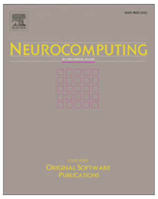
Neurocomputing 494 (2022) 203–223

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom



Survey paper

Meta-learning approaches for learning-to-learn in deep learning: A survey

Yingjie Tian [a,b,c,*], Xiaoxi Zhao [a,b,c], Wei Huang [d,e]

[a] *School of Economics and Management, University of Chinese Academy of Sciences, No.80 of Zhongguancun East Road, Haidian District, Beijing 100190, China*
[b] *Research Center on Fictitious Economy and Data Science, University of Chinese Academy of Sciences, No.80 of Zhongguancun East Road, Haidian District, Beijing 100190, China*
[c] *Key Laboratory of Big Data Mining and Knowledge Management, University of Chinese Academy of Sciences, No.80 of Zhongguancun East Road, Haidian District, Beijing 100190, China*
[d] *College of Business, Southern University of Science and Technology, Shenzhen 518055, China*
[e] *School of Management, Xi'an Jiaotong University, Xi'an 710049, China*

ARTICLE INFO

Article history:
Received 5 September 2021
Revised 5 April 2022
Accepted 17 April 2022
Available online 21 April 2022
Communicated by Zidong Wang

Keywords:
Meta-learning
Learning to learn
Application

ABSTRACT

Compared to traditional machine learning, deep learning can learn deeper abstract data representation and understand scattered data properties. It has gained considerable attention for its extraordinary performances. However, existing deep learning algorithms perform poorly on new tasks. Meta-learning, known as learning to learn, is one of the effective techniques to overcome this issue. Meta-learning's generalization ability to unknown tasks is improved by employing prior knowledge to assist the learning of new tasks. There are mainly three types of meta-learning methods: metric-based, model-based, and optimization-based meta-learning. We investigate classical algorithms and recent meta-learning advances. Second, we survey meta-learning application in real world scenarios. Finally, we discuss present challenges and future research directions of meta-learning.

1. Introduction

Deep learning algorithms have been applied in a variety of tasks, including computer vision [1,2], machine translation [3–5], speech recognition [6,7], and autonomous driving [8,9]. However, a critical limitation of deep learning is its inability to learn new tasks as efficiently as humans do when leveraging previously acquired knowledge. On the one hand, many classifiers require millions of training samples. On the other hand, algorithm performance is highly dependent on labeled data with specified properties, which limits generalization in specific scenarios to some extent [10,11]. Several studies have suggested solutions, such as transfer learning with pre-trained models. Nevertheless, lack of raw data for pre-training, test data different from the pre-training data, samples of specific classes that are too small, and among other factors, might result in poor model performance during the learning process.

Inadequate training samples are a major impediment to the development of deep learning. Numerous studies have been conducted to solve this problem, with an emphasis on unsupervised and semi-supervised model training [12]. Unsupervised techniques involve either learning from unlabeled data using various types of transformed representations or generating data using generative models such as adversarial generative networks. Additionally, self-supervised methods can be utilized to derive knowledge directly from unlabeled data.

Generalizability refers to a model's ability to produce accurate predictions on new tasks. If a model overfits the training data, it will perform poorly in generalization. The existing strategies for improving generalization in deep learning algorithms can be classified perspectives of data-level, model-level, and loss function. When a data set is insufficient, data augmentation might be used to enhance the quantity of data by flipping [13] or rotating [14] existing data, introducing noise [15], or other methods of increasing the amount of data. It is also possible to fit adversarial data into model training process in order to improve a model's resilience to attack [16] and generalization. From a model-level perspective, dropout can be used to cease a neural network operation with a specified probability for the activation value of a particular neuron during forward propagation [17]. Zhang et al. demonstrated that the SGD method has an implicit regular effect [18]. From a loss function perspective, regular terms and label smoothing [19] are often employed to increase a model's robustness, such as sparsify-

[*] Corresponding author at: School of Economics and Management, University of Chinese Academy of Sciences, No. 80 of Zhongguancun East Road, Haidian District, Beijing 100190, China.

E-mail address: tyj@ucas.ac.cn (Y. Tian).

https://doi.org/10.1016/j.neucom.2022.04.078
0925-2312/© 2022 Elsevier B.V. All rights reserved.

ing features with L1 regular terms [20] and boosting generalization with L2 regular terms [21].

The above improvements address the problem of insufficient sample and improve the model's generalization performance, but are not a systematic learning paradigm for the problem. As a result, how to quickly adapt the model to new tasks when there are limited data has become a rising worry for researchers. Meta-learning is a process in which previous knowledge and experience are used to guide the model's learning of a new task, enabling the model to learn to learn. Additionally, it is an effective way to solve the problem of few-shot learning. Meta-learning first appears in the field of educational psychology [22]. It is defined as the comprehension and adaptation to the process of learning, rather than simply the accumulation of subject knowledge. As machine learning progressed, it is realized that meta-learning better performs and enables algorithm selection, parameter tuning, and other functions by leveraging the algorithm's previously acquired knowledge. Not only should a successful meta-learning model be capable of rapidly predicting unknown data and inferring the implicit rules underlying the data, but it ought to solve similar but distinct issues using a model trained on a specific training data set.

According to our investigation into current meta-learning reviews, we discover that the reviews can be divided into two main types: The first elaborates on the technical details of numerous classical methods [23,24]. The other examines promising applications of meta-learning, such as reinforcement learning and few-shot learning [25]. According to the typical classification [24], we divide meta-learning methods into three categories: metric-based methods, model-based methods, and optimization-based methods. Taking into account the rapid development of the field of meta-learning, we then investigate the recent developments in each category separately, in addition to providing a detailed introduction to a variety of classical methods. Significant research methods proposed by researchers in the area are also summarized. More importantly, we discover that there are only few reviews explore the application of meta-learning in realistic circumstances, despite the interest of many. Consequently, we study the application of meta-learning to specific scenarios, elaborate on the evolution of fraud detection and spatial-temporal prediction problems, and offer research findings on natural language processing, image segmentation, and fault diagnosis.

The structure of this paper is as follows: Second 2 introduces the basic concepts of meta-learning and the comparison with transfer learning and multi-task learning. Second 3 presents metric-based, model-based, and optimization-based meta-learning methods, both classical methods and novel research advances. Second 4 details two application scenarios involving meta-learning, fraud detection, and spatial-temporal prediction problems. Moreover, we describe research on natural language processing, image segmentation, and fault diagnosis. Second 5 summarizes the conclusion, challenges, and future research directions.

2. Background

In this section, we briefly introduce the basic learning framework of meta-learning and compare it with two related areas in machine learning (transfer learning and multi-task learning).

2.1. Meta-learning

For general supervised learning, we are first given a task-specific dataset $D = \{x_i, y_i\}_{i=1}^{n}$, typically a large dataset, and a loss function l. Training with D, the goal is to obtain a predictive model of the form $\hat{y} = f_\theta(x)$ parameterized by θ, by solving:

$$\theta' = \mathrm{argmin}_\theta \mathscr{L}(D, \theta) = \mathrm{argmin}_\theta \sum_{i=1}^{n} \ell(f_\theta(x_i), y_i) \tag{1}$$

Meta-learning is the use of previous knowledge experiences to guide new tasks learning, equipping the network with the ability to learn to learn. Specifically, in the meta-training phase, extracting tasks from the distribution of $p(\mathscr{T})$, the meta-goal is to find a common parameter that plays a role in the task distribution.

$$\omega^* = \mathrm{argmin}_\omega \sum_{\substack{\mathscr{T}_i \sim p(\mathscr{T}) \\ D_i \sim \mathscr{T}_i}} \mathscr{L}_i(D_i, \omega) \tag{2}$$

In the meta-testing phase, given a target task (e.g., task 0), we use the learned meta-knowledge ω^* to obtain the optimal parameters for the target task with fewer samples.

$$\theta_0^* = \mathrm{argmin}_\theta \mathscr{L}_0(D_0, \theta | \omega^*) \tag{3}$$

2.2. Compare with other methods

2.2.1. Transfer learning

Transfer learning is a machine learning approach that refers to a pre-trained model being reused in another task [26,27]. For example, the model developed for task A is used as an initial point and reused in the process of training a model for task B. There is no essential difference between meta-learning and transfer learning in terms of their goals, both are to improve the generalization ability of the learner across multiple tasks. The most significant difference is that meta-learning is trained in the task space, assuming the same distribution of training tasks as the target task. Transfer learning, does not have such strict assumptions and optimizes on the target task. Therefore, the meta-learning model performs better on unknown tasks.

2.2.2. Multi-task learning

Multi-task learning is given m learning tasks where all or some of the tasks are related but not identical. And the goal is to use the useful information contained in multiple learning tasks to help get a more accurate learner for each task learning [28,29]. In multi-task learning, information is shared between tasks, and knowledge is transferred to each other across tasks. So multi-task learning is also called parallel transfer learning. Traditional transfer learning emphasizes the sequence of learning, which is the transfer of knowledge learned in one domain to another domain, and the process of knowledge transfer is serial. While multi-task learning aims to solve a fixed number of known tasks, meta-learning aims to find a model that can learn new tasks rapidly, solving tasks that have not been seen before. Nevertheless, meta-learning can also be combined with multi-task learning to produce models with better performance [30,31].

3. Meta-learning

This section summarizes the classical approaches and recent researches in meta-learning according to the typical classification methods, mainly divided into metric-based, model-based, and optimization-based meta-learning [25,24]. The summary is shown in Table 1.

3.1. Metric-based meta-learning

Metric-based learning is known as similarity-based learning. It is to learn a space with good feature descriptions. The images are projected into the space and the similarity between the two images is calculated. In general, if two images are close to each other, then

Table 1
Classical methods and recent research progress.

Class	Methods	Reference	Summary
Metric-Based	Siamese Neural Networks	[32–36]	We show four metric-based meta-learning algorithms, focusing on feature extractors, similarity metrics, and automatic algorithm selection However, the metric-based approaches are sensitive to the dataset and increase the computational expenditure when the number of tasks is large.
	Matching Networks	[37–41]	
	Prototype Networks	[42–46]	
	Relation Networks	[47–53]	
Model-Based	Memory-Augmented Neural Networks	[54–56] [57,58]	We display three model-based approaches. MANN combines neural networks with external memory modules, but the model is complex. Meta-Net is computationally intensive and has high memory requirements. SNAIL is relatively simplified, but has to be optimized in terms of automatic parameter tuning and reducing computation.
	Meta Networks	[59–65]	
	Simple Neural Attentive Meta-Learner	[66–71]	
Optimization-Based	MAML	[72–80]	We present three methods of optimization-based meta-learning. MAML is relatively simple to implement, but the capacity of the model is limited. Meta-LSTM has a large capacity, but a complicated training process. Meta-SGD has improved capacity but still has difficulties in generalization ability.
	META- LSTM	[81–86]	
	META- SGD	[87–93]	

the similarity between the two images is high and they are likely to belong to the same category. The current classical metric-based meta-learning methods are mainly about networks, such as siamese networks [32], matching networks [37], prototypical networks [42], and relation networks [47], etc.

3.1.1. Siamese neural networks

Siamese Neural Networks, a twin network is proposed to evaluate the similarity of the input image pairs [32]. It is able to calculate a distance score by extracting features from both images using the same network and measuring the distance between the features (using the distance metric). And the network weights can be updated according to the loss function. When the network is trained, its discriminative features can be used to generalize to new classes that have not been seen before.

The siamese neural network has L fully-connected layers, each with N_l unit, where $\mathbf{h}_{1,l}$ denotes the hidden vector of the first twin layer in layer l and $\mathbf{h}_{2,l}$ denotes the same vector of the second twin layer. We use specialized rectified linear units (ReLU) in the first $(L-1)$ layers, so that for any layer$l \in \{1, \ldots, L-1\}$:

$$\begin{aligned} h_{1,m} &= \max\left(0, \mathbf{W}_{l-1,l}^{T}\mathbf{h}_{1,(l-1)} + \mathbf{b}_l\right) \\ h_{2,m} &= \max\left(0, \mathbf{W}_{l-1,l}^{T}\mathbf{h}_{2,(l-1)} + \mathbf{b}_l\right) \end{aligned} \tag{4}$$

where $\mathbf{W}_{l-1,l}$ denotes the $N_{l-1} \times N_l$ shared weight matrix connecting the N_{l-1} units of layer $l-1$ and the N_l unit of layer l, and $\mathbf{b}_l$ is denotes the shared deviation vector of layer l.

After the $(L-1)th$ feed-forward layer, we use a fixed distance function to compare each twin's features:

$$p = \sigma\left(\sum_j \alpha_j \left|\mathbf{h}_{1,l}^{(j)} - \mathbf{h}_{2,l}^{(j)}\right|\right) \tag{5}$$

where σ is the sigmoidal activation function.

The last layer generalizes a metric on the feature space of the $(L-1)th$ hidden layer and scores the similarity between the two feature vectors. The α_j are additional parameters that are learned by the model during training, weighing the importance of the component-wise distance. The final L fully-connected layer of the network is defined [33].

Since meta-features do not provide enough information to train meta-learners effectively, Shorfuzzaman et al. proposed a collaborative approach that integrated contrastive learning with a fine-tuned pre-trained ConvNet encoder to capture unbiased feature representations [34]. In [35], Beel et al. proposed a Siamese neural network architecture for automatic algorithm selection, which focused more on "alike performing" instances rather than meta-features. They also proposed a new performance metric called Relative Intra-Instance-Performance*Max-Possible Relative Error (*RIIP_MPRE*). In magnetic resonance imaging (MRI) for automated spinal metastasis detection, Wang et al. developed a siamese deep neural network method that included three identical subnetworks to accommodate the large variation in metastatic lesion size for multi-resolution analysis and detection of spinal metastases [36].

3.1.2. Matching networks

The matching network, as a general network framework, adopts the idea of metric learning based on deep neural features and uses external memory to augment the neural network. And it maps a small labeled support set and an unlabeled example to its label, avoiding the need for fine-tuning to adapt to new class types [37].

In terms of the model architecture, the key to matching networks is the ability to generate reasonable test labels for unobserved classes without changing the network. When given a new support set S', we can obtain the prediction about the appropriate label distribution $\hat{y}$ for each test case $\hat{x}$. The matching network calculates the probability of $\hat{y}$ in its simplest form as follows:

$$P(\hat{y}|\hat{x},S)=\sum_{i=1}^{k}a(\hat{x},x_i)y_i \tag{6}$$

where (x_i,y_i) are the inputs and corresponding label distributions from the support set $S=\{(x_i,y_i)\}_{i=1}^{k}$ and a is the attention mechanism,$a(\hat{x},x_i)=e^{c(f(\hat{x}),g(x_i))}/\sum_{j=1}^{k}e^{c\left(f(\hat{x}),g\left(x_j\right)\right)}$,$f$ and g are two embedding functions.

In terms of the training strategy, the training process explicitly learns to learn from the given support set to minimize the loss of batch processing. Concretely, L is the first sampled from T (L can be the label set{cats, dogs}). Then, we use L to sample the support set S and batch B (both S and B are labeled as cats and dogs for example). The matching network is then trained to minimize the error predicting the labels in the batch B conditioned on the support set S. The Matching Nets training objective is as follows:

$$\theta=\arg\max_{\theta}E_{L\sim T}\left[E_{S\sim L,B\sim L}\left[\sum_{(x,y)\in B}\log P_{\theta}(y|x,S)\right]\right] \tag{7}$$

In [38], the one-shot learning technique of matching network was used to address the problem of insufficient training data in action detection, using correlations to mine and localize actions of previously unseen classes. In order to stop retraining the classifier and run it in real-time, the algorithm proposed by Choi et al. utilized a meta-learning network to provide the matching network with new information about the appearance of the target object by adding the target piece feature space [39]. In dealing with reading comprehension problems, Zhang et al. proposed a dual-matching network that selected short sentences to find the most salient supporting sentences to answer the question. And then the answer options interacted to encode comparative information between the answer options so that the relationship between the short text and the question could be modeled in both directions [40]. Furthermore, a sequence matching network was proposed for polyphonic sound event detection and direction-of-arrival estimation [41].

3.1.3. Prototype networks

The core idea of Prototype Network is to consider the existence of an embedded representation in which the feature points corresponding to each category are clustered near a single prototype representation. Based on this idea, a method similar to clustering is used to implement few-shot classification [42]. In particular, the prototype network learns a metric space. The samples of each class are mapped into the space and their "mean values" are extracted to represent them as class prototypes. The M-dimensional representation $\mathbf{c}_k\in R^M$ of each class is computed by an embedding function $f_{\phi}:R^D\rightarrow R^M$ with learnable parameters ϕ. The formula is as follows:

$$\mathbf{c}_k=\frac{1}{|S_k|}\sum_{(\mathbf{x}_i,y_i)\in S_k}f_{\phi}(\mathbf{x}_i) \tag{8}$$

where $S=\{(\mathbf{x}_1,y_1),\ldots,(\mathbf{x}_N,y_N)\},\mathbf{x}_i\in R^D$ is the D-dimensional feature vector of an example and $y_i\in\{1,\ldots,K\}$ is the corresponding label. S_k denotes the set of examples labeled with class k.

Using the Euclidean distance as a distance measure, the data in this category are trained to be the closest to the prototype representation of this class and farther to the prototype representation of other classes. The distribution of query points x among classes is determined by the softmax of the prototype distance in the embedding space. The formula is as follows:

$$p_{\phi}(y=k|\mathbf{x})=\frac{\exp\left(-d(f_{\phi}(\mathbf{x}),\mathbf{c}_k)\right)}{\sum_{k'}\exp\left(-d(f_{\phi}(\mathbf{x}),\mathbf{c}_{k'})\right)} \tag{9}$$

where $d:R^M\times R^M\rightarrow[0,+\infty)$.

Learning proceeds by minimizing the negative log-probability $J(\phi)=-\log p_{\phi}(y=k|\mathbf{x})$ of the true class k via SGD. Training episodes are formed by randomly selecting a subset of classes from the training set, then choosing a subset of examples within each class as the support set and a subset of the remainder as the query set.

In [43], Boney et al. demonstrated that using larger and better regularized prototype networks could improve classification accuracy. After that, the attention mechanism was gradually introduced into the prototype network [44]. Xiao et al. integrated prototypes, partial matching, and top-down attention regulation into deep neural networks to realize robust object classification under occlusion [45]. This ensemble model not only achieves better generalization ability in case of inconsistent training and test data, but also filters out irrelevant information when features are compared with prototypes using partial matching. To better transfer attribute-based knowledge from known classes to unknown classes, Gao et al. proposed a hybrid attention-based prototype network with less noise lens RC, which made key instances and features more prominent [44]. Xu et al. proposed a new zero-shot representation learning framework that jointly learned discriminative global and local features only using class-level attributes [46].

3.1.4. Relation networks

The idea of the Relation Network is to construct a comparator using a neural network to compare the characteristics of the samples and determine the correlation between the characteristics of the test samples and the support set [47]. To be specific, samples x_i in the sample set $\mathscr{S}$ and samples x_j in the query set $\mathscr{Q}$ are fed through the embedding module f_{φ}, which produces feature maps $f_{\varphi}(x_i)$ and $f_{\varphi}(x_j)$. The feature maps $f_{\varphi}(x_i)$ and $f_{\varphi}(x_j)$ are combined with operator $\mathscr{C}\left(f_{\varphi}(x_i),f_{\varphi}(x_j)\right)$. The combined feature map of the sample and query are fed into the relation module g_{ϕ}, which eventually produces a scalar in range of 0 to 1 representing the similarity between x_i and x_j, referred to as the relation score. Thus, in the C-way one-shot setting, we generate C relation scores $r_{i,j}$ for the relation between one query input x_j and training sample set examples x_i.

$$r_{i,j}=g_{\phi}\left(\mathscr{C}\left(f_{\varphi}(x_i),f_{\varphi}(x_j)\right)\right),\quad i=1,2,\ldots,C \tag{10}$$

The mean square error (MSE) is used in the objective function to train the model, regressing the relation score $r_{i,j}$ to the ground truth: matched pairs have similarity 1 and the mismatched pair have similarity 0.

$$\varphi,\phi\leftarrow\underset{\varphi,\phi}{\operatorname{argmin}}\sum_{i=1}^{m}\sum_{j=1}^{n}\left(r_{i,j}-\mathbf{1}(y_i==y_j)\right)^2 \tag{11}$$

Currently, relational networks have been fully applied to images [48–51], texts [52], and videos [51,53], etc. For image processing, Xu et al. proposed a spatial-aware graph relational network (SGRN) that integrated a graph learning module and a spatial graph inference module with a learnable spatial Gaussian kernel. The former was designed to learn interpretable sparse graph structures to encode relevant contextual regions, and the latter was to perform spatially aware graph inference [48]. Instead of considering only each instance, Memory enhanced relational networks (MRN) considered its relationship with other instances, selected visually similar samples and performed weighted information propagation, concentrating on aggregating useful information from the selected samples to enhance their representational power [49]. For text processing, Chen et al. proposed a gated relation network (GRN), which used CNN to mine the local contextual features of each word. Then, the relationships between words were modeled and used as portals to fuse local context features into global context features to predict labels [52].

3.1.5. Other metric-based meta-learning methods

In addition to the above four classical methods, there are also widely used methods such as graph neural networks(GNN) [94]. The main idea of GNN was to transfer label information from labeled samples to unlabeled samples, considering such information transfer as a posteriori inference given by a graph model based on the input images and labels. Differing from graph neural networks based on node labels, EGNN learned to predict edge labels on the graph, and updated edge labels iteratively using intra-cluster similarity and the inter-cluster dissimilarity [95]. Shyam et al. proposed Attentive Recurrent Comparators(ARCs) for one-shot classification [96]. The key difference between ARCs and the above methods was that it focused more on the local information of the object as opposed to viewing the input as a whole, and realized the combination of two parts of information from the input. As well as the visual learning ability of humans to quickly learn what objects look like, Wang et al. combined a meta-learner with a "hallucinator" that generated additional training samples and optimized both models jointly [97].

Different learning algorithms are not isolated individuals like islands. Combining the ideas of multiple learning algorithms may yield extraordinary gains. From our surveyed numerous literature summaries, metric-based meta-learning methods are more often combined with few-shot learning methods. The method aims to learn from a limited number of training samples that can identify new classes [98]. Reinforce Relation Network (RRN) performed feature extraction using a convolutional neural network and identified similarities between samples under different conditions using a metric learner [99]. After extracting features, Zhang et al. optimized them in the metric space using cosine distance and learnable parameters, and discriminated the unseen classes with an adaptive classifier [100]. Nowadays, there are also investigations on the generalization problem and overfitting problem in few-shot learning. MetaDelta improved model efficiency and generalization by two core components (central controller and meta-ensemble module), respectively [101]. The metric-based meta-learning approach proposed by Guo et al. incorporated attention mechanism and ensemble learning approach to avoid the overfitting problem [102].

Except for the research methods described above, another research point that should not be overlooked is the combination of meta-learning with incremental learning. Liu et al. entertained the idea of incremental learning in the meta-training phase and displayed a method that is usable jointly with metric-based meta-learning methods [103]. As the tasks were solved continuously, the model was optimized. Instead of using direct alignments for all tasks, indirect alignment was more likely to adopt the model to different tasks, since there are different classes between tasks [103]. Resolving the task heterogeneity problem as well, the Task Adaptive Network (TAdaNet) constructed the metric space by learning task embeddings that contained task relations as well as task-specific parameters [104]. The sampling method based on greedy class pairs tackled the cross-task problem by adaptive task sampling [105]. To reduce labeling cost and training time, Wang et al. proposed a few-shot incremental learning search framework based on meta-learning, which was capable of learning the discriminative representation of each sample, and also designed feature adapters to ensure robustness [106]. The Random Fine-Tuning Meta Metric Learning Model (RF-MML) dealt with the class imbalance problem by introducing randomness in scenario training and associating it with fine-tuning of all classes [107]. Furthermore, researchers have investigated metric scaling parameters [108], feature embedding [109], and data scarcity [110] in metric-based meta-learning.

3.1.6. Summary

In this section, we show some classical algorithms for metric-based meta-learning. Based on the above, these methods can be innovated from different perspectives such as feature extractor, similarity measure, loss function, and hyperparameter. Additionally, we offer several novel methods for metric-based meta-learning, including the use of adaptive classifiers, attention mechanisms, and combination with incremental learning. But the metric-based method is not robust, relatively sensitive to the dataset, as well as increases the computational overhead when the number of tasks is large.

3.2. Model-based meta-learning

Model-based meta-learning aims to adapt to new tasks by changing the model's learnable parameters θ. When faced with a different task, the model can capture relevant task-specific information and continuously update θ to make predictions. Thus, model-based techniques are not limited to learning a good feature space, as they can also learn the internal dynamics of the model to process and predict the input data for the task.

3.2.1. Memory-augmented neural networks

In [54], Santoro et al. proposed that memory-augmented neural networks (MANN) can quickly assimilate new data and leverage them to make accurate predictions. The structure is shown in Fig. 1.

The whole training process is divided into several episodes, each episode contains several samples and corresponding labels. All the samples are combined into a sequence and input to the network in a staggered manner, (x_t, y_{t-1}) denotes the samples input at time t. From episode to episode, the associated labels and specific samples of each category are shuffled. The right part of Fig. 1 shows the network strategy uses external memory to store the boundary sample representation-class label information, which can be retrieved at a later point for successful classification when a sample of the seen category appears. More specifically, the sample data x_t from a particular time step should be bound to the appropriate class label y_t that will be displayed at subsequent time steps. Afterwards, when samples are seen in the same class, MANN should retrieve this bound information from external memory to make a prediction. Backpropagated error signals from this prediction step will update the weight from the previous steps to facilitate the binding strategy.

For the external memory module, Gulcehre et al. proposed a novel memory-enhanced neural network model called Temporal Automatic Relation Discovery in Sequences(TARDIS). The controller of the TARDIS could selectively store a set of embeddings of previously hidden states in external memory and revisit them when needed [55]. In the field of NLP, the neural semantic encoder was equipped with novel memory update rules and had variable-sized encoding memory that evolved and maintained understanding of the input sequence through reading, coding, and writing operations. All of the natural language inference, question and answer, sentence classification, document sentiment analysis, and machine translation could be performed [56]. Furthermore, text normalization using memory augmented neural networks could convert text from written to spoken form [57]. For long documents, Vu et al. argued that it was unnecessary to keep all entities

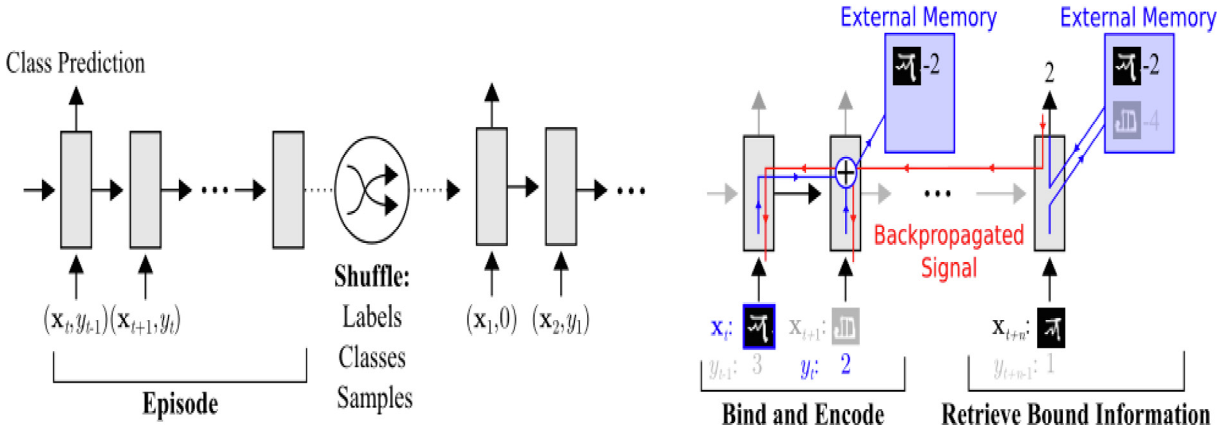


Fig. 1. MANN structure [54].

in memory. And they proposed a memory-enhanced neural network that tracked only a limited number of entities at a time, ensuring a linear runtime of document length [58].

3.2.2. Meta networks

A significant challenge for neural network models is to achieve fast generalization with small training data and maintain the previously learned performance. Munkhdalai et al. proposed a new meta-learning method, meta Networks (MetaNet), which can learn a meta-level knowledge across tasks and transfer its inductive bias by fast parameterization to achieve rapid generalization [59]. The network architecture is shown in Fig. 2.

MetaNet consists of two important parts, Meta learner and Base learner. The Meta learner generates weights quickly by operating across tasks, while the base learner completes each task by capturing the task goals. The generated fast weights are integrated into the base learner and meta learner to change the induction bias of the learners.

More recently, Meta Networks has been applied to style transfer [60,61], image deformation [62,63] and image caption [64,65]. To improve the generalization to new styles, Shen et al. built a meta-network that took in the style image and generated a corresponding image transformation network directly [60]. In [62], Chen et al. combined a meta-learner with an image deformation subnetwork that generated additional training examples and optimized in an end-to-end manner. The deformation subnetwork learned deformed images by fusing a pair of images, a detection image that maintained the visual content and a gallery image that diversified the deformation. Based on this, Cao et al. introduced a feature deformation meta network(FDM-net), which trained on the source data to learn new target features detected by the auxiliary detection model [64].

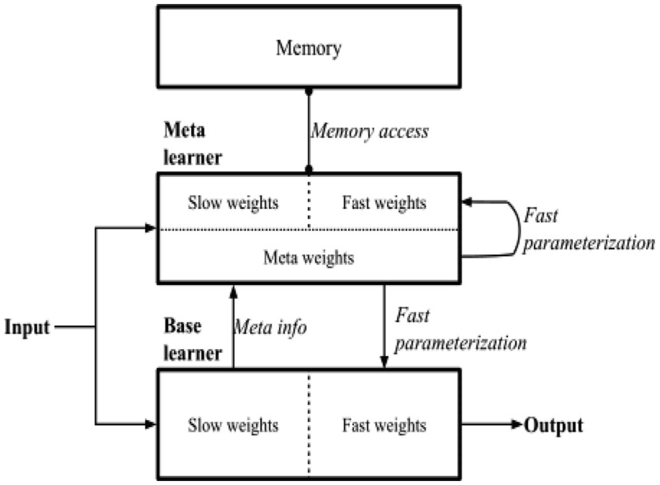


Fig. 2. Overall architecture of Meta Networks [59].

3.2.3. Simple neural attentive meta-learner

For data scarcity or the need to quickly adapt to task changes, Mishra et al. proposed a simple and general meta-learner architecture called Simple Neural Attentive Meta-Learner(SNAIL). SNAIL uses a novel combination of temporal convolution and soft attention mechanism, with the former to gather information from experience and the latter to identify specific information [66].

Temporal convolution is a structure that generates temporal data by one-dimensional convolution inflated in the time dimension, where the values generated at the next time node are only affected by the information from the previous time node. Compared to traditional RNN, it provides a more direct and higher bandwidth way to obtain past information. However, the number of convolution layers needed is logarithmically related to the sequence length, and the expansion rate also grows exponentially. As a result, a large amount of prior experience cannot be fully utilized. Soft attention allows the model to locate information precisely within as large a context as possible, treating the context as an unordered key-value store and allowing queries based on the content of each element. Despite their respective drawbacks, temporal convolution and attention are complementary: the former provides high-bandwidth access at the expense of a limited context size, while the latter provides precise access in an infinitely large context. Hence, SNAIL combines the two: using temporal convolution to process the content that has been extracted with the attention mechanism. By using the attention mechanism in multiple stages, SNAIL learns how to extract the required information from the collected information and represents it appropriately. The architectures of the model can be applied to both supervised learning and reinforcement learning.

Presently, there has been a lot of research on meta-learning for temporal convolutions and attention mechanisms [67–71]. The research of temporal convolutions is mainly in the field of smart city and traffic prediction. Pan et al. proposed a deep meta-learning-based model called ST-MetaNet that collectively predicted the traffic flow in all locations immediately [67]. The meta-learning model proposed by Yao et al. learned a generalized initialized spatial-temporal network that could be effectively adapted to the target city [68]. Also inspired from the Model-Agnostic Meta-Learning framework (MAML) [72], Jiang et al. introduced the Attentive Task-Agnostic Meta-Learning (ATAML) algorithm for text classification [69]. Additionally, a depth metric element approach learned to adaptively generate new FSL task-specific metrics based on the task description (e.g., a few labeled samples) [71].

3.2.4. Other model-based meta-learning methods

Conditional neural processes (CNPs) combined the advantages of deep neural networks and Gaussian processes, allowing both fast inference of the shape of functions with prior knowledge and sound approximation [111]. In meta-learning algorithms, meta-training tasks are mutually exclusive implicitly to prevent memory problems. To cope with this challenge, Yin et al. suggested the use of meta-regularization objective based on CNP(MR-CNP) to adapt to new tasks even when non-mutually exclusive task data are used [112]. In the real world, unexpected interference or unknown objects cause poor model results. More methods adopted in recent years in model-based meta-learning are combining meta-learning with reinforcement learning and online learning to achieve efficient learning and fast adaptive of samples [113–115]. In general, model-based reinforcement learning methods build accurate dynamics models as much as possible and correct for model biases by learning a robust strategy, but struggle to realize the same asymptotic performance as model-free methods. Instead of striving for accuracy, the model-based meta-policy optimization approach (MB-MPO) was implemented by learning a set of dynamics models and treating strategy optimization as a meta-learning problem [116]. MB-MPO contained two levels of loops, the internal loop for generating adaptation strategies for each model and the external loop for generating meta-strategies to adapt to all models.

The effectiveness of deep learning models needs to be supported by extensive experimental data. From our findings, the problem of greater interest to current research in meta-reinforcement learning is how a model trained with fewer data can learn new tasks quickly. Things are both different and related to each other. When confronted again with a problem similar to the previous one, people will have more experience to deal with it. And the search for the optimal exploration strategy can be regarded as a reinforcement learning problem [117]. Animals learn new tasks benefiting from a priori knowledge after several trials. Inspired by this, RL^2 invoked the idea of meta-learning to train a generic model - SLOW - in multiple tasks. And it would be far faster to use this generic model to expand to other tasks to get a new model - FAST [118]. Certainly, we also need to account for forgetting when using prior knowledge. As the bucket has a limited capacity, we do not and cannot use all the prior knowledge, we need to find a mechanism that acts differently on the importance of prior knowledge. Aside from research from a priori knowledge, some works refine from the model itself. Deep meta-reinforcement learning embedded recurrent networks in a reinforcement learning framework [119], and Nagabandi et al. suggested task self-adaptation in the context of model-based reinforcement learning to learn new tasks more efficiently [114]. Keeping in mind the limited number of samples, SMILe was designed to introduce inverse reinforcement learning based on behavioral cloning, which was a scalable framework for meta-inverse reinforcement learning (Meta-IRL) with maximum entropy IRL to deliver better imitation quality on less data [120].

Reinforcement learning algorithms inherently require extensive model configuration and optimization of neural network parameters [121]. Meta-reinforcement learning methods combine meta-learning and reinforcement learning, which will be implemented to a greater extent, require higher costs, and take longer [122]. ENAS leveraged the training results from the previous round of NAS, easing the computational pressure [123]. Further works separated task inference and control to improve the utility of meta-learning algorithms [124].

Much like meta-reinforcement learning, online meta-learning is also focused on rapid adaptation to new tasks, but with a stronger interest in how to achieve rapid adaptation online. As the environment changes over time, the training samples also change dynamically. How to construct models with few parameters and achieve fast online adaptation is a challenging area for online meta-learning research [125]. Nagabandi et al. developed an online learning procedure that updated model parameters using stochastic gradient descent [126]. In the OMPAC method, the training instances of the reinforcement learning algorithm were run in parallel to the initial values of the parameters [127]. Denevi et al. utilized an inner online algorithm to explore the similarity between tasks in order to make the average cumulative error generated over the tasks lower [128]. Other researchers took an orthogonal perspective and presented an online meta-learning framework from transfer learning, which minimized the distance between the source and target domains to enhance the learning performance [129]. However, when dealing with real-world problems, there is a considerable probability of encountering heterogeneous information, which leads to suboptimal results after global search due to its difficulty in sharing. Online structured meta-learning (OSML) decomposed the meta-learner into a meta-hierarchical graph with different meta-knowledge blocks, and constructed meta-knowledge paths using the most relevant meta-knowledge blocks each time [130].

From the above research methods, it is evident that recent studies have focused on how to quickly implement self-adaptation as well as improve the generalization ability of the model [131].

3.2.5. Summary

In this section, we present three classical model-based approaches, MANN, Meta-Net, and SNAIL. MANN combines neural networks with external memory modules and can be used for regression problems, but the models are overly complex. Meta-Net uses meta-learner and base-learner to represent meta-learning and base learning, but the model is computationally intensive with high memory requirements. SNAIL memorizes the tasks into memory through temporal convolution and soft attention mechanisms. The model is more simplified, but the internal modules need to be designed manually. Therefore, there is still further research in automatic parameter adjustment, model simplification, and calculation reduction. In recent years, more model-based meta-learning have been combined with reinforcement learning, and online learning in order to achieve rapid adaptive and efficient sample learning.

3.3. Optimization-based meta-learning

The optimization-based approaches treat meta-learning as an optimization problem, focusing on extracting the meta-knowledge that improves optimization performance and generating a classifier that performs well on the query set with only a few gradient updating steps. In this section, we give a brief introduction to the three classical methods (MAML, META-LSTM, and META-SGD) and some novel research advances.

3.3.1. Model-agnostic meta-learning

A notable model in optimization-based meta-learning is MAML, which obtains model initialization parameters by training. It can produce excellent generalization performance for new tasks with a few gradient update steps and a small amount of training data [72]. Specially, Finn et al. considered a distribution over tasks

$p(\mathscr{T})$, and a model represented by a parametrized function f_θ with parameters θ. Firstly, MAML calculates the gradient of each task in each batch in the support set, and then updates to get the parameter θ'_i, which is the first time the parameters are updated. The parameters are updated as follows:

$$\theta'_i = \theta - \alpha \nabla_\theta \mathscr{L}_{\mathscr{T}_i}(f_\theta) \tag{12}$$

where the step size α may be fixed as a hyperparameter or meta learned.

The authors used gradient descents twice. The second gradient update is used for each task in the query set, and θ is obtained by summing the losses. After executing the batch size update, the parameters can be obtained. The stochastic gradient descent (SGD) method is used to update the model parameters θ as follows:

$$\theta \leftarrow \theta - \beta \nabla_\theta \sum_{\mathscr{T}_i \sim p(\mathscr{T})} \mathscr{L}_{\mathscr{T}_i}\left(f_{\theta'_i}\right) \tag{13}$$

where β is the meta step size.

The MAML algorithm performs well in classification and regression, but requires expensive hyperparameter tuning for training stability. Alpha MAML eliminated the need to adjust hyperparameters by incorporating an online hyperparameter adaptive scheme [73]. In [74], Arnold et al. proposed that an external, separate neural network meta-optimizer could be used to transform the gradient updates of a smaller model. Reptile was one of the most widely used methods based on MAML. It differed most from MAML in that Reptile did not require differentiation through the optimization process, which made it more suitable for optimization problems that required many update steps [75]. As new tasks are learned, catastrophic forgetting might occur, BI-MAML was designed with a balanced learning strategy based on a balanced incremental approach, so that it performed equally well on the existing tasks [76]. Solutions to this problem also came from task agnostic [77,78] and online learning [79,80].

3.3.2. META-LSTM

In [81], Ravi et al. proposed an LSTM-based meta-learning model to learn the exact optimization algorithm. The parameterization of the model allows it to learn appropriate parameter updates, especially when a large number of updates are required, and also to learn the general initialization of the learner (classifier) network for fast convergence of training.

Consider a dataset $D \in \mathscr{D}_{\text{meta-train}}$, and a learning neural network classifier with parameters θ that trained on D_{train}. The general gradient descent optimization algorithm in neural networks is as follows:

$$\theta_t = \theta_{t-1} - \alpha_t \nabla_{\theta_{t-1}} \mathscr{L}_t \tag{14}$$

where θ_{t-1} is the parameter of the learner after $t-1$ updates, α_t is the learning rate at time t. $\mathscr{L}_t$ is the loss optimized by the learner for its t^{th} update, $\nabla_{\theta_{t-1}} \mathscr{L}_t$ is the gradient of that loss with respect to parameters θ_{t-1}, and θ_t is the updated parameters of the learner.

The general update rule of the cell state of LSTM is as follows:

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{15}$$

Based on (15), Ravi et al. made a clever substitution by setting the cell state of the LSTM as a parameter of the learner and the candidate cell state $\tilde{c}_t = -\nabla_{\theta_{t-1}} \mathscr{L}_t$, taking into account the value of gradient information for optimization $f_t = 1, c_{t-1} = \theta_{t-1}, i_t = \alpha_t$, and $\tilde{c}_t = -\nabla_{\theta_{t-1}} \mathscr{L}_t$.

The parameter forms of i_t and f_t are also defined so that the meta-learner can determine the optimal values through the update process. The definitions are as follows:

$$i_t = \sigma\left(\mathbf{W}_I \cdot \left[\nabla_{\theta_{t-1}} \mathscr{L}_t, \mathscr{L}_t, \theta_{t-1}, i_{t-1}\right] + \mathbf{b}_I\right) \tag{16}$$

$$f_t = \sigma\left(\mathbf{W}_F \cdot \left[\nabla_{\theta_{t-1}} \mathscr{L}_t, \mathscr{L}_t, \theta_{t-1}, f_{t-1}\right] + \mathbf{b}_F\right) \tag{17}$$

where i_t corresponds to the updated learning rate, which is a function of the current parameter value θ_{t-1}, the current gradient $\nabla_{\theta_{t-1}} \mathscr{L}_t$, the current loss $\mathscr{L}_t$, and the previous learning rate i_{t-1}. With this information, the meta learner be able to finely control the learning rate so as to train the learner quickly while avoiding divergence. For f_t, the optimal choice of weights for previous round of parameters is not the constant 1. Therefore, the authors proposed the forgetting gate to forget the partial values.

To date, there is a lot of research in natural language processing. In [82], Bohnet et al.proposed the Meta-BiLSTM Model, which integrated context sensitive representations by simultaneously training with a meta-model that learned to combine the states of words. Based on this, Lim et al.proposed a semi-supervised learning method based on consensus facilitation in multi-view learning [83]. For the multi-task underfitting problem, Chen et al. proposed a new sharing scheme of composition function across multiple tasks that used a shared meta-network to capture the meta-knowledge of semantic combinations and generate the parameters of a task-specific [84]. Besides, there is also learning from the perspective of learning rate [85] and kernel function [86].

3.3.3. META-SGD

Based on MAML and Meta-LSTM, Li et al. proposed an easily trainable meta-learner that could initialize and adapt to any differentiable learner [87]. Assume that there is a distribution $p(\mathscr{T})$ over the related task space, from which can randomly sample tasks. A task $\mathscr{T}$ consists of a training set *train*$(\mathscr{T})$ and a testing set *test*$(\mathscr{T})$. Meta-SGD is divided into inner training and outer training. The inner training is performed on the training set. Li et al. proposed the following meta-learners consisting of an initialization term and adaptation terms in the inner-level training:

$$\boldsymbol{\theta}' = \boldsymbol{\theta} - \boldsymbol{\alpha} \circ \nabla \mathscr{L}_{\mathscr{T}}(\boldsymbol{\theta}) \tag{18}$$

where $\boldsymbol{\theta}$ and $\boldsymbol{\alpha}$ are parameters of the meta-learner to be learned, and $\circ$ denotes element-wise product. The adaptation term $\boldsymbol{\alpha} \circ \nabla \mathscr{L}_{\mathscr{T}}(\boldsymbol{\theta})$ is a vector whose direction represents the update direction and whose length represents the learning rate.

The outer level training is performed on the test set, along with learning of initialization parameters θ, learning rate α, and direction of parameter updates. The model goal is to train the meta-learner to maximize the expected generalization power of the meta-learner on the task space. It can be formulated as the following optimization problem:

$$\min_{\boldsymbol{\theta},\boldsymbol{\alpha}} E_{\mathscr{T} \sim p(\mathscr{T})}\left[\mathscr{L}_{\text{test}(\mathscr{T})}(\boldsymbol{\theta}')\right] = E_{\mathscr{T} \sim p(\mathscr{T})}\left[\mathscr{L}_{\text{test}(\mathscr{T})}\left(\boldsymbol{\theta} - \boldsymbol{\alpha} \circ \nabla \mathscr{L}_{\text{train}(\mathscr{T})}(\boldsymbol{\theta})\right)\right] \tag{19}$$

The Meta-SGD method is widely used, and we briefly introduce it from three directions of extension: data deficiency [88,89], performance on generalization [90,91], and self-adaptation [92,93]. When the amount of data is insufficient, on the one hand, optimization-based meta-learning can be performed in this low-dimensional potential space by learning data-dependent potential generative representations of model parameters [88]. And on the other hand, Simon et al. proposed ModGrad, which aimed to circumvent the nature of noisy gradients and achieve fast adaptation [89]. To achieve better generalization, Cheng et al. proposed meta-curvature (MC), a framework for learning curvature information for better generalization and fast adaptation [90]. Masked Meta-SGD (M2SGD) aimed to learn an optimization rule that sparsely updated the learning parameters and removed redundant weights [91]. At last, to achieve self-adaptive, Bartler et al. combined meta-learning, self-supervision, and test-time training to adapt to unseen distributions [92].

3.3.4. Other optimization-based meta-learning methods

Optimization-based meta-learning methods can be grouped into two principal classes: one is to obtain good model initialization parameters, and the other is to learn to optimize the parameters. The first class is represented by MAML which has been rapidly developed during recent years. We have elaborated on two classical approaches of the second class, and this subsection is an orientation to other related methods. Two main areas of research are being done. On the one hand, meta-learning combined with online convex optimization methods to achieve self-adaptation for dynamically changing environments [132,133]. It was also studied to estimate bias vectors online from the task, which required much less memory and time. The method had been proved its effectiveness in practice [134]. Another promising research direction is to integrate meta-learning with federated learning.

The goal of online meta-learning is to enable continuous lifelong learning. It combines meta-learning with online learning, which allows both rapid learning of new tasks using prior knowledge and constant optimization [135]. Other than getting good initialization parameters using MAML, online meta-learning provided another idea to achieve better initialization by weighting the loss of different source tasks [136]. At present, optimization-based online meta-learning approaches look more at how to cope with dynamic changes in the environment and the problem of storing a priori historical knowledge for large amounts [137]. In response to the mismatch between tasks caused by environmental changes, Kaushik et al. demonstrated offline adaptive algorithms to perform fast adaptation to new data, and also worked out online algorithms to enhance the stability of the offline algorithms [138]. Shedivat et al. obtained transfer risk bounds based on algorithmic stability techniques to improve model generalization [139]. Another problem that comes with time is the increasing number of historical behaviors, which poses a great challenge to both the memory of the system and the utilization of historical information [140]. The Long Short Term Temporal Meta-Learning Framework (LSTTM) learned users' short-term preferences separately from their long-term preferences and employed an asynchronous optimization strategy for fast adaptation [141]. Lastly, accounting for the complexity and time consumption in the actual training process, Chen et al. found that the discriminatively trained linear classifier could also achieve comparable generalization with the base learner in meta-learning although with a moderate increase in computational overhead [142].

With data security and privacy become increasingly emphasized, achieving collaborative training across devices with compliance is a critical point of research today. Federated meta-learning also comes to the forefront, and the effectiveness of this learning paradigm for solving edge learning and data heterogeneity problems is widely recognized. For the field of edge learning limited by computational resources and the number of training samples, Lin et al. suggested a platform-assisted collaborative learning framework that was trained at the source edge nodes and then quickly adapted to the target edge nodes [143]. The FML algorithm was also available to accelerate convergence [144]. Federated learning is co-modeling with guaranteed data security and privacy. There is no exchange of data between users, only users communicate with a common central server. Nevertheless, the constructed model is not suitable for each specific user, especially the heterogeneity among user data distributions [145,146]. Fallah et al. investigated a personalized variant of federated learning, where an initial shared model was constructed, and then personalized application to users could be achieved with one or several steps of gradient descent [147]. Meta-HAR was a federated learning representation framework that first performed meta-learning in a federated manner, and then treated each user's problem as a different task, feeding the learned signals back into each user's network so that the shared network could be generalized to each user [148]. Federated meta-learning(FedMeta) provided a shared parameterized algorithm(or meta-learner) that enabled faster convergence, lower cost, and higher accuracy [149]. In parallel, conditional meta-learning methods were available to address the challenge of poor model performance in heterogeneous environments [150]. More generally, federated meta-learning is an equally critical approach for protecting the security and privacy of data [151,152].

3.3.5. Summary

In this section, we present three classical methods of optimization-based meta-learning (MAML, META-LSTM, and META-SGD). MAML is to use meta-learning to obtain better initialization parameters, based on which only a small number of samples need to be fine-tuned for training to obtain better results. The method is relatively simple to implement, but the capacity of the model is limited because only the initialization parameters are learned. Meta-LSTM uses the LSTM network as the outer network to learn various optimization parameters (learning rate, decay rate, etc) of the inner network. This method has a large model capacity, but it is not practical due to the complex training process of LSTM and the slow convergence speed. The model capacity of Meta-SGD is improved relative to MAML, and the training difficulty of this algorithm is significantly reduced relative to Meta-LSTM. Current innovative optimization-based meta-learning approaches emphasize more on the dynamic changes of the environment or integrate with federated learning to ensure data privacy and security. However, there is still room for improvement in handling large-scale meta-learning problems and the generalization ability of meta-learner.

3.4. Summary

In general, meta-learning has received more attention in recent years. Whether metric-based, model-based, or optimization-based meta-learning, the ultimate goal is to learn new tasks faster and better by using previously learned experiences. Metric-based meta-learning is simple and effective, but is generally only applicable to supervised learning. Model-based meta-learning can update its internal state, but lacks generalization performance. Optimization-based meta-learning has broader applicability but is computationally intensive. According to our survey conducted on novel approaches, meta-learning, as a flexible learning framework, is more integrated with different learning paradigms and provides complementary benefits. Thus, how to improve the generalization performance while reducing the computational effort and model complexity is a common concern.

4. Applications of meta-learning

With the focus on meta-learning, more and more attention is being paid to the application of meta-learning in real-world scenarios. In this section, we elaborate on research advances in two application areas of meta-learning, respectively, the fraud detection and spatial-temporal prediction problems. Moreover, we investigate the application of meta-learning to natural language processing, image segmentation, and defect diagnosis to provide a more comprehensive review. With respect to the fraud detection problem, we present it in three stages, from the proposal of meta-learning in the field to the recent research progress. In the case of the spatial-temporal prediction problem, we briefly review the research process on the spatial-temporal prediction problem in addition to the application of meta-learning to this problem, since meta-learning techniques are still relatively new in this problem.

4.1. Fraud detection

With the development of credit derivatives, market volatility has increased and credit fraud has emerged. When a borrower commits fraud, creditors or banks are bound to bear financial losses because they do not receive the expected returns, which can have serious consequences not only for the long-term viability and stability of an organization, but even adversely affect the overall economic environment. Consequently, fraud detection is needed. The non-uniform distribution of data and the mixture of legitimate and fraudulent transactions are the two main reasons for the difficulty of fraud detection. In this section, we sort out the development process of meta-learning for fraud detection into three phases. The summary is shown in Table 2.

4.1.1. First stage

Chan and Stolfo were the first to apply meta-learning to the field of fraud detection. They proposed that meta-learning as a general technique to integrate several different learning processes could greatly increase the amount of data handled by knowledge discovery systems [153]. Following on from this, Stolfo et al. used meta-learning techniques to learn models of fraudulent credit card transactions. They concluded that fraud catching rate (True Positive Rate) and false alarm rate (False Positive Rate) were better metrics in the field of fraud detection to assess the overall accuracy of the learned fraud classifiers [154]. During this phase, two key behaviors of the meta-learning framework were also laid down. First, the meta-learning strategy must produce an accurate final classification system. Second, it must be fast, relative to an individual sequential learning algorithm, and operate in a reasonable amount of time when applied to massive amounts of data [156].

One of the reasons for the complexity of the fraud detection problem is the skewed class distribution. When given the data with skewed distribution, adjusting the training data artificially to be balanced will produce a better classifier. Chan et al. further investigated the effect of class distribution on performance, the different methods of measuring performance based on cost models, and the performance impact of training class distribution for different cost models [155]. Philip et al. designed a multi-classifier meta-learning method to solve large databases with skewed class distributions and non-uniform costs [157].

JAM was the first system to date to use meta-learning for mining distributed databases and had been used to address critical problems in financial information systems. This system required the analysis of large distributed databases (e.g. from different banks) using information about transaction behavior to generate ”probably fraudulent” transaction models [158–161].

Generally, in this phase, the application of meta-learning to the field of fraud detection is first proposed. The first system using meta-learning as a way to mine distributed databases is presented. And the problem of skewed class distribution in fraud detection starts to be noticed. However, this stage is still in the initial stage, mainly providing theoretical explanations with simple applications of meta-learning on fraud detection. Comprehensive and systematic knowledge of the application of meta-learning on fraud detection has not yet been recognized.

4.1.2. Second stage

This phase of studies in the field of fraud detection mainly employs traditional machine learning methods and meta-classifier models using ensemble learning.

Under the assumptions regarding the cost of misclassification and the different ratios of fraudulent to non-fraudulent firms, Perols et al. compared the performance of six popular statistical and machine learning models for detecting financial statement fraud [162]. The results showed that logistic regression and support vec-

Table 2
Application of meta-learning to default detection.

Stage	Theme	Reference	Summary
First Stage	Proposed meta-learning for fraud detection	[153–155] [156,157]	This phase first proposed the application of meta-learning to the field of fraud detection, and also proposed the first system(JAM) using meta-learning as a way to mine distributed databases, and started to investigate the problem of skewed distributions in fraud detection.
	JAM	[158,159] [160,161]	
Second Stage	Traditional machine learning	[162–164] [165–167]	In this phase, the application of meta-learning to fraud detection was gradually spread, mainly using traditional machine learning ensemble methods for processing. After the first phase focused on the problem of skewed class distribution, this phase was investigated in more detail from different strategies and methods. In addition, textual fraud detection was also performed.
	Skewed Class Distribution	[168–170] [171–173]	
	Text Fraud Detection Framework	[174–176] [177]	
Third Stage	Concept Drift	[178,179]	This phase started using deep learning methods for fraud detection. In addition to skewed class distribution, concept drift and real-time fraud detection problems were derived with the development of transactions such as credit cards. After that, fraud detection from a deep learning perspective will be the focus of research, with concerns about information security, model sharing, and data privacy.
	Real-time Detection	[180–182] [173,183]	
	Feature Extraction	[184,185]	
	Meta-learning ensemble technology	[186,187] [188,189]	

tor machines performed well relative to artificial neural networks, bagging, C4.5, and stacking. Cecchini et al. provided a methodology based on support vector machines using basic financial data to detect management fraud. A kernel specific to the financial domain was developed that built the features shown in previous studies to help detect management fraud [163]. [165] was the first to compare the performance of SVM and decision tree methods in credit card fraud detection using real datasets. The authors also applied artificial neural network (ANN) and logistic regression (LR) based classification models to the credit card fraud detection problem [190]. The network approach to risk management (NARM) treated banks as networks connected by financial relationships. It integrated networks and financial principles into a business intelligence (BI) algorithm that analyzed the systemic risk attributable to each bank through simulations based on real-world data from the Federal Deposit Insurance Corporation [164].

With credit card fraud detection systems, one of the problems is that a large percentage of transactions flagged as fraudulent are legitimate. This misinformation delays the detection of fraudulent transactions. Pun et al. proposed the use of a meta-classifier model, which consisted of three base classifiers, k-nearest neighbor, decision tree, and naive Bayes algorithm [166]. The final result was generated by combining the predictions of basic classifiers. There was also research proposed to combine five supervised machine learning algorithms, including Classification and Regression Trees (CART), Adaboost and Logitboost, Bagging and Dagging, for the classification of credit card data [167].

To solve the data mining problem with skewed data distribution, McCarthy et al. considered two basic strategies for handling data with skewed class distributions and non-uniform misclassification costs. One strategy was based on cost-sensitive learning, and the other strategy used sampling to create a more balanced class distribution in the training set [168]. Also in response to this question, Weiss et al. investigated three methods. The first method incorporated misclassification costs into the learning algorithm while the other two methods used oversampling or undersampling to make the training data more balanced [169]. In the DataBoost-IM method, boosting, an ensemble-based learning algorithm was combined with data generation to improve the predictive power of the classifier for an unbalanced dataset [171]. The Adaptive Synthesis (ADASYN) sampling method used a weighted distribution for different minority classes of examples according to their level of learning difficulty, where more synthetic data is generated for minority class examples that are harder to learn compared to those minority examples that are easier to learn [172]. In addition, there was a meta-clustering method with the K-means algorithm. As a preprocessing method, the main purpose of this method was to generate a cluster in a dataset of e-commerce transaction anomalies, with each cluster containing similar instances [173]. From the results of the experiments, it is difficult to say which method is the best. Based on this, Phua et al. proposed to replace a few oversampled data partitions using backpropagation (BP), naive Bayesian (NB), and C4.5 algorithms. Their main contribution was to use a single meta-classifier (stacking) to select the best base classifier and combined the prediction of these base classifiers (bagging) to improve cost savings [170].

MetaFraud, a meta-learning framework for detecting financial fraud classics [177], which had four main components (1) using organizational and industry contextual information, (2) using quarterly and annual data, and more powerful classification methods (3) overlay generalization, and (4) adaptive learning.

Fraud clues may be hidden not only for financial data but also in the text. Goel et al. used natural language processing tools to examine the verbal content and presentation style of qualitative sections of annual reports. They explored the linguistic features that distinguish fraudulent from nonfraudulent annual reports. The results indicated that the use of linguistic features was an effective means for detecting fraud [174]. Subsequent researches had considered qualitative and quantitative perspectives separately [175,176].

In this phase, the application of meta-learning on fraud detection is gradually spread, mainly using traditional machine learning and ensemble learning for processing, such as SVM, decision tree, logistic regression, and so on. After raising the problem of skewed class distribution in the first phase, this phase is studied more carefully from different strategies and methods, including sampling, boosting, adaptive, and so forth. Furthermore, since fraud is not only hidden in financial data, fraud detection on text in financial reports has also been studied. With the development of credit derivatives and deep learning techniques, the issues of concern have subsequently become more diverse and the techniques more sophisticated.

4.1.3. Third stage

Fraud detection faces huge challenges with the development of credit derivatives, such as huge transaction volume, fast transaction speed, imbalance of data, and frequent changes in fraud patterns, which have given rise to many new challenges. Two issues are currently focused on, concept drift and real-time fraud detection. Traditional fraud detection is mainly based on an ensemble approach with multiple machine learning models. With the development of technology, deep learning approaches are starting to be employed.

Concept drift indicates the phenomenon that the statistical properties of the target variable change in an unpredictable manner over time. The predictive accuracy of the model will decrease over time. When learning about the concept drift, the delay in obtaining accurate labels and the interaction between alerts and supervisory information must be carefully considered. Dal Pozzolo et al. showed that investigators' feedback and delayed labeling had to be handled separately. They designed two fraud-detection systems based on the ensemble method and the sliding window method. The strategy that proved successful consisted of training two separate classifiers (for feedback and delay labels, respectively) and aggregating the results [178]. However, the majority of learning algorithms proposed for fraud detection relied on assumptions that hardly held in real-world fraud detection systems. Therefore, a study proposed a formal scheme for the fraud detection problem that realistically described the operating conditions of a fraud detection system [191]. They also used the Hellinger weighted ensemble of HDDT to combat concept drift and improve the accuracy of individual classifiers [179].

Due to the increasing volume and speed of transactions, there is an increasing demand for the timeliness of fraud detection to minimize losses. Kavitha et al. proposed a real-time tree-based meta-classifier (TBMC), which operated based on two levels of prediction. The first level prediction was performed by a random forest classifier and the second level prediction was performed by an ensemble created by a decision tree and a gradient boosting tree. The results of the first and second level prediction models were combined to form the final prediction [180]. The Transaction Window Bagging (TWB) model used a paralleled bagging approach that combined an incremental learning model, a cost-sensitive base learner, and a weighted voting-based combiner to efficiently handle concept drift and data imbalance for real-time fraud detection [181]. At the same time, as online payments open up more and more new opportunities for e-commerce, the use of credit card transactions for online or offline purchases has increased exponentially. Due to this phenomenon, the number of online fraudulent activities is also increasing. Van Vlasselaer et al. proposed APATE, a novel approach for automated credit card transaction fraud detection using network-based extensions. The method combined features from incoming transactions and intrinsic features based on customer spending history, as well as network-based features which were derived by using the credit card holder and merchant networks to derive time-dependent suspiciousness scores for each network object [182]. Tripathi et al. proposed a credit and debit card fraud detection method using local outlier factors [183].

Currently, supervised learning techniques are widely used for credit card fraud detection as the assumption that fraud patterns can be learned from the analysis of past transactions. However, when considering changes in customer behavior and the ability of fraudsters to invent novel fraud patterns, the use of unsupervised learning techniques becomes necessary. Carcillo et al. proposed a hybrid technique that combined supervised and unsupervised techniques to improve the accuracy of fraud detection [186]. Olowookere et al. proposed a framework that combined the potentials of meta-learning ensemble techniques and a cost-sensitive learning paradigm for fraud detection. The framework was to allow base-classifiers to fit traditionally while the cost-sensitive learning is incorporated in the ensemble learning process to fit the cost-sensitive meta-classifier without having to enforce cost-sensitive learning on each of the base-classifiers [187].

As financial services and businesses grow, financial fraud is escalating. Fraud detection is also studied at this stage from the perspective of feature extraction and system upgrade [184,185,189]. It is worth noting that more and more attention is being paid to data security and privacy issues(Federated Meta-Learning). Due to the security and privacy of the data, different banks do not allow to share their transaction datasets. Based on these issues, Zheng et al. proposed a new framework called Federated Meta-Learning for fraud detection. Unlike traditional techniques trained with data centralized in the cloud, this model enabled banks to learn fraud detection models with the training data distributed on their own local databases. By aggregating locally computed fraud detection model updates, a shared whole model was constructed. Banks could collectively reap the benefits of a shared model without sharing datasets and protect the sensitive information of cardholders [188].

As the problem diversifies and escalates in difficulty, this stage is no longer only using traditional machine learning algorithms, but starting to use a combination of supervised and unsupervised learning, deep learning, and other methods. In addition to the skewed class distribution mentioned in the previous phase, there is also a focus on the real-time fraud detection and the concept drift. Also, there is an increasing concern about information security, model sharing, and data privacy. In the subsequent research, more fraud detection will be performed based on deep learning from the perspectives of data security and privacy.

4.1.4. Summary

In this section, we describe the application of meta-learning in the field of fraud detection. In terms of approach, we describe from the first proposal of meta-learning in the field of fraud detection to the use of traditional machine learning to recent deep learning approaches. And in terms of problems, we move from a theoretical exposition of the problem to a skewed class distribution to real-time fraud detection and concept drift problems. In the subsequent research, fraud detection from the perspective of deep learning methods will be the focus of research and gradually starts to focus on the issues of information security, model sharing, and data privacy.

4.2. Spatial-temporal prediction

Spatial-temporal prediction is a fundamental problem in building smart cities and is important for tasks such as traffic control, cab scheduling, and environmental policy making. In this problem, traffic flow, vehicle speed, vehicle demand, and travel time show strong dynamic correlations in both spatial and temporal dimensions. Therefore, how to exploit the spatial-temporal correlation for accurate traffic prediction is an essential issue. On the one hand, it is the complex spatial-temporal correlation of urban traffic,

Table 3
Application to spatial-temporal prediction.

Methods	Theme	Reference	Summary
Traditional Methods	Classical Statistical Models	[192,193] [194,195] [196,197] [198,199]	In classical statistical models, time series analysis methods are mainly used. But such methods are not suitable for dealing with complex and dynamic time series data. Machine learning methods are divided into three main categories, feature-based models, whose performance depends heavily on manually designed features; Gaussian process models, which have high computational and storage pressure; and state-space models, which are not suitable for modeling complex dynamic traffic data.
	Machine Learning Models	[200,201] [202][203] [204][205] [206,207] [208,209] [210,211]	
Deep Learning Methods	Space Dependence	[212,213] [214,215] [216,217] [218,219]	We briefly analyze the spatial dependence and temporal correlation from two perspectives respectively, but since only partial information is considered, more studies are now beginning to focus on the diversity and complexity of spatial-temporal correlations.
	Time Dependence	[220,221] [222,223] [224,225] [213,214]	
	Spatial-temporal Correlation	[226,227] [67,228] [229,230] [231,68] [232][233]	Meta-learning is mainly based on the perspective of spatial-temporal correlation, and currently shows good performance in parameter tuning and algorithm selection.

including spatial correlation between locations and temporal correlation between times; on the other hand, it is the diversity of spatial-temporal correlation, including spatial diversity and temporal diversity. At present, the spatial-temporal prediction problem is mainly divided into traditional methods and deep learning methods. The summary is shown in Table 3.

4.2.1. Traditional methods

Among the traditional methods, they can be further divided into classical statistical models and machine learning models [234].

4.2.1.1. Classical statistical models. In classical statistical models, time series analysis is the main method used for spatial-temporal forecasting problems. One of the most classic methods is the autoregressive integrated moving average model (ARIMA) and its variants [192]. These methods can find out the characteristics, trends, and development patterns of variable changes from time series to effectively predict the future changes of variables and have been widely used in spatial-temporal forecasting problems [193–199]. However, these methods are generally designed for small datasets and are not suitable for dealing with complex and dynamic time-series data. Moreover, these methods only consider the role of temporal factors in forecasting and do not take into account the influence of other external factors, such as the spatial dependence of traffic data.

4.2.2. Machine learning models

Machine learning methods are divided into three main categories: feature-based models, Gaussian process models, and state-space models. Feature-based models solve traffic prediction problems by training regression models based on artificially engineered traffic features [200–203]. Although these methods are easy to implement and can provide predictions in some practical situations, feature-based models have a key limitation: the performance of the model depends heavily on the artificially designed features. Gaussian processes model the intrinsic characteristics of traffic data through different kernel functions that require both spatial and temporal correlation. Although such methods are effective and feasible in traffic prediction [204–206], they use the complete sample/feature information for prediction, have high computational and storage pressure when a large number of training samples or features are available. State space models are dynamic time-domain models with implied time as the independent variable, assuming that the observations are generated by Markov hidden states. This type of method not only naturally models the uncertainty of the system and captures the underlying structure of the spatial-temporal data better, but also saves time and effort by describing the state of the system in the form of minimal information about the present and the past. However, the nonlinearity of the model is limited, and most often they are not optimal for modeling complex and dynamic traffic data [207–211].

4.2.3. Deep learning methods

Deep learning methods use more complex structures and utilize more features for training than traditional machine learning methods, which can achieve better performance. For the spatial-temporal prediction problem, firstly, we briefly analyze it from two perspectives: spatial dependence and temporal correlation. After that, we discuss methods based on spatial-temporal correlation. And the studies presented here are mainly based on meta-learning methods.

4.2.4. General deep learning methods

For the spatial dependence problem, CNNs have been applied to deal with it for a long time [212–214], but traditional CNNs are limited to modeling Euclidean data. GCNs are used to model non-Euclidean spatially structured data, which are more in line with the traffic road network structure [215–217]. Since the traffic condition on one road can be influenced differently by conditions on other roads, this impact is highly dynamic over time. Therefore, the attention mechanism is introduced to capture the correlation in the road network adaptively [218,219].

For temporal correlation, in addition to CNN [220], RNN is also used to model the nonlinear temporal correlation of traffic data, relying mainly on the order of the data to process [221,222]. The length of the semantic vector between encoding and decoding is always fixed regardless of the length of the input and output sequences, so some information will be lost when the input information is too long. The attention mechanism can adaptively select the relevant hidden states of the encoder to generate the output sequence, simulating the nonlinear correlation between different time slices [223,224]. With the development of MLP, CNN, and RNN, DNN is gradually employed in spatial-temporal prediction [225,213,214].

Currently, many advanced methods have been designed with spatial-only (graph neural networks, etc.) and temporal-only (recurrent neural networks, etc.) modules to extract spatial and temporal features, respectively. However, it is less effective to extract complex spatial-temporal relationships with such decomposed modules. These methods only consider partial information and do not take into account spatial-temporal correlations. Therefore, more and more studies are now focusing on the diversity and complexity of spatial-temporal correlations [226,227].

4.2.5. Meta learning

Meta-learning has been gradually developed for spatial-temporal prediction problems in the past two years, which no longer extracts spatial and temporal features separately but focuses more on spatial-temporal correlation, including ST-MetaNet [67], ST-MetaNet+ [228] and Meta-MSNet [229], etc.

ST-MetaNet was built using a sequence-to-sequence architecture, consisting of an encoder that learned historical information and a decoder that made predictions step-by-step. The encoder and decoder had the same network structure, including a meta graph attention network and a meta recurrent neural network to capture different spatial and temporal correlations, respectively [67]. On this basis, it was further extended to ST-MetaNet+ [228]. The main difference between them was that ST-MetaNet did not model the relationships between ST correlations and dynamic traffic states within Meta-GATs and Meta-GRUs. Meta-MSNet was also designed with an encoder-decoder architecture. The encoder captured the temporal dependence of adjacency, while the decoder extracted periodic features, taking advantage of the different functions of short-term adjacency and long-term periodic temporal patterns. Furthermore, two meta-learning-based fusion modules were designed to integrate multiple sources of external data in both temporal and spatial dimensions [229].

Due to the spatial-temporal dependence and traffic uncertainty, Zhang et al. considered a novel continuous spatial-temporal meta-learner (cST-ML) from the perspective of Bayesian meta-learning [230]. cST-ML was trained based on the distribution of traffic prediction tasks segmented by historical traffic data to learn a strategy that can quickly adapt to relevant but invisible traffic prediction tasks. In AutoSTG, spatial graph convolution and temporal convolution operations were employed in the search space to capture the complex spatial-temporal correlations. And meta-learning techniques were used to learn the adjacency matrix of the spatial graph convolution layer and the kernel of the temporal convolution layer from the meta-knowledge of the attribute graph [231]. The model proposed by Yao et al. was designed as a spatial-temporal network with a meta-learning paradigm that used long-term data from other cities through transfer learning to deal with the spatial-temporal prediction problem for cities with a short period of data [68].

Due to the large amount of traffic data generated every day, training with deep learning models needs to face time-consuming as well as tuning problems. Using meta-learning to train hyperparameter tuning for high traffic datasets can improve the automatic learning process and reduce time-consuming tasks [232]. And meta-learning has already achieved impressive performance improvements in algorithm selection [233].

Deep meta-learning models, especially those that consider spatial-temporal dependencies, show better prediction accuracy compared to traditional machine learning models. And the accuracy increases with the increase of data but decreases with the increase of time horizon [235].

4.2.6. Summary

In this section, we briefly analyze the spatial-temporal prediction problem from traditional and deep learning methods, respectively, and introduce the application of meta-learning to the problem. At present, meta-learning mainly addresses the spatial-temporal prediction problem from the perspective of spatial-temporal correlation, including the use of sequence-to-sequence architectures, encoder-decoder architectures, and the combination with GCN. It also shows good performance in parameter tuning and algorithm selection.

4.3. Other applications

Other applications of meta-learning are outlined in this section, mainly natural language processing, image segmentation, and fault diagnosis.

4.3.1. Natural language processing

Natural language processing is a significant area of study within the field of artificial intelligence, which is concerned with the processing and analysis of natural language using computer technology. Meta-learning has been extensively applied to natural language processing problems, especially for few-shot text classification [236]. The hierarchical attention prototype network(HAPN) considered the importance of features, words, and instances from the perspective of semantic space [237]. The Self-Attentive Relational Network (SARN) augmented the features learned by the model with an attention module after the features were extracted [238]. Text classification does not achieve satisfactory results when the amount of data is scarce or new classes need to be adapted [239,240]. Geng et al. proposed induction networks for better generalization by leveraging dynamic routing algorithms in meta-learning to implement a more generalized representation of the data in the support set [241]. Afterwards, they evolved a dynamic routing mechanism on static memory to adapt it to new classes better [242]. It was also possible to capture language features and understand downstream language tasks better by pre-training the language representation [243].

In parallel, the textual diversity restricts the application of meta-learning as texts can be expressed in different ways even for the same class [244]. Existing research augmented the data by generating more samples [245], and also enhanced the prototype network with relational and entity descriptions [246]. Semantic relevance is also the focus and challenge of current research. The rich semantic information under class labels has been neglected in many investigations. It had been indicated that distinguishing features could be extracted by leveraging the class label information using pre-trained models [247]. Simultaneously, to avoid the confusion of semantically relevant labels, a unique generation method could be employed so that each label was represented in its specific way [248].

4.3.2. Image segmentation

Image segmentation is a vital research element in deep learning, and abundant studies have been conducted [249]. With many computer vision tasks requiring intelligent segmentation of images, learning each pixel in an image using deep learning methods facilitates better understanding of the content in the image [250]. From the application point of view, the following image segmentation problems are investigated: significant amounts of labeled data are required to boost the effectiveness of model with attendant cost of labeling and privacy issues involved; another issue is the need for models to learn quickly and improve generalization capabilities as data dynamics change and unseen datasets emerge; and lastly, the choice of the optimal learning algorithm for different datasets.

To cope with the difficulty of needing a large number of labeled training samples, semantic meta-learning (SML) incorporated class-level semantic information into the model [251]. Meta-DRN was a fancy lightweight CNN architecture to address image segmentation [252]. MetaSegNet developed an embedding module architecture consisting of global and local feature branches to solve the sample annotation scarcity problem by extracting meta-knowledge [253]. Rakelly et al. proposed guided networks to extract potential task representations from the training dataset and optimize the architecture end-to-end towards fast and accurate image segmentation [254]. Regarding the high cost of annotation and privacy issues that come with large datasets and the diversity of data distribution, Zhang et al. proposed an enhanced optimization-based meta-learning algorithm with the ability to learn from different segmentation tasks [255]. Nevertheless, it is often ineffective to use the trained model directly on new unseen datasets. For better learning of unseen target domains, optimization-based meta-learning methods as well as gradient-based meta-learning methods have been investigated [256,257]. Other researchers are now beginning to investigate how to make algorithm choices for different datasets [258,259].

4.3.3. Fault diagnosis

Fault diagnosis refers to determining whether a fault occurs in a system, when, where, what kind, and the extent of the occurrence. Deep learning techniques are now broadly used in the field of fault diagnosis [260,261]. The exact experimental results rely on a massive database, but it is extremely difficult to collect enough data for different fault types in real-world scenarios, and the process is time-consuming and unsafe. Moreover, training on small datasets is prone to overfitting and the trained models are not robust [262]. Many scholars have provided solutions to this problem [263]. Based on MAML, Dixit et al. suggested a new conditional auxiliary classifier GAN framework that used GAN to generate new samples [264]. Similarly based on MAML, Zhang et al. presented a few-shot learning framework for fault diagnosis, which not only allowed effective training of limited samples but also learned to identify new task scenarios [100]. Besides, the feature space metric-based meta-learning model (FSM3) combined supervised learning with metric-based meta-learning, utilizing information from both individual samples and sample groups to learn [265].

Apart from the poor number of samples, the problem of fault diagnosis suffers from variations in data distribution [266], changes in working conditions [148,267,268], and data imbalance [269]. [266] suggested training on datasets with different distributions and improving model performance by means of parameter transfer. The novel meta-learning fault diagnosis method (MLFD) based on model-agnostic meta-learning converted signals under different operating conditions into time-frequency images, and then randomly sampled them to form the MLFD task [148]. Further, the identification of faults from the perspective of multi-label attributes was also available in the literature [270].

5. Discussion

Deep learning approaches demand a large amount of data to train efficiently and are difficult to generalize to new tasks. Clearly, this is contrary to the natural purpose of learning from a small number of samples based on previous information and adapting fast to new tasks. Meta-learning, as a learning paradigm, addresses this weakness by utilizing prior knowledge to guide the learning of new tasks, with the goal of rapidly learning. In this section, we summarize the findings of this research and discuss the existing challenges as well as future research directions.

5.1. Conclusion

This paper provides an overview of the development and applications of meta-learning. We divide meta-learning methods into three classes, metric-based, model-based, and optimization-based, which rely on computing input similarity, task embedding with state, and task-specific updates, respectively. The success of deep learning comes from the ability to learn feature representations of data, and metric-based meta-learning learns the common feature representations behind different tasks, with the main focus on feature extractors, similarity measures, and automatic algorithm selection. Such techniques are simple and effective, but are sensitive to the dataset and increase the computational expenditure when the number of tasks is large. Further, we can directly learn the ability to build models for different tasks. This requires a model with a generic model compilation capability. Among others, the framework of neural Turing machines provides a generic model-building capability. The MANN-based learning framework is obtained by deriving from the neural Turing machine. In MANN, updating slower weights enables the creation of models, while reading and writing of the memory module allows for building specific models directly. This is similar to a neural network with memory added, having distinct functions corresponding to different memory situations. This technique is extremely adaptable internally but lacks significant generalization possibilities. Another class of optimization-based meta-learning methods is to learn how to learn new tasks or learn strategies to accelerate the acquisition of new tasks. This approach is more broad and applicable, although it is computationally costly and still requires improvement in model capacity and convergence speed. Furthermore, we observed that meta-learning is increasingly being combined with reinforcement learning, online learning, and federated learning over recent years, and that solutions continue to be created as challenges vary to increase model learning performance.

In terms of application scenarios, we investigate the use of meta-learning in the field of fraud detection, showing the use of traditional machine learning to deep learning methods. In terms of issues, we discuss the problem of skewed class distributions, real-time fraud detection, and concept drift. In subsequent research, fraud detection from the perspective of deep learning methods will be the focus of research. Existing research is gradually shifting its emphasis to topics like data security, model sharing, and data privacy. For the spatial-temporal prediction problem, we briefly analyze the traditional and deep learning methods, and describe the application of meta-learning to the problem. Currently, meta-learning approaches are mainly based on the perspective of spatial-temporal correlation, including the use of sequence-sequence architectures, encoder-decoder architectures, and the combination with GCN. And it also demonstrates superior performance when it comes to parameter tuning and algorithm selection. Along with the two scenarios mentioned above, we give an overview of meta-learning in three other circumstances, including natural language processing, image segmentation, and fault diagnosis, to provide a more comprehensive application scenario. Our study has demonstrated that meta-learning has considerable promise for both academic research and industry concerns for application in real-world scenarios.

Inevitably, our study has some limitations. To begin, this study details several classical approaches under three classifications. There are still other classical approaches with the considerable impact that are not included in this review and deserve further exploration. Second, this paper indicates that meta-learning is more commonly integrated with other learning paradigms when investigating creative research presented under different classification algorithms. And what we provide is a relatively well-researched and pragmatic method, with more intricate studies that need to be analyzed further. Finally, five widely applied domains are chosen for the use of meta-learning in real-world settings in this work. It is evident that meta-learning, as a flexible learning framework, has a much broader range of application situations than this, and readers are encouraged to pursue their areas of interest further.

5.2. Future research

While meta-learning has enormous potential and a wide range of application possibilities, there are still certain challenges, which we explore here along with some future research directions.

The primary difficulty encountered throughout the meta-learning process is the meta-generalization problem [271]. The majority of meta-learning techniques implicitly require that meta-training tasks be mutually exclusive, so that no single model can solve them all concurrently. Meta-learning prevents meta-overfitting through the use of techniques such as intelligent design and meta-regularization [112]. As meta-learning adapts to a new task, its capacity to adapt to a new task decreases with distance from the training task [98]. It's worth researching how meta-learning can boost cross-domain generalization. Another serious factor that meta-learning needs to handle is computational efficiency and memory capacity. For instance, optimization-based meta-learning techniques necessitate both internal and external optimization. While rapid adaptation to novel tasks is possible, each optimization step of the model takes multiple steps, which require significant computational resources and internal memory space. As a result, increasing computing efficiency and reducing the cost of storage continue to be significant challenges. Finally, concerns such as multimodal of tasks and class imbalance of data can affect the model's performance, and various situations demand distinct solutions, which are currently challenging to achieve.

Meta-learning has been inextricably combined with other learning paradigms in recent years, which will be a critical area of future research. Deep reinforcement learning requires a vast amount of training data to be available. Meta-reinforcement learning alleviates this disadvantage and enables rapid adaptation to new tasks. Meta-learning is combined with online learning to accelerate online adaptability. Meta-learning is integrated with federated learning to meet users' needs for data privacy and security. Meta-learning, with its adaptable learning framework, combines with different learning paradigms to complement one another's strengths and enables the development of distinct models for distinct situations to better satisfy the requirements of real-world problems.

Another area of future research is to facilitate meta-learning to automatically learn to learn. Whether or not it is successful, meta-learning will accumulate relevant information each time it attempts to execute a task. This information can be used to process new tasks without having to train from scratch each time. We are exposed to more tasks over time, and the more knowledge we accumulate. When confronted with a new task, the more similar we compare the information with the past, the more likely we are to do it well. Meta-learning's generalization ability depends on the computer's storage capacity. Once given a task and data, AutoML permits models to learn automatically. And its learning effectiveness and generalization capacity are quite effective [272]. Meta-learning is a concept that is somewhat similar to this, and hence we believe that making models learn without human interaction will be the focus of future research efforts.

Ultimately, contemporary research highlights not only the task itself, but also the underlying causal relationship and abstract concept representation. In comparison to earlier, the level of problem solving thinking increases. Causal inference originates in statistics, with the objective of establishing the causal relationship between individual variables based on observed statistical data, and its central concept is counterfactual inference [273]. As a causal inference approach, the meta-learning method first learns the benchmark's strategy outcome estimator and then does strategy effect estimation. By developing an understanding of the causal linkages that exist between things, we can improve our chances of achieving the desired model effect. Moreover, some analysis of meta-abstraction level representation is included.

In short, meta-learning is a promising field of research in deep learning. It compensates for the limits of deep learning that are hindered by the insufficient amount of data through its flexible structure. As opposed to training a model on a specific task, meta-learning empowers the model to learn to learn.

CRediT authorship contribution statement

Yingjie Tian: Conceptualization, Methodology, Writing – review & editing, Supervision, Project administration, Funding acquisition. **Xiaoxi Zhao:** Methodology, Data curation, Formal analysis, Writing – original draft. **Wei Huang:** Supervision, Project administration, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work has been partially supported by grants from: National Natural Science Foundation of China (No. 12071458, 71731009).

References

[1] A. Voulodimos, N. Doulamis, A. Doulamis, E. Protopapadakis, Deep learning for computer vision: A brief review, Computational intelligence and neuroscience 2018 (2018).

[2] N. Akhtar, A. Mian, Threat of adversarial attacks on deep learning in computer vision: A survey, IEEE Access 6 (2018) 14410–14430.

[3] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, arXiv preprint arXiv:1409.0473 (2014).

[4] A. Vaswani, S. Bengio, E. Brevdo, F. Chollet, A.N. Gomez, S. Gouws, L. Jones, Ł. Kaiser, N. Kalchbrenner, N. Parmar, et al., Tensor2tensor for neural machine translation, arXiv preprint arXiv:1803.07416 (2018).

[5] Y. Wu, M. Schuster, Z. Chen, Q.V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, et al., Google's neural machine translation system: Bridging the gap between human and machine translation, arXiv preprint arXiv:1609.08144 (2016).

[6] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen, et al., Deep speech 2: End-to-end speech recognition in english and mandarin, in: International conference on machine learning PMLR, 2016, pp. 173–182.

[7] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, et al., Deep speech: Scaling up end-to-end speech recognition, arXiv preprint arXiv:1412.5567 (2014).

[8] B. Huval, T. Wang, S. Tandon, J. Kiske, W. Song, J. Pazhayampallil, M. Andriluka, P. Rajpurkar, T. Migimatsu, R. Cheng-Yue, et al., An empirical evaluation of deep learning on highway driving, arXiv preprint arXiv:1504.01716 (2015).

[9] A.E. Sallab, M. Abdou, E. Perot, S. Yogamani, Deep reinforcement learning framework for autonomous driving, Electronic Imaging 2017 (19) (2017) 70–76.

[10] D.K. Naik, R.J. Mammone, Meta-neural networks that learn by learning, in: [Proceedings 1992] IJCNN International Joint Conference on Neural Networks, vol. 1, IEEE, 1992, pp. 437–442.

[11] S. Thrun, L. Pratt, Learning to learn, Springer Science & Business Media, 2012.

[12] G.-J. Qi, J. Luo, Small data challenges in big data era: A survey of recent progress on unsupervised and semi-supervised methods, IEEE Trans. Pattern Anal. Mach. Intell. (2020).

[13] S. Zhao, Z. Liu, J. Lin, J.-Y. Zhu, S. Han, Differentiable augmentation for data-efficient gan training, Advances in Neural Information Processing Systems 33 (2020) 7559–7570.

[14] S. Corkery, H. Babinsky, W. Graham, Quantification of added-mass effects using particle image velocimetry data for a translating and rotating flat plate, J. Fluid Mech. 870 (2019) 492–518.

[15] H. Cai, H. Chen, Y. Song, C. Zhang, X. Zhao, D. Yin, Data manipulation: Towards effective instance learning for neural dialogue generation via learning to augment and reweight, arXiv preprint arXiv:2004.02594 (2020).

[16] J. Cong, S. Yang, L. Xie, G. Yu, G. Wan, Data efficient voice cloning from noisy samples with domain adversarial training, arXiv preprint arXiv:2008.04265 (2020).

[17] G.E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R.R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors, arXiv preprint arXiv:1207.0580 (2012).

[18] C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals, Understanding deep learning (still) requires rethinking generalization, Commun. ACM 64 (3) (2021) 107–115.

[19] B. Petrovska, T. Atanasova-Pacemska, R. Corizzo, P. Mignone, P. Lameski, E. Zdravevski, Aerial scene classification through fine-tuning with adaptive learning rates and label smoothing, Appl. Sci. 10 (17) (2020) 5792.

[20] Y. Bengio, Practical recommendations for gradient-based training of deep architectures, in: Neural networks: Tricks of the trade, Springer, 2012, pp. 437–478.

[21] H. Zhang, L. Zhang, Y. Jiang, Overfitting and underfitting analysis for deep learning based end-to-end communication systems, in: 2019 11th International Conference on Wireless Communications and Signal Processing (WCSP), IEEE, 2019, pp. 1–6.

[22] J.B. Biggs, The role of metalearning in study processes, Brit. J. Educ. Psychol. 55 (3) (1985) 185–212.

[23] J. Vanschoren, Meta-learning: A survey, arXiv preprint arXiv:1810.03548 (2018).

[24] M. Huisman, J.N. van Rijn, A. Plaat, A survey of deep meta-learning, Artif. Intell. Rev. (2021) 1–59.

[25] T. Hospedales, A. Antoniou, P. Micaelli, A. Storkey, Meta-learning in neural networks: A survey, arXiv preprint arXiv:2004.05439 (2020).

[26] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (10) (2009) 1345–1359.

[27] K. Weiss, T.M. Khoshgoftaar, D. Wang, A survey of transfer learning, J. Big data 3 (1) (2016) 1–40.

[28] S. Ruder, An overview of multi-task learning in deep neural networks, arXiv preprint arXiv:1706.05098 (2017).

[29] Y. Zhang, Q. Yang, A survey on multi-task learning, arXiv preprint arXiv:1707.08114 (2017).

[30] T. Yu, D. Quillen, Z. He, R. Julian, K. Hausman, C. Finn, S. Levine, Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning, in: Conference on Robot Learning, PMLR, 2020, pp. 1094–1100.

[31] N. Tripuraneni, C. Jin, M. Jordan, Provable meta-learning of linear representations, International Conference on Machine Learning, PMLR (2021) 10434–10443.

[32] G. Koch, R. Zemel, R. Salakhutdinov, Siamese neural networks for one-shot image recognition, in: ICML deep learning workshop, vol. 2, Lille, 2015.

[33] L. Bertinetto, J. Valmadre, J.F. Henriques, A. Vedaldi, P.H. Torr, Fully-convolutional siamese networks for object tracking, in: European conference on computer vision, Springer, 2016, pp. 850–865.

[34] M. Shorfuzzaman, M.S. Hossain, Metacovid: A siamese neural network framework with contrastive loss for n-shot diagnosis of covid-19 patients, Pattern Recogn. 113 (2021) 107700.

[35] J. Beel, B. Tyrell, E. Bergman, A. Collins, S. Nagoor, Siamese meta-learning and algorithm selection with'algorithm-performance personas'[proposal], arXiv preprint arXiv:2006.12328 (2020).

[36] J. Wang, Z. Fang, N. Lang, H. Yuan, M.-Y. Su, P. Baldi, A multi-resolution approach for spinal metastasis detection using deep siamese neural networks, Comput. Biol. Med. 84 (2017) 137–146.

[37] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, D. Wierstra, Matching networks for one shot learning, arXiv preprint arXiv:1606.04080 (2016).

[38] H. Yang, X. He, F. Porikli, One-shot action localization by learning sequence matching network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 1450–1459.

[39] J. Choi, J. Kwon, K.M. Lee, Deep meta learning for real-time target-aware visual tracking, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 911–920.
[40] S. Zhang, H. Zhao, Y. Wu, Z. Zhang, X. Zhou, X. Zhou, Dcmn+: Dual co-matching network for multi-choice reading comprehension, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 9563–9570.
[41] T.N.T. Nguyen, D.L. Jones, W.-S. Gan, A sequence matching network for polyphonic sound event localization and detection, in: ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020, pp. 71–75.
[42] J. Snell, K. Swersky, R.S. Zemel, Prototypical networks for few-shot learning, arXiv preprint arXiv:1703.05175 (2017).
[43] R. Boney, A. Ilin, Semi-supervised few-shot learning with prototypical networks, CoRR abs/1711.10856 (2017).
[44] T. Gao, X. Han, Z. Liu, M. Sun, Hybrid attention-based prototypical networks for noisy few-shot relation classification, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 6407–6414.
[45] M. Xiao, A. Kortylewski, R. Wu, S. Qiao, W. Shen, A. Yuille, Tdapnet: Prototype network with recurrent top-down attention for robust object classification under partial occlusion, arXiv preprint arXiv:1909.03879 (2019).
[46] W. Xu, Y. Xian, J. Wang, B. Schiele, Z. Akata, Attribute prototype network for zero-shot learning, arXiv preprint arXiv:2008.08290 (2020).
[47] F. Sung, Y. Yang, L. Zhang, T. Xiang, P.H. Torr, T.M. Hospedales, Learning to compare: Relation network for few-shot learning, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 1199–1208.
[48] H. Xu, C. Jiang, X. Liang, Z. Li, Spatial-aware graph relation network for large-scale object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 9298–9307.
[49] J. He, R. Hong, X. Liu, M. Xu, Z.-J. Zha, M. Wang, Memory-augmented relation network for few-shot learning, in: Proceedings of the 28th ACM International Conference on Multimedia, 2020, pp. 1236–1244.
[50] H. Hu, Z. Zhang, Z. Xie, S. Lin, Local relation networks for image recognition, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 3464–3473.
[51] C. Sun, A. Shrivastava, C. Vondrick, K. Murphy, R. Sukthankar, C. Schmid, Actor-centric relation network, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 318–334.
[52] H. Chen, Z. Lin, G. Ding, J. Lou, Y. Zhang, B. Karlsson, Grn: Gated relation network to enhance convolutional neural network for named entity recognition, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 6236–6243.
[53] M. Bishay, G. Zoumpourlis, I. Patras, Tarn: Temporal attentive relation network for few-shot and zero-shot action recognition, arXiv preprint arXiv:1907.09021 (2019).
[54] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, T. Lillicrap, Meta-learning with memory-augmented neural networks, in: International conference on machine learning PMLR, 2016, pp. 1842–1850.
[55] C. Gulcehre, S. Chandar, Y. Bengio, Memory augmented neural networks with wormhole connections, arXiv preprint arXiv:1701.08718 (2017).
[56] T. Munkhdalai, H. Yu, Neural semantic encoders, in: Proceedings of the conference. Association for Computational Linguistics. Meeting, vol. 1, NIH Public Access, 2017, p. 397.
[57] S. Pramanik, A. Hussain, Text normalization using memory augmented neural networks, Speech Commun. 109 (2019) 15–23.
[58] T. Vu, B. Hu, T. Munkhdalai, H. Yu, Sentence simplification with memory-augmented neural networks, arXiv preprint arXiv:1804.07445 (2018).
[59] T. Munkhdalai, H. Yu, Meta networks, International Conference on Machine Learning, PMLR (2017) 2554–2563.
[60] F. Shen, S. Yan, G. Zeng, Neural style transfer via meta networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8061–8069.
[61] J. Pan, H. Sun, Y. Kong, Fast human motion transfer based on a meta network, Inf. Sci. 547 (2021) 367–383.
[62] Z. Chen, Y. Fu, Y.-X. Wang, L. Ma, W. Liu, M. Hebert, Image deformation meta-networks for one-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 8680–8689.
[63] S. Tsutsui, Y. Fu, D. Crandall, Meta-reinforced synthetic data for one-shot fine-grained visual recognition, arXiv preprint arXiv:1911.07164 (2019).
[64] T. Cao, K. Han, X. Wang, L. Ma, Y. Fu, Y.-G. Jiang, X. Xue, Feature deformation meta-networks in image captioning of novel objects, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 10494–10501.
[65] N. Li, Z. Chen, S. Liu, Meta learning for image captioning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 8626–8633.
[66] N. Mishra, M. Rohaninejad, X. Chen, P. Abbeel, A simple neural attentive meta-learner, arXiv preprint arXiv:1707.03141 (2017).
[67] Z. Pan, Y. Liang, W. Wang, Y. Yu, Y. Zheng, J. Zhang, Urban traffic prediction from spatio-temporal data using deep meta learning, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, pp. 1720–1730.
[68] H. Yao, Y. Liu, Y. Wei, X. Tang, Z. Li, Learning from multiple cities: A meta-learning approach for spatial-temporal prediction, The World Wide Web Conference (2019) 2181–2191.
[69] X. Jiang, M. Havaei, G. Chartrand, H. Chouaib, T. Vincent, A. Jesson, N. Chapados, S. Matwin, Attentive task-agnostic meta-learning for few-shot text classification (2018).
[70] Y. Guo, N.-M. Cheung, Attentive weights generation for few shot learning via information maximization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 13499–13508.
[71] L. Zhang, F. Zhou, W. Wei, Y. Zhang, Meta-generating deep attentive metric for few-shot classification, arXiv preprint arXiv:2012.01641 (2020).
[72] C. Finn, P. Abbeel, S. Levine, Model-agnostic meta-learning for fast adaptation of deep networks, in: International Conference on Machine Learning PMLR, 2017, pp. 1126–1135.
[73] H.S. Behl, A.G. Baydin, P.H. Torr, Alpha maml: Adaptive model-agnostic meta-learning, arXiv preprint arXiv:1905.07435 (2019).
[74] S. Arnold, S. Iqbal, F. Sha, When maml can adapt fast and how to assist when it cannot, in: International Conference on Artificial Intelligence and Statistics PMLR, 2021, pp. 244–252.
[75] A. Nichol, J. Schulman, Reptile: a scalable metalearning algorithm, arXiv preprint arXiv:1803.02999 2 (2) (2018) 1.
[76] Y. Zheng, J. Xiang, K. Su, E. Shlizerman, Bi-maml: Balanced incremental approach for meta learning, arXiv preprint arXiv:2006.07412 (2020).
[77] M.A. Jamal, G.-J. Qi, Task agnostic meta-learning for few-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 11719–11727.
[78] J. Rajasegaran, S. Khan, M. Hayat, F.S. Khan, M. Shah, itaml: An incremental task-agnostic meta-learning approach, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 13588–13597.
[79] C. Finn, A. Rajeswaran, S. Kakade, S. Levine, Online meta-learning, International Conference on Machine Learning, PMLR (2019) 1920–1930.
[80] G. Gupta, K. Yadav, L. Paull, La-maml: Look-ahead meta learning for continual learning, arXiv preprint arXiv:2007.13904 (2020).
[81] S. Ravi, H. Larochelle, Optimization as a model for few-shot learning (2016).
[82] B. Bohnet, R. McDonald, G. Simoes, D. Andor, E. Pitler, J. Maynez, Morphosyntactic tagging with a meta-bilstm model over context sensitive token encodings, arXiv preprint arXiv:1805.08237 (2018).
[83] K. Lim, J.Y. Lee, J. Carbonell, T. Poibeau, Semi-supervised learning on meta structure: Multi-task tagging and parsing in low-resource scenarios, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 8344–8351.
[84] J. Chen, X. Qiu, P. Liu, X. Huang, Meta multi-task learning for sequence modeling, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 32, 2018.
[85] C. Yu, X. Qi, H. Ma, X. He, C. Wang, Y. Zhao, Llr: Learning learning rates by lstm for training neural networks, Neurocomputing 394 (2020) 41–50.
[86] X. Zhen, H. Sun, Y. Du, J. Xu, Y. Yin, L. Shao, C. Snoek, Learning to learn kernels with variational random features, in: International Conference on Machine Learning PMLR, 2020, pp. 11409–11419.
[87] Z. Li, F. Zhou, F. Chen, H. Li, Meta-sgd: Learning to learn quickly for few-shot learning, arXiv preprint arXiv:1707.09835 (2017).
[88] A.A. Rusu, D. Rao, J. Sygnowski, O. Vinyals, R. Pascanu, S. Osindero, R. Hadsell, Meta-learning with latent embedding optimization, arXiv preprint arXiv:1807.05960 (2018).
[89] C. Simon, P. Koniusz, R. Nock, M. Harandi, On modulating the gradient for meta-learning, European Conference on Computer Vision, Springer (2020) 556–572.
[90] E. Park, J.B. Oliva, Meta-curvature, arXiv preprint arXiv:1902.03356 (2019).
[91] I. Nicholas, H. Kuo, M. Harandi, N. Fourrier, C. Walder, G. Ferraro, H. Suominen, M2sgd: Learning to learn important weights, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE Computer Society, 2020, pp. 957–964.
[92] A. Bartler, A. Bühler, F. Wiewel, M. Döbler, B. Yang, Mt3: Meta test-time training for self-supervised test-time adaption, arXiv preprint arXiv:2103.16201 (2021).
[93] I. Kulikovskikh, S. Prokhorov, T. Legović, T. Šmuc, An sgd-based meta-learner with 'growing' descent, in: Journal of Physics: Conference Series, Vol. 1368, IOP Publishing, 2019, p. 052008.
[94] V. Garcia, J. Bruna, Few-shot learning with graph neural networks, arXiv preprint arXiv:1711.04043 (2017).
[95] J. Kim, T. Kim, S. Kim, C.D. Yoo, Edge-labeling graph neural network for few-shot learning, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 11–20.
[96] P. Shyam, S. Gupta, A. Dukkipati, Attentive recurrent comparators, in: International Conference on Machine Learning, PMLR, 2017, pp. 3173–3181.
[97] Y.-X. Wang, R. Girshick, M. Hebert, B. Hariharan, Low-shot learning from imaginary data, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 7278–7286.
[98] W.-Y. Chen, Y.-C. Liu, Z. Kira, Y.-C.F. Wang, J.-B. Huang, A closer look at few-shot classification, arXiv preprint arXiv:1904.04232 (2019).
[99] S. Wang, D. Wang, D. Kong, J. Wang, W. Li, S. Zhou, Few-shot rolling bearing fault diagnosis with metric-based meta learning, Sensors 20 (22) (2020) 6437.
[100] S. Zhang, F. Ye, B. Wang, T.G. Habetler, Few-shot bearing fault diagnosis based on model-agnostic meta-learning, IEEE Trans. Ind. Appl. 57 (5) (2021) 4754–4764.

[101] Y. Chen, C. Guan, Z. Wei, X. Wang, W. Zhu, Metadelta: A meta-learning system for few-shot image classification, in: AAAI Workshop on Meta-Learning and MetaDL Challenge, PMLR, 2021, pp. 17–28.
[102] N. Guo, K. Di, H. Liu, Y. Wang, J. Qiao, A metric-based meta-learning approach combined attention mechanism and ensemble learning for few-shot learning, Displays 70 (2021) 102065.
[103] Q. Liu, O. Majumder, A. Ravichandran, R. Bhotika, S. Soatto, Incremental learning for metric-based meta-learners (2020).
[104] Q. Suo, J. Chou, W. Zhong, A. Zhang, Tadanet: Task-adaptive network for graph-enriched meta-learning, in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020, pp. 1789–1799.
[105] C. Liu, Z. Wang, D. Sahoo, Y. Fang, K. Zhang, S.C. Hoi, Adaptive task sampling for meta-learning, European Conference on Computer Vision, Springer (2020) 752–769.
[106] Q. Wang, X. Liu, W. Liu, A.-A. Liu, W. Liu, T. Mei, Metasearch: Incremental product search via deep meta-learning, IEEE Trans. Image Process. 29 (2020) 7549–7564.
[107] J. Guan, J. Liu, J. Sun, P. Feng, T. Shuai, W. Wang, Meta metric learning for highly imbalanced aerial scene classification, in: ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) IEEE, 2020, pp. 4047–4051.
[108] J. Chen, L.-M. Zhan, X.-M. Wu, F.-L. Chung, Variational metric scaling for metric-based meta-learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 3478–3485.
[109] H. Tang, Z. Li, Z. Peng, J. Tang, Blockmix: meta regularization and self-calibrated inference for metric-based meta-learning, in: Proceedings of the 28th ACM international conference on multimedia, 2020, pp. 610–618.
[110] Y. Wang, X.-M. Wu, Q. Li, J. Gu, W. Xiang, L. Zhang, V.O. Li, Large margin meta-learning for few-shot classification, Neural Information Processing Systems Foundation, 2018, Workshop on Meta-Learning (MetaLearn 2018)@ NeurIPS 2018.
[111] M. Garnelo, D. Rosenbaum, C. Maddison, T. Ramalho, D. Saxton, M. Shanahan, Y.W. Teh, D. Rezende, S.A. Eslami, Conditional neural processes, in: International Conference on Machine Learning PMLR, 2018, pp. 1704–1713.
[112] M. Yin, G. Tucker, M. Zhou, S. Levine, C. Finn, Meta-learning without memorization, arXiv preprint arXiv:1912.03820 (2019).
[113] S. Belkhale, R. Li, G. Kahn, R. McAllister, R. Calandra, S. Levine, Model-based meta-reinforcement learning for flight with suspended payloads, IEEE Robot. Autom. Lett. 6 (2) (2021) 1471–1478.
[114] A. Nagabandi, I. Clavera, S. Liu, R.S. Fearing, P. Abbeel, S. Levine, C. Finn, Learning to adapt in dynamic, real-world environments through meta-reinforcement learning, arXiv preprint arXiv:1803.11347 (2018).
[115] I. Clavera, A. Nagabandi, R.S. Fearing, P. Abbeel, S. Levine, C. Finn, Learning to adapt: Meta-learning for model-based control, arXiv preprint arXiv:1803.11347 3 (2018) 3.
[116] I. Clavera, J. Rothfuss, J. Schulman, Y. Fujita, T. Asfour, P. Abbeel, Model-based reinforcement learning via meta-policy optimization, Conference on Robot Learning, PMLR (2018) 617–629.
[117] F. Garcia, P.S. Thomas, A meta-mdp approach to exploration for lifelong reinforcement learning, Advances in Neural Information Processing Systems 32 (2019).
[118] Y. Duan, J. Schulman, X. Chen, P.L. Bartlett, I. Sutskever, P. Abbeel, Rl 2: Fast reinforcement learning via slow reinforcement learning, arXiv preprint arXiv:1611.02779 (2016).
[119] J.X. Wang, Z. Kurth-Nelson, D. Tirumala, H. Soyer, J.Z. Leibo, R. Munos, C. Blundell, D. Kumaran, M. Botvinick, Learning to reinforcement learn, 2016, arXiv preprint arXiv:1611.05763.
[120] S.K. Seyed Ghasemipour, S.S. Gu, R. Zemel, Smile: Scalable meta inverse reinforcement learning through context-conditional policies, Advances in Neural Information Processing Systems 32 (2019).
[121] B.H. Abed-alguni, Action-selection method for reinforcement learning based on cuckoo search algorithm, Arab. J. Sci. Eng. 43 (12) (2018) 6771–6785.
[122] B. Baker, O. Gupta, R. Raskar, N. Naik, Accelerating neural architecture search using performance prediction, arXiv preprint arXiv:1705.10823 (2017).
[123] H. Pham, M. Guan, B. Zoph, Q. Le, J. Dean, Efficient neural architecture search via parameters sharing, in: International conference on machine learning PMLR, 2018, pp. 4095–4104.
[124] K. Rakelly, A. Zhou, C. Finn, S. Levine, D. Quillen, Efficient off-policy meta-reinforcement learning via probabilistic context variables, in: International conference on machine learning PMLR, 2019, pp. 5331–5340.
[125] T. Raviv, S. Park, N. Shlezinger, O. Simeone, Y.C. Eldar, J. Kang, Meta-viterbinet: Online meta-learned viterbi equalization for non-stationary channels, in: 2021 IEEE International Conference on Communications Workshops (ICC Workshops) IEEE, 2021, pp. 1–6.
[126] A. Nagabandi, C. Finn, S. Levine, Deep online learning via meta-learning: Continual adaptation for model-based rl, arXiv preprint arXiv:1812.07671 (2018).
[127] S. Elfwing, E. Uchibe, K. Doya, Online meta-learning by parallel algorithm competition, in: Proceedings of the Genetic and Evolutionary Computation Conference, 2018, pp. 426–433.
[128] G. Denevi, D. Stamos, C. Ciliberto, M. Pontil, Online-within-online meta-learning, Advances in Neural Information Processing Systems 32 (2019).
[129] D. Li, T. Hospedales, Online meta-learning for multi-source and semi-supervised domain adaptation, European Conference on Computer Vision, Springer (2020) 382–403.
[130] H. Yao, Y. Zhou, M. Mahdavi, Z.J. Li, R. Socher, C. Xiong, Online structured meta-learning, Advances in Neural Information Processing Systems 33 (2020) 6779–6790.
[131] Q. Wang, H. van Hoof, Model-based meta reinforcement learning using graph structured surrogate models, arXiv preprint arXiv:2102.08291 (2021).
[132] M. Khodak, M.-F.F. Balcan, A.S. Talwalkar, Adaptive gradient-based meta-learning methods, Advances in Neural Information Processing Systems 32 (2019).
[133] M.-F. Balcan, M. Khodak, A. Talwalkar, Provable guarantees for gradient-based meta-learning, International Conference on Machine Learning, PMLR (2019) 424–433.
[134] G. Denevi, C. Ciliberto, R. Grazzi, M. Pontil, Learning-to-learn stochastic gradient descent with biased regularization, in: International Conference on Machine Learning, PMLR, 2019, pp. 1566–1575.
[135] Z. Zhuang, Y. Wang, K. Yu, S. Lu, No-regret non-convex online meta-learning, in: ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2020, pp. 3942–3946.
[136] D. Cai, R. Sheth, L. Mackey, N. Fusi, Weighted meta-learning, arXiv preprint arXiv:2003.09465 (2020).
[137] R. Kaushik, T. Anne, J.-B. Mouret, Fast online adaptation in robotics through meta-learning embeddings of simulated priors, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) IEEE, 2020, pp. 5269–5276.
[138] Y. Yuan, G. Zheng, K.-K. Wong, B. Ottersten, Z.-Q. Luo, Transfer learning and meta learning-based fast downlink beamforming adaptation, IEEE Trans. Wireless Commun. 20 (3) (2020) 1742–1755.
[139] M. Al-Shedivat, L. Li, E. Xing, A. Talwalkar, On data efficiency of meta-learning, in: International Conference on Artificial Intelligence and Statistics PMLR, 2021, pp. 1369–1377.
[140] D.A.E. Acar, R. Zhu, V. Saligrama, Memory efficient online meta learning, in: International Conference on Machine Learning, PMLR, 2021, pp. 32–42.
[141] R. Xie, Y. Wang, R. Wang, Y. Lu, Y. Zou, F. Xia, L. Lin, Long short-term temporal meta-learning in online recommendation, in: Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, 2022, pp. 1168–1176.
[142] K. Lee, S. Maji, A. Ravichandran, S. Soatto, Meta-learning with differentiable convex optimization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 10657–10665.
[143] S. Lin, G. Yang, J. Zhang, A collaborative learning framework via federated meta-learning, in: 2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS) IEEE, 2020, pp. 289–299.
[144] S. Yue, J. Ren, J. Xin, D. Zhang, Y. Zhang, W. Zhuang, Efficient federated meta-learning over multi-access wireless networks, IEEE J. Sel. Areas Commun. (2022).
[145] Y. Jiang, J. Konečný, K. Rush, S. Kannan, Improving federated learning personalization via model agnostic meta learning, arXiv preprint arXiv:1909.12488 (2019).
[146] Z. Chi, Z. Wang, W. Du, Heterogeneous federated meta-learning with mutually constrained propagation, IEEE Intell. Syst. (2022).
[147] A. Fallah, A. Mokhtari, A. Ozdaglar, Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach, Advances in Neural Information Processing Systems 33 (2020) 3557–3568.
[148] C. Li, S. Li, A. Zhang, Q. He, Z. Liao, J. Hu, Meta-learning for few-shot bearing fault diagnosis under complex working conditions, Neurocomputing 439 (2021) 197–211.
[149] F. Chen, M. Luo, Z. Dong, Z. Li, X. He, Federated meta-learning with fast convergence and efficient communication, arXiv preprint arXiv:1802.07876 (2018).
[150] G. Denevi, M. Pontil, C. Ciliberto, The advantage of conditional meta-learning for biased regularization and fine tuning, Advances in Neural Information Processing Systems 33 (2020) 964–974.
[151] W. Li, S. Wang, Federated meta-learning for spatial-temporal prediction, Neural Comput. Appl. (2022) 1–20.
[152] W. Zheng, L. Yan, C. Gou, F.-Y. Wang, Federated meta-learning for fraudulent credit card detection, in: Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, 2021, pp. 4654–4660.
[153] P.K. Chan, S.J. Stolfo, et al., Toward parallel and distributed learning by meta-learning, in: AAAI workshop in Knowledge Discovery in Databases, 1993, pp. 227–240.
[154] S. Stolfo, D.W. Fan, W. Lee, A. Prodromidis, P. Chan, Credit card fraud detection using meta-learning: Issues and initial results, AAAI-97 Workshop on Fraud Detection and Risk Management (1997) 83–90.
[155] P.K. Chan, S.J. Stolfo, Learning with non-uniform class and cost distributions: Effects and a distributed multi-classifier approach, In Workshop Notes KDD-98 Workshop on Distributed Data Mining, Citeseer (1998).
[156] P.K. Chan, S.J. Stolfo, On the accuracy of meta-learning for scalable data mining, Journal of Intelligent Information Systems 8 (1) (1997) 5–28.
[157] K. Philip, S. Chan, Toward scalable learning with non-uniform class and cost distributions: A case study in credit card fraud detection, in: Proceeding of the Fourth International Conference on Knowledge Discovery and Data Mining, 1998, pp. 164–168.
[158] S.J. Stolfo, A.L. Prodromidis, S. Tselepis, W. Lee, D.W. Fan, P.K. Chan, Jam: Java agents for meta-learning over distributed databases., in: KDD, Vol. 97, 1997, pp. 74–81.

[159] S.J. Stolfo, P.K. Chan, D. Fan, W. Lee, A. Prodromidis, Meta-learning agents for fraud and intrusion detection in financial information systems (1996).
[160] S. Stolfo, W. Fan, A.P.W. Lee, S. Tselepis, P. Chan, Agentbased fraud and intrusion detection in financial information systems, in: Submitted to IEEE Symposium on Security and Privacy, 1998.
[161] A. Prodromidis, P. Chan, S. Stolfo, et al., Meta-learning in distributed data mining systems: Issues and approaches, Advances in distributed and parallel knowledge discovery 3 (2000) 81–114.
[162] J. Perols, Financial statement fraud detection: An analysis of statistical and machine learning algorithms, Auditing: A Journal of Practice & Theory 30 (2) (2011) 19–50.
[163] M. Cecchini, H. Aytug, G.J. Koehler, P. Pathak, Detecting management fraud in public companies, Manage. Sci. 56 (7) (2010) 1146–1160.
[164] D. Hu, J.L. Zhao, Z. Hua, M.C. Wong, Network-based modeling and analysis of systemic risk in banking systems, MIS quarterly (2012) 1269–1291.
[165] Y.G. Şahin, E. Duman, Detecting credit card fraud by decision trees and support vector machines (2011).
[166] J.K.-F. Pun, Improving credit card fraud detection using a meta-learning strategy, Ph.D. thesis (2011).
[167] S.K. Sen, S. Dash, Meta learning algorithms for credit card fraud detection, International Journal of Engineering Research and Development 6 (6) (2013) 16–20.
[168] K. McCarthy, B. Zabar, G. Weiss, Does cost-sensitive learning beat sampling for classifying rare classes?, in: Proceedings of the 1st international workshop on Utility-based data mining, 2005, pp 69–77.
[169] G.M. Weiss, K. McCarthy, B. Zabar, Cost-sensitive learning vs. sampling: Which is best for handling unbalanced classes with unequal error costs?, Dmin 7 (35–41) (2007) 24
[170] C. Phua, D. Alahakoon, V. Lee, Minority report in fraud detection: classification of skewed data, Acm sigkdd explorations newsletter 6 (1) (2004) 50–59.
[171] H. Guo, H.L. Viktor, Learning from imbalanced data sets with boosting and data generation: the databoost-im approach, ACM Sigkdd Explorations Newsletter 6 (1) (2004) 30–39.
[172] H. He, Y. Bai, E.A. Garcia, S. Li, Adasyn: Adaptive synthetic sampling approach for imbalanced learning, in: 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence), IEEE 2008 (2008) 1322–1328.
[173] X. Tan, Integrating classification with k-means to detect e-commerce transaction anomaly (2015).
[174] S. Goel, J. Gangolly, S.R. Faerman, O. Uzuner, Can linguistic predictors detect fraudulent financial filings?, Journal of Emerging Technologies in Accounting 7 (1) (2010) 25–46
[175] F.H. Glancy, S.B. Yadav, A computational model for financial reporting fraud detection, Decis. Support Syst. 50 (3) (2011) 595–601.
[176] S. Goel, J. Gangolly, Beyond the numbers: Mining the annual reports for hidden cues indicative of financial statement fraud, Intelligent Systems in Accounting, Finance and Management 19 (2) (2012) 75–89.
[177] A. Abbasi, C. Albrecht, A. Vance, J. Hansen, Metafraud: a meta-learning framework for detecting financial fraud, Mis Quarterly (2012) 1293–1327.
[178] A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi, G. Bontempi, Credit card fraud detection and concept-drift adaptation with delayed supervised information, in: 2015 international joint conference on Neural networks (IJCNN), IEEE 2015 (2015) 1–8.
[179] A. Dal Pozzolo, R. Johnson, O. Caelen, S. Waterschoot, N.V. Chawla, G. Bontempi, Using hddt to avoid instances propagation in unbalanced and evolving data streams, in: 2014 International Joint Conference on Neural Networks (IJCNN) IEEE, 2014, pp. 588–594.
[180] M. Kavitha, M. Suriakala, Real time credit card fraud detection on huge imbalanced data using meta-classifiers, in: 2017 international conference on inventive computing and informatics (ICICI), IEEE 2017 (2017) 881–887.
[181] A. Somasundaram, S. Reddy, Parallel and incremental credit card fraud detection model to handle concept drift and data imbalance, Neural Comput. Appl. 31 (1) (2019) 3–14.
[182] V. Van Vlasselaer, C. Bravo, O. Caelen, T. Eliassi-Rad, L. Akoglu, M. Snoeck, B. Baesens, Apate: A novel approach for automated credit card transaction fraud detection using network-based extensions, Decis. Support Syst. 75 (2015) 38–48.
[183] D. Tripathi, Y. Sharma, T. Lone, S. Dwivedi, Credit card fraud detection using local outlier factor, International Journal of Pure and Applied Mathematics 118 (7) (2018) 229–234.
[184] S. Nami, M. Shajari, Cost-sensitive payment card fraud detection based on dynamic random forest and k-nearest neighbors, Expert Syst. Appl. 110 (2018) 381–392.
[185] X. Zhang, Y. Han, W. Xu, Q. Wang, Hoba: A novel feature engineering methodology for credit card fraud detection with a deep learning architecture, Inf. Sci. (2019).
[186] F. Carcillo, Y.-A. Le Borgne, O. Caelen, Y. Kessaci, F. Oblé, G. Bontempi, Combining unsupervised and supervised learning in credit card fraud detection, Inf. Sci. (2019).
[187] T.A. Olowookere, O.S. Adewale, A framework for detecting credit card fraud with cost-sensitive meta-learning ensemble approach, Scientific African 8 (2020) e00464.
[188] W. Zheng, L. Yan, C. Gou, F.-Y. Wang, Federated meta-learning for fraudulent credit card detection, in: Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20), 2020.
[189] J. Błaszczyński, A.T. de Almeida Filho, A. Matuszyk, M. Szelg, R. Słowiński, Auto loan fraud detection using dominance-based rough set approach versus machine learning methods, Expert Systems with Applications 163 (2021) 113740.
[190] Y. Sahin, E. Duman, Detecting credit card fraud by ann and logistic regression, in: 2011 International Symposium on Innovations in Intelligent Systems and Applications, IEEE 2011 (2011) 315–319.
[191] A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi, G. Bontempi, Credit card fraud detection: a realistic modeling and a novel learning strategy, IEEE transactions on neural networks and learning systems 29 (8) (2017) 3784–3797.
[192] B.M. Williams, L.A. Hoel, Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results, Journal of transportation engineering 129 (6) (2003) 664–672.
[193] M. Sabry, H. Abd-El-Latif, N. Badra, Comparison between regression and arima models in forecasting traffic volume, Aust. J. Basic Appl. Sci. 1 (2) (2007) 126–136.
[194] S.V. Kumar, L. Vanajakshi, Short-term traffic flow prediction using seasonal arima model with limited input data, European Transport Research Review 7 (3) (2015) 1–9.
[195] B. Moghimi, A. Safikhani, C. Kamga, W. Hao, Cycle-length prediction in actuated traffic-signal control using arima model, Journal of Computing in Civil Engineering 32 (2) (2018) 04017083.
[196] V.B. Gavirangaswamy, G. Gupta, A. Gupta, R. Agrawal, Assessment of arima-based prediction techniques for road-traffic volume, in: Proceedings of the fifth international conference on management of emergent digital EcoSystems, 2013, pp. 246–251.
[197] K. Makatjane, N. Moroke, Comparative study of holt-winters triple exponential smoothing and seasonal arima: forecasting short term seasonal car sales in south africa, Makatjane KD, Moroke ND, 2016.
[198] M.S. Jamil, S. Akbar, Taxi passenger hotspot prediction using automatic arima model, in: 2017 3rd International Conference on Science in Information Technology (ICSITech) IEEE, 2017, pp. 23–28.
[199] S. Shahriari, M. Ghasri, S. Sisson, T. Rashidi, Ensemble of arima: combining parametric and bootstrapping technique for traffic flow prediction, Transportmetrica A: Transport Science 16 (3) (2020) 1552–1573.
[200] J. Guan, W. Wang, W. Li, S. Zhou, A unified framework for predicting kpis of on-demand transport services, IEEE access 6 (2018) 32005–32014.
[201] W. Li, J. Cao, J. Guan, S. Zhou, G. Liang, W.K. So, M. Szczecinski, A general framework for unmet demand prediction in on-demand transport services, IEEE Trans. Intell. Transp. Syst. 20 (8) (2018) 2820–2830.
[202] A. Vahedian, X. Zhou, L. Tong, W.N. Street, Y. Li, Predicting urban dispersal events: A two-stage framework through deep survival analysis on mobility data, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 5199–5206.
[203] Z. Wu, G. Lian, A novel dynamically adjusted regressor chain for taxi demand prediction, in: 2020 International Joint Conference on Neural Networks (IJCNN), IEEE, 2020, pp. 1–10.
[204] L. Lin, J. Li, F. Chen, J. Ye, J. Huai, Road traffic speed prediction: a probabilistic model fusing multi-source data, IEEE Trans. Knowl. Data Eng. 30 (7) (2017) 1310–1323.
[205] D. Salinas, M. Bohlke-Schneider, L. Callot, R. Medico, J. Gasthaus, High-dimensional multivariate forecasting with low-rank gaussian copula processes, arXiv preprint arXiv:1910.03002 (2019).
[206] H. Yu, T. Nghia, B.K.H. Low, P. Jaillet, Stochastic variational inference for bayesian sparse gaussian process regression, in: 2019 International Joint Conference on Neural Networks (IJCNN) IEEE, 2019, pp. 1–8.
[207] D. Deng, C. Shahabi, U. Demiryurek, L. Zhu, R. Yu, Y. Liu, Latent space model for road networks to predict time-varying traffic, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 1525–1534.
[208] H.-F. Yu, N. Rao, I.S. Dhillon, Temporal regularized matrix factorization for high-dimensional time series prediction, Advances in neural information processing systems 29 (2016) 847–855.
[209] N. Polson, V. Sokolov, Bayesian particle tracking of traffic flows, IEEE Trans. Intell. Transp. Syst. 19 (2) (2017) 345–356.
[210] P. Duan, G. Mao, W. Liang, D. Zhang, A unified spatio-temporal model for short-term traffic flow prediction, IEEE Trans. Intell. Transp. Syst. 20 (9) (2018) 3212–3223.
[211] K. Ishibashi, S. Harada, R. Kawahara, Inferring latent traffic demand offered to an overloaded link with modeling qos-degradation effect, IEICE Transactions on Communications (2018).
[212] J. Wang, Q. Gu, J. Wu, G. Liu, Z. Xiong, Traffic speed prediction and congestion source exploration: A deep learning method, in: 2016 IEEE 16th international conference on data mining (ICDM), IEEE 2016 (2016) 499–508.
[213] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, Y. Wang, Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction, Sensors 17 (4) (2017) 818.
[214] S. Sun, H. Wu, L. Xiang, City-wide traffic flow forecasting using a deep convolutional neural network, Sensors 20 (2) (2020) 421.
[215] K. Lee, W. Rhee, Ddp-gcn: Multi-graph convolutional network for spatiotemporal traffic forecasting, arXiv preprint arXiv:1905.12256 (2019).
[216] K. Lee, W. Rhee, Graph convolutional modules for traffic forecasting (2018).
[217] H. Peng, H. Wang, B. Du, M.Z.A. Bhuiyan, H. Ma, J. Liu, L. Wang, Z. Yang, L. Du, S. Wang, et al., Spatial temporal incidence dynamic graph neural networks for traffic flow forecasting, Inf. Sci. 521 (2020) 277–290.

[218] W. Chen, L. Chen, Y. Xie, W. Cao, Y. Gao, X. Feng, Multi-range attentive bicomponent graph convolutional network for traffic forecasting, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 3529–3536.

[219] R. Huang, C. Huang, Y. Liu, G. Dai, W. Kong, Lsgcn: Long short-term traffic prediction with graph convolutional networks., IJCAI (2020) 2355–2361.

[220] B. Yu, H. Yin, Z. Zhu, Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting, 2017, arXiv preprint arXiv:1709.04875.

[221] Y. Li, R. Yu, C. Shahabi, Y. Liu, Diffusion convolutional recurrent neural network: Data-driven traffic forecasting, arXiv preprint arXiv:1707.01926 (2017).

[222] C. Chen, K. Li, S.G. Teo, X. Zou, K. Wang, J. Wang, Z. Zeng, Gated residual recurrent graph neural networks for traffic prediction, in: Proceedings of the AAAI conference on artificial intelligence, Vol. 33, 2019, pp. 485–492.

[223] S. Guo, Y. Lin, N. Feng, C. Song, H. Wan, Attention based spatial-temporal graph convolutional networks for traffic flow forecasting, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 922–929.

[224] C. Zheng, X. Fan, C. Wang, J. Qi, Gman: A graph multi-attention network for traffic prediction, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 1234–1241.

[225] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in neural information processing systems 25 (2012) 1097–1105.

[226] A. Roy, K.K. Roy, A.A. Ali, M.A. Amin, A. Rahman, Unified spatio-temporal modeling for traffic forecasting using graph neural network, arXiv preprint arXiv:2104.12518 (2021).

[227] Y. Lin, H. Hong, X. Yang, X. Yang, P. Gong, J. Ye, Meta graph attention on heterogeneous graph with node-edge co-evolution, arXiv preprint arXiv:2010.04554 (2020).

[228] Z. Pan, W. Zhang, Y. Liang, W. Zhang, Y. Yu, J. Zhang, Y. Zheng, Spatio-temporal meta learning for urban traffic prediction, IEEE Trans. Knowl. Data Eng. (2020).

[229] S. Fang, X. Pan, S. Xiang, C. Pan, Meta-msnet: Meta-learning based multi-source data fusion for traffic flow prediction, IEEE Signal Process. Lett. 28 (2020) 6–10.

[230] Y. Zhang, Y. Li, X. Zhou, J. Luo, cst-ml: Continuous spatial-temporal meta-learning for traffic dynamics prediction, in: 2020 IEEE International Conference on Data Mining (ICDM) IEEE, 2020, pp. 1418–1423.

[231] Z. Pan, S. Ke, X. Yang, Y. Liang, Y. Yu, J. Zhang, Y. Zheng, Autostg: Neural architecture search for predictions of spatio-temporal graph, in: Proceedings of the Web Conference 2021, 2021, pp. 1846–1855.

[232] K.-H.N. Bui, H. Yi, Optimal hyperparameter tuning using meta-learning for big traffic datasets, in: 2020 IEEE International Conference on Big Data and Smart Computing (BigComp) IEEE, 2020, pp. 48–54.

[233] J. Beel, L. Kotthoff, Preface: The 1st interdisciplinary workshop on algorithm selection and meta-learning in information retrieval (amir)., in: AMIR@ ECIR, 2019, pp. 1–9.

[234] X. Yin, G. Wu, J. Wei, Y. Shen, H. Qi, B. Yin, A comprehensive survey on traffic prediction, arXiv preprint arXiv:2004.08555 (2020).

[235] V. Varghese, M. Chikaraishi, J. Urata, Deep learning in transport studies: A meta-analysis on the prediction accuracy, Journal of Big Data Analytics in Transportation (2020) 1–22.

[236] Y. Bao, M. Wu, S. Chang, R. Barzilay, Few-shot text classification with distributional signatures, arXiv preprint arXiv:1908.06039 (2019).

[237] S. Sun, Q. Sun, K. Zhou, T. Lv, Hierarchical attention prototypical networks for few-shot text classification, in: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), 2019, pp. 476–485.

[238] B. Hui, P. Zhu, Q. Hu, Q. Wang, Self-attention relation network for few-shot learning, in: 2019 IEEE International Conference on Multimedia & Expo Workshops (ICMEW) IEEE, 2019, pp. 198–203.

[239] S. Deng, N. Zhang, Z. Sun, J. Chen, H. Chen, When low resource nlp meets unsupervised language model: Meta-pretraining then meta-learning for few-shot text classification (student abstract), in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 13773–13774.

[240] N. Holla, P. Mishra, H. Yannakoudakis, E. Shutova, Learning to learn to disambiguate: Meta-learning for few-shot word sense disambiguation, arXiv preprint arXiv:2004.14355 (2020).

[241] R. Geng, B. Li, Y. Li, X. Zhu, P. Jian, J. Sun, Induction networks for few-shot text classification, arXiv preprint arXiv:1902.10482 (2019).

[242] R. Geng, B. Li, Y. Li, J. Sun, X. Zhu, Dynamic memory induction networks for few-shot text classification, arXiv preprint arXiv:2005.05727 (2020).

[243] N. Zhang, Z. Sun, S. Deng, J. Chen, H. Chen, Improving few-shot text classification via pretrained language representations, arXiv preprint arXiv:1908.08788 (2019).

[244] T. Bansal, R. Jha, A. McCallum, Learning to few-shot learn across diverse natural language classification tasks, arXiv preprint arXiv:1911.03863 (2019).

[245] P. Sun, Y. Ouyang, W. Zhang, X.-Y. Dai, Meda: Meta-learning with data augmentation for few-shot text classification.

[246] K. Yang, N. Zheng, X. Dai, L. He, S. Huang, J. Chen, Enhance prototypical network with text descriptions for few-shot relation classification, in: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, 2020, pp. 2273–2276.

[247] Q. Luo, L. Liu, Y. Lin, W. Zhang, Dont miss the labels: Label-semantic augmented meta-learner for few-shot text classification, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, 2021, pp. 2773–2782.

[248] S. Ohashi, J. Takayama, T. Kajiwara, Y. Arase, Distinct label representations for few-shot text classification, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), 2021, pp. 831–836.

[249] S. Minaee, Y.Y. Boykov, F. Porikli, A.J. Plaza, N. Kehtarnavaz, D. Terzopoulos, Image segmentation using deep learning: A survey, IEEE transactions on pattern analysis and machine intelligence (2021).

[250] M.H. Hesamian, W. Jia, X. He, P. Kennedy, Deep learning techniques for medical image segmentation: achievements and challenges, Journal of digital imaging 32 (4) (2019) 582–596.

[251] A.K. Pambala, T. Dutta, S. Biswas, Sml: Semantic meta-learning for few-shot semantic segmentation, Pattern Recogn. Lett. 147 (2021) 93–99.

[252] A. Banerjee, Meta-drn: Meta-learning for 1-shot image segmentation, in: 2020 IEEE 17th India Council International Conference (INDICON), IEEE 2020 (2020) 1–6.

[253] P. Tian, Z. Wu, L. Qi, L. Wang, Y. Shi, Y. Gao, Differentiable meta-learning model for few-shot semantic segmentation, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 12087–12094.

[254] K. Rakelly, E. Shelhamer, T. Darrell, A.A. Efros, S. Levine, Few-shot segmentation propagation with guided networks, arXiv preprint arXiv:1806.07373 (2018).

[255] P. Zhang, J. Li, Y. Wang, J. Pan, Domain adaptation for medical image segmentation: a meta-learning method, Journal of Imaging 7 (2) (2021) 31.

[256] R. Khadka, D. Jha, S. Hicks, V. Thambawita, M.A. Riegler, S. Ali, P. Halvorsen, Meta-learning with implicit gradients in a few-shot setting for medical image segmentation, Comput. Biol. Med. 105227 (2022).

[257] Q. Liu, Q. Dou, P.-A. Heng, Shape-aware meta-learning for generalizing prostate mri segmentation to unseen domains, in: International Conference on Medical Image Computing and Computer-Assisted Intervention Springer, 2020, pp. 475–485.

[258] G.F. Campos, S. Barbon, R.G. Mantovani, A meta-learning approach for recommendation of image segmentation algorithms, in: 2016 29th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI) IEEE, 2016, pp. 370–377.

[259] G.J. Aguiar, R.G. Mantovani, S.M. Mastelini, A.C. de Carvalho, G.F. Campos, S.B. Junior, A meta-learning approach for selecting image segmentation algorithm, Pattern Recogn. Lett. 128 (2019) 480–487.

[260] M. He, D. He, Deep learning based approach for bearing fault diagnosis, IEEE Trans. Ind. Appl. 53 (3) (2017) 3057–3065.

[261] F. Jia, Y. Lei, L. Guo, J. Lin, S. Xing, A neural network constructed by deep learning technique and its application to intelligent fault diagnosis of machines, Neurocomputing 272 (2018) 619–628.

[262] Y. Dong, Y. Li, H. Zheng, R. Wang, M. Xu, A new dynamic model and transfer learning based intelligent fault diagnosis framework for rolling element bearings race faults: Solving the small sample problem, ISA transactions (2021).

[263] Y. Yang, H. Wang, Z. Liu, Z. Yang, Few-shot learning for rolling bearing fault diagnosis via siamese two-dimensional convolutional neural network, in: 2020 11th International Conference on Prognostics and System Health Management (PHM-2020 Jinan) IEEE, 2020, pp. 373–378.

[264] S. Dixit, N.K. Verma, A. Ghosh, Intelligent fault diagnosis of rotary machines: Conditional auxiliary classifier gan coupled with meta learning using limited data, IEEE Trans. Instrum. Meas. 70 (2021) 1–11.

[265] D. Wang, M. Zhang, Y. Xu, W. Lu, J. Yang, T. Zhang, Metric-based meta-learning model for few-shot fault diagnosis under multiple limited data conditions, Mechanical Systems and Signal Processing 155 (2021) 107510.

[266] F. Li, J. Chen, J. Pan, T. Pan, Cross-domain learning in rotating machinery fault diagnosis under various operating conditions based on parameter transfer, Meas. Sci. Technol. 31 (8) (2020) 085104.

[267] Y. Feng, J. Chen, Z. Yang, X. Song, Y. Chang, S. He, E. Xu, Z. Zhou, Similarity-based meta-learning network with adversarial domain adaptation for cross-domain fault identification, Knowl.-Based Syst. 217 (2021) 106829.

[268] Y. Feng, J. Chen, J. Xie, T. Zhang, H. Lv, T. Pan, Meta-learning as a promising approach for few-shot cross-domain fault diagnosis: Algorithms, applications, and prospects, Knowl.-Based Syst. 235 (2022) 107646.

[269] T. Zhang, J. Chen, F. Li, K. Zhang, H. Lv, S. He, E. Xu, Intelligent fault diagnosis of machines with small & imbalanced data: A state-of-the-art review and possible extensions, ISA transactions 119 (2022) 152–171.

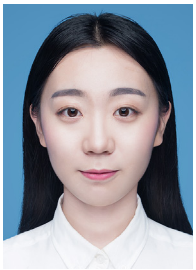
[270] C. Yu, Y. Ning, Y. Qin, W. Su, X. Zhao, Multi-label fault diagnosis of rolling bearing based on meta-learning, Neural Comput. Appl. 33 (10) (2021) 5393–5407.

[271] S.T. Jose, O. Simeone, Information-theoretic generalization bounds for meta-learning and applications, Entropy 23 (1) (2021) 126.

[272] Q. Yao, M. Wang, Y. Chen, W. Dai, Y.-F. Li, W.-W. Tu, Q. Yang, Y. Yu, Taking human out of learning applications: A survey on automated machine learning, arXiv preprint arXiv:1810.13306 (2018).

[273] J. Pearl, Causal inference in statistics: An overview, Statistics surveys 3 (2009) 96–146.

Yingjie Tian is a Professor of Research Center on Fictitious Economy and Data Science, Chinese Academy of Sciences. He received the bachelor?s degree in mathematics (1994), the master?s degree in applied mathematics (1997), and the Ph.D. degree in Management Science and Engineering (2005). His current research interests include artificial intelligent, machine learning and optimization.



Xiaoxi Zhao is pursuing the Ph.D. degree with the School of Economics and Management, University of Chinese Academy of Sciences, Beijing, China. She received the bachelor?s degree in Economic statistics from Shandong University of Finance and Economics, Jinan, China, in 2018, and the master?s degree in applied statistics from the University of Chinese Academy of Sciences, Beijing, China, in 2020. Her current research interests include data mining, and machine learning.

Wei Huang (Wayne) is an AIS Fellow, a chair professor at the College of Business, Southern University of Science and Technology. His professional career includes worked as a chair professor of Xian Jiaotong university, a Fellow at Harvard University, a tenured Armbruster-Scholar Professor at Ohio University and a Sir Anthony Mason Fellow at the University of New South Wales in Australia. His research interests include business intelligence/analytics, big data management and data quality, e-government/e-commerce, IT and service outsourcing (PPP), key issues of IT/IS management, etc. He has published more than 12 academic books (including book chapters) in the United States, Germany and other countries, as well as more than 200 refereed research papers in international journals, book chapters, and international quality conference proceedings.