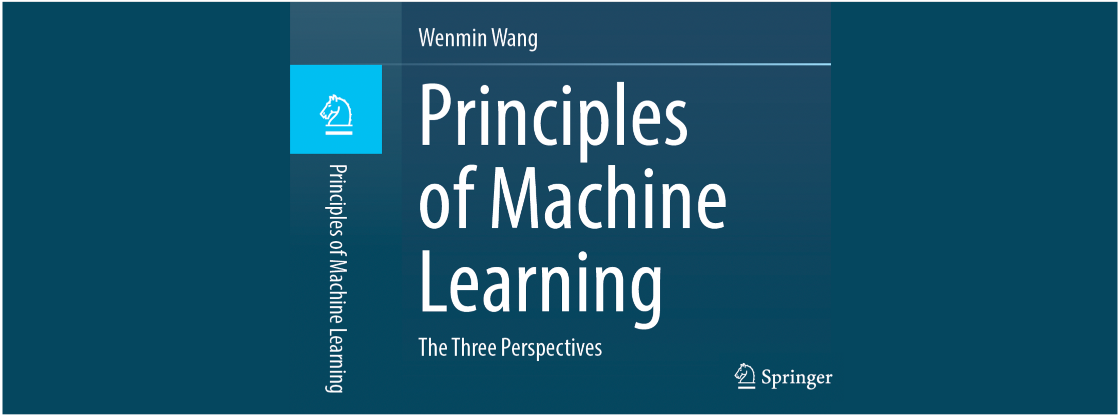


Principles of Machine Learning

Wenmin Wang

Principles of Machine Learning

The Three Perspectives

Springer

Part III Paradigms

8 Supervised Learning Paradigm

9 Unsupervised Learning Paradigm

10 Reinforcement Learning Paradigm

11 Other Learning Quasi-Paradigm

11 Other Learning Quasi-Paradigm

Contents

11.1 Ensemble Learning

11.2 Meta-Learning

11.3 Transfer Learning

11.4 Self-Supervised Learning

11.5 n-Shot Learning

About the Ensemble Learning

- Ensemble learning can be traced back to the extensions of the theory of learnability: *strong learnability* and *weak learnability*.
- The formal definitions of strong learnability and weak learnability are provided by “Thoughts on Hypothesis Boosting” in 1988.
- A detailed proof to the hypothesis boosting problem, and a method to boost weak learning algorithms to any strong learner are provided by “The Strength of Weak Learnability” in 1990.
- The 1990s was a period for classic ensemble learning methods.
- The 2000s was a period that ensemble learning combines with neural networks.

Definition

Definition: (Ensemble Learning)

Ensemble learning is the approach of organically combining several base learners to form a strong learner, whose performance surpasses any of the base learners before the combination.

Base learners: The basic models used to solve machine learning tasks such as classification and regression. Which are regarded as weak learners, with simplicity, combinability, and complementarity.

Strong learner: The ensemble model with better predictive performance obtained by multiple base learners.

Combination Modes

Parallel combination and Sequential combination

(1) **Parallel combination**
It is to connect T base learners $\{h_t(\boldsymbol{x})\}_{t=1}^{T}$ parallelly, so that for a common machine learning task, it can process parallelly and independently, and then integrate the results of each base learner into a strong learner $\boldsymbol{h}(\boldsymbol{x})$.

(2) **Sequential combination**
It is to connect T base learner $\{h_t(\boldsymbol{x})\}_{t=1}^{T}$ sequentially, so that for a common machine learning task, it learns in stages and iteratively, finally resulting in a strong learner $\boldsymbol{h}(\boldsymbol{x})$.

Combination Modes

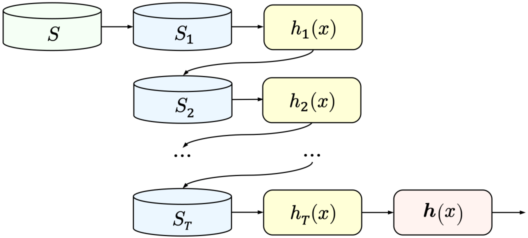
Homogeneous and Heterogeneous

(1) **Homogeneous combination**
It refers to a strong learner $\boldsymbol{h}(\boldsymbol{x})$ that is composed of base learners of the same type and used for the same learning task.
E.g., T base learners $\{h_t(\boldsymbol{x})\}_{t=1}^{T}$ all use same logistic regression algorithm for binary classification.

(2) **Heterogeneous combination**
It refers to the strong learner $\boldsymbol{h}(\boldsymbol{x})$ can be composed of different types of base learners, but used for the same learning task.
E.g., T base learners $\{h_t(\boldsymbol{x})\}_{t=1}^{T}$ respectively use logistic regression, naive Bayes, decision tree, and support vector machine for binary classification.

Ensemble Methods

Boosting: It uses the sequential combination of ensemble learning, aiming to reduce bias and variance through the iteration of base learners.



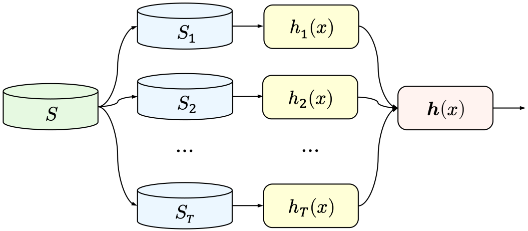
(1) It assigns the same weight to each sample in S, and generates boosted S_1 to train the base learner $h_1(\boldsymbol{x})$.

(2) The samples by $h_1(\boldsymbol{x})$ are weighted to generate boosted S_2, and to train the base learner $h_2(\boldsymbol{x})$.

(3) Repeat the above process, the prediction results of each base learner are combined by weighted voting to obtain the strong learner $\boldsymbol{h}(\boldsymbol{x})$.

Ensemble Methods

Bagging: The abbreviation of “bootstrap aggregating”.



(1) It randomly selects samples from the original samples S using sampling with replacement, and generates T bootstrap samples $\{S_t\}_{t=1}^{T}$.

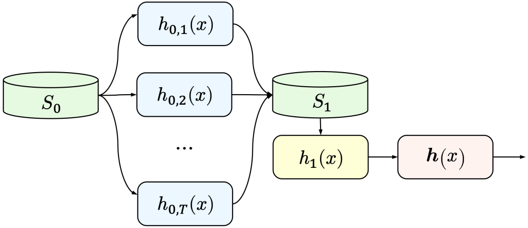
(2) Each bootstrap sample S_t is used to train each base learner $h_t(\boldsymbol{x})$.

(3) By aggregating the prediction results of each base learner, the required strong learner $\boldsymbol{h}(\boldsymbol{x})$ is obtained.

Random forest is an ensemble learning algorithm for classification and regression that combines the bagging strategy with random feature selection of decision trees.

Ensemble Methods

Stacking: It allows several heterogeneous base learners to be combined into a strong learner.



(1) It uses the original samples as the *level*-0 samples S_0, and for the samples S_0 to use the *level*-0 base learner $\{h_{0,t}(\boldsymbol{x})\}_{t=1}^{T}$ for learning.

(2) The output of the *level*-0 base learner forms the *level*-1 samples S_1, which are then learned by the *level*-1 meta-learner $h_1(\boldsymbol{x})$.

(3) The result of the *level*-1 meta-learner forms a strong learner $\boldsymbol{h}(\boldsymbol{x})$.

Averaging Scheme

Simple Averaging: The simple averaging scheme calculates the sum of T base learners $\{h_t(\boldsymbol{x})\}_{t=1}^T$, and then divides by the number of base learners T to get a strong learner $\boldsymbol{h}(\boldsymbol{x})$. Its expression is as follows:

$$\boldsymbol{h}(\boldsymbol{x}) = \frac{1}{T}\sum_{t=1}^{T} h_t(\boldsymbol{x}).$$

Weighted Averaging: In the weighted averaging scheme, it calculates the sum of T *weighted* base learners $\{w_t h_t(\boldsymbol{x})\}_{t=1}^T$ to derive a strong learner $\boldsymbol{h}(\boldsymbol{x})$. The calculation equation is as follows:

$$\boldsymbol{h}(\boldsymbol{x}) = \sum_{t=1}^{T} w_t h_t(\boldsymbol{x}), \quad w_t \geq 0 \text{ and } \sum_{t=1}^{T} w_t = 1.$$

Voting Scheme

Hard Voting: The schemes of majority voting, and weighted majority voting.

$$\boldsymbol{h}\left(\boldsymbol{x}_i\right) = \arg\max_j \sum_{t=1}^{T} \mathbb{I}\left(h_t\left(\boldsymbol{x}_i\right) = c_j\right), \quad \boldsymbol{h}\left(\boldsymbol{x}_i\right) = \arg\max_j \sum_{t=1}^{T} w_t \mathbb{I}\left(h_t\left(\boldsymbol{x}_i\right) = c_j\right).$$

Where, indicator function $\mathbb{I}\left(\omega\right) = 1$ if ω is true, otherwise 0; and $w_t \geq 0$, $\sum_{t=1}^{T} w_t = 1$.

Soft Voting: The schemes of probabilistic voting, and weighted probabilistic voting.

$$\boldsymbol{h}\left(\boldsymbol{x}_i\right) = \arg\max_j \sum_{t=1}^{T} P\left(h_t\left(\boldsymbol{x}_i\right) = c_j | \boldsymbol{x}_i\right), \quad \boldsymbol{h}(\boldsymbol{x}_i) = \arg\max_j \sum_{t=1}^{T} w_t P\left(h_t(\boldsymbol{x}_i) = c_j | \boldsymbol{x}_i\right).$$

Similarly, $w_t \geq 0$, and $\sum_{t=1}^{T} w_t = 1$.

About the Meta-Learning

- The prefix “meta-” originates from Greek, meaning “above” or “beyond”.
- In epistemology, it is defined in an abstract recursive way: “X about X”.
- Meta-learning is a branch of meta-cognition, known as “learning about one’s own learning and learning processes”.
- Humans can learn and need to continue learning. Not only learning new concepts and skills but also learning their inductive bias, i.e., learning how to obtain a hypothesis and how to make a generalization.
- Meta-learning is also considered as “learning to learn”.

Definition

Definition: (Meta-Learning)

Meta-learning, also known as “learning to learn” or “learning how to learn”, is a quasi-paradigm of machine learning, which dedicated to acquiring meta-knowledge, using it to train the meta-model, and then applying the meta-model to solve new problems and tasks in machine learning.

Based on the recursive definition of meta-learning, it can be thought of as “learning about learning”.

Therefore it is logical to call meta-learning “learning to learn”, or “learning how to learn”.

Related Discourses

Workshop on Meta-Learning: The Statement:

> Recent years have seen rapid progress in meta-learning methods, which transfer knowledge across tasks and domains to efficiently learn new tasks, optimize the learning process itself, and even generate new learning methods from scratch. Meta-learning can be seen as the logical conclusion of the arc that machine learning has undergone in the last decade, from learning classifiers, to learning representations, and finally to learning algorithms that themselves acquire representations, classifiers, and policies for acting in environments.

https://meta-learn.github.io/2022/

Note, the phrase "transfer knowledge" is used in the above description.

Related Discourses

Workshop on Meta-Learning: The Questions:

- What are the meta-learning processes in nature (e.g., in humans), and how can we take inspiration from them?
- What is the relationship between meta-learning, continual learning, and transfer learning?
- What interactions exist between meta-learning and large pretrained / foundation models?
- What principles can we learn from meta-learning to help us design the next generation of learning systems?
- What kind of theoretical principles can we develop for meta-learning?
- How can we exploit our domain knowledge to effectively guide the meta-learning process and make it more efficient?
- How can be design better benchmarks for different meta-learning scenarios?

Related Discourses

Workshop on Learning to Learn: The Statement:

> Recent years have seen a lot of interest in the use and development of learning-to-learn algorithms. Research on learning-to-learn, or meta-learning, algorithms is often motivated by the hope to learn representations that can be easily transferred to the learning of new skills, and lead to faster learning. Yet, current meta-learned representations often struggle to generalize to novel task settings. In this workshop, we'd like to discuss how humans meta-learn, and what we can and should expect from learning-to-learn in the field of machine learning.

https://sites.google.com/view/learning-2-learn/

Related Terminologies

Meta-Data: The data for machine learning, such as datasets, data configurations, annotated samples, hyperparameters, and performance metrics.

Meta-knowledge: The knowledge about machine learning model knowledge, including datasets, meta-data, learning algorithms, hardware configurations, training techniques, evaluation methods, and ablation experiments.

Meta-Model: The model that can be used to build other machine learning models, including the model's documentation, source code, datasets, and evaluation metrics.

In a sense, it is similar to a baseline, a reference model, or foundational model for further development of machine learning.

About the Transfer Learning

- Transfer learning originates from the theory of transfer of learning in psychology, is also one of research contents in cognitive science.
- Learning transfer occurs when people apply the information, strategies, and skills they have learned to new scenes or environments.
- Transfer learning is to apply the knowledge or skills learned in solving a certain problem to another different but related task.
- In machine learning, transfer learning occurs between the source model and the target model.

Definition

Definition: (Transfer Learning)

In machine learning, transfer learning is a quasi-paradigm that transfers the knowledge or functionality of a source model of machine learning to a different but related target model.

Why transfer learning needed, as in the paradigm of supervised learning:

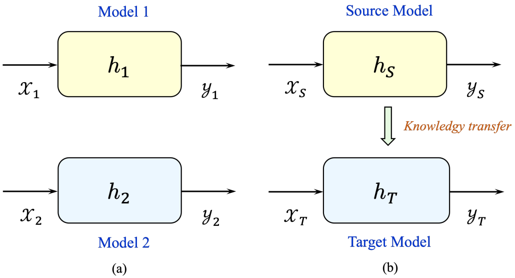
- A machine learning model can only solve a specific task.
- A lot of manually labeled samples are required for extensive training.
- Training and test data must be in same distribution, same data type.

Supervised Learning vs. Transfer Learning

(a) In supervised learning, h_1 and h_2 in “Model 1” and “Model 2” are trained separately, and employ their respective tasks independently. And $\mathcal{X}_1$ with $\mathcal{X}_2$, and $\mathcal{Y}_1$ with $\mathcal{Y}_2$ are without any relation.

(b) In transfer learning, h_S in “Source model” can be transferred to h_T in “Target model”, while $\mathcal{X}_S$ with $\mathcal{X}_T$, and $\mathcal{Y}_S$ with $\mathcal{Y}_T$ are different but related.

Thereby it is able to reduce sample annotation cost and shorten training time for target model.



Working Principle

Source model (SM) and target model (TM) can be represented as 5-tuple respectively:

$$\text{SM} = \langle \mathcal{X}_S, \mathcal{Y}_S, S_S, H_S, \mathcal{L}_S \rangle \,, \quad \text{and} \quad \text{TM} = \langle \mathcal{X}_T, \mathcal{Y}_T, S_T, H_T, \mathcal{L}_T \rangle \,.$$

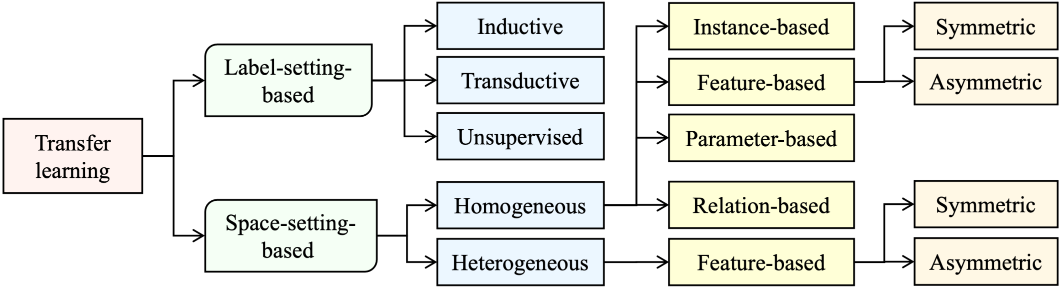
Where: $\mathcal{X}_S$ with $\mathcal{X}_T$, $\mathcal{Y}_S$ with $\mathcal{Y}_T$ are input, output spaces, $S_S = \{(\boldsymbol{x}_i, y_i) \mid i = 1, \ldots, n_S\}$ with $S_T = \{(\boldsymbol{x}_i, y_i) \mid i = 1, \ldots, n_T\}$ are training samples but $n_T \ll n_S$, H_S with H_S are hypothesis sets, and $\mathcal{L}_S$ with $\mathcal{L}_T$ are loss functions.

Transfer learning is to transfer the knowledge of source model to a different but related target model, can be formally expressed as:

$$\mathcal{X}_S \neq \mathcal{X}_T \;\text{ but }\; \mathcal{X}_S \sim \mathcal{X}_T, \quad \text{and} \quad \mathcal{Y}_S \neq \mathcal{Y}_T \;\text{ but }\; \mathcal{Y}_S \sim \mathcal{Y}_T.$$

The most common approach is to transfer $h_S \in H_S$ to $h_T \in H_T$.

Categorization



Two categories of transfer learning:

(1) Label-setting-based: inductive, transductive, and unsupervised transfer learning.

(2) Space-setting-based: homogeneous and heterogeneous transfer learning.

About the Self-Supervised Learning

- Supervised learning often requires thousands or tens of thousands of training samples, and these training samples are often manually annotated.
- Some fields, such as medical images, require domain experts to annotate, so the annotation cost is laborious and expensive, becoming the bottleneck of supervised learning.
- Especially some data in the real world are almost impossible to annotate, that is, they cannot be labeled.
- Humans mainly acquire knowledge through self-study. The self-supervised learning, therefore came into being.

Definition

Definition: (Self-Supervised Learning)

Self-supervised learning (SSL) is a quasi-paradigm of machine learning that automatically extracts *pseudo-labels* from raw data, and then uses them for supervised learning in the next stage.

- The “pseudo-labels” are different from “true labels”:

  pseudo-labels: automatically extracted in the pretext stage;
  true labels: manually annotated by professionals.

- The “pseudo-labels”can be seen as the “representation” of original data, and therefore can also be referred to as self-supervised representation learning.

Related Discourses

Self-Supervised Learning is Key to Human-Level Intelligence

> "Turing Award recipients Yann LeCun and Yoshua Bengio say that self-supervised learning could lead to the creation of artificial intelligence (AI) programs that are more humanlike in their reasoning."

https://cacmb4.acm.org/news/244720-yann-lecun-yoshua-bengio-self-supervised-learning-is-key-to-human-level-intelligence/fulltext

Self-supervised learning: The dark matter of intelligence

> "We believe that self-supervised learning (SSL) is one of the most promising ways to build such background knowledge and approximate a form of common sense in AI systems."

https://ai.meta.com/blog/self-supervised-learning-the-dark-matter-of-intelligence/

Typical Methods

Generative:

The generative model-based self-supervised learning. The generative models adopted include autoregressive models, flow-based models, and autoencoders.

Contrastive:

The contrastive learning model-based self-supervised learning. It uses positive instances and negative instances to train the model, and its loss function is to minimize the distance between positive instances while maximizing the distance between negative instances. The contrastive models adopted include context-instance contrast, and instance-instance contrast.

Typical Methods

Adversarial:

The adversarial model-based self-supervised learning. The approaches adopted include adversarial self-supervised contrastive learning, adversarial self-supervised learning for semi-supervised learning, and self-supervised learning based on adversarial enhanced neural networks.

General framework:

The unified self-supervised learning framework that can support multiple modalities. Such as: the general framework for self-supervised learning—Data2vec, which can be used for computer vision, speech, and natural language processing; and the upgraded version of this framework—Data2vec 2.0.

About the n-Shot Learning

- The ability to quickly learn object classification from a few samples has been confirmed in humans, who can also use existing knowledge to distinguish some new object categories without any training samples.
- A child around the age of 6 has basically learned the tens of thousands of object categories that exist in the world. This is because the child possesses the remarkable ability to rapidly learn new visual concepts by observing only one or a few visual instances.
- The n-shot learning is an umbrella term that covers *few-shot* learning, *one-shot* learning, and even *zero-shot* learning.
- Since n-shot learning does not require a lot of manual annotations, it has expanded from computer vision to other fields.

Related Subtasks

One-Shot vs. Few-Shot vs. Zero-Shot Learning

Subtasks	Brief Statements
One-shot learning (OSL)	It aims to learn some information from one, or only a few, training data.
Few-shot learning (FSL)	It aims to learn some information from a very small amount of training data.
Zero-shot learning (ZSL)	It is able to solve a task despite not having received any training examples of that task.

Working Principle

Without loss of generality, n-shot learning (NSL) can be represented as the 2-tuple:

$$\text{NSL}_n = \langle S_m, H \rangle .$$

Where, S_m is a set of labeled samples with m training data, $S_m = \{(\boldsymbol{x}_i, y_i) \mid i = 1, \ldots, m\}$, and H denote the hypothesis set.

Our aim is to find a hypothesis $h \in H$, so that we can use $h(\boldsymbol{x}) = y$ for n-shot learning.

Therefore, the relationship of zero-shot learning (ZSL), one-shot learning (OSL), and few-shot learning (FSL) is as follows:

- If n is “zero” and $m = 0$, it is zero-shot learning $\text{NSL}_{\text{zero}} = \text{ZSL}$.
- If n is “one” and $m = 1$, it is one-shot learning $\text{NSL}_{\text{one}} = \text{OSL}$.
- If n is “few” and m is very small, it is few-shot learning $\text{NSL}_{\text{few}} = \text{FSL}$.

Case Study of One-Shot Learning

We select a paper published in *Science* in Dec. 2015, titled “Human-Level Concept Learning Through Probabilistic Program Induction”.

This paper addresses the learning problem of one-shot classification.

Supervised learning often requires a large amount of training data, yet humans can learn rich concepts from limited data.

The framework proposed in this paper can learn many visual concepts from a single example and generalize them in a way that is almost indistinguishable from humans.

It combines three ideas: compositionality, causality, and learning to learn. These ideas are very important in cognitive science and machine learning.

Case Study of One-Shot Learning

The Bayesian program learning learns concepts through simple random programs.

(a) Start with primitive tokens.
(b) Combine them to sub-parts
(c) Synthesize parts
(d) Generate object templates with relations
(e) Combine and generate new token exemplars
(f) Finally render them as raw token data.

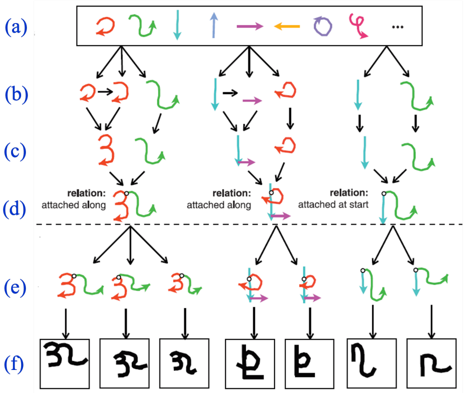


Image Source: https://www.science.org/doi/10.1126/science.aab3050

Case Study of One-Shot Learning

To compare Bayesian program learning with several deep learning models and humans, this paper proposes the following two tasks:

(1) The one-shot classification, i.e., given a character image, human testers and each machine learning model are required to select the same type of image from 20 images;

(2) Generating new samples, including standard, dynamic, and new concept samples.

This paper also proposes several visual Turing tests to detect the creative generalization abilities of the model.

The results show that in many cases, the model is almost indistinguishable from human behavior.

Thank You