Data Optimization in Deep Learning: A Survey

Ou Wu, Rujing Yao

Abstract—Large-scale, high-quality data are considered an essential factor for the successful application of many deep learning techniques. Meanwhile, numerous real-world deep learning tasks still have to contend with the lack of sufficient amounts of high-quality data. Additionally, issues such as model robustness, fairness, and trustworthiness are also closely related to training data. Consequently, a huge number of studies in the existing literature have focused on the data aspect in deep learning tasks. Some typical data optimization techniques include data augmentation, logit perturbation, sample weighting, and data condensation. These techniques usually come from different deep learning divisions and their theoretical inspirations or heuristic motivations may seem unrelated to each other. This study aims to organize a wide range of existing data optimization methodologies for deep learning from the previous literature, and makes the effort to construct a comprehensive taxonomy for them. The constructed taxonomy considers the diversity of split dimensions, and deep sub-taxonomies are constructed for each dimension. On the basis of the taxonomy, connections among the extensive data optimization methods for deep learning are built in terms of four aspects. We probe into rendering several promising and interesting future directions. The constructed taxonomy and the revealed connections will enlighten the better understanding of existing methods and the design of novel data optimization techniques. Furthermore, our aspiration for this survey is to promote data optimization as an independent subdivision of deep learning. A curated, up-to-date list of resources related to data optimization in deep learning is available at https://github.com/YaoRujing/Data-Optimization.

Index Terms—Deep learning, data optimization, data augmentation, sample weighting, data perturbation.

I. INTRODUCTION

DEEP learning has received increasing attention in both the AI community and many application domains due to its superior performance in various machine-learning tasks in recent years. A successful application of deep learning cannot leave the main factors, which include a properly designed deep neural network (DNN), a set of high-quality training data, and a well-suited learning strategy (e.g., initialization schemes for hyper-parameters). Among the main factors, training data is of great importance and it usually plays a decisive role in the entire training process [1]. The concept of data-centric AI is rising, which breaks away from the widespread model-centric perspective [2]. Large models like GPT-4 show significant potential in the direction of achieving general artificial intelligence (AGI). It is widely accepted that the training for large models requires a huge size of high-quality training data.

However, most real applications lack ideal training data. Real training data usually encounters one or several of the

Ou Wu is with National Center for Applied Mathematics, Tianjin University, Tianjin, China, 300072. Rujing Yao is with Department of Information Resources Management, Nankai University, Tianjin, China, 300071. E-mail: wuou@tju.edu.cn, rjyao@mail.nankai.edu.cn.
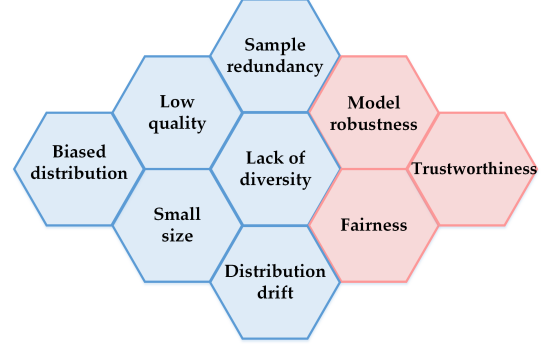

Fig. 1. Nine issues around real training data.

nine common issues as shown in Fig. 1. The following six issues are directly related to training data:

(1) Biased distribution: This issue denotes that the distribution of the training data does not conform to the true distribution of the data in a learning task. One typical bias is class imbalance, in which the proportions of different categories in the training data are not identical due to reasons such as data collection difficulties, whereas the proportions of different categories in test data are identical.

(2) Low quality: This issue corresponds to at least two scenarios. The first refers to data noise that either partial training samples or partial training labels contain noises. As for sample noises, partial samples themselves are corrupted by noises. Taking optical character recognition (OCR) for example, some scanned images may contain serious background noises. The second typical case occurs in multi-model/multi-view learning scenarios. Inconsistency and information missing may exist [3], [4]. For instance, the text title for an image may be mistakenly provided, or it may contain limited words with little information.

(3) Small size: The training size surely impacts the training performance [5]. The larger the training data, the better the training performance usually being attained. Due to insufficient data collection budget or technique limitation, the training data will be relatively small for real use. Therefore, learning under small-size training data is a serious concern in deep learning. This study does not discuss the extreme cases of small size, such as few/one/zero-shot learning.

(4) Sample redundancy: Even though large training data is expected, it does not mean that every datum is useful. There are still learning tasks that the training set contains redundant data [6]. Two typical cases exist. First, the training size is relatively large and exceeds the processing capacity of the available computing hardware. Second,

some regions of training samples may be sampled excessively, and the deletion of such excessive training samples does not affect the training performance. In this case, sample redundancy may occur in certain subsets of some categories.

(5) Lack of diversity: This issue refers to the fact that some attributes for certain categories concentrate excessively in the training corpus. Data diversity is also crucial for DNN training [7]. Taking object classification as an example, the backgrounds in images of the dog category may usually be green grass. However, the "dog" category is not necessarily related to green grass. The lack of diversity in some non-essential attributes can lead to a spurious correlation between some non-essential attributes and the category. This issue is similar to the second case of sample redundancy. Nevertheless, lack of diversity does not necessarily imply the presence of redundant samples.

(6) Distribution drift: This issue denotes that the distribution of the involved data varies over time. Indeed, distribution drift may occur in most real learning applications, as either the concept or the form (e.g., object appearances, text styles) of samples varies fast or slow. Concept drift [8] is the research focus in distribution drift.

The above summary of data issues is not mutually exclusive, as there are overlaps among different issues. For example, small size may only occur in several categories, which can also be attributed to a type of biased distribution. Besides these data issues, there are also some other (not exhaustive) issues closely related to the training data:

(7) Model robustness: This issue concerns the resistance ability of a DNN model to adversarial attacks [9]. Model robustness is highly important for applications related to health, finance, and human life. If DNN models for these applications are compromised by adversarial attacks, serious consequences may ensue.

(8) Fairness: This issue concerns the performance differences among different categories or different attributes in a learning task [10]. For example, the recognition accuracy of faces in different color groups should be at the same level.

(9) Trustworthiness. This issue has emerged in recent years as deep learning has been gradually applied in many safety-critical applications such as autonomous driving and medical assistance [11]. This issue is closely related to robustness and fairness. It mainly refers to the explainability and calibration of DNN models.

To address the above-mentioned issues, numerous theoretical explorations have been conducted and tremendous new methodologies have been proposed in previous literature. Most of these existing methods directly optimize the involved data in learning rather than explore new DNN structures, which is referred to as **data optimization** for deep learning in this paper. As the listed issues belong to different machine learning divisions, the inspirations and focuses of these methods are usually distinct and seem unrelated to each other. For instance, the primary learning strategy for imbalanced learning (belonging to the biased distribution issue) is sample weighting which assigns different weights to training samples in deep learning training epochs. The primary manipulation for the small-size issue is to employ the data augmentation technique such as image resize and mixup [12] for image classification. When dealing with label noise in deep learning, one strategy is to identify noisy labels and then remove them during training. In cases where training data for certain categories lack sufficient diversity, causal learning is employed to break down the spurious correlations among labels and some irrelevant attributes such as certain backgrounds. Due to the apparent lack of connection, these studies typically do not mutually cite or discuss each other.

Our previous study [13] partially reveals that one technique, namely, data perturbation, has been leveraged to deal with most aforementioned issues. This observation illuminates us to explore the data optimization methodologies for those issues in a more broad view. In this study, a comprehensive review for a wide range of data optimization methods is conducted. First, a systematic data optimization taxonomy is established in terms of eight dimensions, including pipeline, object, technical path, and so on. Second, the intrinsic connections among some classical methods are explored according to four aspects, including data perception, application scenario, similarity/opposite, and theory. Third, theoretical studies are summarized for the existing data optimization techniques. Lastly, several future directions are presented according to our analysis.

The differences between our survey and existing surveys in relevant areas, including imbalanced learning, noisy-label learning, data augmentation, adversarial training, and distillation, lie in two aspects. First, this survey takes a data-centric view for studies from a wide range of distinct deep learning realms. Therefore, our focus is merely on the data optimization studies for the listed issues. Methods that do not belong to data optimization for the listed issues are not referred to in this study. Second, the split dimensions (e.g., data perception and theory) which facilitate the establishment of connections among seemingly unrelated methods are considered in our taxonomy. These dimensions are usually not referred to in the existing surveys.

The contributions of this study are summarized as follows.

- Methodologies related to data enhancement for dealing with distinct deep learning issues are reviewed with a new taxonomy. To our knowledge, this is the first work that aims to construct a data-centric taxonomy focusing on data optimization across multiple deep learning divisions and applications.
- The connections among many seemingly unrelated methods are built according to our constructed taxonomy. The connections can inspire researchers to design more potential new techniques.
- Theoretical studies for data optimization are summarized and interesting future directions are discussed.

This paper is organized as follows. Section II introduces main survey studies related to data optimization techniques. Section III describes the main framework of our constructed data optimization taxonomy. Sections IV, V, VI, and VII introduce the details of our taxonomy. Section VIII explores the connections among different data optimization techniques.

Section IX presents several future directions, and conclusions are presented in Section X.

II. RELATED STUDIES

The issues listed in the previous section gradually spawn numerous independent research realms of deep learning. Subsequently, there have been many survey studies conducted for these issues. The following introduces related surveys in several typical research topics.

Imbalanced learning. It is a hot research area in deep learning [14]. He and Garcia [15] conducted the first comprehensive yet deep survey study on imbalanced learning. They explored the intrinsic characteristics of learning tasks incurred by imbalanced data. It is noteworthy that He and Garcia pointed out that an imbalanced dataset is "a high-complexity dataset with both between-class and within-class imbalances, multiple concepts, overlapping, noise, and a lack of representative data". This statement refers to most data issues listed in Section I. For instance, the lack of diversity and sample redundancy can be considered as a lack of representativeness. Recent studies have focused on the extreme case of imbalanced learning, namely, long-tailed classification. Zhang et al. [16] summarized the recent developments in deep long-tailed classification. In their constructed taxonomy, module improvement such as a new classifier is listed as one of the three main techniques. In this study, module improvement is not considered, as it does not fall under data optimization.

Noisy-label learning. This is another research area gaining tremendous attention in recent years as label noise is nearly unavoidable in real learning tasks. Algan and Ulisory [17] summarized the methods in noisy-label learning for image classification. Song et al. [18] elaborately designed taxonomy for noisy-label learning along with three categories, including "data", "objective", and "optimization". Their taxonomy facilitates the understanding of a huge number of existing techniques. Nevertheless, overlap exists between the three dimensions. For example, reweighting locates in the "objective" category, whereas learning to reweight locates in the "optimization" category. The taxonomy introduces in this study may aid the construction a more appropriate taxonomy for noisy-label learning.

Learning with small data. Big data has achieved great success in deep learning tasks. Meanwhile, many real learning tasks still confront with the challenge of small-size training data. Cao et al. [19] performed rigorous theoretical analysis for the generalization error and label complexity of learning on small data. They categorized the small-data learning methods into those with the Euclidean or non-Euclidean mean representation. Wang et al. [20] constructed a few-shot learning taxonomy with three folds, including "data", "model", and "algorithm". Data-centric learning methods are also among the primary choices for few-shot learning.

Concept drift. Lu et al. [8] investigated the learning for concept drift under three components, including concept drift detection, concept drift understanding, and concept drift adaptation. Yuan et al. [21] divided existing studies into two categories, namely, model parameter updating and model structure updating in concept drift adaptation. This division is from the viewpoint of the model. Indeed, pure data-based strategy is also employed in learning under concept drift. For example, Diez-Olivan et al. [22] leveraged data augmentation to fine-tune the last layer of DNNs for quickly concept drift adaptation. This study may motivate researchers on distribution drift to focus more on data optimization manners. Some early surveys can be found in [23], [24].

Adversarial robustness. In many studies, model robustness is limited to adversarial robustness. Silva and Najafirad [25] explored challenges and future directions for model robustness in deep learning. They divided existing adversarial robust learning methods into three categories, including adversarial training, regularization, and certified defenses. Xu et al. [26] summarized the studies for model robustness on graphs. Goyal et al. [27] reviewed the adversarial defense and robustness in the NLP community. Their constructed taxonomy contains four categories, including adversarial training, perturbation control, certification, and miscellaneous. In this study, adversarial training is taken as a data optimization strategy, as it enhances the training set by adding or virtually adding new data.

Fairness-aware learning. It receives increasingly attention in recent years. Mehrabi et al. [28] explored different sources of biases that can affect the fairness of learning models. They revealed that each of the three factors, namely, data, learning algorithms, and involved users may result in bias. Petrović et al. [29] pointed out that sample reweighting and adversarial training are two common strategies for fair machine learning.

Trustworthy learning. It is the key of trustworthy AI, which aims to ensure that an AI system is worthy of being trusted. Trust is a complex phenomenon [30] highly related to fairness, explainability, reliability, etc. Kaur et al. [31] summarized studies on trustworthy artificial intelligence in a quite broad view. Wu et al. [32] provided an in-depth review for studies about trustworthy learning on graphs.

There are also studies that focus on learning tasks with more than one of the listed data issues. For example, Fang et al. [33] addressed noisy-label learning under the long-tailed distributions of training data. Singh et al. [34] conducted an empirical study concerning fairness, adversarial robustness, and concept drift, simultaneously. To our knowledge, no survey study pays attention to the intersection of the research areas related to the listed issues. The unified taxonomy constructed in this survey would enlighten the study on the intersection of multiple research areas.

The most similar study to this work is the survey presented by Wan et al. [35], which focuses on data optimization in computer vision. There are significant differences between our and Wan et al.'s study. First, the covered technical scopes of ours are much broader than those of Wan et al.'s study. Their study limits the scope merely in data selection, including resampling, subset selection, and active learning-based selection. Nevertheless, perturbation, weighting, dataset distillation, and augmentation which attempt to optimize the training data without modifying the backbone network are considered as the data optimization in this study. Second, the split dimensions of ours are quite different from those in Wan et al.'s study
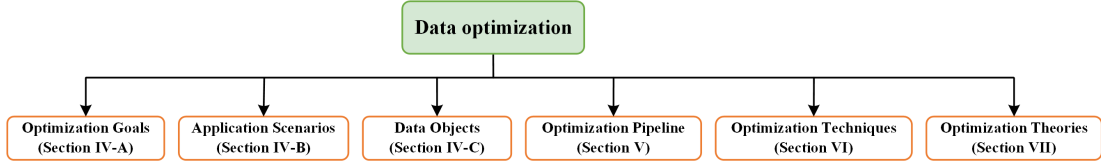
Fig. 2. The six split dimensions of our constructed taxonomy for data optimization.
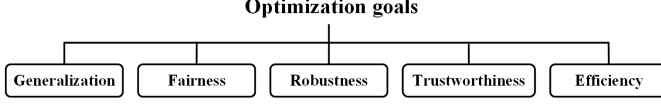

Fig. 3. The sub-taxonomy for optimization goals.

for the overlapped methods. For example, curriculum learning is divided into the resampling category, whereas it is divided into the weighting category in our study. Dataset distillation is merged into one division, namely, data pruning. In contrast, dataset distillation is not included in Wan et al.'s study. Lastly, additional important parts including data perception, connections among different paths, and theoretical investigation are introduced and discussed in this study. Zheng et al. [36] holds a data-centric perspective to review studies on graph machine learning, which is also similar in spirit with our study. Most data issues summarized in this study are also discussed in their study. They divided existing studies into graph data collection, enhancement, exploration, maintenance (for privacy and security), and graph operations, which are not applicable for our taxonomy.

It is noteworthy that classical data pre-processing methodologies such as data cleaning (e.g., missing data imputation), standardization (e.g., z-score), and transformation (e.g., data discretization) also aim to make data better for learning. Considering that these methodologies are mature and mainly utilized in shallow learning, they are not introduced in this survey.

III. OVERALL OF THE PROPOSED TAXONOMY

To ensure our constructed taxonomy well organized and comprehensive coverage on previous data optimization techniques about the issues listed in Section I as much as possible, the following principles for the design of split dimensions are considered.

(1) The first layer of the taxonomy should consider multiple views, with each view corresponding to a sub-taxonomy. Most existing taxonomies for specific research realms adopt only a single view. In this study, only a single view is inadequate for systematically arranging studies from various deep learning realms.

(2) The dividing dimension should be general so as to embrace existing studies as much as possible. Therefore, the dimensions designed in existing taxonomies for specific research areas should not be directly followed. A new comprehensive taxonomy is required.

(3) The new taxonomy should be compatible with existing taxonomies. That is, inconsistency between our and existing taxonomies is allowed. However, contradiction between them should be avoided.

On the basis of these principles, the first layer[1] of our taxonomy is designed as shown in Fig. fig2. This layer consists of six dimensions for data optimization as follows:

- Ultimate goals. This dimension refers to the final goal of a data optimization method used in a deep learning task. We divide the optimization goals into five main aspects[2], including generalization, robustness, fairness, trustworthy, and efficiency.
- Application scenarios. This dimension refers to the deep learning applications that utilize data optimization. Nine applications are involved, including learning under biased distribution, noisy-label learning, learning with redundant training data, learning with limited training data, model safety, fairness-aware learning, learning under distribution drift, trustworthy learning, and learning for large models.
- Data objects. This dimension refers to the objects to optimize in the employed data optimization method. Most studies focus on the raw training data. There are also methods concentrating in other data objects such as hyper-parameters and meta data.
- Optimization pipeline. This dimension refers to the common steps for a concrete data optimization method for a deep learning procedure. We divide the pipeline into three common steps, namely, data perception, analysis, and optimizing.
- Optimization techniques. This dimension refers to the employed technique paths in data optimization. This study summarizes five main paths, namely, data resampling, data augmentation, data perturbation, data weighting, and dataset pruning. Each path also contains sub-divisions. The introduction for this part is the focus of this survey.
- Optimization theories. This dimension refers to the theoretical analysis and exploration for data optimization in deep learning. We divided this dimension into two aspects: formulation and explanation.

Section IV introduces the ultimate goals, application scenarios, and data objects. Sections V, VI, and VII introduces the optimization pipeline, technique, and theory, respectively.

IV. GOALS, SCENARIOS, AND DATA OBJECTS

This section introduces ultimate goals, targeted applications, and data objects.

A. Optimization goals

Fig. 14 describes the sub-taxonomy for the dimension of optimization goals, including generalization, fairness, robustness, trustworthiness, and efficiency.

---

[1]The fine-granularity layers are detailed in the succeeding sections.

[2]It should be noted that these five aspects are not exhaustive and there are overlaps among them as revealed by the previous literature.

Generalization is the primary optimization goal in most data optimization techniques, as it is almost the sole goal in most deep learning tasks. According to the generalization theory studied in shallow learning, generalization of a category is highly related to class margin, inter-class distance, and class compactness [37]. A larger margin/larger inter-class distance/higher class compactness of a category indicates a better generalization performance on the learned model on the category. The data augmentation strategy that injects noise to training samples is proven to increase the generalization [38]. The implicit data augmentation method ISDA [39] actually aims to improve the class compactness³ of each category. Adaptive margin loss [40] also aims to improve the class compactness by perturbing the logits. Fujii et al. [41] modified the classical data augmentation method mixup [12] by considering the "between-class distance", which finally increases the inter-class distance. In addition, some studies explore the compiling of an optimal batch in the training process of deep learning [42]. The ultimate goal is also the generalization. Nevertheless, the direct goal of the batch compiling may consist of balance, diversity, and others.

As previously stated, fairness is also an important learning goal in many deep learning tasks. To combat unfairness on samples with certain attributes, techniques such as data augmentation [43], perturbation [44], and sample weighting [45] have been used in previous literature. Indeed, imbalanced learning also pursues fairness among different categories. A category with a small prior probability, denoted as a minor category, will receive more attention in the employed data optimization. For example, larger weights [46], larger degrees of data perturbation [47], or more augmented quantities [48] are exerted on minor categories than others.

Adversarial robustness is an essential goal in deep learning tasks that are quite sensitive to model safety. Adversarial training is usually leveraged to improve the adversarial robustness of a model. It can be attributed to a special type of data augmentation. Thus, adversarial training is actually a data optimization technique, which aims to improve the quality of training data such that the models trained on the optimized training data have better adversarial robustness.

Trustworthiness is a goal that has recently been highly valued. Explainability and calibration are two crucial requirements for the trustworthiness of a deep learning model. Explainability mainly relies on methodologies such as feature attribution and causal reasoning rather than pure data optimization technique. Nevertheless, data optimization is widely used in model calibration. Calibration mainly concerns the trustworthiness of the predicted probability of a probabilistic model [49]. Liu et al. [50] introduced margin-aware label smoothing to improve the calibration of trained models. Mukhoti et al. [51] leveraged sample weighting to achieve better calibration.

Efficiency is crucial for real applications as many learning tasks are sensitive to both time complexity and storage. Therefore, how to optimally reduce the redundant training

---

³Some methods such as center loss also aim to increase the class compactness. These methods are considered not data optimization.
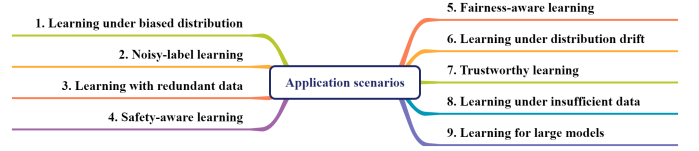

Fig. 4. The sub-taxonomy for targeted application scenarios.

data and remain the diverse important training data deserves further investigation. The time complexity can be significantly reduced after data pruning.

B. Application scenarios

Fig. fig4 describes the sub-taxonomy for the dimension of targeted application scenarios. The first eight scenarios have been referred to in previous sections, so they are not further introduced in this subsection.

Learning under insufficient data contains the case that the training data are not as diverse as possible. Data diversity affects the model generalization [7]. Dunlap et al. [52] utilized large vision and language models to automatically generate visually consistent yet significantly diversified training data. Some studies [53], [54] consider that data augmentation is actually a widely-used technique to increase data diversity. These studies develop new data augmentation methods for deep learning tasks.

Large models, such as large language models (LLMs), have made remarkable advancements in nearly each AI field. The data quality is crucial for the training or fine-tuning of a large model. Therefore, data optimization techniques also prevail in learning for large models. Yang et al. [55] utilized flip operation on the training corpus to balance the two-way translation in language pairs in their building of a large multilingual translation model. Liu et al. [56] applied adversarial training in both the pre-training and fine-tuning stages. Results show that the error rate of the trained model is also reduced.
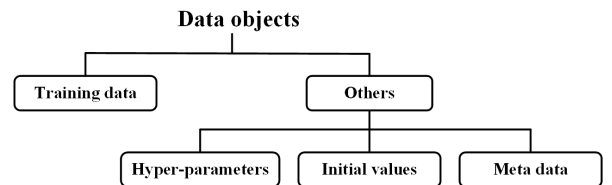

Fig. 5. The sub-taxonomy for data objects.

C. Data objects

1) Primary objects: The primary objects of data optimization for deep learning are the training data. Some studies optimize raw samples, while some others optimize labels. There are also studies focusing on the data transformed by DNNs, e.g., features and logits. In Sections VI-B and VI-C, more details will be presented.

2) Other objects: There are also numerous studies concerning other data objects in deep learning. Fig. 5 lists three other data objects, namely, hyper-parameters, initial values, and meta data.

Hyper-parameters highly affect the final performance of trained models. They are determined either by grid searching

in a pre-defined scope or directly being set as fixed values. Consequently, setting a proper searching scope or fixed initial values is a crucial step in DNN training.

Network initialization is also important for DNN training. Gaussian distribution-based initialization is the primary choice in most learning tasks. Other effective strategies are also investigated and applied. Glorot and Bengio [57] adopted a scaled uniform distribution for initialization which is called "Xavier" initialization. He et al. [58] proposed a robust initialization method for rectifier nonlinearities, which is called "Kaiming" initialization.

Meta learning offers a powerful manner to optimize hyperparameters of independent modules in deep learning. It relies on an unbiased meta dataset. Nevertheless, in most learning tasks there are no independent high-quality meta data and constructing a high-quality unbiased meta dataset is challenging. Su et al. [59] conducted a theoretical analysis for the compiling of a high-quality meta dataset from the training set. Four criteria, namely, balance, uncertainty, clean, and diversity, are selected in their proposed compiling method.

There are numerous classical studies for the optimization of the three types of data which are not mentioned in this study. The placing of these studies into data optimization for deep learning may facilitate the further development of the optimization for the three types of data. The focus of this survey is the training data. Therefore, the following parts will be limited in the scope of training data optimization.
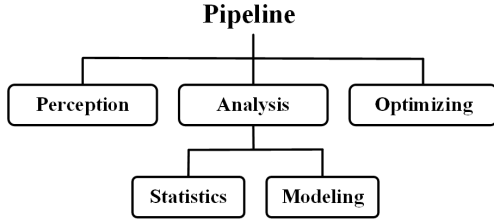

Fig. 6. Three main steps in data optimization pipeline.

V. OPTIMIZATION PIPELINE

The pipeline mainly consists of three steps, namely, data perception, analysis, and optimizing, as shown in Fig. 6. Some notations and symbols are defined as follows. Let $D = \{x_i, y_i\}_{i=1}^N$ be a set of N training samples, where x_i is the feature and y_i is the label. Let C be the number of categories and N_c be the number of the samples in the cth category in D. $\pi_c = N_c/N$ is the proportion of the cth category. Let p_c and $p(x|y = c)$ be the prior and the class conditional probability density for the cth class, respectively. When there is no ambiguity, x_i represents the feature output by the last feature encoding layer and u_i represents the logit vector output by the Softmax layer for x_i. Let l be the loss for x. In this study, the cross-entropy loss is assumed and Θ represents the network parameters.

A. Data perception

In this study, data perception refers to all possible methods aimed at sensing and diagnosing the training data to capture
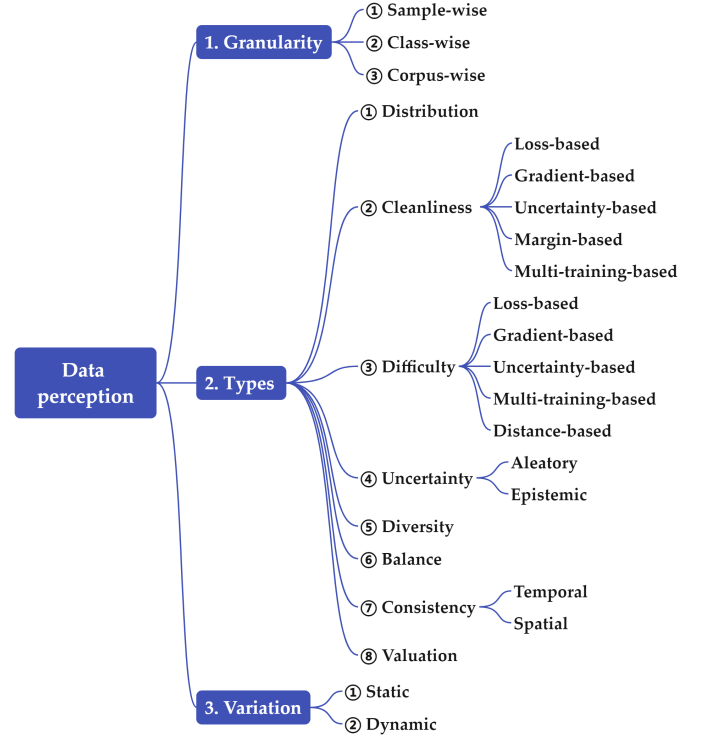

Fig. 7. The sub-taxonomy for data perception.

the intrinsic data characteristics and patterns that affect learning performance. It serves as the first step in the pipeline, and an effective data optimization method cannot work well without accurate perception of the training corpus

Generally, data perception for training data quantifies the factors related to the true distribution, training data distribution, cleanliness, diversity, etc. We construct a sub-taxonomy for data perception in three dimensions as shown in Fig. 7. First, in terms of quantifying granularity, there are three levels, namely, sample-wise, category-wise, and corpus-wise. Secondly, in terms of perception types, there are eight divisions, namely, distribution, cleanliness, difficulty, diversity, balance, consistency, neighborhood, and valuation. Thirdly, in terms of quantifying variation, there are two divisions, namely, static and dynamics. Each of the above divisions and partial representative studies are introduced as follows.

1) Perception on different granularity levels: There are three granularity levels, including sample-wise, category-wise, and corpus-wise.

Sample-wise data perception. It denotes that the perceived quantities reflect or influence a sample's positive/negative or trivial/important role in training. For example, most noisy-label learning methods employ sample-wise data perception, e.g., training loss [60] and gradient norm [61], to infer the noisy degree of a training sample.

Category-wise data perception. It denotes that the perceived quantities reflect or influence a category's positive/negative or trivial/important role in training. In category-wise perception, the learning performance of each category is usually monitored to return feedback for the entire scheme [62]. Therefore, the average outcome (e.g., average

loss or precision) is also used to infer a reasonable category-wise weight in the next training epoch [63], [64]. Another popular quantity is the category proportion (π_c) for imbalanced learning. Some studies [65] measure the compactness of a category as it reflects the generalization of the features for a category. Studies on/leveraging category-wise perception are fewer than sample-wise studies.

Corpus-wise data perception. It denotes that the perceived quantities reflect or influence a training corpus' positive/negative or trivial/important role in training. Limited studies fall into this division. Lin et al. [66] used the query score to measure the utility of a training dataset.

These three levels can be used together to more comprehensively perceive the training data [67].

2) Perception on different types: The eight quantifying types are introduced as follows:

- Distribution. This type aims to quantify the true data distribution for a learning task and the training data distribution. An effective quantification of these two distributions is significantly beneficial for training. Nevertheless, it is nearly impossible to obtain a clear picture of them. Therefore, the true distribution is usually assumed to conform to several some basic assumptions, such as Gaussian distribution for each category [39]. For the training data distribution, some studies [68], [69] apply clustering to deduce the intrinsic structure of the training data. These studies concern the global distribution of a category. Recently, researchers have investigated local distributions of training samples. One typical characteristic is about the neighborhood of each training sample. In deep learning on graphs, the distribution of neighborhood samples with heterogeneous labels negatively impacts the training or prediction for the sample. Wang et al. [70] defined a label difference index to quantify the difference between a node and its neighborhood in a graph as follows:

$$LDI(x_i) = \frac{1}{\sqrt{2}}||p_{x_i} - p_{N_i}||_2, \tag{1}$$

where p_{x_i} and p_{N_i} are the category distributions of x_i and its neighborhood N_i.

- Cleanliness. This type aims to identify the degree of noise in each training sample. This study primarily focuses on label noise, as it garners more attention than sample noise. There are numerous metrics for noise measurement. As illustrated in Fig. 7, typical measures include loss-based, gradient-based, uncertainty-based, margin-based, and multi-training-based techniques. Samples with large losses, gradient norms, or uncertainties are more likely to be noisy. In margin-based measures, a small margin indicates a high probability of being noise. Huang et al. [60] conducted multiple training procedures to identify noisy labels.

- Difficulty. This type aims to infer the degree of learning difficulty for a training sample or a category. The accurate measurement of learning difficulty for each training sample is of great importance because several deep learning paradigms employ adaptive learning strategies based on the level of learning difficulty. For instance,

curriculum learning [71] holds the perspective that easy samples should receive more focus in the early training stages, while hard samples should be given more attention in the later stages of training. Some other studies [72] hold the opposite perspective that hard samples should be prioritized throughout the training procedure. As shown in Fig. 7, there are five major manners to measure learning difficulty of samples, namely, loss-based, gradient-based, uncertainty-based, multi-training-based, and distance-based. Obviously, the measures for learning difficulty are quite similar to those for cleanliness. In fact, some studies consider that noisy samples are those quite difficult to learn and divide samples into easy/medium/hard/noisy. Paul et al. [73] proposed the error l2-norm score to measure difficulty. Zhu et al. [74] established a formal definition for learning difficulty of samples inspired by the bias-variance trade-off theorem and proposed a new learning difficulty measures. Sorscher et al. [75] defined the cosine distance of a sample to its nearest cluster center as the sample's difficulty measure and applied it in sample selection.

- Uncertainty. This type contains two sub-types, namely, aleatory uncertainty and epistemic uncertainty [76]. The former is also called data uncertainty and occurs when training samples are imperfect, e.g., noisy. Therefore, the cleanliness degree can be used as a measure of data uncertainty [77]. Epistemic uncertainty is also called model uncertainty. It appears when the learning strategy is imperfect. Model uncertainty can be calculated based on information entropy of the DNN prediction or the variance of multiple predictions output by a DNN with the dropout trick [78].

- Diversity. This type aims to identify the diversity of a subset of training samples. The subset is usually a category. The measurement for subset diversity is useful in the design of data augmentation strategy for the subset [79] and data selection [59]. Friedman and Dieng [80] leveraged the exponential of the Shannon entropy of the eigenvalues of a similarity matrix, namely, vendi score to measure diversity. Salimans et al. [81] utilized a pre-trained Inception model to measure diversity called inception score.

- Balance. This type aims to measure the balance between/within categories. The balance between categories belongs to global balance, while that within a category belongs to local balance. Global balance can be simply measured by the proportion of the training sample of a category. Nevertheless, our previous study [82] reveals that other factors such as variance and distance may also result in serious imbalance. Local balance is relatively difficult to measure. Some studies define local balance as attribute balance [68].

- Consistency. This type aims to identify the consistency of the training dynamics of a training sample along the temporal or spatial dimensions. In the temporal dimension, the variations of the training dynamics between the previous and the current epochs are recorded [83]. In the spatial dimension, the differences in the training

dynamics between a sample and other samples such as neighbors [84] or samples within the same category are recorded. A classical measure called "forgetting" [85] quantifies the number of variations in the prediction between adjacent epochs. Singh et al. [86] investigated class-wise forgetting. Maini et al. [87] utilized forgetting to distinguish among examples that are hard for distinct reasons, such as membership in a rare subpopulation, being mislabeled, or belonging to a complex subpopulation. Wang et al. [88] provided a comprehensive summary for sample forgetting in learning. Kim et al. [89] focused on the dynamics of each sample's latent representation and measured the alignment between the latent distributions.

- Valuation. This value is usually measured by the Shapley value, which is a concept from the game theory [90]. Ghorbani and Zou firstly introduced Shapley value for data valuation [91] as follows:

$$\phi(x_i) = \sum_{S \in D - x_i} \frac{1}{C_{|D|-1}^{|S|}} [V(S \cup \{x_i\}) - V(S)] \quad (2)$$

where $V(\cdot)$ is the utility function of a dataset and S is a subset of the training corpus D. Their values are different when modeling the clean and the noisy samples. Nevertheless, the calculation for the Shapley value as shown in Eq. (2) is NP-hard, thereby hindering its use in real applications. Yoon et al. [92] proposed a reinforcement learning-based method for data valuation. Their inferred weights reflect the importance of a sample in learning, which is not equal to the Shapley value. Some other studies [66], [93] proposed more practical methods to approximate the Shapley value. Jiang et al. [94] established an easy-to-use and unified framework that facilitates researchers and practitioners to apply and compare existing data valuation algorithms. Compared with the aforementioned perception quantities such as cleanliness and difficulty, the Shapley value has a more solid theoretical basis. Therefore, establishing a direct connection among the Shapley value and the quantities listed above deserves further study.

This study only lists commonly used measures for data perception. Additionally, there are some other important measures which will be explored in our future work. For example, the neighborhoods for each training sample may vary as the feature encoding network is updated at each epoch. In shallow learning, neighborhood is an important information and many classical methods are based on the utilization of neighborhood. However, previous deep learning methodologies rarely leverage neighborhood information, as the computational complexity for neighborhood identification is high. Some studies adopt a simplified manner to construct the neighborhood. For example, Bahri and Heinrich Jiang [95] employed the logit vectors rather than the features to construct neighborhood. The dimension of logit vector is usually much smaller than the feature dimension, so the computational complexity is significantly reduced. It is believable that with the advancement of related computational techniques, neighborhood information will receive increasingly attention in deep learning. Some other
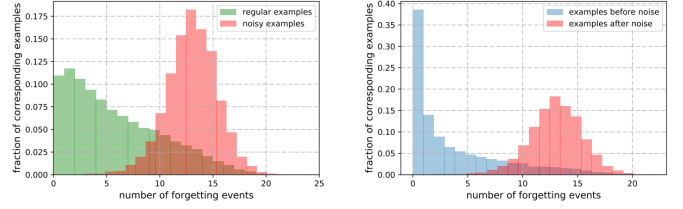

Fig. 8. The statistics for the forgetting numbers for training samples [85].

important quantities such as problematic score [96] and data influence [97] in learning, which have large overlaps with the aforementioned quantities, also deserve further exploration.

If the perceived quantities are required to fed into a model, hidden representation for the raw quantities is favored. In meta learning-based sample weighting or perturbation [98], the weights or perturbation vectors for each sample are derived based on the hidden representations of the the perceived quantities. For example, Shu et al. [99] extracted the training loss for each sample as the input and fed it into an MLP network containing 100 hidden nodes. In other words, the raw training loss is represented by a 100-dimensional hidden vector. Zhou et al. [67] utilized six quantities to perceive the character of a training sample, including loss, margin, gradient norm, entropy of the Sofxmax prediction, class proportion, and average categorical loss. Likewise, these six quantities are also transformed into a 100-dimensional feature vector through an MLP network.

3) Static and dynamic perception: Static perception denotes that the perceived quantities remain unchanged during optimization, whereas dynamic perception denotes that the quantities vary.

In imbalanced learning, category proportion is widely used to quantify a category. It belongs to static perception because this quantity remains unchanged. In noisy-label learning, many studies adopt a two-stage strategy in which the noisy degree of each training sample is measured and the degrees are used in the second training stage [60]. In this two-stage strategy, the perception for label noise is static.

The impact of a training sample usually varies during training. Therefore, compared with static perception, dynamic perception is more prevailing in deep learning tasks. Many studies utilize training dynamics of training samples for the successive sample weighting or perturbation. Such training dynamics also belong to the dynamic perception. The training dynamics including loss, prediction, uncertainty, margin, and neighborhood vary at each epoch. For example, self-paced learning [100] determines the weights of each training sample according to their losses in the previous epoch and a varied threshold. Therefore, the weight may also vary in each epoch.

B. Analysis on perceived quantities

Analysis on perceived quantities contains two manners, namely, statistics and modeling, as shown in Fig. 6.

Statistical analysis. Most studies employ this manner for the perceived data quantities. These studies considered only one or two quantities. For example, Toneva et al. [85] made a statistics for the forgetting numbers of training samples
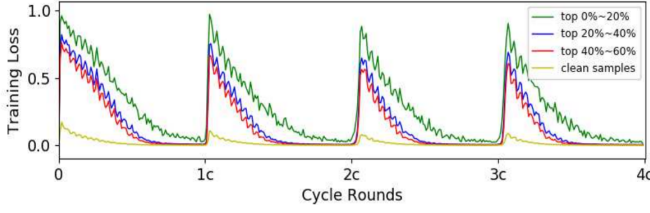
Fig. 9. The statistics for loss along the training cycle rounds [60].

(a) Bias of pair-flip-type label noisy samples

(b) Variance of pair-flip-type label noisy samples

(c) Bias of symmetric-type label noisy samples

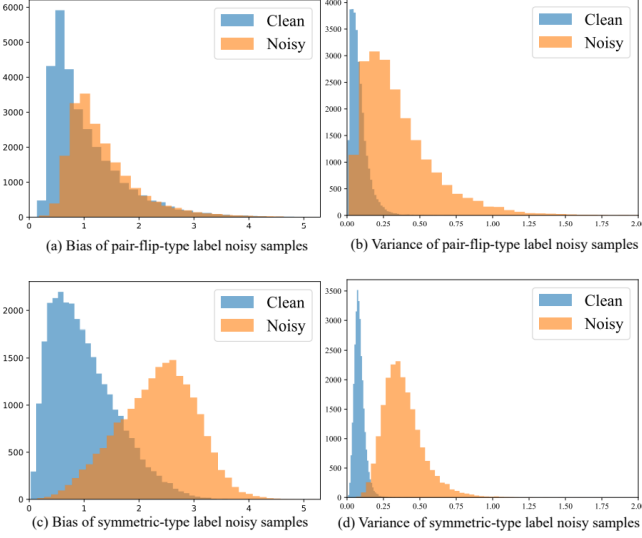(d) Variance of symmetric-type label noisy samples

Fig. 10. The statistics for the mean and the variance of the loss under the cross validation-based training. [74].

as shown in Fig. 8. The left figure shows the distributions of forgetting numbers for clean and noisy samples, while the right one shows the distributions of forgetting numbers before and after the noise is added. Distinct difference exists between the distributions of clean and noisy samples. Huang et al. [60] proposed a cycle training strategy that the model is trained from overfitting to underfitting cyclically. The epoch-wise loss for each training sample is recorded. Fig. 9 shows the differences between the average losses for the noisy and clean samples. Noisy samples have larger training losses. Therefore, they leveraged the average loss as an indicator for noisy labels. Zhu et al. [74] proposed a cross validation-based training strategy. Multiple training losses are also recorded for each training sample. They revealed that the variance of the multiple losses for each sample is also useful in identifying noisy labels as shown in Fig. 10.

Modeling. This manner refers to the statistical modeling on the perceived quantities for training data. Arazo et al. [101] assumed that the traning loss conforms to the following distribution:

$$p(l|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} l^{\alpha-1}(1-l)^{\beta-1}. \tag{3}$$

where $\Gamma(\cdot)$ is the Gamma function; α and β are parameters to infer. Their values are different when modeling the clean and the noisy samples. Hu et al. [102] leveraged the Weibull mixture distribution to model the memorization-forgetting value of each training sample. The mixture distribution contains two components which suit for the clean and noisy samples, respectively.

Both manners are transparent and thus the entire data optimization approach is explainable. Nevertheless, these two divisions usually rely on appropriate prior distributions about the involved quantities. If the prior distributions are incorrect, the successive optimizing will negatively influence the model training.

C. Optimizing

The data perception and analysis act as the pre-processing for data operation. This step is the key processing of the entire data optimization pipeline. The successive section will introduce current optimization techniques in detail.

VI. DATA OPTIMIZATION TECHNIQUES

This section describes the most important dimension for the presented taxonomy, namely, data optimization techniques for deep learning. Fig. 11 presents the sub-taxonomy along this dimension. We summarized six sub-divisions for existing data optimization techniques, including resampling, augmentation, perturbation, weighting, pruning, and others. It is noteworthy that this survey covers numerous technique/methodology divisions and leaves a through comparison for them as our future work. The reason lies in two folds. First, each division has its own merits and defects and their effectiveness have been verified in previous literature, so it is difficulty to judge which one is absolutely the best in universal learning tasks. Second, a thorough theoretical or empirical comparison is not a trivial task.

A. Data resampling

Data resampling compiles a new training set in which samples are randomly sampled from the raw training set. It is widely used in tasks encountering the issues, including biased distribution [103] and redundancy. This study summarizes two split dimensions for this division. The first dimension concerns the size of the sampled datasets, while the second dimension concerns the sampling rates.

In the first dimension, resampling is divided into under-sampling and oversampling. undersampling compiles a new training set whose size is smaller than that of the raw training set. Contrarily, oversampling compiles a new training set whose size is larger than that of the raw training set. Both manners are widely used in previous machine learning tasks, including imbalanced learning, bagging, and cost-sensitive learning. Meanwhile, tremendous theoretical studies have been conducted to explain the effectiveness of these two manners in both the statistics and the machine learning communities. Nevertheless, there is currently no consensus on which manner is more effective. Some studies concluded that undersampling should be the primary choice when dealing with imbalanced datasets [104]. However, some other studies hold the opposite view [105].

In the second dimension, resampling is divided into uniform, proportion-based, importance-based, learning difficulty-based, and uncertainty-based. They are detailed as follows:

- Uniform sampling. This manner is quite intuitive. It treats samples definitely equal regardless of their distributions,
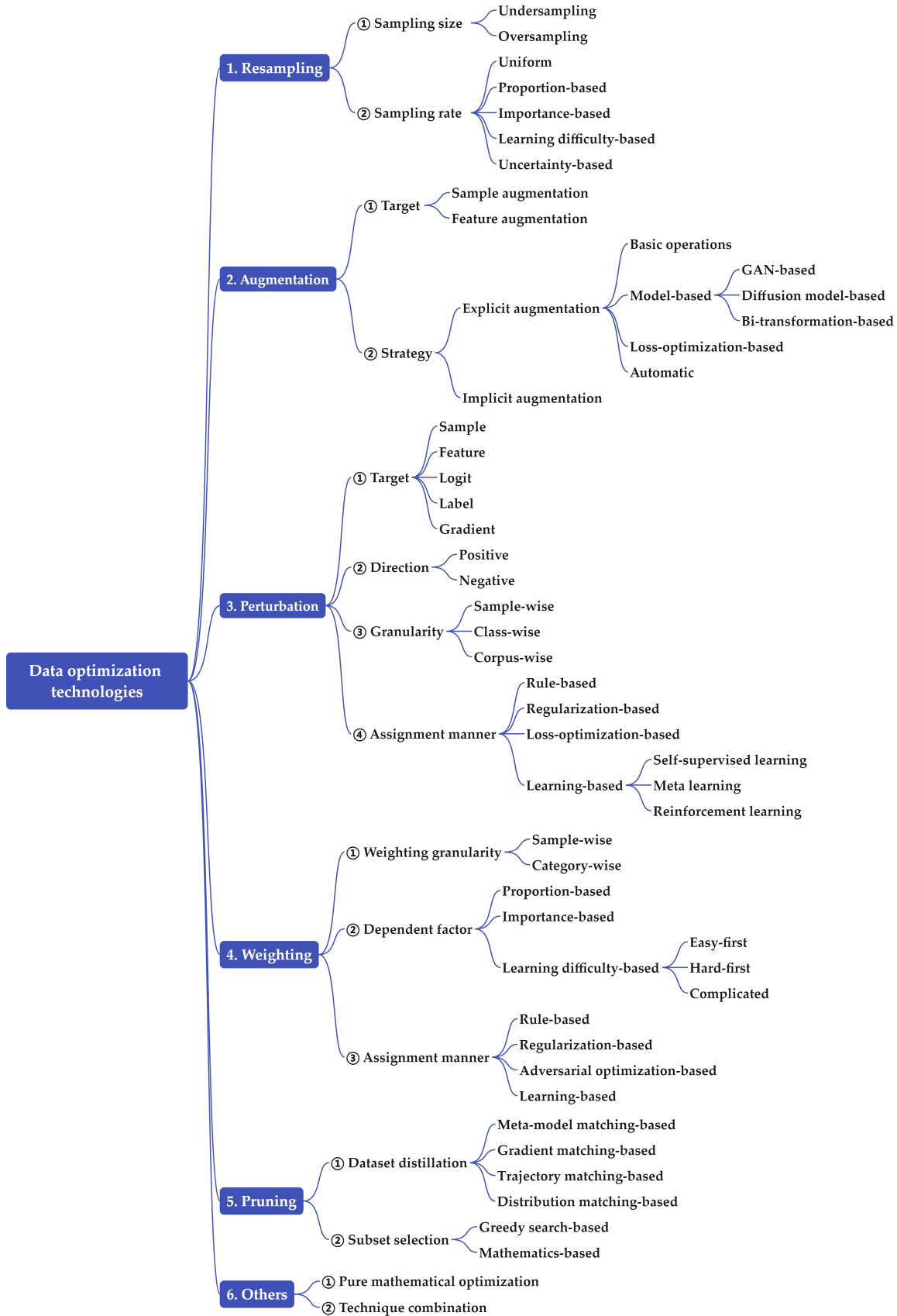
Fig. 11. The sub-taxonomy of data optimization techniques.

location, categories, and training performances. In fact, in nearly all existing deep learning tasks, the batch is constructed by uniformly sampling from the training corpus. Kirsch et al. [106] claimed that the independent selection of a batch of samples leads to data inefficiency due to the correlations between samples. Some studies explore alternative sampling strategies. For example, Loshchilov and Hutter [42] proposed a rank-based batch selection strategy according to the training loss in previous epochs and samples with large losses have high probabilities to be sampled. Experiments reveal that this new strategy accelerates the training speed by a factor of five.

- Proportion-based sampling. This manner simply assigns the total sampling rate for each category with its proportion (π_c) in the corpus. It is mainly used in imbalanced learning in which the minor categories are assigned with large sampling rates [15].

- Importance-based sampling. This manner assigns sampling probabilities according to samples' importance. In this study, the definition for importance sampling follows several classical studies [107] [108]. Given a target distribution $q(x, y)$ and a source distribution on training data $p(x, y)$, the importance (sampling rate) for a training sample $\{x, y\}$ in importance sampling is defined as

$$w(x) = \frac{q(x, y)}{p(x, y)}. \qquad (4)$$

As the target distribution is unknown, some studies [109] utilize the kernel trick to generate sampling rates. In some importance sampling studies, the sampling rates are not based on the probability density ration as presented in Eq. (4). For example, Atharopoulos and Fleuret [110] took the gradient norms of each training sample as their importance. These methods actually belong to learning difficulty-based sampling.

- Learning difficulty-based sampling. This manner assigns sampling rates according to samples' learning difficulties. As summarized in Section V-A2, learning difficulty is usually measured by loss and gradient norm. For instance, Li et al. [61] applied the gradient norm of logit vectors as the difficulty measurement. Similar with the study conducted by Atharopoulos and Fleuret [110], Johnson and Guestrin [111] proposed the O-SGD sampling method with the following sampling rate:

$$w(x, y) = \frac{||\nabla l(x, y)||}{\sum_x ||\nabla l(x, y)||}. \qquad (5)$$

They claimed that this "importance sampling" can reduce the stochastic gradient's variance and thus accelerate the training speed. Jiang et al. [112] introduced a selective back propagation strategy, in which samples with large losses have relative large probabilities to be selected in back propagation. Gui et al. [113] utilized sampling strategy for noisy-label learning. They calculated the sampling weights based on the mean loss of each example along the training process. The training samples with large mean losses are assigned low weights. Liu et al. [114] proposed adaptive data sampling, in which the sampling rates of the samples correctly classified in previous epochs are reduced. Xu et al. [115] conducted a theoretical analysis and concluded that inverse margin-based sampling may accelerate gradient descent in finite-step optimization by matching weights with the inverse margin.

- Uncertainty-based sampling. This manner assigns sampling rates according to samples' uncertainties. It is widely used in active learning, in which a subset of data is sampled for human labeling [116], [117]. Aljuhani et al. [118] presented an uncertainty-aware sampling framework for robust histopathology image analysis. The uncertainty is calculated by predictive entropy.

There are also some other sampling manners. For instance, Ting and Brochu [119] calculated the sample influence for optimal data sampling. Li and Vasconcelos [120] proposed the adversarial sampling to improve OOD detection performance of an image classifier. In their adversarial sampling, the sampling weights are pursued by maximizing the OOD loss. Wang and Wang [121] sampled sentences according to their semantic characteristics. Zhang et al. [122] sampled training data of the majority categories by considering the samples' sensitivities. Samples with low sensitivities may be noisy or safe ones, while those with high sensitivities are borderline ones. Sun et al. [123] explored an automatic scheme for effective data resampling.

B. Data augmentation

Data augmentation compiles a new training set in which samples (or features) are generated based on the raw training set or sometimes other relevant sets. It is a powerful tool to improve the generalization capability [124], [125] and even adversarial robustness [126], [127] of DNNs. Illuminated by related surveys on data augmentation [128]–[131], two split dimensions are considered, namely, sample/feature and explicit/implicit, as shown in Fig. 11.

1) Sample/feature augmentation: In sample augmentation, the new training set consists of generated new samples, while in feature augmentation, the new training set consists of generated new features.

Sample augmentation. This division is subject to data types (e.g., image, text, or others). For image corpus, augmentation methods adopt noise adding, color transformation, geometric transformation, or other basic operations such as cropping to augment new images [129]. For texts, new samples can be generated by noise adding, paraphrasing, or other basic operations such as word swapping [132].

Feature augmentation. This division is performed on the feature space, so learning tasks for different data types may utilize the same or similar augmentation strategies. Some intuitive feature augmentation methods include adding noise, interpolating, or extrapolating [133], which are applicable for general data types, including both image and text data. Li et al. [134] revealed that the simply perturbing the feature embedding with Gaussian noise in training leads to comparable domain-generalization performance compared with the SOTA methods. Ye et al. [135] proposed novel domain-agnostic augmentation strategies on feature space. Cui et al. [136]

decomposed features into the class-generic and the class-specific components. They generated samples by combining these two components for minor categories. A classical robust learning paradigm, namely, adversarial training, is actually a feature-wise augmentation strategy when it is run on the feature space [137]–[139].

Some studies augment other data targets such as label and gradient. For example, Lee et al. [140] rotated a training image and the rotation angle is also used as supervised information. Elezi et al. [141] proposed a transductive label augmentation method to generate labels for unlabeled large set using graph transduction techniques. Some other studies [142] investigated gradient augmentation. Compared with sample/feature augmentation, label/gradient augmentation receives quite limited attention.

2) Explicit/implicit augmentation: Explicit augmentation directly generates new samples/features. Meanwhile, implicit augmentation conducts data augmentation only theoretically yet do not generate any new samples/features actually.

Explicit augmentation. According to the employed techniques, existing explicit data augmentation can be divided into basic operation, model-based (GAN, diffusion model), loss-optimization-based, and automatic augmentation, as described in Fig. 11. They are introduced as follows:

- Basic operations. This technique is widely used in practical learning tasks as basic operations conform to human intuitions. The popular deep learning platforms such as pyTorch provide several common basic operations such as cropping, rotation, replacement, masking, cutout, etc. One of the most popular data augmentation method used for shallow learning tasks, namely, SMOTE [143] has been utilized in deep learning tasks [144]. Dablain et al. [145] designed more sophisticated improvement for SMOTE for deep learning tasks. Among the basic operations, mixup is a simple yet quite effective augmentation manner [12], [146]. It generates a new sample with a new label that does not belong to the raw label space.
- Model-based augmentation. This technique generates new samples by leveraging independent models. There are three main schemes:
  ① GAN-based scheme. Generative adversarial network (GAN) trains a generative model and a discriminative model simultaneously in a well designed two-player min-max game [147]. The trained generative model can be used to generate new samples conforming to the distribution of the involved training data. A large number of variations have been designed in the previous literature [148]. Mariani et al. [149] proposed balancing GAN for imbalanced learning tasks. Huang et al. [150] developed AugGAN for the data augmentation in cross domain adaptation. Yang et al. [151] investigated the GAN-based augmentation for time series.
  ② Diffusion model-based scheme. Diffusion models are a new class of generative models and achieve SOTA performance in many applications [152]. Xiao et al. [153] leveraged a text-to-image stable diffusion model to expand the training set. Dunlap et al. [52] utilized large vision and language models to automatically generate natural language descriptions of a dataset's domains and augment the training data via language-guided image editing.
  ③ Bi-transformation-based scheme. This scheme usually relies on two transformation models. The first model transforms a training sample into a new type of data. The second model transforms the new type of data into a new sample. In natural language processing (NLP), back-translation is a popular data augmentation technique [154], which translates the raw text sample into new texts in another language and back translates the new texts into a new sample in the same language with the raw sample. Dong et al. [155] proposed a new augmentation technique called Image-Text-Image (I2T2I) which integrates text-to-image and image-to-text (image captioning) models. There are also augmentation attempts about Text-Image-Text (T2I2T) [156] and Text-Text-Image (T2T2I) [157]. Theoretically, tri-transformation-based augmentation may be also applicable. We leave it our future work.
- Loss-optimization-based augmentation. This manner generates new sample/features by minimizing or maximizing a defined loss with heuristic or theoretical inspirations. Adversarial training is a typical loss-optimization-based manner. It generates a new sample for x by solving the following optimization problem:

$$x_{\text{adv}} = x + \arg \max_{\|\boldsymbol{\delta}\| \leq \epsilon} \ell(f(x + \boldsymbol{\delta}), y), \qquad (6)$$

where $\boldsymbol{\delta}$ and ϵ are the perturbation term and bound, respectively. Zhou et al. [67] proposed anti-adversaries by solving the following optimization problem:

$$x_{\text{anti-adv}} = x + \arg \min_{\|\boldsymbol{\delta}\| \leq \epsilon} \ell(f(x + \boldsymbol{\delta}), y). \qquad (7)$$

Pagliardini et al. [158] obtained new samples by maximizing an uncertainty-based loss.
- Automatic augmentation. This manner investigates automated data augmentation techniques [159] based on meta learning [160] or reinforcement learning [161]. Nishi et al. [162] proposed new automated data augmentation method and validated its usefulness in noisy-label learning. Some studies focus on differentiable automatic data augmentation which can dramatically reduces the computational complexity of existing methods [163].

Implicit augmentation. Wang et al. [39] proposed the first implicit augmentation method called ISDA. It establishes a Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_y, \Sigma_y)$ for each category. New samples can be generated (i.e., sampled) from its corresponding Gaussian distribution. An upper bound of the loss with augmented samples can then be derived when the number of generated samples for each training sample approaches to infinity. Finally, the upper bound of the loss is used for the final training loss. There are several variations for ISDA, such as IRDA [164] and ICDA [165]. Li et al. [166] also proposed an implicit data augmentation approach mainly based on heuristic inspirations.

Explicit augmentation is the primary choice in data augmentation tasks. Nevertheless, implicit augmentation is more efficient than explicit augmentation as it does not actually generate new samples or features. There are also theoretical studies about data augmentation. A mainstream perspective is that data augmentation performs regularization in training [167]–[170]. Chen et al. [171] conducted a probabilistic analysis and concluded that data augmentation can result in variance reduction and thus prevent overfitting according to the bias-variance theory. In fact, the regularizer in machine learning also reduces model variance.

C. Data perturbation

Given a datum x (x can be the raw sample, feature, logit, label, or others), data perturbation will generate a perturbation $\triangle x$ such that $x' = x + \triangle x$ can replace x or be used as a new datum. Therefore, some data augmentation methods, such as adversarial perturbation, cropping, and masking, can also be viewed as data perturbation. In our previous work [13], we constructed a taxonomy for compensation learning, which is actually learning with perturbation. This study follows our previous taxonomy in [13] with slight improvements. The sub-taxonomy for data perturbation is presented in Fig. 11. Four split dimensions are considered, namely, target, direction, granularity, and assignment manner.

1) Perturbation target: The perturbation targets can be raw sample, feature, logit vector, label, and gradient.

- Sample perturbation. This division adds the perturbation directly to the raw samples. The basic operations in data augmentation can be placed into this division. For instance, noise addition and masking used in image classification actually exert a small perturbation on the raw image.
- Feature perturbation. This division adds the perturbation on the hidden features. Jeddi et al. [172] perturbed the feature space at each layer to increase uncertainty in the network. Their perturbation conforms to the Gaussian distribution. Shu et al. [173] designed a single network layer that can generate worst-case feature perturbations during training to improve the robustness of DNNs.
- Logit perturbation. This division adds the perturbation on the logit vectors in the involved DNNs. Li et al. [47] analyzed several classical learning methods such as logit adjustment [174], LDAM [175], and ISDA [39] in a unified logit perturbation viewpoint. They proposed a new logit perturbation method and extended it to the multi-label learning tasks [176].
- Label perturbation. This division adds the perturbation on either the ground-truth label of the predicted label. One classical learning skill, namely, label smoothing [177], is a kind of label perturbation method. Let C be the number of categories and λ be a hyper-parameter. Label smoothing perturbs the label y (one-hot type) with the following perturbation $\triangle y = \lambda(\frac{I}{C} - y)$, where I is a C-dimensional vector and its each element is 1. A large number of variations have been proposed for label smoothing [62], [178], [179].

- Gradient perturbation. This division adds the perturbation directly on gradient. Studies on gradient perturbation are few. Orvieto et al. [180] proposed a gradient perturbation method and verified its effectiveness both theoretically and empirically.

There are also studies [181] which perturb other data such as network weights in training, which is not the focus of this study. Wang et al. [182] proposed a reward perturbation method for noisy reinforcement learning.

2) Perturbation direction: Data perturbation will either increase or decrease the loss values of training samples in the learning process. Based on whether the loss increases or decreases, existing methods can be categorized as positive or negative augmentations.

Positive perturbation. It increases the training losses of perturbed training samples. Obviously, adversarial perturbation belongs to positive perturbation, as it maximizes the training loss with the adversarial perturbations. ISDA [39] also belongs to positive perturbation as it adds positive real numbers to the denominator of the Softmax function.

Negative perturbation. It reduces the training losses. Anti-adversarial perturbation [67] belongs to negative perturbation, as it minimizes the training loss with the adversarial perturbations. Bootstrapping [183] is a typical robust loss based on label perturbation. It also belongs to negative perturbation as its perturbation is $\triangle y = \lambda(p - y)$, where p is the prediction of the current trained model.

Some methods increase the losses of some samples and decrease those of others simultaneously. For instance, the losses of noisy-label training samples may be reduced, while those of clean samples may be increased in label smoothing. Li et al. [47] proposed a conjecture for the relationship between loss increment/decrement and data augmentation.

3) Perturbation granularity: According to perturbation granularity, existing methods can be divided into sample-wise, class-wise, and corpus-wise.

- Sample-wise perturbation. In this division, each training sample has its own perturbation and different samples usually have distinct perturbations. The aforementioned Bootstrapping and adversarial perturbation all belong to this division. The random cropping and masking also belong to this division.
- Class-wise perturbation. In this division, all the training samples in a category share the same perturbation, and different categories usually have distinct perturbations. Benz et al. [184] proposed a class-wise adversarial perturbation method. Wang et al. [185] introduced class-wise logit perturbation for the training in semantic segmentation. Label smoothing also belongs to this division.
- Corpus-wise perturbation. In this division, all the training samples in the training corpus share only one perturbation. Shafahi et al. [186] pursued the universal adversarial perturbation for all the training samples, which has proven to be effective in various applications [187]. Wu et al. [188] proposed a corpus-wise logit perturbation method for multi-label learning tasks.

4) Assignment manner: The perturbation variables should be assigned before or during training. As presented in Fig. 11,

there are four typical assignment manners to determine the perturbations.

Rule-based assignment. In this manner, the perturbation is assigned according to pre-fixed rules. These rules are usually based on prior knowledge or statistical inspirations. In both label smoothing and Boostrapping loss, the label perturbation is determined according to manually defined formulas. In text classification, word replacement and random masking also obey rules.

Regularization-based assignment. In this manner, a regularizer for the perturbation is usually added in the total loss. Take the logit perturbation as an example. A loss function with regularization for logit perturbation can be defined as follows:

$$\mathcal{L} = \sum_i l(\mathcal{S}(v_i + \triangle v_i), y_i) + \lambda Reg(\triangle v_i). \tag{8}$$

where $\mathcal{S}$ is the Softmax function, v_i is the logit vector for x_i, $\triangle v_i$ is the perturbation vector for v_i, and $Reg(\cdot)$ is the regularizer. Zhou et al. [189] introduced a novel perturbation way for adversarial examples by leveraging smoothing regularization on adversarial perturbations. Wei et al. [190] took the notation that adversarial perturbations are temporally sparse for videos and then proposed a sparse-regularized adversarial perturbation method. Zhu et al. [191] proposed a Bayesian neural network with non-zero mean of Gaussian noise. The mean is actually a feature perturbation and inferred with l_2 regularization.

Loss-optimization-based assignment. This division is similar to the loss-optimization-based augmentation introduced in Section VI-B2. A new loss containing the perturbations is defined and the perturbation is pursued by optimizing the loss. In the optimization procedure, only the perturbations are the variables to be optimized, while the model parameters are fixed.

Learning-based assignment. In this manner, the perturbation is assigned by leveraging a learning method. Three learning paradigms are usually applied, including self-supervised learning, meta learning, and reinforcement learning.

- Self-supervised learning. This paradigm leverages self-supervised learning methodologies such as contrastive learning [192] to pursue the perturbations. Naseer et al. [193] constructed a self-supervised perturbation framework to optimize the feature distortion for a training image. Zhang et al. [194] proposed a generative adversarial network-based self-supervised method to generate EEG signals.

- Meta learning. This paradigm leverages meta-learning methodologies to pursue the perturbations using an additional meta dataset. It assumes that the perturbation $\triangle x$ (or $\triangle y$) for a training sample x (or its label y) is determined by the representation of x or factors such as training dynamics for x, which is described as follows:

$$\triangle x = g(x, \eta(x)), \tag{9}$$

where $g(\cdot)$ can be a black-box neural network such as MLP; $\eta(x)$ represents the training dynamics for x. Li et al. [195] applied meta learning to directly optimize the covariant matrix used in ISDA, which is used to calculate

the logit perturbation. Qiao and Peng [196] utilized the meta learning to learn an independent DNN for both features and label perturbation.

- Reinforcement learning. This paradigm leverages reinforcement learning to pursue the perturbations without relying on additional data. Many data augmentation methods [197], [198], which also belong to data perturbation, are based on reinforcement learning. Giovanni et al. [199] leveraged deep reinforcement learning to automatically generate realistic attack samples that can evade detection and train producing hardened models. Lin et al. [200] formulated the perturbation generation as a Markov decision process and optimized it by reinforcement learning to generate perturbed instructions sequentially.

Given a learning task, it is difficulty to directly judge which assignment manner is the most appropriate without a thorough and comprehensive understanding for the task. Each assignment manner has its own merits and defects.

D. Data weighting

Data weighting assigns a weight for each training sample in loss calculation. It is among the most popular data optimization techniques in many learning scenarios, including fraud detection [201], portfolio selection [202], medical diagnosis [203], and fairness-aware learning [45], [204]. Three dividing dimensions are considered, namely, granularity, dependent factor, and assignment manner for weights.

1) Weighting granularity: According to the granularity of weights, existing weighting methods can be divided into sample-wise and category-wise. Noisy-label learning usually adopts sample-wise weighting methods [205], [206], while imbalanced learning usually adopts category-wise ones [46], [207], [208]. Data weighting is also widely used in standard learning [209], [210], which are usually sample-wise.

2) Dependent factor: Dependent factor in this study denotes the factors that are leveraged to calculate the sample weights. Similar with the resampling introduced in Section VI-A, three factor types are usually considered, namely, category proportion, importance, and learning difficulty. As these concepts are introduced in Section VI-A and quite similar procedures are adopted, these factors are not detailed in this part. There are an increasing number of studies employing learning difficulty-based weighting. They can be further summarized according to which samples are learned first.

As samples with larger weights than others can be considered as having priority in training, learning difficulty-based weighting contains three basic folds, namely, easy-first, hard-first, and complicated.

- Easy-first. Easy samples are given higher weights than hard ones in this fold. There are a huge number of easy-first weighting methods, which mainly belong to two paradigms: curriculum learning [71] and self-paced learning [100]. These two paradigms assign larger weights to easy samples during the early training stage and gradually increase the weights of hard samples. Numerous studies have been conducted on the design of the weighting formulas [211]–[213]. Easy-first weighting is usually

used in noisy-label learning. Extensive experiments on curriculum learning indicate that it mainly takes effects on noisy-label learning tasks [214].

- Hard-first. Hard samples have higher weights than easy ones in this fold. Focal loss is a typical hard-first strategy [72]. Zhang et al. [210] also assigned large weights on hard samples. Santiagoa et al. [215] utilized the gradient norm to measure learning difficulty and exerted large weights on samples with large gradient norms.
- Complicated. In some weighting methods, the easy-first or the hard-first is combined with other weighting inspirations. In Balanced CL [216](should be replaced), on the basis of the easy-first mode, the selection of samples has to be balanced under certain constraints to ensure diversity across image regions or categories. Therefore, Balanced CL adopts the complicated mode.

Besides the three general ways, Zhou et al. [217] revealed some other priority types including both-ends-first and varied manners during training. There also other dependent factors such as misclassified cost and those reflecting other concerns such as fairness and confidence [218], [219].

3) Assignment manner: Generally, there are four manners to assign weights for training samples as shown in Fig. 11.

Rule-based assignment. This manner determines the sample weights according to theoretical or heuristic rules. For example, many methods assume that the category proportion is the prior probability. Consequently, the inverse of the category proportion is used as the weight based on the Bayesian rule. Cui et al. [46] established a theoretical framework for weight calculation based on the effective number theory in computation geometry. The classical Focal loss [72] heuristic defines the weight using $w = (1 - p)^\gamma$, where p is the prediction on the ground-truth label and γ is a hyper-parameter. Han et al. [220] defined an uncertainty-based weighting manner for the two random samples in mixup. Importance weighting [221] is also placed in this division.

Regularization-based assignment. This method defines a new loss function which contains a weighted loss and a regularizer ($Reg(W)$) on the weights as follows:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} w_i l(f(x_i), y_i) + \lambda Reg(W), \qquad (10)$$

where $W = \{w_1, \cdots, w_N\}^T$ is the vector of sample weights. The classical self-paced learning, which mimics the mechanism of human learning from easy to hard gradually, is actually the regularization method defined as $Reg(W) = -|W|_1$ ($w_i \in \{0, 1\}$) [100]. Fan et al. [222] presented a new group of self-paced regularizers deduced from robust loss functions and further analyzed the relation between the presented regularizer-based optimization and half-quadratic optimization.

Adversarial optimization-based assignment. This manner pursues the sample weights by optimizing a defined objective function, which is similar to the pursing of the adversarial perturbation. For instance, Gu et al. [223] adversarially learned the weights of source domain samples to align the source and target domain distributions by maximizing the Wasserstein distance. Yi et al. [224] defined a maximal expected loss and

obtained a simple and interpretable closed-form solution for samples' weights: larger weights should be given to augmented samples with large loss values.

Learning-based assignment. Similar with that in data perturbation, learning-based assignment also usually applies meta learning or reinforcement learning to infer the sample weights. Ren et al. [225] firstly introduced meta learning for sample weighting in imbalanced learning and noisy-label learning. Shu et al. [99] utilized an MLP network to model the relationship between samples' characters and their weights, and then trained the network using meta learning. Zhao et al. [226] further proposed a probabilistic formulation for meta learning-based weighting. Trung et al. [227] leveraged meta learning to train a neural network-based self-paced learning for unsupervised domain adaption. Wei et al. [228] also combined meta learning and self-paced network to automatically generate a weighting scheme from data for cross-modal matching. Li et al. [229] proposed meta learning-based weighting for pseudo-labeled target samples in unsupervised domain adaptation. Meta learning requires additional meta data, whereas reinforcement learning does not require additional data. Zhou et al. [230] leveraged an augmentation policy network which takes a transformation and the corresponding augmented image as inputs to generate the loss weight of an augmented. Ge et al. [231] used a delicately designed controller network to generate sample weights and combined the weights with the loss of each input data to train a recommendation system. The controller network is optimized by reinforcement learning.

Weights assignment can also be divided into static and dynamic. There are a few methods adopting static weighting [46], whereas most methods adopting dynamic. Fang et al. [232] proposed dynamic importance weighting to train the models.

E. Data pruning

Data pruning is contrary to data augmentation. In this study, it is divided into dataset distillation and subset selection.

1) Dataset distillation: Dataset distillation is firstly proposed by Wang et al. [233] and it aims to synthesize a small typical training set from substantial data [234]. The synthesized dataset replaces the given dataset for efficient and accurate data-usage for the learning task. Following the division established by Sachdeva and McAuley [235], existing data distillation methods can be placed in four folds.

Meta-model matching-based strategy. This strategy is firstly proposed by Wang et al. [233]. It performs an inner-loop optimization for a temporal optimal model based on the synthesized set and an outer-loop optimization for a temporal subset (i.e., the synthesized set) by turns. Some recent studies discussed its drawbacks such as the ineffectiveness of the TBPTT-based optimization [236] and proposed new solutions such as momentum-based optimizers [237]. Loo et al. [238] utilized the light-weight empirical neural network Gaussian process kernel for the inner-loop optimization and a new loss for outer-loop optimization. Zhou et al. [239] combined feature extractor in the distillation procedure.

Gradient matching-based strategy. This strategy [236], [240] does not require to perform the inner-loop optimization

as used in the meta-model matching-based strategy. Therefore, it is more efficient than the meta-model matching-based strategy. Numerous approaches have been proposed along this division. Kim et al. [241] further utilized spatial redundancy removing to accelerate the optimization process and gradients matching on the original dataset.

Trajectory matching-based strategy. This strategy performs distillation by matching the training trajectories of models trained on the original and the pursued datasets [242]. Cui et al. [243] proposed a memory-efficient method which is avaliable for large datasets.

Distribution matching-based strategy. This strategy performs the distillation by directly matching the distribution of the original dataset and the pursued dataset [244]. Wang et al. [245] constructed a bilevel optimization strategy to jointly optimize a single encoder and summarize data.

There are some solutions [237], [246]–[250] which take alternative technical strategies. Zhou et al. [246] introduced reinforcement learning to solve the bi-level optimization in data distillation. Zhao and Bilen [247] learned a series of low-dimensional codes to generate highly informative images through the GAN generator.

2) Subset selection: Different from dataset distillation that generates a new training set, subset selection aims to select the most useful samples from the original training set [35]. It does not generate new samples and can be used in any training stages that require to select samples from the original training set. In Fig. 11, there are two divisions, including greedy search-based and mathematics-based.

In the greedy search-based strategy, the utility of each training sample is measured, and the subset is searched based on the utility rankings. According to the employed measures, existing methods can be divided into four categories, including difficulty-based, influence-based, value-based, and confidence-based. Meding et al. [251] utilized the misclassified rate by multiple classifiers as the learning difficulty of a training sample to select samples. Feldman and Zhang [252] defined an influence score and a memorization score to measure the usefulness of a training sample. Samples with low influence and memorization scores are redundant and can be deleted. Birodkar et al. [6] employed clustering to select most valuable samples which are close to the cluster centers and delete the rest redundant ones. Northcutt et al. [253] leveraged the confidence score to prune training samples. There are also many studies combining the measures and active learning to select samples [254].

Different from the greedy search strategy, some other methods seek a global optimal subset according to a mathematical approach. Yang et al. [255] proposed a scalable framework to iteratively extract multiple mini-batch coresets from larger random subsets of training data by solving a submodular cover problem. Mirzasoleiman et al. [256] defined a monotonic function for coreset selection and proposed a generic algorithm with approximately linear complexity.

F. Other typical techniques

This study lists two representative technical paths, including pure mathematical optimization and the combination of more

than one aforementioned methods described in Sections VI-A to VI-E.

1) Pure mathematical optimization: This division refers to the manners that perform data optimization via a pure mathematical optimization procedure in the above-mentioned divisions.

The first typical scenario for pure mathematical optimization is the construction of a small-size yet high-quality dataset from the original training set. The tasks involving batch construction, meta data compiling in meta learning, or dataset distillation usually adopt mathematical optimization. Liu et al. [257] constructed a set variance diversity-based objective function for data augmentation and pursued the selection for a set of augmented samples via the maximization of the objective function in batch construction. Joseph et al. [258] proposed a submodular optimization-based method to construct a mini-batch in DNN training. Significant improvements in convergence and accuracy with their constructed mini-batches have been observed. Su et al. [59] established an objective function for meta data compiling. The objective consists of four criterion, including cleanliness, diversity, balance, and informative. As introduced in Section VI-E, data pruning is usually performed based on pure mathematical optimization.

The second typical scenario is the regularized sample weighting or perturbation. The details are described in Sections VI-C4 and VI-D3. For instance, Li et al. [259] devised a new objective function for the label perturbation strength, which can also reduce the Bayes error rate during training. Meister et al. [260] constructed a general form of regularization that can derive a series of label perturbation methods.

The third typical scenario is the constrained optimization, which embeds prior knowledge or conditions in data weighting, perturbation, or pruning into the constraints. For instance, Chai et al. [261] defined an optimization objective function with the constraints that each demographic group should have equal total weights in fairness-aware learning. The adversarial perturbation of multi-label learning is usually attained by solving constrained optimization problems [262]–[264]. Hu et al. [263] developed a novel loss for multi-label top-k attack with the constraints that considers top-k ranking relation among labels.

2) Technique combination: Indeed, many learning algorithms do not employ a single data optimization technique. Instead, they combine different data optimization techniques. The following lists a few combination examples.

In data augmentation, many methods choose to generate samples in the first step and resample or reweight the samples in the second step. For instance, Cao et al. [265] dealt with grammatical error correction by using a data augmentation method during training and a data weighting method to automatically balance the importance of each kind of augmented samples. Liu et al. [266] generated new source phrases from a masked language model then sampled an aligned counterfactual target phrase for neural machine translation. Zang et al. [267] combined data augmentation and resampling for a long-tailed learning task.

In data perturbation, different directions/granularity levels are usually combined in the same method. For example,

adversarial perturbation belongs to the positive direction, while anti-adversarial perturbation belongs to the negative one. Zhao et al. [268] considered both category-wise and sample-wise factors to define the logit perturbation for imbalanced learning. Zhou et al. [67] combined both adversarial and anti-adversarial perturbations and theoretically revealed the superiority of the combination than the adversarial perturbation only.

In data weighting, numerous methods combine it with data augmentation. Han et al. [220] combined uncertainty-based weighting and the classical augmentation method mixup. Chen et al. [164] combined effective number-based weighting and logit perturbation for long-tail learning tasks. In addition, some methods combine different granularity levels or different priority models. For example, Focal loss [72] employs both category-wise and sample-wise weight coefficients for each sample.

VII. Data optimization theories

There are a large amount of studies focusing on the theoretical aspects of data optimization. It is quite challenging to arrange existing theoretical studies into a clear roadmap. This study summarizes existing studies in the following two dimensions, including formalization and explanation.

A. Formalization

In order to theoretically analyze and understand the data optimization methods, it is essential to establish mathematical formulations. Statistical modeling is the primary tool for their formalization [269]–[271]. Basic assumptions are usually relied on. The most widely used assumptions for the statistical modeling include the following.

- Gaussian distribution assumption. Many studies [29], [272]–[274] assume that data in each category conforms to a Gaussian distribution, which simplifies computation and inference compared to other complicated distributions [275].
- Equal class CPD assumption. In many learning studies [276], [277] excepting those for distribution drift, the class-conditional probability densities (CPD) of the training and testing sets are assumed to be identical.
- Uniform distribution assumption. In many studies [175], [278], the distribution over categories in the testing set is assumed to be uniform. Some studies implicitly use this assumption by using modified losses such as the balanced accuracy or balanced test error [174], [279], even if the category proportions in the test corpus are not identical.
- Linear boundary assumption. In many studies [115], [280], the decision boundary of the involved classifier is assumed to be linear. The decision boundary between two categories under the cross-entropy loss is linear.

Based on these assumptions, the data optimization problems are usually formalized into probabilistic, constrained optimization, or regularization-based problems. For example, Xu et al. [281] investigated importance weighting for covariate-shift generalization based on probabilistic analysis. Chen et al. [282] defined the classification accuracy based on posterior probability for zero-shot learning. Qraitem et al. [283] formalized a constrained linear program problem to investigate the effect of data resampling. Roh et al. [284] formulated a combinatorial optimization problem for the unbiased selection of samples in the presence of data corruption. In classical weighting paradigm such as SPL, data weighting is directly formalized in the optimization object consisting of the weighted loss and a regularizer. Zhang et al. [285] defined a re-weighted score function consisting of weighted loss and a sparsity regularization for causal discovery. Jiang et al. [286] proposed a new adversarial perturbation generation method by adding a diversity-based regularization which measures the diversity of candidates. Hounie et al. [287] proposed a constrained learning problem for automatic data augmentation by combining conventional training loss and the constraints for invariance risk. Blum and Stangl [288] investigated the utility of fairness constraints in fair machine learning.

B. Explanation

Most theoretical studies on data optimization aim to explain why the existing methods are effective or ineffective.

In data perception, researchers usually conducted theoretical analysis on the role of one typical data measure or leveraged the measure to understand the training process of DNNs. Doan et al. [289] conducted a theoretical analysis of catastrophic forgetting in continuous learning with neural-tangent-kernel overlap matrix. Chatterjee et al. [290] utilized the perception on gradients to explain the generalization of deep learning.

In data resampling, existing theoretical studies focus on importance sampling for deep learning. Katharopoulos and Fleuret [110] derived an estimator of the variance reduction achieved with importance sampling in deep learning. Katharopoulos and Fleuret [291] theoretically revealed that the loss value can be used as an alternative importance metric, and propose an efficient way to perform importance sampling for a deep model. Wang et al. [292] proposed an unweighted data sub-sampling method, and proved that the subset-model acquired through the method outperforms the full-set-model.

In data augmentation, more and more theoretical studies are performed. Dao et al. [293] established a theoretical framework for understanding data augmentation. According to their framework, data augmentation is approximated by two components, namely, first-order feature averaging and second-order variance regularization. Zhao et al. [125] defined an effective regularization term for adversarial data augmentation and theoretically derived it from the information bottleneck principle. Wu and He [294] investigated the theoretical issues for adversarial perturbations for multi-source domain adaptation. Gilmer et al. [295] also attempted to explain the adversarial samples.

In data perturbation, most theoretical studies focus on the adversarial perturbation. Yi et al. [296] investigated the models trained by adversarial training on OOD data and justified that the input perturbation robust model in pre-training provides an initialization that generalizes well on downstream OOD data. Peck et al. [297] formally characterized adversarial perturbations by deriving lower bounds on the magnitudes of perturbations required to change the classification of neural

TABLE I
SOME DATA OPTIMIZATION METHODS IN NOISY-LABEL LEARNING.

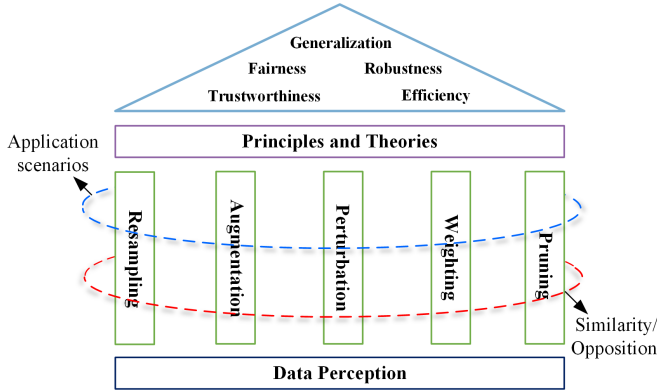| Datasets	Resampling	Augmentation	Perturbation	Weighting	Pruning
CIFAR10	[113], [92], [292]	[162], [305], [306], [307]	[177], [183]	[100], [308], [309], [310]	[311], [312], [313]
CIFAR100	[113], [92], [292]	[162], [305], [306], [307]	[177], [183]	[100], [309], [310]	[312], [313]
Clothing1M	[292]	[162], [307], [305]	[177], [183]	[100], [310]	[313]
SVHN	[120]	[314]	[67], [177], [183]	[100]	[245]
WebVision	[315]	[307], [305]	[84]	[308], [309], [310]	[312]

Fig. 12. High-level connections for existing data optimization studies.

networks. Some studies delved into the theoretical justification for label and logit perturbation. Xu et al. [298] analyzed the convergence SGD with label smoothing regularization and revealed that an appropriate LSR can help to speed up the convergence of SGD. Li et al. [176] theoretically analyzed the usefulness of logit adjustment in dealing with class imbalanced issues.

In data weighting, Byrd and Lipton [108] investigated the role of importance in deep learning. Fang et al. [232] discussed the limitations of importance weighting and found that it suffers from a circular dependency. Meng et al. [299] analyzed the capability of the self-paced learning and provided an insightful interpretation of the effectiveness of several classical SPL variations. Weinshall et al. [300] proved that the rate of convergence of an ideal curriculum learning method is monotonically increasing with the learning difficulty of the training samples.

In data pruning, theoretical studies are relatively limited. Zhu et al. [301] revealed that distilled data lead to networks that are not calibratable. The reason lies in two folds, including a more concentrated distribution of the maximum logits and the loss of information that is semantically meaningful but unrelated to classification tasks. Dong et al. [302] emerged dataset distillation into the privacy community and theoretically revealed the connection between dataset distillation and differential privacy.

There are also studies which aim to reveal the intrinsic connections between two different technical paths. For instance, regularization is a widely used technique in deep learning [169], and several typical data optimization techniques are revealed to be a regularization method [39], [303], [304]. Therefore, intrinsic connections among these techniques can be established, which enlightens a better understanding of the involved technical paths and can envision novel inspirations or methods.

VIII. CONNECTIONS AMONG DIFFERENT TECHNIQUES

The connections among different data optimizations techniques can be described by Fig. 12. Four aspects, namely, perception, application scenarios, similarity/opposition, and theories, connect different methods within a technical path or across different paths.

A. Connections via data perception

Data perception is the first (explicit or implicit) step in the data optimization pipeline. Methods along different technical paths introduced in Section VI may choose the same or similar quantities in perception. Therefore, quantities for data perception connect different methods. For example, many data optimization methods are on the basis of training loss in resampling [320], augmentation [321], perturbation [47], weighting [299], and subset selection [322]. Gradient is widely used in resampling [110], augmentation [323], perturbation [67], weighting [61], and dataset distillation [236]. Other quantities including margin and uncertainty are also used in different optimization techniques.

The utilization of the same or similar perception quantities demonstrates that these methods have the same or similar heuristic observations or theoretical inspirations.

B. Connections via application scenarios

Most data optimization methods can be leveraged for the application scenarios discussed in Section IV-B.

One of the most focused scenarios of data optimization methods is noisy-label learning. Many classical methods are from resampling [324], augmentation, weighting, or perturbation. These are also dataset distillation studies for noisy-label datasets [325]. Table I shows some representative data optimization methods for noisy-label learning on five benchmark datasets CIFAR10 [326], CIFAR100 [326], Clothing1M [327], SVHN [328], and WebVision [329].

Imbalanced learning is also among the most focused scenarios. Nearly all the listed data optimization technical paths have been used in imbalanced learning. Table II shows some representative data optimization methods for imbalanced learning on four benchmark datasets CIFAR10-LT [46], CIFAR100-LT [46], iNaturalist [330], and ImageNet-LT [331]. There are some studies employing more than one type of data optimization techniques such as ReMix [316], which combines resampling and augmentation, in Table II.

Robust learning for adversarial attacks is another typical scenario. Karimireddy and Jaggi [332] employed resampling to design robust model in distributed learning. Data weighting [210] and dataset distillation [333] are also used in robust learning.

TABLE II
SOME DATA OPTIMIZATION METHODS IN IMBALANCED LEARNING.

Datasets	Resampling	Augmentation	Perturbation	Weighting	Dataset pruning
CIFAR10-LT	[316]	[316], [317]	[174], [195]	[72], [46]	[59], [318]
CIFAR100-LT	[175]	[164], [317]	[174], [195]	[72], [46]	[59], [318]
iNaturallist	[175]	[164], [317]	[174], [195]	[72], [46]	[59]
ImageNet-LT	[175]	[164], [317]	[174], [195]	[72], [46]	[59], [319]

C. Connections via similarity/opposition

The similar and opposite relationships existing among the five technical paths are introduced in Section VI.

Data resampling and weighting are closely related techniques, as their key steps are nearly the same. Therefore, in many studies on noisy-label learning and imbalanced learning, these two techniques are often considered as a single strategy.

Although data pruning and augmentation are opposite to each other, they have consistent ultimate goals in learning tasks. They are overlapped in terms of employed methodologies as shown in Fig. 13. It is believable that more intrinsic connections can be explored for them.
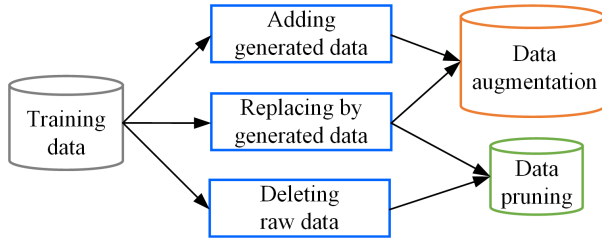

Fig. 13. Connection between augmentation and pruning.

In the data resampling, weighting, and perturbation, the assignment manners for the sampling rate, weighting score, and perturbation variable are quite similar. In addition to the classical importance score, both meta learning [334] and adversarial strategy [120] have also been used in data resampling. Regularization-based manner is used in nearly all the data optimization paths except resampling. Due to space constraints, methods with different assignment manners are not summarized in a table as those in Section VIII-B.

There are other opposite relationships, such as undersampling vs. oversampling, easy-first weighting vs. hard-first weighting, positive perturbation vs. negative perturbation, and explicit augmentation vs. implicit augmentation. Both methodologies in these opposite relationships have been demonstrated to be effective, aligning with the proverb "All roads lead to Rome".

D. Connections via theory

There are some common theoretical issues, analyses, and conclusions heavily influencing most data optimization techniques. They are the natural connections among different techniques. Several examples are listed as follows:

- Theoretical issues in data perception. A solid theoretical basis for data perception in data optimization is lacking, even though most data optimization methods implicitly or explicitly rely on the perception for the training data. For instance, many methods from resampling, weighting, and perturbation are based on dividing samples into easy and hard. Nevertheless, there is not yet a widely accepted learning difficulty measure with a rigorous theoretical basis in the literature. More than ten types of learning difficulty measures are utilized to distinguish easy from hard samples in previous literature. A theoretical formulation for data perception is of great importance.

- Probabilistic density (ratio) estimation. Many data optimization methods, especially data resampling and weighting, heavily rely on the probabilistic density (ratio) estimation. The most representative method is the importance sampling. In learning difficulty-based weighting, the probabilistic density ratio, in terms of learning difficulty, is revealed to determine the priority mode [217], namely, easy/medium/hard-first.

- Regularization-based explanation. Many data optimization methods are considered as a type of regularization, including data augmentation and perturbation. In these methods, data optimization performs implicit model regularization other than explicit regularization that directly works on model parameters. Regularization is not always beneficial as over-regularization may occur. Li et al. [335] pointed out that large amount of augmented noisy data could lead to over-regularization and proposed an adaptive augmentation method. Adversarial training may result in robust overfitting [336].

- Generalization bound for data optimization. Many studies choose to deduce a mathematical bound for the generalization risk in terms of the empirical risk and variables related to the data optimization. This manner can theoretically explain the utility of the involved data optimization. Xiao et al. [337] derived stability-based generalization bounds for stochastic gradient descent (SGD) on the loss with adversarial perturbations. Xu et al. [115] established a new generalization bound that reflects how importance weighting leads to the interplay between the empirical risk and the deviation between the source and target distributions.

The progress in each of the above theoretical aspects will promote the advancement of many data optimization methods in different technical paths. Hopefully, this survey will promote the mutual understanding of the referred technical paths.

IX. FUTURE DIRECTIONS

This section summarizes some research directions deserving further exploring.

A. Principles of data optimization

Up till now, there has been no consensus theoretical framework that is suitable for all or most technical paths.

There are some studies aiming to establish the connection between two different technical paths, such as resampling vs. weighting [338]. Many open problems or controversies remain unsolved. For example, there is no ideal answer for which resampling strategy should be employed first: over-sampling or undersampling? Megahed et al. [104] suggested that undersampling should be used firstly, whereas Xie et al. [275] demonstrated that oversampling is effective. Like-wise, although Zhou et al. [217] provided an initial answer for the choice of easy-first and hard-first weighting strategies, a solid theoretical framework is still lacking in their study.

Moreover, even for a single data optimization method, multiple explanations from different views may exist. The explanation for label smoothing is a typical example. At least four studies provide empirical or theoretical explanations for it [260], [298], [339], [340]. Regarding the effectiveness of adversarial samples, some researchers have pointed out that adversarial samples are useful features [270], while some other researchers investigated it in terms of gradient regularization [341].

Consequently, the construction of the data optimization principles is of great importance, as it can promote the establishing of a unified and solid theoretical framework which can be used to analyze and understand of each data optimization technical path. There have been studies on the first principle for the design of DNNs [342]. To explore the principles for data optimization, a unified mathematical formalization tool is required and large-scale empirical studies (e.g., [343], [344]) will also be helpful.

B. Interpretable data optimization

Interpretable data optimization refers to the explanation for the involved data optimization techniques in terms of how and which aspects they affect the training process of DNNs. Although interpretable deep learning receives much attention in recent years [345], it focuses on DNN models other than the training processing in which data optimization techniques are involved. Interpretable data optimization is an under-explored research topic and there are limited studies on this topic [346]. The well explanation of how and which aspects of a data optimization method affects a specific training process is significant beneficial for the design or selecting of more effective optimization methods.

The aforementioned theoretical studies on data optimization provide partial explanations for the corresponding data optimization method. However, the partial explanations are concentrated in common aspects across different learning tasks. How and which aspects for the involved method on a specific learning task remain unexplored.

The interpretable deep learning area has raised many effective methodologies. Recently, researchers have attempted to introduce interpretable methodologies to explore the data optimization methods. Zelaya and Vladimiro [347] explored metrics to quantify the effect of some data-processing steps such as undersampling and data augmentation on the model performance. Hopefully, more and more studies on explainable data optimization appear in the future.

C. Human-in-the-loop data optimization

Recently, human-in-the-loop (HITL) deep learning receives increasing attention in the AI community [348]. With out human's participants, high-quality training samples are not intractable to obtain. Naturally, HITL data optimization can also be beneficial for deep learning. Collins et al. [349] investigated HITL mixup and indicated that collating humans' perceptions on augmented samples could impact model performance. Wallace et al. [350] proposed HITL adversarial generation, where human authors are guided to break models. Agarwal et al. [351] proposed Variance of Gradients (VoG) to measure samples' learning difficulty and ranked samples by VoG. Then, a tractable subset of the most difficult samples is selected for HITL auditing. Overall, research on HITL data optimization is in the early stage.

D. Data optimization for new challenges

New challenges are constantly emerging in deep learning applications. We take the following three recent challenges as examples to illustrate the future direction of data optimization:

- Open-world learning. This learning scenario confronts the challenge of out-of-distribution (OOD) samples. Wu et al. [352] investigated noisy-label learning under the open-world setting, in which both OOD and noisy samples exist. Some other studies investigate cases when ODD meets imbalanced learning [353] and adversarial robustness [354].
- Large-model training. Large models especially the large language models [355] have achieved great success in recent years. Data optimization can also take effect in the training of large models. Wei et al. [356] investigated the condensation of prompts and promising results are obtained. Contrarily, Jiang et al. [357] leveraged prompt augmentation to calibrate large language models. Many issues investigated in conventional deep learning tasks may also exist for large-model training, e.g., prompt valuation.
- Multi-modal learning. With the development of data sensing and collection technology, multi-modal data are avaliable in more and more real tasks [358]. Consequently, many learning tasks are actually multi-modal learning. As each sample consists of raw data/features from different modalities, the data perception for multi-modal samples should be different from that for conventional single-modal samples. The data optimization methods are likewise different from conventional methods [359].

E. Data optimization agent

Given a concrete learning task, a selection dilemma occurs for the tremendous data optimization techniques. There have been studies on the automatic data optimization such as automatic data augmentation [287]. Nevertheless, existing automatic data optimization methods still focus on a particular type of technical path rather than the types across different technical paths [360], [361]. A more general data optimization agent can be trained by iteratively training on a large number of deep learning tasks via reinforcement learning.
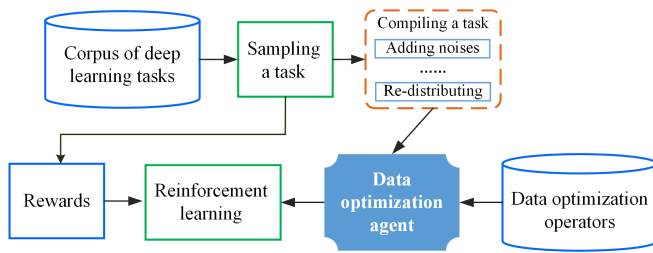
Fig. 14. The training of a data optimization agent.

Fig. 14 shows a possible mean to construct a data optimization agent. New learning tasks are compiled based on existing classical tasks via operations such as noise adding, and class proportion re-distributing. The candidate of data optimization operators are from arbitrary optimization techniques or a single technique introduced in Section VI. A powerful data optimization agent can then be trained via reinforcement learning based on compiled learning tasks and their rewards.

X. CONCLUSIONS

This paper aims to summarize a wide range of learning methods within an independent deep learning realm, namely, data optimization. A taxonomy for data optimization, as well as fine-granularity sub-taxonomies, is established for existing studies on data optimization. Connections among different methods are discussed, and potential future directions are presented. It is noteworthy that many classical methods, such as dropout, are essentially data optimization methods. In our future work, we will explore a more fundamental and unified viewpoint on data optimization, and develop a more comprehensive taxonomy to incorporate more classical methods. We hope that this study can inspire more researchers to gain insight into data-centric AI.

REFERENCES

[1] S. E. Whang, Y. Roh, H. Song, and J.-G. Lee, "Data collection and quality challenges in deep learning: A data-centric ai perspective," *The VLDB Journal*, vol. 32, no. 4, pp. 791–813, 2023.

[2] M. H. Jarrahi, A. Memariani, and S. Guha, "The principles of data-centric ai," *Communications of the ACM*, vol. 66, no. 8, pp. 84–92, 2023.

[3] Y. Liang, D. Huang, C.-D. Wang, and P. S. Yu, "Multi-view graph learning by joint modeling of consistency and inconsistency," *IEEE TNNLS*, pp. 1–15, 2022.

[4] P. Zhu, X. Yao, Y. Wang, M. Cao, B. Hui, S. Zhao, and Q. Hu, "Latent heterogeneous graph network for incomplete multi-view learning," *IEEE TMM*, vol. 25, pp. 3033–3045, 2023.

[5] L. Brigato and L. Iocchi, "A close look at deep learning with small data," in *ICPR*, 2021, pp. 2490–2497.

[6] V. Birodkar, H. Mobahi, and S. Bengio, "Semantic redundancies in image-classification datasets: The 10% you don't need," *arXiv:1901.11409*, 2019.

[7] Y. Yu, S. Khadivi, and J. Xu, "Can data diversity enhance learning generalization?" in *COLING*, 2022, pp. 4933–4945.

[8] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, "Learning under concept drift: A review," *IEEE TKDE*, vol. 31, no. 12, pp. 2346–2363, 2018.

[9] W. Wang, R. Wang, L. Wang, Z. Wang, and A. Ye, "Towards a robust deep neural network against adversarial texts: A survey," *IEEE TKDE*, vol. 35, no. 3, pp. 3159–3179, 2023.

[10] Y. Wu, L. Zhang, and X. Wu, "On convexity and bounds of fairness-aware classification," in *WWW*, 2019, pp. 3356–3362.

[11] P. Xiong, S. Buffett, S. Iqbal, P. Lamontagne, M. Mamun, and H. Molyneaux, "Towards a robust and trustworthy machine learning system development: An engineering perspective," *JISA*, vol. 65, p. 103121, 2022.

[12] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *ICLR*, 2018.

[13] R. Yao and O. Wu, "Compensation learning," *arXiv:2107.11921*, 2022.

[14] X. Wang, L. Jing, Y. Lyu, M. Guo, J. Wang, H. Liu, J. Yu, and T. Zeng, "Deep generative mixture model for robust imbalance classification," *IEEE TPAMI*, vol. 45, no. 3, pp. 2897–2912, 2023.

[15] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE TKDE*, vol. 21, no. 9, pp. 1263–1284, 2009.

[16] Y. Zhang, B. Kang, B. Hooi, S. Yan, and J. Feng, "Deep long-tailed learning: A survey," *IEEE TPAMI*, vol. 45, no. 9, pp. 10 795–10 816, 2023.

[17] G. Algan and I. Ulusoy, "Image classification with deep learning in the presence of noisy labels: A survey," *Knowledge-Based Systems*, vol. 215, no. 5, p. 106771, 2021.

[18] H. Song, M. Kim, D. Park, Y. Shin, and J.-G. Lee, "Learning from noisy labels with deep neural networks: A survey," *IEEE TNNLS*, pp. 1–19, 2022.

[19] X. Cao, W. Bu, S. Huang, M. Zhang, I. W. Tsang, Y. S. Ong, and J. T. Kwok, "A survey of learning on small data," *arXiv:2207.14443*, 2022.

[20] Y. Wang, Q. Yao, J. Kwok, and L. M. Ni, "Generalizing from a few examples: A survey on few-shot learning," *ACM computing surveys*, vol. 53, no. 3, pp. 1–34, 2020.

[21] L. Yuan, H. Li, B. Xia, C. Gao, M. Liu, W. Yuan, and X. You, "Recent advances in concept drift adaptation methods for deep learning," in *IJCAI*, 2022, pp. 5654–5661.

[22] A. Diez-Olivan, P. Ortego, J. D. Ser, I. Landa-Torres, D. Galar, D. Camacho, and B. Sierra, "Adaptive dendritic cell-deep learning approach for industrial prognosis under changing conditions," *IEEE TII*, vol. 17, no. 11, pp. 7760–7770, 2021.

[23] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Computing Surveys*, vol. 46, no. 4, pp. 1–37, 2014.

[24] A. S. Iwashita and J. P. Papa, "An overview on concept drift learning," *IEEE Access*, vol. 7, pp. 1532–1547, 2019.

[25] S. H. Silva and P. Najafirad, "Opportunities and challenges in deep learning adversarial robustness: A survey," *arXiv:2007.00753*, 2020.

[26] J. Xu, J. Chen, S. You, Z. Xiao, Y. Yang, and J. Lu, "Robustness of deep learning models on graphs: A survey," *AI Open*, vol. 2, pp. 69–78, 2021.

[27] S. Goyal, S. Doddapaneni, M. M. Khapra, and B. Ravindran, "A survey of adversarial defenses and robustness in nlp," *ACM Computing Surveys*, vol. 55, no. 14s, pp. 1–39, 2023.

[28] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, "A survey on bias and fairness in machine learning," *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–35, 2021.

[29] A. Petrović, M. Nikolić, S. Radovanović, B. Delibašić, and M. Jovanović, "Fair: Fair adversarial instance re-weighting," *Neurocomputing*, vol. 476, pp. 14–37, 2022.

[30] S. K. Devitt, "Trustworthiness of autonomous systems," *Foundations of trusted autonomy (Studies in Systems, Decision and Control, Volume 117)*, pp. 161–184, 2018.

[31] D. Kaur, S. Uslu, K. J. Rittichier, and A. Durresi, "Trustworthy artificial intelligence: A review," *ACM Computing Surveys*, vol. 55, no. 2, pp. 1–38, 2022.

[32] B. Wu, Y. Bian, H. Zhang, J. Li, J. Yu, L. Chen, C. Chen, and J. Huang, "Trustworthy graph learning: Reliability, explainability, and privacy protection," *ACM KDD*, pp. 4838–4839, 2022.

[33] C. Fang, L. Cheng, H. Qi, and D. Zhang, "Combating noisy labels in long-tailed image classification," *arXiv:2209.00273*, 2022.

[34] M. Singh, G. Ghalachyan, K. R. Varshney, and R. E. Bryant, "An empirical study of accuracy, fairness, explainability, distributional robustness, and adversarial robustness," in *KDD Workshop*, 2021.

[35] Z. Wan, Z. Wang, C. Chung, and Z. Wang, "A survey of data optimization for problems in computer vision datasets," *arXiv:2210.11717*, 2022.

[36] X. Zheng, Y. Liu, Z. Bao, M. Fang, X. Hu, A. W.-C. Liew, and S. Pan, "Towards data-centric graph machine learning: Review and outlook," *arXiv:2309.10979*, 2023.

[37] Y. Luo, Y. Wong, M. Kankanhalli, and Q. Zhao, "$\mathcal{G}$-softmax: Improving intraclass compactness and interclass separability of features," *IEEE TNNLS*, vol. 31, no. 2, pp. 685–699, 2020.

[38] A. Damian, T. Ma, and J. D. Lee, "Label noise sgd provably prefers flat global minimizers," in *NeurIPS*, 2021, pp. 27 449–27 461.

[39] Y. Wang, X. Pan, S. Song, H. Zhang, C. Wu, and G. Huang, "Implicit semantic data augmentation for deep networks," in *NeurIPS*, 2019, pp. 12 635–12 644.

[40] M. Wang, Y. Zhang, and W. Deng, "Meta balanced network for fair face recognition," *IEEE TPAMI*, 2021.

[41] S. Fujii, Y. Ishii, K. Kozuka, T. Hirakawa, T. Yamashita, and H. Fujiyoshi, "Data augmentation by selecting mixed classes considering distance between classes," *arXiv:2209.05122*, 2022.

[42] I. L. . F. Hutter, "Online batch selection for faster training of neural networks," *ICLR Workshop*, 2016.

[43] C.-Y. Chuang and Y. Mroueh, "Fair mixup: Fairness via interpolation," in *ICLR*, 2021.

[44] L. E. Celis, A. Mehrotra, and N. Vishnoi, "Fair classification with adversarial perturbations," in *NeurIPS*, 2021, pp. 8158–8171.

[45] B. Yan, S. Seto, and N. Apostoloff, "Forml: Learning to reweight data for fairness," *arXiv:2202.01719*, 2022.

[46] Y. Cui, M. Jia, T. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *CVPR*, 2019, pp. 9260–9269.

[47] M. Li, F. Su, O. Wu, and J. Zhang, "Logit perturbation," in *AAAI*, 2022, pp. 10 388–10 396.

[48] Y. Chen, P. Zhang, T. Kong, Y. Li, X. Zhang, L. Qi, J. Sun, and J. Jia, "Scale-aware automatic augmentations for object detection with dynamic training," *IEEE TPAMI*, vol. 45, no. 2, pp. 2367–2383, 2023.

[49] M. P. Naeini, G. F. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using bayesian binning," in *AAAI*, 2015.

[50] B. Liu, I. Ben Ayed, A. Galdran, and J. Dolz, "The devil is in the margin: Margin-based label smoothing for network calibration," in *CVPR*, 2022, pp. 80–88.

[51] J. Mukhoti, V. Kulharia, A. Sanyal, S. Golodetz, P. Torr, and P. Dokania, "Calibrating deep neural networks using focal loss," in *NeurIPS*, 2020, pp. 15 288–15 299.

[52] L. Dunlap, A. Umino, H. Zhang, J. Yang, J. E. Gonzalez, and T. Darrell, "Diversify your vision datasets with automatic diffusion-based augmentation," *arXiv:2305.16289*, 2023.

[53] Z. Ye, Y. Dai, C. Hong, Z. Cao, and H. Lu, "Infusing definiteness into randomness: Rethinking composition styles for deep image matting," *AAAI*, 2023.

[54] S. Yang, W. Xiao, M. Zhang, S. Guo, J. Zhao, and F. Shen, "Image data augmentation for deep learning: A survey," *arXiv:2204.08610*, 2022.

[55] W. Yang, C. Li, J. Zhang, and C. Zong, "Bigtranslate: Augmenting large language models with multilingual translation capability over 100 languages," *arXiv:2305.18098*, 2023.

[56] X. Liu, H. Cheng, P. He, W. Chen, Y. Wang, H. Poon, and J. Gao, "Adversarial training for large neural language models," *arXiv:2004.08994*, 2020.

[57] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," *AISTATS*, pp. 249–256, 2010.

[58] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," *ICCV*, pp. 1026–1034, 2015.

[59] F. Su, Y. Zhu, O. Wu, and Y. Deng, "Submodular meta data compiling for meta optimization," *ECML/PKDD*, 2022.

[60] J. Huang, L. Qu, R. Jia, and B. Zhao, "O2u-net: A simple noisy label detection approach for deep neural networks," in *ICCV*, 2019, pp. 3326–3334.

[61] B. Li, Y. Liu, and X. Wang, "Gradient harmonized single-stage detector," in *AAAI*, 2019, pp. 8577–8584.

[62] C.-B. Zhang, P.-T. Jiang, Q. Hou, Y. Wei, Q. Han, Z. Li, and M.-M. Cheng, "Delving deep into label smoothing," *IEEE TIP*, vol. 30, pp. 5984–5996, 2021.

[63] S. Sinha, H. Ohashi, and K. Nakamura, "Class-wise difficulty-balanced loss for solving class-imbalance," in *ACCV*, 2020.

[64] M. Escudero-Viñolo and A. López-Cifuentes, "Ccl: Class-wise curriculum learning for class imbalance problems," in *ICIP*, 2022, pp. 1476–1480.

[65] X. Ning, W. Tian, F. He, X. Bai, L. Sun, and W. Li, "Hyper-sausage coverage function neuron model and learning algorithm for image classification," *Pattern Recognition*, vol. 136, p. 109216, 2023.

[66] J. Lin, A. Zhang, M. Lécuyer, J. Li, A. Panda, and S. Sen, "Measuring the effect of training data on deep learning predictions via randomized experiments," in *ICML*, 2022, pp. 13 468–13 504.

[67] X. Zhou, N. Yang, and O. Wu, "Combining adversaries with anti-adversaries in training," in *AAAI*, 2023.

[68] K. Tang, M. Tao, J. Qi, Z. Liu, and H. Zhang, "Invariant feature learning for generalized long-tailed classification," in *ECCV*, 2022, pp. 709–726.

[69] S. Shrivastava, X. Zhang, S. Nagesh, and A. Parchami, "Datasetequity: Are all samples created equal? in the quest for equity within datasets," in *ICCV*, 2023, pp. 4417–4426.

[70] R. Wang, W. Xiong, Q. Hou, and O. Wu, "Tackling the imbalance for gnns," in *IJCNN*, 2022.

[71] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *ICML*, 2009, pp. 41–48.

[72] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *CVPR*, 2017, pp. 2999–3007.

[73] M. Paul, S. Ganguli, and G. K. Dziugaite, "Deep learning on a data diet: Finding important examples early in training," in *NeurIPS*, 2021, pp. 20 596–20 607.

[74] W. Zhu, O. Wu, F. Su, and Y. Deng, "Exploring the learning difficulty of data: Theory and measure," *arXiv:2205.07427*, 2022.

[75] B. Sorscher, R. Geirhos, S. Shekhar, S. Ganguli, and A. S. Morcos, "Beyond neural scaling laws: beating power law scaling via data pruning," in *NeurIPS*, 2022.

[76] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya, V. Makarenkov, and S. Nahavandi, "A review of uncertainty quantification in deep learning: Techniques, applications and challenges," *Information fusion*, vol. 76, pp. 243–297, 2021.

[77] D. D'souza, Z. Nussbaum, C. Agarwal, and S. Hooker, "A tale of two long tails," *arXiv:2107.13098*, 2021.

[78] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" in *NeurIPS*, 2017, pp. 5574–5584.

[79] A. Kumar, S. Bhattamishra, M. Bhandari, and P. Talukdar, "Submodular optimization-based diverse paraphrasing and its effectiveness in data augmentation," in *NAACL*, 2019, pp. 3609–3619.

[80] D. Friedman and A. B. Dieng, "The vendi score: A diversity evaluation metric for machine learning," *arXiv:2210.02410*, 2023.

[81] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," *NeurIPS*, pp. 2234–2242, 2016.

[82] O. Wu, "Rethinking class imbalance in machine learning," *arXiv:2305.03900*, 2023.

[83] S. Swayamdipta, R. Schwartz, N. Lourie, Y. Wang, H. Hajishirzi, N. A. Smith, and Y. Choi, "Dataset cartography: Mapping and diagnosing datasets with training dynamics," in *EMNLP*, 2020.

[84] A. Iscen, J. Valmadre, A. Arnab, and C. Schmid, "Learning with neighbor consistency for noisy labels," in *CVPR*, 2022, pp. 4672–4681.

[85] M. Toneva, A. Sordoni, R. T. des Combes, A. Trischler, Y. Bengio, and G. J. Gordon, "An empirical study of example forgetting during deep neural network learning," *ICLR*, 2019.

[86] P. Singh, P. Mazumder, and M. A. Karim, "Attaining class-level forgetting in pretrained model using few samples," in *ECCV*, 2022, pp. 433–448.

[87] P. Maini, S. Garg, Z. Lipton, and J. Z. Kolter, "Characterizing datapoints via second-split forgetting," in *NeurIPS*, 2022, pp. 30 044–30 057.

[88] Z. Wang, E. Yang, L. Shen, and H. Huang, "A comprehensive survey of forgetting in deep learning beyond continual learning," *arXiv:2307.09218*, 2023.

[89] T. Kim, J. Ko, s. Cho, J. Choi, and S.-Y. Yun, "Fine samples for learning with noisy labels," in *NeurIPS*, 2021, pp. 24 137–24 149.

[90] L. S. Shapley, "A value for n-person games," in *In Contributions to the Theory of Games*, 1953, pp. 307–317.

[91] A. Ghorbani and J. Zou, "Data shapley: Equitable valuation of data for machine learning," in *ICML*, 2019, pp. 2242–2251.

[92] J. Yoon, S. Arik, and T. Pfister, "Data valuation using reinforcement learning," in *ICML*, 2020, pp. 10 842–10 851.

[93] Y. Bian, Y. Rong, T. Xu, J. Wu, A. Krause, and J. Huang, "Energy-based learning for cooperative games, with applications to valuation problems in machine learning," in *ICLR*, 2022.

[94] K. F. Jiang, W. Liang, J. Zou, and Y. Kwon, "Opendataval: a unified benchmark for data valuation," in *NeurIPS*, 2023.

[95] D. Bahri and H. Jiang, "Locally adaptive label smoothing improves predictive churn," in *ICML*, 2021, pp. 532–542.

[96] C. Dong, L. Liu, and J. Shang, "Data profiling for adversarial training: On the ruin of problematic data," *arXiv:2102.07437v1*, 2021.

[97] Z. Hammoudeh and D. Lowd, "Training data influence analysisand estimation: A survey," *arXiv:2212.04612*, 2023.

[98] Y. Wu, J. Shu, Q. Xie, Q. Zhao, and D. Meng, "Learning to purify noisy labels via meta soft label corrector," in *AAAI*, 2021, pp. 10 388–10 396.

[99] J. Shu, Q. Xie, L. Yi, Q. Zhao, S. Zhou, Z. Xu, and D. Meng, "Meta-Weight-Net: Learning an explicit mapping for sample weighting," in *NeurIPS*, 2019, pp. 1917–1928.

[100] M. P. Kumar, B. Packer, and D. Koller, "Self-paced learning for latent variable models," *NeurIPS*, pp. 1–9, 2010.

[101] E. Arazo, D. Ortego, P. Albert, N. E. O'Connor, and K. McGuinness, "Unsupervised label noise modeling and loss correction," in *ICML*, 2019.

[102] C. Hu, S. Yan, Z. Gao, and X. He, "Mild: Modeling the instance learning dynamics for learning with noisy labels," *arXiv:2306.11560*, 2023.

[103] Y. Li and N. Vasconcelos, "Repair: Removing representation bias by dataset resampling," *CVPR*, 2019.

[104] F. M. Megahed, Y.-J. Chen, A. Megahed, Y. Ong, N. Altman, and M. Krzywinski, "The class imbalance problem," *Nature Methods*, vol. 18, pp. 1270–1272, 2021.

[105] J. Cui, S. Liu, Z. Tian, Z. Zhong, and J. Jia, "Reslt: Residual learning for long-tailed recognition," *IEEE TPAMI*, vol. 45, no. 3, pp. 3695–3706, 2023.

[106] A. Kirsch, J. van Amersfoort, and Y. Gal, "Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning," in *NeurIPS*, 2019.

[107] H. Shimodaira, "Improving predictive inference under covariate shift by weighting the log-likelihood function," *Journal of statistical planning and inference*, 2000.

[108] J. Byrd and Z. Lipton, "What is the effect of importance weighting in deep learning?" in *ICML*, 2019, pp. 872–881.

[109] Q. Liu and J. Lee, "Black-box Importance Sampling," in *AISTATS*, 2017, pp. 952–961.

[110] A. Katharopoulos and F. Fleuret, "Not all samples are created equal: Deep learning with importance sampling," in *ICML*, 2018, pp. 2525–2534.

[111] T. B. Johnson and C. Guestrin, "Training deep models faster with robust, approximate importance sampling," in *NeurIPS*, 2018.

[112] A. H. Jiang, D. L.-K. Wong, G. Zhou, D. G. Andersen, J. Dean, G. R. Ganger, G. Joshi, M. Kaminksy, M. Kozuch, Z. C. Lipton, and P. Pillai, "Accelerating deep learning by focusing on the biggest losers," *arXiv:1910.00762*, 2019.

[113] X. J. Gui, W. Wang, and Z. H. Tian, "Towards understanding deep learning from noisy labels with small-loss criterion," *IJCAI*, 2021.

[114] H. Liu, X. Zhu, Z. Lei, and S. Z. Li, "Adaptiveface: Adaptive margin and sampling for face recognition," in *CVPR*, 2019, pp. 11 947–11 956.

[115] D. Xu, Y. Ye, and C. Ruan, "Understanding the role of importance weighting for deep learning," *arXiv:2103.15209*, 2021.

[116] V. Nguyen, M. Shaker, and E. Hüllermeier, "How to measure uncertainty in uncertainty sampling for active learning," *Machine Learning*, vol. 111, pp. 89–122, 2022.

[117] J. Mena, O. Pujol, and J. Vitrià, "A survey on uncertainty estimation in deep learning classification systems from a bayesian perspective," *ACM Computing Surveys*, vol. 54, no. 9, pp. 1–35, 2021.

[118] A. Aljuhani, I. Casukhela, J. Chan, D. Liebner, and R. Machiraju, "Uncertainty aware sampling framework of weak-label learning for histology image classification," in *MICCAI*, 2022, pp. 366–376.

[119] D. Ting and E. Brochu, "Optimal subsampling with influence functions," in *NeurIPS*, 2018, pp. 3650–3659.

[120] Y. Li and N. Vasconcelos, "Background data resampling for outlier-aware classification," in *CVPR*, 2020, pp. 13 218–13 227.

[121] X. Wang and Y. Wang, "Sentence-level resampling for named entity recognition," in *NAACL*, 2022, pp. 2151–2165.

[122] J. Zhang, T. Wang, W. W. Y. Ng, S. Zhang, and C. D. Nugent, "Undersampling near decision boundary for imbalance problems," in *ICMLC*, 2019, pp. 1–8.

[123] M. Sun, H. Dou, B. Li, J. Yan, W. Ouyang, and L. Cui, "Autosampling: Search for effective data sampling schedules," in *ICML*, 2017, p. 9923–9933.

[124] G. Li, L. Liu, G. Huang, C. Zhu, and T. Zhao, "Understanding data augmentation in neural machine translation: Two perspectives towards generalization," in *EMNLP-IJCNLP*, 2019, pp. 5689–56 958.

[125] L. Zhao, T. Liu, X. Peng, and D. Metaxas, "Maximum-entropy adversarial data augmentation for improved generalization and robustness," in *NeurIPS*, vol. 33, 2020, pp. 14 435–14 447.

[126] S.-A. Rebuffi, S. Gowal, D. A. Calian, F. Stimberg, O. Wiles, and T. A. Mann, "Data augmentation can improve robustness," in *NeurIPS*, 2021, pp. 29 935–29 948.

[127] L. Li and M. Spratling, "Data augmentation alone can improve adversarial training," in *ICLR*, 2023.

[128] M. Bayer, M.-A. Kaufhold, and C. Reuter, "A survey on data augmentation for text classification," *ACM Computing Surveys*, vol. 55, no. 7, pp. 1–39, 2022.

[129] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 60, 2019.

[130] K. Ding, Z. Xu, H. Tong, and H. Liu, "Data augmentation for deep graph learning: A survey," *ACM SIGKDD Explorations Newsletter*, vol. 24, no. 2, 2022.

[131] Q. Wen, L. Sun, F. Yang, X. Song, J. Gao, X. Wang, and H. Xu, "Time series data augmentation for deep learning: A survey," in *IJCAI*, 2021, pp. 1–8.

[132] B. Li, Y. Hou, and W. Che, "Data augmentation approaches in natural language processing: A survey," *AI Open*, vol. 3, pp. 71–90, 2022.

[133] T. DeVries and G. W. Taylor, "Dataset augmentation in feature space," *ICLR Workshop*, 2017.

[134] P. Li, D. Li, W. Li, S. Gong, Y. Fu, and T. M. Hospedales, "A simple feature augmentation for domain generalization," in *ICCV*, 2021, pp. 8886–8895.

[135] M. Ye, J. Shen, X. Zhang, P. C. Yuen, and S.-F. Chang, "Augmentation invariant and instance spreading feature for softmax embedding," *IEEE TPAMI*, vol. 44, no. 2, pp. 924–939, 2022.

[136] P. Chu, X. Bian, S. Liu, and H. Ling, "Feature space augmentation for long-tailed data," in *ECCV*, 2020, pp. 694–710.

[137] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *ICLR*, 2018.

[138] T. Bai and J. Luo, "Recent advances in adversarial training for adversarial robustness," in *IJCAI*, 2021, pp. 4312–4321.

[139] S. Lee, H. Kim, and J. Lee, "Graddiv: Adversarial robustness of randomized neural networks via gradient diversity regularization," *TPAMI*, vol. 45, no. 2, pp. 2645–2651, 2023.

[140] H. Lee, S. J. Hwang, and J. Shin, "Self-supervised label augmentation via input transformations," in *ICML*, 2020, pp. 5714–5724.

[141] I. Elezi, A. Torcinovich, S. Vascon, and M. Pelillo, "Transductive label augmentation for improved deep network learning," in *ICPR*, 2018, pp. 1432–1437.

[142] F. Huang, L. Zhang, Y. Zhou, and X. Gao, "Adversarial and isotropic gradient augmentation for image retrieval with text feedback," *IEEE TMM*, pp. 1–12, 2022.

[143] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, no. 1, pp. 321–357, 2002.

[144] A. Telikani, A. H. Gandomi, K.-K. R. Choo, and J. Shen, "A cost-sensitive deep learning-based approach for network traffic classification," *IEEE TNSE*, vol. 19, no. 1, pp. 661–670, 2022.

[145] D. Dablain, B. Krawczyk, and N. V. Chawla, "Deepsmote: Fusing deep learning and smote for imbalanced data," *IEEE TNNLS*, vol. 34, no. 9, pp. 6390–6404, 2023.

[146] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, and Y. Bengio, "Manifold mixup: Better representations by interpolating hidden states," in *ICML*, 2019, pp. 6438–6447.

[147] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *NeurIPS*, 2014.

[148] J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye, "A review on generative adversarial networks: Algorithms, theory, and applications," *IEEE TKDE*, vol. 35, no. 4, pp. 3313–3332, 2023.

[149] G. Mariani, F. Scheidegger, R. Istrate, C. Bekas, and C. Malossi, "Bagan: Data augmentation with balancing gan," *arXiv:1803.09655*, 2018.

[150] S.-W. Huang, C.-T. Lin, S.-P. Chen, Y.-Y. Wu, P.-H. Hsu, and S.-H. Lai, "Auggan: Cross domain adaptation with gan-based data augmentation," in *ECCV*, 2018.

[151] Z. Yang, Y. Li, and G. Zhou, "Ts-gan: Time-series gan for sensor-based health data augmentation," *ACM TOCH*, vol. 4, no. 2, pp. 1–21, 2022.

[152] L. Yang, Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, W. Zhang, B. Cui, and M.-H. Yang, "Diffusion models: A comprehensive survey of methods and applications," *ACM Computing Surveys*, 2023.

[153] C. Xiao, S. X. Xu, and K. Zhang, "Multimodal data augmentation for image captioning using diffusion models," *arXiv:2305.01855*, 2023.

[154] P. McNamee and K. Duh, "An extensive exploration of back-translation in 60 languages," in *Findings of ACL*, 2023, pp. 8166–8183.

[155] H. Dong, J. Zhang, D. McIlwraith, and Y. Guo, "I2t2i: Learning text to image synthesis with textual data augmentation," in *ICIP*, 2017, pp. 2015–2019.

[156] D. Lu, Z. Wang, T. Wang, W. Guan, H. Gao, and F. Zheng, "Set-level guidance attack: Boosting adversarial transferability of vision-language pre-training models," in *ICCV*, 2023.

[157] Y. Yin, J. Kaddour, X. Zhang, Y. Nie, Z. Liu, L. Kong, and Q. Liu, "Ttida: Controllable generative data augmentation via text-to-text and text-to-image models," *arXiv:2304.08821*, 2023.

[158] M. Pagliardini, G. Manunza, M. Jaggi, M. I. Jordan, and T. Chavdarova, "Improving generalization via uncertainty driven perturbations," *arXiv:2202.05737*, 2022.

[159] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, "Randaugment: Practical automated data augmentation with a reduced search space," *arXiv:1909.13719*, 2019.

[160] Z. Mai, G. Hu, D. Chen, F. Shen, and H. T. Shen, "Metamixup: Learning adaptive interpolation policy of mixup with metalearning," *IEEE TNNLS*, vol. 33, no. 7, pp. 3050–3064, 2021.

[161] T. Qin, Z. Wang, K. He, Y. Shi, Y. Gao, and D. Shen, "Automatic data augmentation via deep reinforcement learning for effective kidney tumor segmentation," in *ICASSP*, 2020, pp. 1419–1423.

[162] K. Nishi, Y. Ding, A. Rich, and T. Hollerer, "Augmentation strategies for learning with noisy labels," in *CVPR*, 2021, pp. 8022–8031.

[163] Y. Li, G. Hu, Y. Wang, T. Hospedales, N. M. Robertson, and Y. Yang, "Differentiable automatic data augmentation," in *ECCV*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds., 2020, pp. 580–595.

[164] X. Chen, Y. Zhou, D. Wu, W. Zhang, Y. Zhou, B. Li, and W. Wang, "Imagine by reasoning: A reasoning-based implicit semantic data augmentation for long-tailed classification," in *AAAI*, Online, February 2022, pp. 356–364.

[165] X. Zhou and O. Wu, "Implicit counterfactual data augmentation for deep neural networks," *arXiv:2304.13431*, 2023.

[166] B. Li, F. Wu, S.-N. Lim, S. Belongie, and K. Q. Weinberger, "On feature normalization and data augmentation," in *CVPR*, 2021, pp. 12 383–12 392.

[167] D. LeJeune, R. Balestriero, H. Javadi, and R. G. Baraniuk, "Implicit rugosity regularization via data augmentation," *arXiv:1905.11639*, 2019.

[168] H. Guo, Y. Mao, and R. Zhang, "Mixup as locally linear out-of-manifold regularization," *AAAI*, pp. 3714–3722, 2019.

[169] C. F. G. D. Santos and J. P. Papa, "Avoiding overfitting: A survey on regularization methods for convolutional neural networks," *ACM Computing Surveys*, vol. 54, no. 10, pp. 1–25, 2022.

[170] C.-H. Lin, C. Kaushik, E. L. Dyer, and V. Muthukumar, "The good, the bad and the ugly sides of data augmentation: An implicit spectral regularization perspective," *arXiv:2210.05021*, 2022.

[171] S. Chen, E. Dobriban, and J. Lee, "A group-theoretic framework for data augmentation," *Journal of Machine Learning Research*, vol. 21, no. 1, pp. 9885–9955, 2020.

[172] A. Jeddi, M. J. Shafiee, M. Karg, C. Scharfenberger, and A. Wong, "Learn2perturb: An end-to-end feature perturbation learning to improve adversarial robustness," in *CVPR*, 2020.

[173] M. Shu, Z. Wu, M. Goldblum, and T. Goldstein, "Encoding robustness to image style via adversarial feature perturbations," in *NeurIPS*, 2021, pp. 28 042–28 053.

[174] A. K. Menon, S. Jayasumana, A. S. Rawat, H. Jain, A. Veit, and S. Kumar, "Long-tail learning via logit adjustment," in *ICLR*, 2021.

[175] K. Cao, C. Wei, A. Gaidon, N. Arechiga, and T. Ma, "Learning imbalanced datasets with label-distribution-aware margin loss," in *NeurIPS*, 2019, pp. 1567–1578.

[176] M. Li, F. Su, O. Wu, and J. Zhang, "Class-level logit perturbation," *IEEE TNNLS*, 2023.

[177] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *CVPR*, 2016, pp. 2818–2826.

[178] M. Goibert and E. Dohmatob, "Adversarial robustness via label-smoothing," *arXiv:1906.11567*, 2019.

[179] J. Lienen and E. Hüllermeier, "From label smoothing to label relaxation," in *AAAI*, 2021, pp. 8583–8591.

[180] A. Orvieto, H. Kersting, F. Proske, F. Bach, and A. Lucchi, "Anticorrelated noise injection for improved generalization," in *ICML*, 2022, pp. 17 094–17 116.

[181] D. Wu, S.-T. Xia, and Y. Wang, "Adversarial weight perturbation helps robust generalization," in *NeurIPS*, 2020, pp. 2958–2969.

[182] J. Wang, Y. Liu, and B. Li, "Reinforcement learning with perturbed rewards," in *AAAI*, 2020, pp. 6202–6209.

[183] S. Reed, H. Lee, D. Anguelov, C. Szegedy, D. Erhan, and A. Rabinovich, "Training deep neural networks on noisy labels with bootstrapping," in *ICLR Workshop*, 2015.

[184] P. Benz, C. Zhang, A. Karjauv, and I. S. Kweon, "Universal adversarial training with class-wise perturbations," in *ICME*, 2021, pp. 1–6.

[185] Y. Wang, J. Fei, H. Wang, W. Li, T. Bao, L. Wu, R. Zhao, and Y. Shen, "Balancing logit variation for long-tailed semantic segmentation," in *CVPR*, 2023, pp. 19 561–19 573.

[186] A. Shafahi, M. Najibi, Z. Xu, J. Dickerson, L. S. Davis, and T. Goldstein, "Universal adversarial training," in *CVPR*, 2017, pp. 5636–5643.

[187] A. Chaubey, N. Agrawal, K. Barnwal, K. K. Guliani, and P. Mehta, "Universal adversarial perturbations: A survey," *arXiv:2005.08087*, 2020.

[188] T. Wu, Q. Huang, Z. Liu, Y. Wang, and D. Lin, "Distribution-balanced loss for multi-label classification in long-tailed datasets," in *ECCV*, 2020, pp. 162–178.

[189] W. Zhou, X. Hou, Y. Chen, M. Tang, X. Huang, X. Gan, and Y. Yang, "Transferable adversarial perturbations," in *ECCV*, 2018, pp. 452–467.

[190] X. Wei, J. Zhu, S. Yuan, and H. Su, "Sparse adversarial perturbations for videos," in *AAAI*, 2019, pp. 8973–8980.

[191] Y. Zhu, Y. Ye, M. Li, J. Zhang, and O. Wu, "Investigating annotation noise for named entity recognition," *Neural Comput. Appl.*, vol. 35, no. 1, pp. 993–1007, 2023.

[192] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *ICML*, 2020, pp. 1597–1607.

[193] M. Naseer, S. Khan, M. Hayat, F. S. Khan, and F. Porikli, "A self-supervised approach for adversarial robustness," in *CVPR*, 2020, pp. 262–271.

[194] Z. Zhang, S.-h. Zhong, and Y. Liu, "Ganser: A self-supervised data augmentation framework for eeg-based emotion recognition," *IEEE TAC*, 2022.

[195] S. Li, K. Gong, C. H. Liu, Y. Wang, F. Qiao, and X. Cheng, "Metasaug: Meta semantic augmentation for long-tailed visual recognition," in *CVPR*, 2021, pp. 5212–5221.

[196] F. Qiao and X. Peng, "Uncertainty-guided model generalization to unseen domains," in *CVPR*, 2021, pp. 6790–6800.

[197] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "Autoaugment: Learning augmentation policies from data," in *CVPR*, 2019.

[198] T. Niu and M. Bansal, "Automatically learning data augmentation policies for dialogue tasks," in *EMNLP*, 2019.

[199] G. Apruzzese, M. Andreolini, M. Marchetti, A. Venturi, and M. Colajanni, "Deep reinforcement adversarial learning against botnet evasion attacks," *IEEE TNSE*, vol. 17, no. 4, pp. 1975–1987, 2020.

[200] B. Lin, Y. Zhu, Y. Long, X. Liang, Q. Ye, and L. Lin, "Adversarial reinforced instruction attacker for robust vision-language navigation," *IEEE TPAMI*, vol. 44, no. 10, pp. 7175–7189, 2022.

[201] A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi, and G. Bontempi, "Credit card fraud detection: A realistic modeling and a novel learning strategy," *IEEE TNNLS*, vol. 29, no. 8, pp. 3784–3797, 2017.

[202] Y. Zhang, P. Zhao, Q. Wu, B. Li, J. Huang, and M. Tan, "Cost-sensitive portfolio selection via deep reinforcement learning," *IEEE TKDE*, vol. 34, no. 1, pp. 236–248, 2022.

[203] D. Gan, J. Shen, B. An, M. Xu, and N. Liu, "Integrating tanbn with cost sensitive classification algorithm for imbalanced data in medical diagnosis," *Computers & Industrial Engineering*, vol. 140, p. 106266, 2020.

[204] Y. Dong, J. Ma, S. Wang, C. Chen, and J. Li, "Fairness in graph mining: A survey," *IEEE TKDE*, pp. 1–22, 2023.

[205] W. Wang, F. Feng, X. He, L. Nie, and T. Chua, "Denoising implicit feedback for recommendation," in *WSDM*, 2021, pp. 373–381.

[206] T. Castells, P. Weinzaepfel, and J. Revaud, "Superloss: A generic loss for robust curriculum learning," in *NeurIPS*, 2020, pp. 1–12.

[207] S. Zhang, Z. Li, S. Yan, X. He, , and J. Sun, "Distribution alignment: A unified framework for long-tail visual recognition," in *CVPR*, 2021, pp. 2361–2370.

[208] K. R. M. Fernando and C. P. Tsokos, "Dynamically weighted balanced loss: Class imbalanced learning and confidence calibration of deep neural networks," *IEEE TNNLS*, vol. 33, no. 7, pp. 2940–2951, 2021.

[209] E. Z. Liu, B. Haghgoo, A. S. Chen, A. Raghunathan, P. W. Koh, S. Sagawa, P. Liang, and C. Finn, "Just train twice: Improving group robustness without training group information," in *ICML*, 2021, pp. 6781–6792.

[210] J. Zhang, J. Zhu, G. Niu, B. Han, M. Sugiyama, and M. Kankanhalli, "Geometry-aware instance-reweighted adversarial training," in *ICLR*, 2021.

[211] L. Jiang, D. Meng, T. Mitamural, and A. G. Hauptmann, "Easy samples first: Self-paced reranking for zero-example multimedia search," in *ACM MM*, 2014, pp. 547–556.

[212] L. Jiang, D. Meng, S. Yu, Z. Lan, S. Shan, and A.-G. Hauptmann, "Self-paced learning with diversity," in *NeurIPS*, 2014, pp. 2078–2086.

[213] D. Zhang, D. Meng, C. Li, L. Jiang, Q. Zhao, and J. Han, "A self-paced multiple-instance learning framework for co-saliency detection," in *ICCV*, 2015, pp. 594–602.

[214] P. Soviany, R. T. Ionescu, P. Rota, and N. Sebe, "Curriculum learning: A survey," *IJCV*, vol. 130, no. 6, pp. 1526–1565, 2022.

[215] C. Santiagoa, C. Barataa, M. Sasdellib, G. Carneirob, and J. C.Nasciment, "Low: Training deep neural networks by learning optimal sample weights," *Pattern Recognition*, vol. 110, no. 1, pp. 1–12, 2021.

[216] P. Soviany, "Curriculum learning with diversity for supervised computer vision tasks," in *ICML Workshop*, 2020.

[217] X. Zhou and O. Wu, "Which samples should be learned first: Easy or hard?" *IEEE TNNLS*, pp. 1–15, 2023.

[218] W. Zhang, Y. Wang, and Y. Qiao, "Metacleaner: Learning to hallucinate clean representations for noisy-labeled visual recognition," in *CVPR*, June 2019, pp. 7373–7382.

[219] C. Northcutt, L. Jiang, and I. Chuang, "Confident learning: Estimating uncertainty in dataset labels," *JAIR*, vol. 70, pp. 1373–1411, 2021.

[220] Z. Han, Z. Liang, F. Yang, L. Liu, L. Li, Y. Bian, P. Zhao, B. Wu, C. Zhang, and J. Yao, "Umix: Improving importance weighting for subpopulation shift via uncertainty-aware mixup," in *NeurIPS*, 2022, pp. 37 704–37 718.

[221] T. Liu and D. Tao, "Classification with noisy labels by importance reweighting," *IEEE TPAMI*, vol. 38, no. 3, p. 447–461, 2023.

[222] Y. Fan, R. He, J. Liang, and B. Hu, "Self-paced learning: An implicit regularization perspective," in *AAAI*, 2017.

[223] X. Gu, X. Yu, Y. Yang, J. Sun, and Z. Xu, "Adversarial reweighting for partial domain adaptation," in *NeurIPS*, 2021, pp. 14 860–14 872.

[224] M. Yi, L. Hou, L. Shang, X. Jiang, Q. Liu, and Z.-M. Ma, "Reweighting augmented samples by minimizing the maximal expected loss," in *ICLR*, 2021.

[225] M. Ren, W. Zeng, B. Yang, and R. Urtasun, "Learning to reweight examples for robust deep learning," in *ICML*, 2018, pp. 4334–4343.

[226] Q. Zhao, J. Shu, X. Yuan, Z. Liu, and D. Meng, "A probabilistic formulation for meta-weight-net," *IEEE TNNLS*, vol. 34, no. 3, pp. 1194–1208, 2023.

[227] N. N. Trung, L. N. Van, and T. H. Nguyen, "Unsupervised domain adaptation for text classification via meta self-paced learning," in *COLING*, 2022, pp. 4741–4752.

[228] J. Wei, X. Xu, Z. Wang, and G. Wang, "Meta self-paced learning for cross-modal matching," in *ACM MM*, 2021, pp. 3835–3843.

[229] S. Li, W. Ma, J. Zhang, C. H. Liu, J. Liang, and G. Wang, "Meta-reweighted regularization for unsupervised domain adaptation," *IEEE TKDE*, vol. 35, no. 3, pp. 2781–2795, 2023.

[230] F. Zhou, J. Li, C. Xie, F. Chen, L. Hong, R. Sun, and Z. Li, "Metaaugment: Sample-aware data augmentation policy learning," in *AAAI*, 2021, pp. 11 097–11 105.

[231] Y. Ge, M. Rahmani, A. Irissappane, J. Sepulveda, J. Caverlee, and F. Wang, "Automated data denoising for recommendation," *arXiv:2305.07070*, 2023.

[232] T. Fang, N. Lu, G. Niu, and M. Sugiyama, "Rethinking importance weighting for deep learning under distribution shift," in *NeurIPS*, 2020, pp. 11 996–12 007.

[233] T. Wang, J. Y. Zhu, A. Torralba, and A. A. Efros, "Dataset distillation," *arXiv:1811.10959*, 2018.

[234] S. Lei and D. Tao, "A comprehensive survey of dataset distillation," *arXiv:2301.05603*, 2023.

[235] N. Sachdeva and J. McAuley, "Data distillation: A survey," *arXiv:2301.04272v1*, 2023.

[236] B. Zhao, K. R. Mopuri, and H. Bilen, "Dataset condensation with gradient matching," in *ICLR*, 2021.

[237] Z. Deng and O. Russakovsky, "Remember the past: Distilling datasets into addressable memories for neural networks," in *NeurIPS*, 2022.

[238] N. Loo, R. Hasani, A. Amini, and D. Rus, "Efficient dataset distillation using random feature approximation," in *NeurIPS*, 2022.

[239] Y. Zhou, E. Nezhadarya, and J. Ba, "Dataset distillation using neural feature regression," in *NeurIPS*, 2022.

[240] B. Zhao and H. Bilen, "Dataset condensation with differentiable siamese augmentation," in *ICML*, 2021, pp. 12 674–12 685.

[241] J.-H. Kim, J. Kim, S. J. Oh, S. Yun, H. Song, J. Jeong, J.-W. Ha, and H. O. Song, "Dataset condensation via efficient synthetic-data parameterization," in *ICML*, 2022.

[242] G. Cazenavette, T. Wang, A. Torralba, A. A. Efros, and J.-Y. Zhu, "Dataset distillation by matching training trajectories," in *CVPR*, 2022.

[243] J. Cui, R. Wang, S. Si, and C.-J. Hsieh, "Scaling up dataset distillation to imagenet-1k with constant memory," *arXiv:2211.10586*, 2022.

[244] B. Zhao and H. Bilen, "Dataset condensation with distribution matching," in *WACV*, 2023.

[245] K. Wang, B. Zhao, X. Peng, Z. Zhu, S. Yang, S. Wang, G. Huang, H. Bilen, X. Wang, and Y. You, "Cafe: Learning to condense dataset by aligning features," in *CVPR*, 2022, pp. 12 196–12 205.

[246] X. Zhou, R. Pi, W. Zhang, Y. Lin, Z. Chen, and T. Zhang, "Probabilistic bilevel coreset selection," in *ICML*, 2022, pp. 27 287–27 302.

[247] B. Zhao and H. Bilen, "Synthesizing informative training samples with gan," in *NeurIPS Workshop*, 2022.

[248] B. Sorscher, R. Geirhos, S. Shekhar, S. Ganguli, and A. S. Morcos., "Beyond neural scaling laws: beating power law scaling via data pruning," in *NeurIPS*, 2022.

[249] Z. Qin, K. Wang, Z. Zheng, J. Gu, X. Peng, D. Zhou, and Y. You, "Infobatch: Lossless training speed up by unbiased dynamic data pruning," *arXiv:2303.04947*, 2023.

[250] S. Liu, K. Wang, X. Yang, J. Ye, and X. Wang, "Dataset distillation via factorization," in *NeurIPS*, 2022.

[251] K. Meding, L. M. S. Buschoff, R. Geirhos, and F. A. Wichmann, "Trivial or impossible – dichotomous data difficulty masks model differences (on imagenet and beyond)," in *ICLR*, 2022.

[252] V. Feldman and C. Zhang, "What neural networks memorize and why: Discovering the long tail via influence estimation," in *NeurIPS*, 2020, pp. 2881–2891.

[253] C. G. Northcutt, T. Wu, and I. L. Chuang, "Learning with confident examples: Rank pruning for robust classification with noisy labels," *arXiv:1705.01936*, 2017.

[254] V. Kaushal, R. Iyer, S. Kothawade, R. Mahadev, K. Doctor, and G. Ramakrishnan, "Learning from less data: A unified data subset selection and active learning framework for computer vision," in *IEEE WACV*, 2019, pp. 1289–1299.

[255] Y. Yang, H. Kang, and B. Mirzasoleiman, "Towards sustainable learning: Coresets for data-efficient deep learning," in *ICML*, 2023.

[256] B. Mirzasoleiman, J. Bilmes, and J. Leskovec, "Coresets for data-efficient training of machine learning models," in *ICML*, 2020, pp. 6950–6960.

[257] Z. Liu, H. Jin, T.-H. Wang, K. Zhou, and X. Hu, "Divaug: Plug-in automated data augmentation with explicit diversity maximization," in *ICCV*, 2021, pp. 4762–4770.

[258] K. J. Joseph, K. Singh, and V. N. Balasubramanian, "Submodular batch selection for training deep neural networks," in *IJCAI*, 2019, pp. 2677–2683.

[259] W. Li, G. Dasarathy, and V. Berisha, "Regularization via structural label smoothing," in *AISTATS*, 2020, pp. 1453–1463.

[260] C. Meister, E. Salesky, and R. Cotterell, "Generalized entropy regularization or: There's nothing special about label smoothing," in *ACL*, 2020.

[261] J. Chai and X. Wang, "Fairness with adaptive weights," in *ICML*, 2022, pp. 2853–2866.

[262] Q. Song, J. Hin, X. Huang, and X. Hu, "Multi-label adversarial perturbations," in *ICDM*. IEEE, 2018, pp. 1242–1247.

[263] S. Hu, L. Ke, X. Wang, and S. Lyu, "Tkml-ap: Adversarial attacks to top-k multi-label learning," in *ICCV*, 2021, pp. 7649–7657.

[264] L. Kong, W. Luo, H. Zhang, Y. Liu, and Y. Shi, "Evolutionary multilabel adversarial examples: An effective black-box attack," *IEEE TAI*, vol. 4, no. 3, pp. 562–572, 2023.

[265] H. Cao, W. Yang, and H. T. Ng, "Mitigating exposure bias in grammatical error correction with data augmentation and reweighting," in *EACL*, 2023, pp. 2123–2135.

[266] Q. Liu, M. Kusner, and P. Blunsom, "Counterfactual data augmentation for neural machine translation," in *NAACL*, 2021, pp. 187–197.

[267] Y. Zang, C. Huang, and C. C. Loy, "Fasa: Feature augmentation and sampling adaptation for long-tailed instance segmentation," in *ICCV*, 2021.

[268] Y. Zhao, W. Chen, X. Tan, K. Huang, and J. Zhu, "Adaptive logit adjustment loss for long-tailed visual recognition," in *AAAI*, 2022, pp. 3472–3480.

[269] J.-H. Xue and P. Hall, "Why does rebalancing class-unbalanced data improve auc for linear discriminant analysis?" *IEEE TPAMI*, vol. 37, no. 5, pp. 1109–1112, 2015.

[270] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry, "Adversarial examples are not bugs, they are features," in *NeurIPS*, 2019.

[271] D. Elreedy, A. F. Atiya, and F. Kamalov, "A theoretical distribution analysis of synthetic minority oversampling technique (smote) for imbalanced learning," *Machine Learning*, 2023.

[272] Y. Yang, K. Zha, Y. Chen, H. Wang, and D. Katabi, "Delving into deep imbalanced regression," in *ICML)*, 2021, pp. 11 842–11 851.

[273] M. Li, Y.-m. Cheung, and Y. Lu, "Long-tailed visual recognition via gaussian clouded logit adjustment," in *CVPR*, 2022, pp. 6929–6938.

[274] J. Ren, M. Zhang, C. Yu, and Z. Liu, "Balanced mse for imbalanced visual regression," in *CVPR*, 2022, pp. 7926–7935.

[275] Y. Xie, M. Qiu, H. Zhang, L. Peng, and Z. Chen, "Gaussian distribution based oversampling for imbalanced data classification," *IEEE TKDE*, vol. 32, no. 2, pp. 667–679, 2022.

[276] Y. Yang and Z. Xu, "Rethinking the value of labels for improving class-imbalanced learning," in *NeurIPS*, 2020, pp. 19 290–19 301.

[277] H. Liu, J. Z. HaoChen, A. Gaidon, and T. Ma, "Self-supervised learning is more robust to dataset imbalance," *arXiv:2110.05025*, 2022.

[278] L. Jin, D. Lang, and N. Lei, "An optimal transport view of class-imbalanced visual recognition," *International Journal of Computer Vision*, 2023.

[279] Q. Dong, S. Gong, and X. Zhu, "Imbalanced deep learning by minority class incremental rectification," *IEEE TPAMI*, vol. 41, no. 6, pp. 1367–1381, 2019.

[280] K. A. Wang, N. S. Chatterji, S. Haque, and T. Hashimoto, "Is importance weighting incompatible with interpolating classifiers?" in *ICLR*, 2022.

[281] R. Xu, X. Zhang, Z. Shen, T. Zhang, and P. Cui, "A theoretical analysis on independence-driven importance weighting for covariate-shift generalization," in *ICML*, 2022, pp. 24 803–24 829.

[282] D. Chen, Y. Shen, H. Zhang, and P. H. Torr, "Zero-shot logit adjustment," in *IJCAI*, 2022.

[283] M. Qraitem, K. Saenko, and B. A. Plummer, "Bias mimicking: A simple sampling approach for bias mitigation," in *CVPR*, 2023, pp. 20 311–20 320.

[284] Y. Roh, K. Lee, S. Whang, and C. Suh, "Sample selection for fair and robust training," in *NeurIPS*, 2021, pp. 815–827.

[285] A. Zhang, F. Liu, W. Ma, Z. Cai, X. Wang, and T.-S. Chua, "Boosting causal discovery via adaptive sample reweighting," in *ICLR*, 2023.

[286] Y. Jang, T. Zhao, S. Hong, and H. Lee, "Adversarial defense via learning to generate diverse attacks," in *ICCV*, 2019.

[287] I. Hounie, L. F. O. Chamon, and A. Ribeiro, "Automatic data augmentation via invariance-constrained learning," in *ICML*, 2023, pp. 13 410–13 433.

[288] A. Blum and K. Stangl, "Recovering from biased data: Can fairness constraints improve accuracy?" *arXiv:1912.01094*, 2019.

[289] T. Doan, M. Abbana Bennani, B. Mazoure, G. Rabusseau, and P. Alquier, "A theoretical analysis of catastrophic forgetting through the ntk overlap matrix," in *AISTATS*, 2021, pp. 1072–1080.

[290] S. Chatterjee and P. Zielinski, "On the generalization mystery in deep learning," *arXiv:2203.10036*, 2022.

[291] A. Katharopoulos and F. Fleuret, "Biased importance sampling for deep neural network training," *arXiv:1706.00043*, 2017.

[292] Z. Wang, H. Zhu, Z. Dong, X. He, and S.-L. Huang, "Less is better: Unweighted data subsampling via influence function," in *AAAI*, 2020, pp. 6340–6347.

[293] T. Dao, A. Gu, A. Ratner, V. Smith, C. De Sa, and C. Re, "A kernel theory of modern data augmentation," in *ICML*, 2019, pp. 1528–1537.

[294] J. Wu and J. He, "A unified framework for adversarial attacks on multi-source domain adaptation," *IEEE TKDE*, pp. 1–12, 2022.

[295] J. Gilmer, N. Ford, N. Carlini, and E. Cubuk, "Adversarial examples are a natural consequence of test error in noise," in *ICML*, 2019, p. 2280–2289.

[296] M. Yi, L. Hou, J. Sun, L. Shang, X. Jiang, Q. Liu, and Z. Ma, "Improved ood generalization via adversarial training and pretraing," in *ICML*, 2021, pp. 11 987–11 997.

[297] J. Peck, J. Roels, B. Goossens, and Y. Saeys, "Lower bounds on the robustness to adversarial perturbations," in *NeurIPS*, 2017.

[298] Y. Xu, Y. Xu, Q. Qian, H. Li, and R. Jin, "Towards understanding label smoothing," *arXiv:2006.11653*, 2017.

[299] D. Meng, Q. Zhao, and L. Jiang, "A theoretical understanding of self-paced learning," *Information Sciences*, vol. 414, pp. 319–328, 2017.

[300] D. Weinshall, G. Cohen, and D. Amir, "Curriculum learning by transfer learning: Theory and experiments with deep networks," in *ICML*, 2018, pp. 5238–5246.

[301] D. Zhu, B. Lei, J. Zhang, Y. Fang, R. Zhang, Y. Xie, and D. Xu, "Rethinking data distillation: Do not overlook calibration," in *ICCV*, 2023.

[302] T. Dong, B. Zhao, and L. Lyu, "Privacy for free: How does dataset condensation help privacy?" *arXiv:2206.00240*, 2022.

[303] L. Yuan, F. E. Tay, G. Li, T. Wang, and J. Feng, "Revisiting knowledge distillation via label smoothing regularization," in *CVPR*, 2020.

[304] A. D. Assis, L. C. B. Torres, L. R. G. Araújo, V. M. Hanriot, and A. P. Braga, "Neural networks regularization with graph-based local resampling," *IEEE Access*, vol. 9, pp. 50 727–50 737, 2021.

[305] F. R. Cordeiro, V. Belagiannis, I. Reid, and G. Carneiro, "Propmix: Hard sample filtering and proportional mixup for learning with noisy labels," in *BMVC*, 2021.

[306] K. Yang, Y. Sun, J. Su, F. He, X. Tian, F. Huang, T. Zhou, and D. Tao, "Adversarial auto-augment with label preservation: A representation learning principle guided approach," in *NeurIPS*, 2022, pp. 22 035–22 048.

[307] J. Li, R. Socher, and S. C. Hoi, "Dividemix: Learning with noisy labels as semi-supervised learning," in *ICLR*, 2020.

[308] J. Shu, X. Yuan, D. Meng, and Z. Xu, "Cmw-net: Learning a class-aware sample weighting mapping for robust deep learning," *IEEE TPAMI*, vol. 45, no. 10, 2023.

[309] Z. Zhang and T. Pfister, "Learning fast sample re-weighting without reward data," in *ICCV*, 2021, pp. 725–734.

[310] X. Wang, E. Kodirov, Y. Hua, and N. M. Robertson, "Derivative manipulation for general example weighting," *arXiv:1905.11233*, 2020.

[311] E. Yang, T. Liu, C. Deng, W. Liu, and D. Tao, "Distillhash: Unsupervised deep hashing by distilling data pairs," in *CVPR*, 2019, pp. 2946–2955.

[312] B. Mirzasoleiman, K. Cao, and J. Leskovec, "Coresets for robust training of deep neural networks with noisy labels," in *NeurIPS*, 2020, pp. 11 465–11 477.

[313] S. Mindermann, J. M. Brauner, M. T. Razzak, M. Sharma, A. Kirsch, W. Xu, B. Höltgen, A. N. Gomez, A. Morisot, S. Farquhar, and Y. Gal, "Prioritized training on points that are learnable, worth learning, and not yet learnt," in *ICML*, 2022, pp. 15 630–15 649.

[314] X. Hu, Y. Zeng, X. Xu, S. Zhou, and L. Liu, "Robust semi-supervised classification based on data augmented online elms with deep features," *Knowledge-Based Systems*, vol. 229, p. 107307, 2021.

[315] S. Kim, S. Bae, and S.-Y. Yun, "Coreset sampling from open-set for fine-grained self-supervised learning," in *CVPR*, 2023, pp. 7537–7547.

[316] C. Bellinger, R. Corizzo, and N. Japkowicz, "Remix: Calibrated resampling for class imbalance in deep learning," *arXiv:2012.02312*, 2020.

[317] F. Du, P. Yang, Q. Jia, F. Nan, X. Chen, and Y. Yang, "Global and local mixture consistency cumulative learning for long-tailed visual recognitions," in *CVPR*, June 2023, pp. 15 814–15 823.

[318] O. Pooladzandi, D. Davini, and B. Mirzasoleiman, "Adaptive second order coresets for data-efficient machine learning," in *ICML*, 2022, pp. 17 848–17 869.

[319] G. Zhao, G. Li, Y. Qin, and Y. Yu, "Improved distribution matching for dataset condensation," in *CVPR*, 2023, pp. 7856–7865.

[320] K. Kim and H. S. Lee, "Probabilistic anchor assignment with iou prediction for object detection," in *ECCV*, 2020, pp. 355–371.

[321] T. Takase, R. Karakida, and H. Asoh, "Self-paced data augmentation for training neural networks," *Neurocomputing*, vol. 442, pp. 296–306, 2021.

[322] S. A. Siddiqui, N. Rajkumar, T. Maharaj, D. Krueger, and S. Hooker, "Metadata archaeology: Unearthing data subsets by leveraging training dynamics," in *ICLR*, 2023.

[323] X. Peng, F.-Y. Wang, and L. Li, "Mixgradient: A gradient-based re-weighting scheme with mixup for imbalanced data streams," *Neural Networks*, vol. 161, pp. 525–534, 2023.

[324] W. Xu, L. Jiang, and C. Li, "Resampling-based noise correction for crowdsourcing," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 33, no. 6, pp. 985–999, 2020.

[325] C. Huang and S. Zhang, "Generative dataset distillation," in *BigCom*, 2021, pp. 212–218.

[326] A. Krizhevsky, "Learning multiple layers of features from tiny images." MIT, 2009.

[327] T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang, "Learning from massive noisy labeled data for image classification," in *CVPR*, 2015, p. 2691–2699.

[328] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *NeurIPSW*, 2011.

[329] W. Li, L. Wang, W. Li, E. Agustsson, and L. Van Gool, "Webvision database: Visual learning and understanding from web data," *arXiv:1708.02862*, 2017.

[330] G. Van Horn, O. Mac Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, and S. Belongie, "The inaturalist species classification and detection dataset," in *CVPR*, 2018, pp. 8769–8778.

[331] Z. Liu, Z. Miao, X. Zhan, J. Wang, B. Gong, and S. X. Yu, "Largescale long-tailed recognition in an open world," in *CVPR*, 2019, p. 2537–2546.

[332] S. P. Karimireddy, L. He, and M. Jaggi, "Byzantine-robust learning on heterogeneous datasets via bucketing," *arXiv:2006.09365*, 2022.

[333] N. Tsilivis, J. Su, and J. Kempe, "Can we achieve robustness from data alone?" *arXiv:2006.09365*, 2022.

[334] Z. Liu, P. Wei, J. Jiang, W. Cao, J. Bian, and Y. Chang, "Mesa: Boost ensemble imbalanced learning with meta-sampler," in *NeurIPS*, 2020, pp. 14 463–14 474.

[335] Y. Li, X. Liu, and F. Liu, "Adaptive noisy data augmentation for regularization of undirected graphical models," *arXiv:1810.04851*, 2019.

[336] C. Yu, B. Han, L. Shen, J. Yu, C. Gong, M. Gong, and T. Liu, "Understanding robust overfitting of adversarial training and beyond," in *ICML*, 2022, pp. 25 595–25 610.

[337] J. Xiao, Y. Fan, R. Sun, J. Wang, and Z.-Q. Luo, "Stability analysis and generalization bounds of adversarial training," in *NeurIPS*, 2022, pp. 15 446–15 459.

[338] J. An, L. Ying, and Y. Zhu, "Why resampling outperforms reweighting for correcting sampling bias with stochastic gradients," *arXiv:2009.13447*, 2021.

[339] R. Müller, S. Kornblith, and G. E. Hinton, "When does label smoothing help?" in *NeurIPS*, 2019.

[340] B. Chen, L. Ziyin, Z. Wang, and P. P. Liang, "An investigation of how label smoothing affects generalization," *arXiv:2010.12648*, 2020.

[341] Z. Qian, K. Huang, Q.-F. Wang, and X.-Y. Zhang, "A survey of robust adversarial training in pattern recognition: Fundamental, theory, and methodologies," *Pattern Recognition*, vol. 131, p. 108889, 2022.

[342] K. H. R. Chan, Y. Yu, C. You, H. Qi, J. Wright, and Y. Ma, "Redunet: a white-box deep network from the principle of maximizing rate reduction," *Journal of Machine Learning Research*, vol. 23, no. 1, pp. 4907–5009, 2022.

[343] Y. Wen, G. Jerfel, R. Muller, M. W. Dusenberry, J. Snoek, B. Lakshminarayanan, and D. Tran, "Combining ensembles and data augmentation can harm your calibration," in *ICLR*, 2021.

[344] M. Lukasik, S. Bhojanapalli, A. K. Menon, and S. Kumar, "Does label smoothing mitigate label noise?" in *ICML*, 2020.

[345] M. Du, N. Liu, and X. Hu, "Techniques for interpretable machine learning," *Communications of the ACM*, vol. 63, pp. 68–77, 2019.

[346] C. R. Pochimireddy, A. T. Siripuram, and S. S. Channappayya, "Can perceptual guidance lead to semantically explainable adversarial perturbations?" *IEEE J-STSP*, pp. 1–11, 2023.

[347] G. Zelaya and C. Vladimiro, "Towards explaining the effects of data preprocessing on machine learning," in *ICDE*, 2019, pp. 2086–2090.

[348] E. Mosqueira-Rey, E. Hernández-Pereira, D. Alonso-Ríos, J. Bobes-Bascarán, and Ángel Fernández-Leal, "Human-in-the-loop machine learning: a state of the art," *Journal of Machine Learning Research*, vol. 56, pp. 3005–3054, 2023.

[349] K. M. Collins, U. Bhatt, W. Liu, V. Piratla, I. Sucholutsky, B. Love, and A. Weller, "Human-in-the-loop mixup," in *UAI*, 2023, pp. 454–464.

[350] E. Wallace, P. Rodriguez, S. Feng, I. Yamada, and J. Boyd-Graber, "Trick me if you can: Human-in-the-loop generation of adversarial examples for question answering," *TACL*, vol. 7, pp. 387–401, 2019.

[351] C. Agarwal, D. D'souza, and S. Hooker, "Estimating example difficulty using variance of gradients," in *CVPR*, 2022, pp. 10 368–10 378.

[352] Z.-F. Wu, T. Wei, J. Jiang, C. Mao, M. Tang, and Y.-F. Li, "Ngc: a unified framework for learning with open-world noisy data," in *ICCV*, 2021, p. 62–71.

[353] Z. Jiang, T. Chen, T. Chen, and Z. Wang, "Improving contrastive learning on imbalanced data via open-world sampling," in *NeurIPS*, 2021, pp. 5997–6009.

[354] T. Gokhale, S. Mishra, M. Luo, B. S. Sachdeva, and C. Baral, "Generalized but not robust? comparing the effects of data modification methods on out-of-domain generalization and adversarial robustness," in *ACL Findings*, 2022.

[355] W. X. Zhao, K. Zhou, and et al., "A survey of large language models," *arXiv:2303.18223*, 2023.

[356] L. Wei, Z. Jiang, W. Huang, and L. Sun, "Instructiongpt-4: A 200-instruction paradigm for fine-tuning minigpt-4," *arXiv:2308.12067*, 2023.

[357] M. Jiang, Y. Ruan, S. Huang, S. Liao, S. Pitis, R. Grosse, and J. Ba, "Calibrating language models via augmented prompt ensembles," in *ICML*, 2023.

[358] T. Baltrušaitis, C. Ahuja, and L.-P. Morency, "Multimodal machine learning: A survey and taxonomy," *IEEE TPAMI*, vol. 41, no. 2, pp. 423–443, 2019.

[359] S. Ge, Z. Jiang, Z. Cheng, C. Wang, Y. Yin, and Q. Gu, "Learning robust multi-modal representation for multi-label emotion recognition via adversarial masking and perturbation," in *The Web Conference*, 2023, pp. 1510–1518.

[360] V. A. Trinh, H. Salami Kavaki, and M. I. Mandel, "Importantaug: A data augmentation agent for speech," in *ICASSP*, 2022, pp. 8592–8596.

[361] M. Li, X. Zhang, C. Thrampoulidis, J. Chen, and S. Oymak, "Auto-balance: Optimized loss functions for imbalanced data," in *NeurIPS*, 2021, pp. 3163–3177.