems that meta-learning aims to address.

Some ways with which existing benchmarks can be improved to better reflect real-world problems and challenges in meta-learning are: 1) to increase the diversity and complexity of tasks that are included; 2) to consider more realistic task distributions that can change over time; and 3) to include real-world data that is representative of the challenges faced in real-world applications of meta-learning. For example, including medical data, financial data, time-series data, or other challenging types of data (besides images and text) can help improve the realism and relevance of benchmarks.

Furthermore, developing benchmarks that reflect these more realistic scenarios can help improve the generalization and robustness of algorithms. This ensures that algorithms are tested on a range of scenarios and that they are robust and generalizable across a wide range of tasks. Better benchmarks are essential for progress in machine learning and AI, as they challenge current algorithms to find common structures, reflect real-world problems, and have a significant impact in the real world.

C. Improving Core Algorithms

The last category of challenges in meta-learning is centered around improving the core algorithms.

A major obstacle is the large-scale bi-level optimization problem encountered in popular meta-learning methods such as MAML. The computational and memory costs of such approaches can be significant, and there is a need to make them more practical, particularly for very large-scale problems, like *learning effective optimizers* [199].

In addition, a deeper theoretical understanding of various meta-learning methods and their performance is critical to driving progress and pushing the boundaries of the field. Such insights can inform and inspire further advancements in the field and lead to more effective and efficient algorithms. To achieve these goals, several fundamental questions can be explored, including:

1) Can we develop theoretical guarantees on the sample complexity and generalization performance of meta-learning algorithms? Understanding these aspects can help us design more efficient and effective meta-learning algorithms that require less data or less tasks. While recent investigations [200], [201], [202] have made notable strides in this domain, they represent just the initial steps toward a more extensive theoretical comprehension. Further research is imperative to completely harness the potential of meta-learning.
2) Can we gain a better understanding of the optimization landscape of meta-learning algorithms? For instance, can we identify the properties of the objective function that

make it easier or harder to optimize? Can we design optimization algorithms that are better suited to the bi-level optimization problem inherent in various meta-learning approaches?

3) Can we design meta-learning algorithms that can better incorporate task-specific or domain-specific expert knowledge, in a principled way, to learn more effective meta-parameters?

Addressing such questions could enhance the design and performance of meta-learning algorithms, and help us tackle increasingly complex and challenging learning problems.

VII. Conclusion

In conclusion, the field of artificial intelligence (AI) has witnessed significant advancements in developing specialized systems for specific tasks. However, the pursuit of generality and adaptability in AI across multiple tasks remains a fundamental challenge.

Meta-learning emerges as a promising research area that seeks to bridge this gap by enabling algorithms to learn how to learn. Meta-learning algorithms offer the ability to learn from limited data, transfer knowledge across tasks and domains, and rapidly adapt to new environments. This review article has explored various meta-learning approaches that have demonstrated promising results in applications with scarce data. Nonetheless, numerous challenges and unanswered questions persist, calling for further investigation.

A key area of focus lies in unifying various fields such as meta-learning, self-supervised learning, domain generalization, and continual learning. Integrating and collaborating across these domains can generate synergistic advancements and foster a more comprehensive approach to developing AI systems. By leveraging insights and techniques from these different areas, we can construct more versatile and adaptive algorithms capable of learning from multiple tasks, generalizing across domains, and continuously accumulating knowledge.

This review article serves as a starting point for encouraging research in this direction. By examining the current state of meta-learning and illuminating the challenges and opportunities, we aim to inspire researchers to explore interdisciplinary connections and contribute to the progress of meta-learning while integrating it with other AI research fields. Through collective efforts and collaboration, we can surmount existing challenges and unlock the full potential of meta-learning to address a broad spectrum of complex problems.

References

[1] S. K. Karmaker, M. M. Hassan, M. J. Smith, L. Xu, C. Zhai, and K. Veeramachaneni, "AutoML to date and beyond: Challenges and opportunities," *ACM Comput. Surv.*, vol. 54, no. 8, pp. 1–36, 2021.

[2] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. Int. Conf. Mach. Learn*, 2017, pp. 1126–1135.

[3] J. Beck et al., "A survey of meta-reinforcement learning," 2023, *arXiv:2301.08028*.

[4] J. Vanschoren, *Meta-Learning: A Survey*. Berlin, Germany: Springer, 2019, pp. 35–61.

[5] T. Hospedales, A. Antoniou, P. Micaelli, and A. Storkey, "Meta-learning in neural networks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5149–5169, Sep. 2022.

[6] R. Vilalta and Y. Drissi, "A perspective view and survey of meta-learning," *Artif. Intell. Rev.*, vol. 18, pp. 77–95, 2002.

[7] M. Huisman, J. N. Van Rijn, and A. Plaat, "A survey of deep meta-learning," *Artif. Intell. Rev.*, vol. 54, no. 6, pp. 4483–4541, 2021.

[8] L. Zou, "Chapter 1 - meta-learning basics and background," in *Meta-Learning*, L. Zou, Ed. Cambridge, MA, USA: Academic Press, 2023, pp. 1–22.

[9] O. Vinyals, "Talk: Model vs optimization meta learning," in *Proc. Int. conf. Neural Inf. Process. Syst.*, 2017. [Online]. Available: https://evolution.ml/pdf/vinyals.pdf

[10] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 4080–4090.

[11] R. Geng, B. Li, Y. Li, X. Zhu, P. Jian, and J. Sun, "Induction networks for few-shot text classification," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process.*, 2019, pp. 3904–3913.

[12] B. Liang et al., "Few-shot aspect category sentiment analysis via meta-learning," *ACM Trans. Inf. Syst.*, vol. 41, no. 1, pp. 1–31, 2023.

[13] J. Gu, Y. Wang, Y. Chen, K. Cho, and V. O. Li, "Meta-learning for low-resource neural machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2020, pp. 3622–3631.

[14] A. Madotto, Z. Lin, C.-S. Wu, and P. Fung, "Personalizing dialogue agents via meta-learning," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 5454–5459.

[15] K. Qian and Z. Yu, "Domain adaptive dialog generation via meta learning," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 2639–2649.

[16] F. Mi, M. Huang, J. Zhang, and B. Faltings, "Meta-learning for low-resource natural language generation in task-oriented dialogue systems," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, 2019, pp. 3151–3157.

[17] C. Finn, T. Yu, T. Zhang, P. Abbeel, and S. Levine, "One-shot visual imitation learning via meta-learning," in *Proc. Conf. Robot Learn.*, 2017, pp. 357–368.

[18] M. Wu, N. Goodman, C. Piech, and C. Finn, "ProtoTransformer: A meta-learning approach to providing student feedback," 2021, *arXiv:2107.14035*.

[19] O. Sener and V. Koltun, "Multi-task learning as multi-objective optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018.

[20] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich, "GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 794–803.

[21] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7482–7491.

[22] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert, "Cross-stitch networks for multi-task learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3994–4003.

[23] S. Ruder, J. Bingel, I. Augenstein, and A. Søgaard, "Latent multi-task architecture learning," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 4822–4829.

[24] Y. Gao, J. Ma, M. Zhao, W. Liu, and A. L. Yuille, "NDDR-CNN: Layerwise feature fusing in multi-task CNNs by neural discriminative dimensionality reduction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3205–3214.

[25] V. Dumoulin et al., "Feature-wise transformations," *Distill*, 2018. [Online]. Available: https://distill.pub/2018/feature-wise-transformations

[26] S. Liu, E. Johns, and A. J. Davison, "End-to-end multi-task learning with attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1871–1880.

[27] M. Long, Z. Cao, J. Wang, and P. S. Yu, "Learning multiple tasks with multilinear relationship networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1593–1602.

[28] A. Jaegle et al., "Perceiver IO: A general architecture for structured inputs & outputs," in *Proc. Int. Conf. Learn. Representations*, 2022, pp. 1–16.

[29] C. Fifty, E. Amid, Z. Zhao, T. Yu, R. Anil, and C. Finn, "Efficiently identifying task groupings for multi-task learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 27 503–27 516.

[30] Y. Zhang and Q. Yang, "A survey on multi-task learning," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 12, pp. 5586–5609, Dec. 2022.

[31] M. Crawshaw, "Multi-task learning with deep neural networks: A survey," 2020, *arXiv:2009.09796*.

[32] M. Huh, P. Agrawal, and A. A. Efros, “What makes imagenet good for transfer learning?,” 2016, *arXiv:1608.08614*.

[33] J. D.M.-W. C. Kenton and L. K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics - Hum. Lang. Technol.*, 2019, pp. 4171–4186.

[34] A. Chowdhery et al., “PaLM: Scaling language modeling with pathways,” 2022, *arXiv:2204.02311*.

[35] H. Touvron et al., “LLaMA: Open and efficient foundation language models,” 2023, *arXiv:2302.13971*.

[36] OpenAI, “GPT-4 technical report,” 2023, *arXiv:2303.08774*.

[37] A. Kumar, A. Raghunathan, R. M. Jones, T. Ma, and P. Liang, “Fine-tuning can distort pretrained features and underperform out-of-distribution,” in *Proc. Int. Conf. Learn. Representations*, 2022, pp. 1–15.

[38] Y. Lee et al., “Surgical fine-tuning improves adaptation to distribution shifts,” in *Proc. Workshop Distrib. Shifts: Connecting Methods Appl.*, 2023, pp. 1–14.

[39] J. Phang, T. Févry, and S. R. Bowman, “Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks,” 2018, *arXiv:1811.01088*.

[40] J. Howard and S. Ruder, “Universal language model fine-tuning for text classification,” in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 328–339.

[41] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. Lillicrap, “Meta-learning with memory-augmented neural networks,” in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1842–1850.

[42] N. Mishra, M. Rohaninejad, X. Chen, and P. Abbeel, “A simple neural attentive meta-learner,” in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–12.

[43] T. Munkhdalai and H. Yu, “Meta networks,” in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2554–2563.

[44] M. Garnelo et al., “Conditional neural processes,” in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1704–1713.

[45] T. Brown et al., “Language models are few-shot learners,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 1877–1901.

[46] S. Garg, D. Tsipras, P. S. Liang, and G. Valiant, ”What can transformers learn in-context? A case study of simple function classes,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 30583–30598.

[47] L. Kirsch, J. Harrison, J. Sohl-Dickstein, and L. Metz, “General-purpose in-context learning by meta-learning transformers,” in *Proc. Workshop Distrib. Shifts: Connecting Methods Appl.*, 2022, pp. 1–14.

[48] E. Akyürek, D. Schuurmans, J. Andreas, T. Ma, and D. Zhou, “What learning algorithm is in-context learning? Investigations with linear models,” in *proc. 11th Int. Conf. Learn. Representations*, 2022, pp. 1–12.

[49] A. Vaswani et al., “Attention is all you need,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.

[50] S. Ravi and H. Larochelle, “Optimization as a model for few-shot learning,” in *Proc. Int. Conf. Learn. Representations*, 2017, pp. 1–11.

[51] M. Andrychowicz et al., “Learning to learn by gradient descent by gradient descent,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3988–3996.

[52] K. Li and J. Malik, “Learning to optimize,” in *Proc. Int. Conf. Learn. Representations*, 2017.

[53] O. Wichrowska et al., “Learned optimizers that scale and generalize,” in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 3751–3760.

[54] A. Shaw, W. Wei, W. Liu, L. Song, and B. Dai, “Meta architecture search,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 11227–11237.

[55] D. Lian et al., “Towards fast adaptation of neural architectures with meta learning,” in *Proc. Int. Conf. Learn. Representations*, 2019, pp. 1–13.

[56] L. Franceschi, P. Frasconi, S. Salzo, R. Grazzi, and M. Pontil, “Bilevel programming for hyperparameter optimization and meta-learning,” in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1568–1577.

[57] T. Kim, J. Yoon, O. Dia, S. Kim, Y. Bengio, and S. Ahn, “Bayesian model-agnostic meta-learning,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 1–11.

[58] C. Finn and S. Levine, “Meta-learning and universality: Deep representations and gradient descent can approximate any learning algorithm,” in *Proc. Int. Conf. Learn. Representations*, 2018.

[59] Z. Li, F. Zhou, F. Chen, and H. Li, “Meta-SGd: Learning to learn quickly for few-shot learning,” 2017, *arXiv:1707.09835*.

[60] H. S. Behl, A. G. Baydin, and P. H. Torr, “Alpha MAML: Adaptive model-agnostic meta-learning,” in *Proc. 6th ICML Workshop Automated Mach. Learn.*, 2019, pp. 1–10.

[61] F. Zhou, B. Wu, and Z. Li, “Deep meta-learning: Learning to learn in the concept space,” 2018, *arXiv:1802.03596*.

[62] A. Raghu, M. Raghu, S. Bengio, and O. Vinyals, “Rapid learning or feature reuse? Towards understanding the effectiveness of MAML,” in *Proc. Int. Conf. Learn. Representations*, 2023, pp. 1–12.

[63] J. Oh, H. Yoo, C. Kim, and S.-Y. Yun, “BOIL: Towards representation change for few-shot learning,” in *Proc. Int. Conf. Learn. Representations*, 2021, pp. 1–12.

[64] A. Antoniou, H. Edwards, and A. Storkey, “How to train your MAML,” in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–10.

[65] L. Zintgraf, K. Shiarli, V. Kurin, K. Hofmann, and S. Whiteson, “Fast context adaptation via meta-learning,” in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7693–7702.

[66] M. Hiller, M. Harandi, and T. Drummond, “On enforcing better conditioned meta-learning for rapid few-shot adaptation,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 4059–4071.

[67] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[68] A. Nichol, J. Achiam, and J. Schulman, “On first-order meta-learning algorithms,” 2018, *arXiv:1803.02999*.

[69] L. Bertinetto, J. F. Henriques, P. H. Torr, and A. Vedaldi, “Meta-learning with differentiable closed-form solvers,” in *Proc. Int. Conf. Learn. Representations*, 2019, pp. 1–13.

[70] K. Lee, S. Maji, A. Ravichandran, and S. Soatto, “Meta-learning with differentiable convex optimization,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 10657–10665.

[71] A. Rajeswaran, C. Finn, S. M. Kakade, and S. Levine, “Meta-learning with implicit gradients,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2019.

[72] T. Cover and P. Hart, “Nearest neighbor pattern classification,” *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, Jan. 1967.

[73] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 586–595.

[74] G. Koch et al., “Siamese neural networks for one-shot image recognition,” in *Proc. ICML Deep Learn. Workshop*, 2015, pp. 1–8.

[75] O. Vinyals et al., “Matching networks for one shot learning,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3637–3645.

[76] S. Laenen and L. Bertinetto, “On episodes, prototypical networks, and few-shot learning,” in *Proc. Adv. Neural Inf. Process. Syst.*, pp. 24581–24592, 2021.

[77] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales, “Learning to compare: Relation network for few-shot learning,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1199–1208.

[78] V. Garcia and J. Bruna, “Few-shot learning with graph neural networks,” in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–12.

[79] K. Allen, E. Shelhamer, H. Shin, and J. Tenenbaum, “Infinite mixture prototypes for few-shot learning,” in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 232–241.

[80] X. Jiang, M. Havaei, F. Varno, G. Chartrand, N. Chapados, and S. Matwin, “Learning to learn with conditional class dependencies,” in *Proc. Int. Conf. Learn. Representations*, 2019, pp. 1–11.

[81] A. A. Rusu et al., “Meta-learning with latent embedding optimization,” in *Proc. Int. Conf. Learn. Representations*, 2019.

[82] E. Triantafillou et al., “Meta-dataset: A dataset of datasets for learning to learn from few examples,” in *Proc. Int. Conf. Learn. Representations*, 2019.

[83] D. Wang, Y. Cheng, M. Yu, X. Guo, and T. Zhang, “A hybrid approach with optimization-based and metric-based meta-learner for few-shot learning,” *Neurocomputing*, vol. 349, pp. 202–211, 2019.

[84] Y. Tian, Y. Wang, D. Krishnan, J. B. Tenenbaum, and P. Isola, “Rethinking few-shot image classification: A good embedding is all you need?,” in *Proc. 16th Eur. Conf. Comput. Vis.*, Glasgow, U.K., 2020, pp. 266–282.

[85] Y. Chen, Z. Liu, H. Xu, T. Darrell, and X. Wang, ”Meta-baseline: Exploring simple meta-learning for few-shot learning,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 9062–9071.

[86] Y. Guo et al., “A broader study of cross-domain few-shot learning,” in *Proc. 16th Eur. Conf. Comput. Vis.*, Glasgow, U.K., 2020, pp. 124–141.

[87] W.-Y. Chen, Y.-C. Liu, Z. Kira, Y.-C. F. Wang, and J.-B. Huang, “A closer look at few-shot classification,” in *Proc. Int. Conf. Learn. Representations*, 2019, pp. 1–11.

[88] R. Vuorio, S.-H. Sun, H. Hu, and J. J. Lim, “Multimodal model-agnostic meta-learning via task-aware modulation,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 1–12.

[89] G. Denevi, M. Pontil, and C. Ciliberto, "The advantage of conditional meta-learning for biased regularization and fine tuning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 964–974.
[90] H. Yao, Y. Wei, J. Huang, and Z. Li, "Hierarchically structured meta-learning," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7045–7054.
[91] W. Jiang, J. Kwok, and Y. Zhang, "Subspace learning for effective meta-learning," in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 10177–10194.
[92] G. Jerfel, E. Grant, T. Griffiths, and K. A. Heller, "Reconciling meta-learning and continual learning with online mixtures of tasks," *Adv. Neural Inf. Process. Syst.*, 2019, pp. 9122–9133.
[93] P. Zhou, Y. Zou, X.-T. Yuan, J. Feng, C. Xiong, and S. Hoi, "Task similarity aware meta learning: Theory-inspired improvement on MAML," in *Proc. 37th Conf. Uncertainty Artif. Intell.*, 2021, pp. 23–33.
[94] A. Vettoruzzo, M.-R. Bouguelia, and T. Rögnvaldsson, "Meta-learning from multimodal task distributions using multiple sets of meta-parameters," in *Proc. Int. Joint Conf. Neural Netw.*, 2023, pp. 1–8.
[95] H. Li, W. Dong, X. Mei, C. Ma, F. Huang, and B.-G. Hu, "LGM-Net: Learning to generate matching networks for few-shot learning," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 3825–3834.
[96] L. Liu, W. L. Hamilton, G. Long, J. Jiang, and H. Larochelle, "A universal representation transformer layer for few-shot image classification," in *Proc. Int. Conf. Learn. Representations*, 2020, pp. 1–11.
[97] W.-H. Li, X. Liu, and H. Bilen, "Universal representation learning from multiple domains for few-shot classification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 9526–9535.
[98] J. Requeima, J. Gordon, J. Bronskill, S. Nowozin, and R. E. Turner, "Fast and flexible multi-task classification using conditional neural adaptive processes," *Adv. Neural Inf. Process. Syst.*, 2019, pp. 7959–7970.
[99] N. Dvornik, C. Schmid, and J. Mairal, "Selecting relevant features from a multi-domain representation for few-shot classification," in *Proc. 16th Eur. Conf. Comput. Vis.*, Glasgow, U.K., 2020, pp. 769–786.
[100] E. Triantafillou, H. Larochelle, R. Zemel, and V. Dumoulin, "Learning a universal template for few-shot dataset generalization," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 10424–10433.
[101] W.-Y. Lee, J.-Y. Wang, and Y.-C. F. Wang, "Domain-agnostic meta-learning for cross-domain few-shot classification," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2022, pp. 1715–1719.
[102] B. Kang and J. Feng, "Transferable meta learning across domains," in *Proc. Conf. Uncertainty Artif. Intell.*, 2018, pp. 177–187.
[103] Y. Li and J. Zhang, "Semi-supervised meta-learning for cross-domain few-shot intent classification," in *Proc. 1st Workshop Meta Learn. Its Appl. Natural Lang. Process.*, 2021, pp. 67–75.
[104] W. Yuan, Z. Zhang, C. Wang, H. Song, Y. Xie, and L. Ma, "Task-level self-supervision for cross-domain few-shot learning," in *Proc. AAAI Conf. Artif. Intell.*, 2022, pp. 3215–3223.
[105] T. Bansal, R. Jha, T. Munkhdalai, and A. McCallum, "Self-supervised meta-learning for few-shot natural language classification tasks," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2020, pp. 522–534.
[106] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2017, pp. 1273–1282.
[107] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proc. Mach. Learn. Syst.*, vol. 2, pp. 429–450, 2020.
[108] Q. Li, B. He, and D. Song, "Model-contrastive federated learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 10713–10722.
[109] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 5132–5143.
[110] F. Hanzely and P. Richtárik, "Federated learning of a mixture of global and local models," 2020, *arXiv:2002.05516*.
[111] C. T. Dinh, N. Tran, and J. Nguyen, "Personalized federated learning with Moreau envelopes," *Adv. Neural Inf. Process. Syst.*, 2020, pp. 21394–21405.
[112] T. Li, S. Hu, A. Beirami, and V. Smith, "Ditto: Fair and robust federated learning through personalization," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 6357–6368.
[113] J. Xu, X. Tong, and H. Shao-Lun, "Personalized federated learning with feature alignment and classifier collaboration," in *Proc. Int. Conf. Learn. Representations*, 2023, pp. 1–14.
[114] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran, "An efficient framework for clustered federated learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 19586–19597.
[115] M. Duan et al., "Flexible clustered federated learning for client-level data distribution shift," *IEEE Trans. Parallel Distrib. Syst.*, vol. 33, no. 11, pp. 2661–2674, Nov. 2022.
[116] F. Sattler, K.-R. Müller, and W. Samek, "Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 8, pp. 3710–3722, Aug. 2021.
[117] L. Yang, J. Huang, W. Lin, and J. Cao, "Personalized federated learning on non-IID data via group-based meta-learning," *ACM Trans. Knowl. Discov. Data*, vol. 17, no. 4, pp. 1–20, 2023.
[118] Y. Jiang, J. Konečn, K. Rush, and S. Kannan, "Improving federated learning personalization via model agnostic meta learning," 2019, *arXiv:1909.12488*.
[119] A. Fallah, A. Mokhtari, and A. Ozdaglar, "Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 3557–3568.
[120] F. Chen, M. Luo, Z. Dong, Z. Li, and X. He, "Federated meta-learning with fast convergence and efficient communication," 2018, *arXiv:1802.07876*.
[121] M. Khodak, M.-F. F. Balcan, and A. S. Talwalkar, "Adaptive gradient-based meta-learning methods," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 5917–5928.
[122] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.
[123] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9729–9738.
[124] J.-B. Grill et al., "Bootstrap your own latent-a new approach to self-supervised learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 21271–21284.
[125] A. V. D. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.
[126] K. Hsu, S. Levine, and C. Finn, "Unsupervised learning via meta-learning," in *Proc. Int. Conf. Learn. Representations*, 2019, pp. 1–14.
[127] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial feature learning," in *Proc. Int. Conf. Learn. Representations*, 2017, pp. 1–12.
[128] J. Donahue and K. Simonyan, "Large scale adversarial representation learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 10542–10552.
[129] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 132–149.
[130] S. Khodadadeh, L. Boloni, and M. Shah, "Unsupervised meta-learning for few-shot image classification," in *Proc. Adv. Neural Inf. Processi. Syst.*, 2019, pp. 10132–10142.
[131] S. Khodadadeh, S. Zehtabian, S. Vahidian, W. Wang, B. Lin, and L. Bölöni, "Unsupervised meta-learning through latent-space interpolation in generative models," in *Proc. Int. Conf. Learn. Representations*, 2021.
[132] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "Autoaugment: Learning augmentation strategies from data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 113–123.
[133] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, "R3M: A universal visual representation for robot manipulation," in *Proc. Conf. Robot Learn.*, 2023, pp. 892–909.
[134] A. Tamkin, M. Wu, and N. Goodman, "Viewmaker networks: Learning views for unsupervised representation learning," in *Proc. Int. Conf. Learn. Representations*, 2020.
[135] D. B. Lee, D. Min, S. Lee, and S. J. Hwang, "Meta-GMVAE: Mixture of Gaussian VAE for unsupervised meta-learning," in *Proc. Int. Conf. Learn. Representations*, 2021, pp. 1–13.
[136] D. Kong, B. Pang, and Y. N. Wu, "Unsupervised meta-learning via latent space energy-based model of symbol vector coupling," in *Proc. 5th Workshop Meta- Learn. Conf. Neural Inf. Process. Syst.*, 2021, pp. 1–9.
[137] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. Int. Conf. Learn. Representations*, 2014, pp. 1–9.
[138] Y. W. Teh, M. Welling, S. Osindero, and G. E. Hinton, "Energy-based models for sparse overcomplete representations," *J. Mach. Learn. Res.*, vol. 4, no. Dec, pp. 1235–1260, 2003.
[139] R. Ni, M. Shu, H. Souri, M. Goldblum, and T. Goldstein, "The close relationship between contrastive learning and meta-learning," in *Proc. Int. Conf. Learn. Representations*, 2021, pp. 1–12.

[140] Z. Yang, J. Wang, and Y. Zhu, "Few-shot classification with contrastive learning," in *Proc. 17th Eur. Conf. Comput. Vis.*, 2022, pp. 293–309.

[141] D. B. Lee et al., "Self-supervised set representation learning for unsupervised meta-learning," in *Proc. Int. Conf. Learn. Representations*, 2023, pp. 1–13.

[142] H. Jang, H. Lee, and J. Shin, "Unsupervised meta-learning via few-shot pseudo-supervised contrastive learning," in *Proc. Int. Conf. Learn. Representations*, 2023, pp. 1–13.

[143] C. Wu, F. Wu, and Y. Huang, "One teacher is enough? Pre-trained language model distillation from multiple teachers," in*Proc. Findings Annu. Meeting Assoc. Comput. Linguistics*, 2021, pp. 4408–4413.

[144] T. Bansal, R. Jha, and A. McCallum, "Learning to few-shot learn across diverse natural language classification tasks," in *Proc. Int. Conf. Comput. Linguistics*, 2020, pp. 5108–5123.

[145] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," 2014, *arXiv:1412.3474*.

[146] Y. Ganin et al., "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2096–2030, 2016.

[147] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2223–2232.

[148] K. Rao, C. Harris, A. Irpan, S. Levine, J. Ibarz, and M. Khansari, "RL-CycleGAN: Reinforcement learning aware simulation-to-real," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11 157–11 166.

[149] L. Smith, N. Dhawan, M. Zhang, P. Abbeel, and S. Levine, "Avid: Learning multi-stage tasks via pixel-level translation of human videos," 2019, *arXiv:1912.04443*.

[150] J. Hoffman et al., "Cycada: Cycle-consistent adversarial domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1989–1998.

[151] H. Zhao, S. Zhang, G. Wu, J. M. Moura, J. P. Costeira, and G. J. Gordon, "Adversarial multiple source domain adaptation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 8568–8579.

[152] L. T. Nguyen-Meidine, A. Belal, M. Kiran, J. Dolz, L.-A. Blais-Morin, and E. Granger, "Unsupervised multi-target domain adaptation through knowledge distillation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2021, pp. 1339–1347.

[153] Z. Chen, J. Zhuang, X. Liang, and L. Lin, "Blending-target domain adaptation by adversarial meta-adaptation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2248–2257.

[154] B. Gholami, P. Sahu, O. Rudovic, K. Bousmalis, and V. Pavlovic, "Unsupervised multi-target domain adaptation: An information theoretic approach," *IEEE Trans. Image Process.*, vol. 29, pp. 3993–4002, 2020.

[155] M. Zhang, H. Marklund, N. Dhawan, A. Gupta, S. Levine, and C. Finn, "Adaptive risk minimization: Learning to adapt to domain shift," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 23 664–23 678.

[156] W. Yang, C. Yang, S. Huang, L. Wang, and M. Yang, "Few-shot unsupervised domain adaptation via meta learning," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2022, pp. 1–6.

[157] Y. Feng et al., "Similarity-based meta-learning network with adversarial domain adaptation for cross-domain fault identification," *Knowl.-Based Syst.*, vol. 217, 2021, Art. no. 106829.

[158] A. Sicilia, X. Zhao, and S. J. Hwang, "Domain adversarial neural networks for domain generalization: When it works and how to improve," *Mach. Learn.*, vol. 112, pp. 2685–2721, 2023.

[159] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *Proc. Eur. Conf. Comput. Vis.*, Amsterdam, The Netherlands, 2016, pp. 443–450.

[160] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–13.

[161] V. Verma et al., "Manifold mixup: Better representations by interpolating hidden states," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6438–6447.

[162] H. Yao et al., "Improving out-of-distribution robustness via selective augmentation," in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 25 407–25 437.

[163] Q. Dou, D. Coelho de Castro, K. Kamnitsas, and B. Glocker, "Domain generalization via model-agnostic learning of semantic features," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 6450–6461.

[164] D. Li, J. Zhang, Y. Yang, C. Liu, Y.-Z. Song, and T. M. Hospedales, "Episodic training for domain generalization," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 1446–1455.

[165] Y. Li, Y. Yang, W. Zhou, and T. Hospedales, "Feature-critic networks for heterogeneous domain generalization," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 3915–3924.

[166] D. Li, Y. Yang, Y.-Z. Song, and T. Hospedales, "Learning to generalize: Meta-learning for domain generalization," in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 3490–3497.

[167] Y. Balaji, S. Sankaranarayanan, and R. Chellappa, "MetaReg: Towards domain generalization using meta-regularization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 1006–1016.

[168] Y. Shu, Z. Cao, C. Wang, J. Wang, and M. Long, "Open domain generalization with domain-augmented meta-learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 9624–9633.

[169] K. Chen, D. Zhuang, and J. M. Chang, "Discriminative adversarial domain generalization with meta-learning based cross-domain validation," *Neurocomputing*, vol. 467, pp. 418–426, 2022.

[170] G. M. van de Ven, T. Tuytelaars, and A. S. Tolias, "Three types of incremental learning," *Nature Mach. Intell.*, vol. 4, no. 12, pp. 1185–1197, 2022.

[171] D. Lopez-Paz and M. Ranzato, "Gradient episodic memory for continual learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6470–6479.

[172] A. Chaudhry, M. Ranzato, M. Rohrbach, and M. Elhoseiny, "Efficient lifelong learning with A-GEM," in *Proc. Int. Conf. Learn. Representations*, 2019, pp. 1–12.

[173] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "iCaRL: Incremental classifier and representation learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2001–2010.

[174] R. Aljundi, M. Rohrbach, and T. Tuytelaars, "Selfless sequential learning," in *Proc. Int. Conf. Learn. Representations*, 2019, pp. 1–12.

[175] J. Kirkpatrick et al., "Overcoming catastrophic forgetting in neural networks," *Proc. Nat. Acad. Sci.*, vol. 114, no. 13, pp. 3521–3526, 2017.

[176] J. Serra, D. Suris, M. Miron, and A. Karatzoglou, "Overcoming catastrophic forgetting with hard attention to the task," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 4548–4557.

[177] X. Li, Y. Zhou, T. Wu, R. Socher, and C. Xiong, "Learn to grow: A continual structure learning framework for overcoming catastrophic forgetting," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 3925–3934.

[178] Q. Pham, C. Liu, D. Sahoo, and H. Steven, "Contextual transformation networks for online continual learning," in *Proc. Int. Conf. Learn. Representations*, 2021, pp. 1–13.

[179] A. A. Rusu et al., "Progressive neural networks," 2016, *arXiv:1606.04671*.

[180] S. Beaulieu et al., "Learning to continually learn," in *Proc. 24th Eur. Conf. Artif. Intell.*, 2020, pp. 992–1001.

[181] M. Riemer et al., "Learning to learn without forgetting by maximizing transfer and minimizing interference," in *Proc. Int. Conf. Learn. Representations*, 2019, pp. 1–14.

[182] K. Javed and M. White, "Meta-learning representations for continual learning," in *Proc. Adv. Neural Inf. Process. Sys.*, 2019, pp. 1820–1830.

[183] G. Gupta, K. Yadav, and L. Paull, "Look-ahead meta learning for continual learning," in *Proc. Adv. Neural Inf. Process. Syst.*, pp. 11 588–11 598, 2020.

[184] V. Prabhu, A. Kannan, M. Ravuri, M. Chaplain, D. Sontag, and X. Amatriain, "Few-shot learning for dermatological disease diagnosis," in *Proc. Mach. Learn. Healthcare Conf.*, 2019, pp. 532–552.

[185] P. P. Liang, P. Wu, L. Ziyin, L.-P. Morency, and R. Salakhutdinov, "Cross-modal generalization: Learning in low resource modalities via meta-alignment," in *Proc. 29th ACM Int. Conf. Multimedia*, 2021, pp. 2680–2689.

[186] J.-B. Alayrac et al., "Flamingo: A visual language model for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 23 716–23 736.

[187] S. Reed et al., "A generalist agent," *Trans. Mach. Learn. Res.*, 2022, pp. 1–42.

[188] M. Rußwurm, S. Wang, M. Korner, and D. Lobell, "Meta-learning for few-shot land cover classification," in *Proc. IIII/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2020, pp. 200–201.

[189] T. Yu et al., "One-shot imitation from observing humans via domain-adaptive meta-learning," Robotics: Science and Systems XIV, 2018, pp. 1–10.

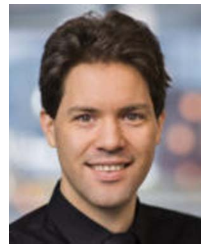
[190] C. Q. Nguyen, C. Kreatsoulas, and K. M. Branson, "Meta-learning GNN initializations for low-resource molecular property prediction," in *Proc. 4th Lifelong Mach. Learn. Workshop*, 2020, pp. 1–6.

[191] L.-Y. Gui, Y.-X. Wang, D. Ramanan, and J. M. Moura, "Few-shot human motion prediction via meta-learning," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 432–450.

[192] I. Ullah et al., “Meta-album: Multi-domain meta-dataset for few-shot image classification,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 3232–3247.
[193] J. Bornschein et al., “Nevis’22: A stream of 100 tasks sampled from 30 years of computer vision research,” 2022, *arXiv:2211.11747*.
[194] T. Yu et al., “Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning,” *Conf. Robot Learn.*, 2020, pp. 1094–1100.
[195] X. Zhai et al., “A large-scale study of representation learning with the visual task adaptation benchmark,” 2019, *arXiv:1910.04867*.
[196] A. R. Zamir, A. Sax, W. Shen, L. J. Guibas, J. Malik, and S. Savarese, “Taskonomy: Disentangling task transfer learning,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3712–3722.
[197] L. Li et al., “Value: A multi-task benchmark for video-and-language understanding evaluation,” in *Proc. 35th Conf. Neural Inf. Process. Syst. Datasets Benchmarks Track*, 2021, pp. 1–13.
[198] A. Srivastava et al., “Beyond the imitation game: Quantifying and extrapolating the capabilities of language models,” *Trans. Mach. Learn. Res.*, 2023, pp. 1–42.
[199] L. Metz, N. Maheswaranathan, C. D. Freeman, B. Poole, and J. Sohl-Dickstein, “Tasks, stability, architecture, and compute: Training more effective learned optimizers, and using them to train themselves,” 2020, *arXiv:2009.11243*.
[200] J. Chen, X.-M. Wu, Y. Li, Q. Li, L.-M. Zhan, and F.-L. Chung, “A closer look at the training strategy for modern meta-learning,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 396–406.
[201] J. Lucas, M. Ren, I. R. K. KAMENI, T. Pitassi, and R. Zemel, “Theoretical bounds on estimation error for meta-learning,” in *Proc. Int. Conf. Learn. Representations*, 2020, pp. 1–12.
[202] J. Guan and Z. Lu, “Fast-rate pac-Bayesian generalization bounds for meta-learning,” in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 7930–7948.

Anna Vettoruzzo received the MSc degree in ICT for Internet and Multimedia from the University of Padova, in 2021, with focus on Machine Learning for Healthcare. She is currently working toward the PhD degree with the Center for Applied and Intelligence Systems Research (CAISR), Halmstad University (Sweden). Her research interests include the areas of meta-learning, few-shot learning, and continuous learning.

Mohamed-Rafik Bouguelia received the MSc degree in computer science from USTHB University, Algeria, and the PhD degree in computer science with a focus on Machine Learning from the University of Lorraine, France. He is an associate professor and docent in machine learning with Halmstad University, Sweden. He is also the program manager for the Applied AI program. Previously, he conducted research with the University of Lorraine and the INRIA research center in France. His current research interests include interactive machine learning, representation learning with deep neural networks, transfer learning, and meta-learning.



Joaquin Vanschoren is an associate professor of machine learning with the Eindhoven University of Technology (TU/e). His research focuses on understanding and automating machine learning, meta-learning, and continual learning. He founded and leads OpenML.org, an open science platform for machine learning research used all over the world. He obtained several demonstration and application awards, the Dutch Data Prize, and has been invited speaker at ECDA, StatComp, AutoML@ICML, CiML@NIPS, Reproducibility@ICML, DEEM@SIGMOD and many other conferences. He also co-organized machine learning conferences (e.g. ECMLPKDD 2013, LION 2016) and many workshops, including the AutoML workshop series at ICML.

Thorsteinn Rögnvaldsson (Senior Member, IEEE) received the PhD degree in theoretical physics from Lund University, 1994. He is a professor of computer science with Halmstad University, Sweden. From 2012, he started and directed the Center for Applied Intelligent Systems Research (CAISR), Halmstad University. He did his postdoc with the Oregon Graduate Institute. His research interests include autonomous knowledge creation, machine learning, and self-organization.

KC Santosh (Senior Member, IEEE) received the PhD degree in computer science - AI from INRIA Nancy Grand East Research Centre, France. He is a highly accomplished AI expert - is the chair with the Department of Computer, University of South Dakota (USD). Immediately after his postdoc with the LORIA reseach centre, University of Lorraine, he joined the National Institutes of Health as a research fellow. With funding of more than \$1.3 million, including a \$1 million grant from DEPSCOR (2023) for AI/ML capacity building at USD, he has authored ten books and published more than 240 peer-reviewed research articles. He is also an editor of multiple prestigious journals, such as *IEEE Transactions on AI*, *International Journal of Machine Learning & Cybernetics*, and *International Journal of Pattern Recognition & AI*. As founder of USD’s AI programs, he significantly boosted graduate program enrollment by over 2,000% in three years, establishing inter-disciplinary AI/Data Science programs with various departments.