Foundations and Trends® in Privacy and Security

Trustworthy Machine Learning: From Data to Models

Suggested Citation: Bo Han, Jiangchao Yao, Tongliang Liu, Bo Li, Sanmi Koyejo and Feng Liu (2025), "Trustworthy Machine Learning: From Data to Models", Foundations and Trends® in Privacy and Security: Vol. 7, No. 2-3, pp 74–246. DOI: 10.1561/3300000043.

Bo Han
Hong Kong Baptist University
bhanml@comp.hkbu.edu.hk

Jiangchao Yao
Shanghai Jiao Tong University
sunarker@sjtu.edu.cn

Tongliang Liu
The University of Sydney
tongliang.liu@sydney.edu.au

Bo Li
University of Illinois Urbana-Champaign
lbo@illinois.edu

Sanmi Koyejo
Stanford University
sanmi@cs.stanford.edu

Feng Liu
The University of Melbourne
feng.liu1@unimelb.edu.au

now
the essence of knowledge
Boston — Delft

Contents

Trustworthy Machine Learning: From Data to Models

Bo Han[1], Jiangchao Yao[2], Tongliang Liu[3], Bo Li[4], Sanmi Koyejo[5] and Feng Liu[6]

[1]*Hong Kong Baptist University, Hong Kong and RIKEN, Japan; bhanml@comp.hkbu.edu.hk*
[2]*Shanghai Jiao Tong University, China; sunarker@sjtu.edu.cn*
[3]*The University of Sydney, Australia, MBZUAI, UAE and RIKEN, Japan; tongliang.liu@sydney.edu.au*
[4]*University of Illinois Urbana-Champaign, USA; lbo@illinois.edu*
[5]*Stanford University, USA; sanmi@cs.stanford.edu*
[6]*The University of Melbourne, Australia and RIKEN, Japan; feng.liu1@unimelb.edu.au*

ABSTRACT

The success of machine learning algorithms relies not only on achieving good performance but also on ensuring trustworthiness across diverse applications and scenarios. Trustworthy machine learning seeks to handle critical problems in addressing the issues of robustness, privacy, security, reliability, and other desirable properties. The broad research area has achieved remarkable advancement and brings various emerging topics along with the progress. We present this survey to provide a systematic overview of the research problems under trustworthy machine learning covering the perspectives from data to model. Starting with fundamental data-centric learning, the survey reviews learning with noisy data, long-tailed distribution, out-of-distribution data,

Bo Han, Jiangchao Yao, Tongliang Liu, Bo Li, Sanmi Koyejo and Feng Liu (2025), "Trustworthy Machine Learning: From Data to Models", Foundations and Trends® in Privacy and Security: Vol. 7, No. 2-3, pp 74–246. DOI: 10.1561/3300000043.

and adversarial examples to achieve robustness. Delving into private and secured learning, the survey elaborates on core methodologies differential privacy, different attacking threats, and learning paradigms, to realize privacy protection and enhance security. Finally, it introduces several trendy issues related to the foundation models, including jailbreak prompts, watermarking, and hallucination, as well as causal learning and reasoning. The survey integrates commonly isolated research problems in a unified manner, which provides general problem setups, detailed sub-directions, and further discussion on its challenges or future developments. We hope the comprehensive investigation presented in this survey can serve as a clear introduction for the problem evolution from data to models and also bring new insight for developing trustworthy machine learning.

1

Introduction

As artificial intelligence (AI) and machine learning (ML) experience advancements rapidly, remarkable breakthroughs have been achieved across a variety of scenarios and applications (Jordan and Mitchell, 2015). These technologies of AI and ML have increasingly become cornerstones of innovation, driving progress in the fields such as healthcare diagnostics (Alowais *et al.*, 2023), autonomous vehicles (Betz *et al.*, 2022), financial modeling (Cao, 2022), protein structure prediction (Abramson *et al.*, 2024), and numerous other domains. Despite these impressive achievements, the trustworthiness of AI systems has come under scrutiny, particularly in security-critical and privacy-sensitive domains. Ensuring that AI systems and ML models are reliable, secure, and trustworthy is not merely desirable but essential for their deployment in large-scale real-world applications.

The heart of machine learning is built upon two crucial aspects: data and model. Data serves as the fundamental resource, representing the diverse, complex, and often noisy real-world phenomena. Meanwhile, the model functions as the learner, with specific model architectures that absorb patterns and knowledge from the data, empowering it to make predictions or decisions in previously unseen scenarios (LeCun

et al., 2015). Notably, the emergence of data privacy and security issues arises at the intersection of data and model, as training data may inadvertently include malicious information that implants backdoors in models (Li *et al.*, 2022b), or contain sensitive privacy details that models could unintentionally expose during inference (Liu *et al.*, 2021a). Overall, these challenges highlight the need to understand and develop trustworthy machine learning from data to model, including perspectives of data-centric methods, privacy and security, and foundation models.

In this monograph, we first discuss the trustworthy data-centric learning, which emphasizes the risks associated with noisy (Song *et al.*, 2022a), long-tailed (Zhang *et al.*, 2023e), out-of-distribution (Yang *et al.*, 2024), and adversarial data (Wang *et al.*, 2019). As the foundation of any ML models, the data directly impacts the model's reliability and generalization ability. Trustworthy data-centric learning focuses on exploring the essential mechanisms by which data influences the trustworthiness of ML models, and designing robust approaches to adapt to, defend against, and mitigate the negative effects of such data challenges. This includes developing robust algorithms for learning from noisy data, handling long-tailed distributions, detecting and generalizing across out-of-distribution data, and improving adversarial robustness and defense strategies. Generally, the goal of trustworthy data-centric learning is to ensure the trustworthiness of ML models from data perspectives, enabling them to handle diverse and complex real-world scenarios while maintaining highly accurate performance.

Privacy and security problems are paramount in the deployment of machine learning models, particularly when dealing with sensitive data (Liu *et al.*, 2021a), such as finance and healthcare. In this monograph, we delve deeply into the approaches that attacking and safeguarding ML systems, addressing challenges from both the data and model perspectives. Specifically, we start with discussing differential privacy (Abadi *et al.*, 2016), a key technique that adds controlled noise to protect privacy while maintaining data utility. We then review two major privacy threats: membership inference attacks (Hu *et al.*, 2022b) and model inversion attacks (Song and Namiot, 2022), both of which attempt to aim to leak information from the training data. Next, we cover data poisoning attacks (Fan *et al.*, 2022) that degrade the model perfor-

mance by manipulating the training data. Additionally, we discuss three types of promising approaches, including machine unlearning (Nguyen *et al.*, 2022), non-transfer learning (Niu *et al.*, 2020) and federated learning (Zhang *et al.*, 2021a), all of which offer solutions to alleviate these risks and enhance trustworthiness of models in defending against such attacks. Overall, these research efforts highlight the vulnerabilities of models and contribute to enhancing the robustness and privacy protection.

Recently, the development of large foundation models, such as ChatGPT, Llama, and Gemini, has revolutionized the field of ML, paving the pathway to artificial general intelligence (Zhou *et al.*, 2024a). Despite their remarkable capacities, these foundation models still face various safety concerns. In this monograph, we discuss several potential risks and vulnerabilities of foundation models, aiming to highlight their weaknesses and provide insights for constructing trustworthy foundation models. In particular, we discuss jailbreak prompts that inveigle foundation models to generate harmful content (Yi *et al.*, 2024), and then review watermarking techniques to ensure content provenance and copyright (Liu *et al.*, 2024a). We next introduce hallucination, which is a critical issue for foundation models in generating unreliable and spurious content (Rawte *et al.*, 2023). Moreover, we discuss causal learning and reasoning methods (Chi *et al.*, 2024a) to enhance reliability of the content generated by foundation models. Finally, we compare different trustworthy concerns in open and proprietary foundation models with their distinct properties. In short, this monograph provides a comprehensive review and discussion of the key challenges and advancements in developing trustworthy machine learning systems, from data-centric approaches, privacy and security concerns to foundation models.

Overview. The monograph is organized around core aspects of trustworthy machine learning from data to models, including *data-centric learning*, *private and secured learning*, and *foundation models*.

- **Trustworthy Data-centric Learning.** First, we start with a systematic review of trustworthy data-centric learning, covering *data-noise learning*, *long-tailed learning*, *out-of-distribution learn-*

ing, as well as *adversarial examples and defense*. These research topics cover fundamental problems regarding data-level issues, e.g., label noise, shifted distribution, outliers, and worst-case corruption. We further categorize specific research problems under the general learning paradigms for discussion.

- **Trustworthy Private and Secured Learning.** Second, we focus on aspects of private and secured learning, covering *differential privacy*, *membership inference attack*, *model inversion attack*, *data poisoning attack*, *machine unlearning*, *non-transferable learning*, and *federated learning*. Considering the trustworthy expectation of privacy, security, and usage or ownership protection, we review a series of critical technologies and research problems.

- **Trustworthy Foundation Models.** Finally, we explore building the trustworthy foundation models, covering *jailbreak prompts*, *watermarking*, *hallucination*, *casual learning and reasoning*, as well as comparison on *open and proprietary foundation models*. These research problems reveal the vulnerability of foundation models in usage control and point the way to developing robust and reliable model learning and reasoning.

In each part, we elaborate on the detailed problem setup and methodology or research directions, for which we conduct a further discussion on promising future development in the problem. We hope this monograph can provide a comprehensive investigation from data to models in trustworthy machine learning and more new insights.

Target Audience and Reading Guidelines. This monograph is intended for researchers, professionals, and graduate students working in the fields of ML and AI. Some sections may contain technical descriptions and discussions that assume a basic knowledge of core ML concepts. Target readers are expected to have a foundational understanding of these key concepts, including supervised, semi-supervised and unsupervised learning, optimization methods, representation learning, federated learning, among others. For undergraduate students or

early-year graduate students new to the field, we recommend first referring to conventional textbooks on ML (Jordan and Mitchell, 2015) and deep learning (LeCun *et al.*, 2015) before delving into the specialized topics covered in this monograph. Additionally, readers unfamiliar with trustworthy ML may benefit from the monograph to establish a solid background of key challenges, methodologies, and emerging research directions in the field. Each section is organized to offer a thorough review and perceptive discussions of its respective topic. This structure makes the monograph suitable for both newcomers looking for a thorough grasp of trustworthy ML and seasoned researchers hoping to further advance the field. We hope this monograph serves as a valuable resource for a broad audience interested in developing more trustworthy and reliable ML systems.

Discussion on the Topic Coverage. In this monograph, our primary focus is on the critical technical aspects of trustworthy machine learning, specifically addressing issues of robustness, privacy, security, and reliability. Our goal is to provide a systematic overview of the evolving research landscape from a data-centric perspective, and consider the privacy and security challenges to building trustworthy foundation models. The overall discussion is with a particular emphasis on data quality, privacy protection, and security challenges. However, we should acknowledge that we don't explicitly cover all the topics that are also important and highly relevant to the field of trustworthy machine learning, such as fairness and bias (Mehrabi *et al.*, 2021; Pessach and Shmueli, 2022), ethics (Chen *et al.*, 2021a; Holmes *et al.*, 2022), and explainability (Došilović *et al.*, 2018; Murdoch *et al.*, 2019). For instance, ML systems can often mirror and amplify biases present in their training data and reflect stereotypes in their outputs, causing a broader concern of algorithm discrimination. In essence, without properly examining the training data, the training stage of the model would reinforce historical or societal prejudices that favor or over-present dominant groups and under-present or mischaracterize minorities.

Moreover, the non-transparency property of ML models increases the difficulty of effective auditing for untrustworthy issues. We do not explicitly discuss fairness and bias, ethics, or interpretability in separated

sections because these issues often require a distinct and deep dive into the societal and moral implications of technologies. However, we recognize the strong connections between these research topics and the core themes of our content. For example, issues of fairness and bias are often intertwined with the robustness considerations discussed in the context of adversarial examples and noisy data, as biased data can lead to unfair and unreliable model outcomes. Similarly, the methodologies explored for privacy protection, such as differential privacy, can contribute to ethical practices by safeguarding sensitive information and ensuring compliance with ethical standards in data handling. Finally, the discussion on casual learning and reasoning for trustworthy foundation models also touches on explainability in the sense that building reliable models often requires transparent and understandable methodologies. We hope this work will complement existing literature and encourage further exploration of these important areas within the context of trustworthy machine learning.

2

Trustworthy Data-centric Learning

Models are trained using data, which serves as the foundation for their learning and performance. The quality and distribution of this data play a pivotal role in determining the effectiveness of machine learning systems. This is where the data-centric approach comes into play, shifting the focus from optimizing model architectures to ensuring the integrity and reliability of the data itself. By emphasizing the importance of data quality, fairness, and robustness, the data-centric paradigm seeks to build more trustworthy, transparent, and reliable AI systems. Trustworthy data-centric learning involves addressing key dimensions such as data quality, fairness, transparency, and robustness. It tackles challenges like noise in data, long-tailed distributions, shifts in data distributions, and vulnerabilities to adversarial attacks, all while upholding stringent standards for data curation and validation.

This section provides a cohesive exploration of four critical aspects in data-level that can undermine the reliability of modern machine learning systems. First, Section 2.1 examines how various forms of noisy data—such as correspondence noise, graph noise, and group-structured noise—can affect model performance. These noise-handling paradigms lay the groundwork for understanding broader data imperfections, set-

ting the stage for subsequent sections. Building on this discussion of imperfect data, Section 2.2 shifts the focus to long-tailed distributions. While noise presents one source of difficulty for learning, heavily skewed class distributions pose another significant challenge from an optimization perspective. The methods introduced here—ranging from supervised to weakly-supervised and self-supervised strategies—benefit from the insights gained in the previous part regarding robust model training under imperfect conditions. Extending beyond issues of label noise and imbalance, Section 2.3 addresses the problem of out-of-distribution detection and generalization. The techniques presented here are complementary to those in noisy and imbalance learning, collectively emphasizing the importance of robust feature representations and adaptable optimization strategies. Finally, Section 2.4 highlights the ultimate test of adversarial attacks. These attacks exploit vulnerabilities often revealed by the same factors—noise, imbalance, and distributional shifts—discussed in the earlier sections. By examining adversarial examples and related defense mechanisms, this last section rounds out the section's coverage of strategies for building truly robust machine learning models.

2.1 Data-noise Learning

Data-noise learning considers scenarios in which some of the supervision signals used during training are incorrect. This challenge is prevalent in practical applications; even meticulously conducted manual annotations can result in a significant number of errors. In this section, we explore various types of data-noise setups, including correspondence noise learning, graph-noise learning, and group-structured noise learning. These categories reflect the key paradigms for data-noise learning, as illustrated in Figure 2.1.

Overall, correspondence noise learning focuses on the mismatches between data pairs or wrong correspondence across different modalities. Graph-noise learning examines errors within graph neural networks, which may involve incorrect relational data or mislabeled nodes. Group-structured noise learning encompasses scenarios where noise affects a group of instances or labels, including multiple-instance learning, partial

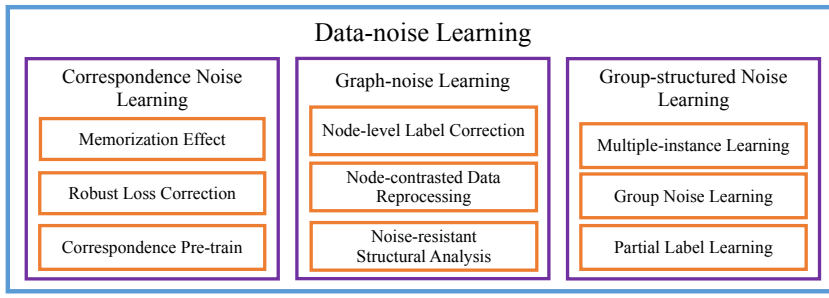

Figure 2.1: The overall framework for data-noise learning.

label learning, and group noise learning. These learning paradigms represent the cutting edge of settings for data-noise learning, offering a comprehensive overview for the current progress in the literature.

2.1.1 Correspondence Noise Learning

While the issue of learning with noisy data has been extensively studied in the context of unimodal learning (Han *et al.*, 2018b; Han *et al.*, 2018a; Li *et al.*, 2020b; Liu *et al.*, 2020c), it remains relatively under-explored in the domain of multimodal learning. Different from unimodal settings where noisy label serving as a primary concern, real-world multimodal datasets which contain non-expert annotations or collected by web crawling are prone to a specific challenge known as the noisy correspondence problem. Noisy correspondence refers to the mismatches in cross-modal data pairs, of which the discrepancy can lead to severe degradation on performance of downstream tasks like cross-modal retrieval (Zhen *et al.*, 2019; Diao *et al.*, 2021) and vision-language pre-training (Radford *et al.*, 2021a; Li *et al.*, 2022a; Liu *et al.*, 2024f). To intuitively present the correspondence noise learning, we illustrate the problem comparison in Figure 2.2.

Problem Setup. Consider a multimodal dataset $\mathcal{D} = \{x_1^i, x_2^i, c\}_{i=1}^N$, where x_1^i and x_2^i represents separate modalities from the i-th multimodal sample pair. Standard multimodal frameworks including SGRAF (Diao *et al.*, 2021) and CLIP (Radford *et al.*, 2021a), aim to project these data pairs into a shared representation space using separate encoders

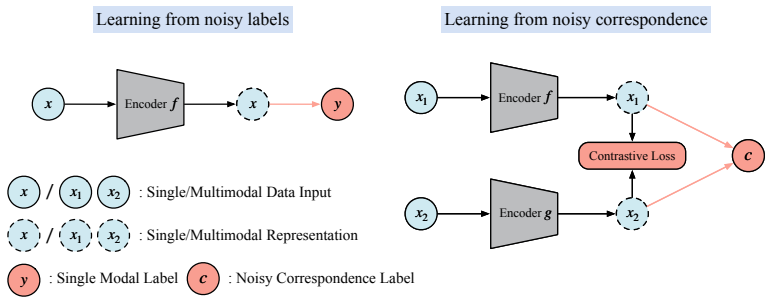

Figure 2.2: Problem setting of correspondence noise learning compared to learning from noisy labels. Standard framework includes separate encoders f and g extracting modality features into representations. Label c indicates the correspondence relation, which can be noisy given by the dataset.

f for input x_1 and g for input x_2. Then, similarity scores between the representations are computed through cosine similarity or an inference model, denoting as $S(f(x_1), g(x_2))$. The associated label c_i indicates whether the pair is positively correlated ($c_i = 1$) or not ($c_i = 0$), which may contain incorrect annotations when the pairs are noisily matched.

Memorization-effect-driven Methods. The noisy correspondence problem was first aroused by NCR (Huang *et al.*, 2021b), which explores to leverage the memorization effect[1] in deep learning to help distinguish samples with noisy correspondence. However, due to the distinct training paradigm in multimodal learning and the additional complexity of noisy correspondence, uni-modal robust methods can hardly be applied directly. To address this, NCR combines the DivideMix (Li *et al.*, 2020b) framework with unsupervised contrastive multimodal training. Specifically, NCR splits the training data into clean and noisy subsets through a two-component Gaussian Mixture Model (GMM), in which the clean subset can be directly learned, while the noisy one is adaptively optimized by estimating a soft margin for the triplet loss, which is further stabilized by introducing dual networks in a co-training curriculum.

[1]The memorization effect refers to that deep neural networks tend to initially learn the clean patterns within the dataset before over-fitting on noise, which has long been extensively studied in learning from noisy labels (Li *et al.*, 2020b; Liu *et al.*, 2020c; Liu *et al.*, 2022b).

Although NCR provides a well-established paradigm for learning from noisy correspondence, it still achieves unsatisfactory accuracy in distinguishing noisy samples. BiCro (Yang *et al.*, 2023a) improves upon NCR by adopting a Beta Mixture Model and estimating soft correspondence labels by sample-wise comparison. MSCN (Han *et al.*, 2023b) introduces a meta similarity correction network that reinterprets the binary classification of correspondence as a meta process, enhancing the data purification process. CTPR (Feng *et al.*, 2023) refines the original GMM by incorporating three components, which helps identify challenging samples that are beneficial for multimodal contrastive learning. CREAM (Ma *et al.*, 2024) further advances this approach by employing a collaborative learning paradigm to detect positive samples with a negative mining approach to maintain consistency.

However, the above methods all depend on the DivideMix framework, which exhibits high training costs and limited effectiveness in addressing real-world noisy correspondence issues. Besides, real-world noisy correspondences are often caused by lopsided observations, where data pairs from separate modalities are frequently merely partially mismatched. Such discrepancy can hardly be captured and split by simply fitting Gaussian distributions to the training losses. To address this, ACL (Qin *et al.*, 2023) proposed an active complementary loss and a self-refining correspondence correction mechanism to enhance robustness and reduce error accumulation. GSC (Zhao *et al.*, 2024c) explored the structural differences from both cross-modal and intra-modal aspects to accurately predict true correspondence labels and counteract the adverse effects of noisy correspondences. It incorporates a temporal ensembling strategy, ensuring stable performance with both single and dual networks.

Robust-loss-based Methods. In addition to leveraging the memorization effect, another line of works introduce robust loss functions to mitigate noisy correspondences with lower computational costs. RINCE (Chuang *et al.*, 2022) proposed a robust symmetric losses for combating noisy views in binary classification tasks. DECL (Qin *et al.*, 2022) combined the concept of evidential learning with noisy correspondence and puts forward a confidence-based method. RCL (Hu *et al.*, 2023) adopted a complementary contrastive learning paradigm to address both problems of overfitting issues on partially mismatched

pairs and under-fitting in weakly supervision settings. In the context of video-language pre-training, video clips are often split into frames and aligned with synchronized subtitles for computational efficiency. However, dividing videos into short clips would inevitably lead to temporal misalignment between subtitles and visual clips. In this scenario, Norton (Lin *et al.*, 2024c) proposed a video-paragraph and a clip-caption contrastive loss based on optimal transport (OT), which explores temporal correlations between video and text. Similarly, the OT principle can also be used to solve the partially mismatched pairs problem. For example, L2RM (Han *et al.*, 2024) proposed generating refined alignments by finding a minimal-cost transport plan across modalities.

Noisy Correspondence in LVLM Pre-training. Learning from noisy correspondence is a relatively new issue in the field of data-noise learning, but it has emerged as a promising research direction in real-world applications. With the success of Large Vision-Language Models (LVLMs), the pre-training of such models relies on vast amounts of paired vision-language data, which are often collected through web crawling and can contain noisy correspondences. For example, the popular CC3M dataset (Sharma *et al.*, 2018) was filtered from a 5-billion collection, and yet was noted to contain noisy correspondence in 3% to 20% of the data pairs. This phenomenon highlights that noisy correspondence learning has a wide range of application scenarios. For instance, CLIPScore (Hessel *et al.*, 2021) discovered that the pre-trained CLIP could be utilized for robust automatic evaluation of image captioning without reference captions, enabling the filtering of multimodal data. BLIP (Li *et al.*, 2022a) introduced a CapFilt module pre-trained on clean dataset to identify and refine low-quality data for pre-training. NLIP (Huang *et al.*, 2023a) estimated the noise probability of each pair based on the memorization effect and incorporated a concept-conditioned cross-modal decoder to generate complete captions for noisy samples. NEVER (Huang *et al.*, 2024e) explored to first split original dataset into clean and noisy subsets, and then employed the positive and negative learning strategies to enhance the model convergence and the noise robustness.

Further Discussion. Learning from noisy correspondence continues to evolve, but several challenges remain unresolved. First, many

modern robust methods rely on the memorization effect to distinguish noisy samples during the early stages of training. However, this strategy demonstrates limited effectiveness in real-world scenarios. In practice, real-world noisy correspondences are often more complex and harder to distinguish from challenging samples (Li *et al.*, 2022a). Second, current robust methods typically require high computational costs, which become unaffordable when training large models on large-scale datasets. Most existing approaches rely on a dual-network co-training framework, which doubles the training costs. Lastly, unlike unimodal learning, multimodal methods face unique challenges, such as modality asymmetry and missing modalities (Zhang *et al.*, 2022g; Chen *et al.*, 2023a). These issues complicate learning from noisy correspondences and demand new strategies specifically designed to address multimodal difficulties. With the rapid growth of deep neural networks and large vision-language models, future strategies for handling noisy correspondences should be designed in pace with LVLMs to address these challenges more efficiently. Furthermore, the development of noisy correspondence robust methods may facilitate better high-quality data selection for foundation model pre-training and fine-tuning.

2.1.2 Graph-Noise Learning

Graph-noise learning has emerged as a significant focus of study in the field of graph neural networks (GNNs). It largely tackles the problems of label noise, which can significantly influence GNN performance in realistic applications (NT *et al.*, 2019; Dai *et al.*, 2021; Wang *et al.*, 2024e). The purpose of this concept is to enhance the robustness of GNN in the face of noisy labels, which are often presented in a large number of datasets as a result of adversarial attacks or erroneous data sources (Wang *et al.*, 2024e). These initiatives are particularly crucial because the quality of node labels is under threat in many fields such as bioinformatics, knowledge graphs, social networks, and recommender systems (NT *et al.*, 2019; Dai *et al.*, 2021; Wang *et al.*, 2024e).

Problem Setup. Research has indicated that adding loss correction methods to GNN training processes can improve test accuracy under synthetic symmetric label noise settings (Dai *et al.*, 2021). Given a

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes, $\mathcal{E}$ is the set of edges, $\mathbf{A} \in \mathbf{R}^{N \times N}$ is the adjacency matrix, and $\mathbf{X} \in \mathbf{R}^{N \times d}$ denotes the node features. Let $\mathcal{Y}_{\text{noise}}$ represent a small set of nodes $\mathcal{V}_{\text{noise}} \in \mathcal{V}$ with noisy labels. As shown in Figure 2.3, the goal is to train a GNN classifier f_θ to predict the labels $\mathcal{Y}_{\text{pred}}$ of the unlabeled nodes: $f_\theta(\mathcal{G}, \mathcal{Y}_{\text{noise}}) \rightarrow \mathcal{Y}_{\text{pred}}$.

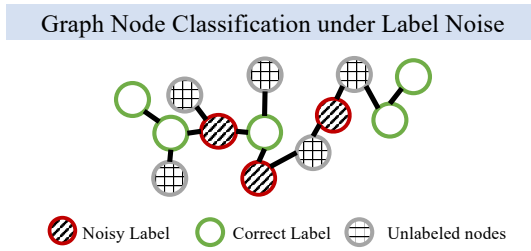

Figure 2.3: A demonstration of the graph noisy label problem.

Node-level Label Correction are crucial for enhancing the performance of GNNs. A prevalent method is label propagation, which has shown considerable efficacy in mitigating label noise, especially in graphs characterized by heterophily and elevated label noise (Cheng *et al.*, 2024a). Recent research has reinvigorated label propagation approaches to address these specific difficulties, using their ability to propagate labels throughout the graph structure to reduce noise and enhance accuracy (Cheng *et al.*, 2024b).

Robust heterophilic graph learning approaches have been devised to combat label noise, especially in anomaly detection applications. These techniques are centered on producing consistent representations even in noisy environments, which ultimately increases the durability of GNNs in noisy environments (Wu *et al.*, 2024). UnionNET (Li *et al.*, 2021c) attempts to prevent gradient passage of mislabeled samples using neighborhood labeling, such as node representation similarity weighted neighborhood voting. RNCGLN (Zhu *et al.*, 2024b) use pseudo graphs and pseudo labels to handle graph and label noise. CLNode (Wei *et al.*, 2023b) implements a curricular learning technique to alleviate the effects of label noise. CLNode first uses a multi-perspective difficulty assessor to evaluate the quality of training nodes accurately. Subsequently, a training scheduler will be employed to determine appropriate

training nodes for GNN training in each epoch based on the evaluated qualities.

Node-contrasted Data Reprocessing is crucial for minimizing the effects of label noise in graph datasets. Maintaining the accuracy and robustness of GNNs under noisy labeling depends on using efficient preprocessing techniques. The robustness of the model is enhanced by contrasting learning methods, which concentrate on data point similarities and contrasts. Under label noise, this concentration helps to extract important representations. Li *et al.* (2024d) learns under the expected label distribution's supervision, boosting the generalizing capacity of text classification models. CGNN (Yuan *et al.*, 2023a) tackles label noise in GNNs by integrating neighborhood-based label rectification with contrastive learning. PI-GNN (Du *et al.*, 2021) increases GNN label noise resilience by adding pair-wise labels since pair-wise labels are more robust than node-wise labels. Complete benchmarks such as NoisyGL have been developed to assess GNN performance under several degrees and kinds of label noise, supporting the building of stronger GNNs (Wang *et al.*, 2024e).

Noise-resistant Structural Analysis for graph noise learning have been developed to address the unique problems of noisy data within graph topologies. Unlike more traditional noise-reduction methods used in other machine learning fields, these systems sometimes integrate specific algorithms addressing the complexity of graph-based data. One such strategy focuses on developing GNNs that are free of label noise. The NR-GNN (Dai *et al.*, 2021) can tolerate label noise in sparsely and loudly labeled graphs to deal with graph-specific noise issues. Another work investigates the particular methods applied in graph noise learning to grasp and reduce graph structural noise (Dong and Kluger, 2023).

Further Discussion. GNNs encounter substantial difficulties stemming from label noise, which directly affects their advancement and efficacy. GNNs are struggling to generalize suitably from noisy data, compromising both predictive powers and general robustness (Li *et al.*, 2024a; Wang *et al.*, 2024e; Dai *et al.*, 2021), which can significantly reduce performance. Graph-noise learning research has mostly focused on node categorization. Other significant objectives in graph learning include link prediction, edge property prediction, and graph categorization. Graph classification and graph transfer learning with label

noise are understudied. Other graph learning research outside node classification is still in its infancy and needs more attention.

2.1.3 Group-structured Noise Learning

In many real-world data acquisition processes, data are organized into groups where each collection of instances is collectively associated with a single class label. Alternatively, individual instances may be assigned multiple labels, though only one of these labels is correct. A typical example of the multi-instance-one-label scenario involves crawling images from search engines using keywords. The retrieved images form a group, linked by those keywords as labels. Moreover, in the one-instance-multi-label case, a representative scenario is to use crowdsourcing to gather multiple labels for each data point, provided by annotators. In general, due to potential failures of search engines or human annotators, there may be cases where there are wrong correlations between individual data and labels within groups, i.e., the group noise exists. The related research problems are multiple-instance learning, partial label learning, and group noise learning. In Figure 2.4, we illustrate the comparison of group structured noise with pair-wise noise.

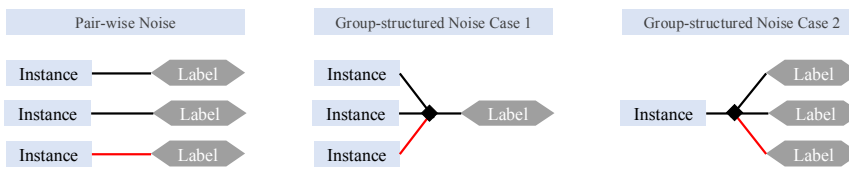

Figure 2.4: Illustration of the supervised learning with pair-wise noise (a) and two settings with group-structured noise (b)-(c), in which the objects are realized by instances and labels, respectively. In the figure, black lines represent the correct relations while red lines mean the incorrect relations.

Problem Setup. Formally, in conventional supervised learning, we have a dataset of instances $\{x_i, y_i\}$, where each x_i should be assigned by $\{x_i, y_i\}$. However, in group-structured noise learning, the dataset consists of a set of data $\{X_i, Y_i\}$, where either X_i or Y_i denote a set of instances or labels. Due to the noisy nature, there are some individuals inside X_i or Y_i that do not have a correct relationship with each other. Then, the general goal of group-structed noise learning is to find a

proper model f such that it can use individual instances as inputs and return its correct instance-level labels.

Multiple-instance Learning considers the multi-label-one-instance setup, where a collection of instances (termed bags) is assigned with a binary label. The basic assumption in multiple-instance learning is that there is at least one positive instance inside a positively labeled group, whereas all individual instances inside a negative group are negative. Broadly speaking, the primary objective of multiple-instance learning can vary between bag-level and instance-level predictions. Our focus here in mainly on the instance-level prediction, which is intricately linked to learning in the presence of noise. Multiple-instance learning has applications in various fields such as bio-informatics and drug activity predictions.

Formally, in conventional supervised learning, we have a dataset of instances $\{x_i, y_i\}$, where each x_i should be assigned by $\{x_i, y_i\}$. However, in multiple-instance learning, the dataset consists a set of bags $\{X_i, Y_i\}$, where $X_i = \{x_{i1}, \ldots, x_{in_i}\}$ is a set of instances and Y_i is the associated label of X_i. In multiple-instance learning, we typically consider the existential assumption: if $Y_i = 1$, at least one instance inside X_i should be positive; if $Y_i = 0$, all instances inside X_i are negative. Then, our goal is to find a proper model f such that it can it individual instances as inputs and return its correct instance-level labels.

Here we introduce two representative works for the task of multi-instance learning. Key Instance Detection (KID) (Liu *et al.*, 2012) aims to figure out those key instances that are responsible for the bag label. To find the key instance, KID leverages the relationships among instance represented by a k-nearest neighbor graph, and suggests an iterative voting-based method that can continuously refine the estimation of instance labels. Peng and Zhang (2019) devise a loss function tailored for instance-level label prediction, despite the absence of instance labels. The main idea is based on unbiased estimation: The authors derive an unbiased estimator for the instance-level label prediction without direct label information, assuming that the proportion of negative instances is known as a priori and that instances are independently and identically distributed.

Group Noise Learning expands upon the limited setup of multiple-instance learning. It accommodates multi-class learning scenarios and assumes that each group, irrespective of the assigned labels, may consist of instances that are wrongly labeled. The framework of group noise learning is more broadly applicable to a variety of practical applications than multiple-instance learning, including search engine-based data crawling and recommender systems.

Similar to multiple-instance learning, the dataset in group noise learning consists a set of bags $\{X_i, Y_i\}$, where $X_i = \{x_{i1}, \ldots, x_{in_i}\}$ is a set of instances and Y_i is the associated label for X_i. However, in group noise learning, Y_i is not restricted to binary value. Moreover, within each bag X_i, there may be some instances x_{ij} that should not be labeled by Y_i, regardless the value of Y_i.

To tackle this challenging issue, Kotzias *et al.* (2015) especially explore the inherent capacity of deep learning with a new learning objective tailored for instance-level prediction. The key assumption for the success of this work is that similar instances in the manifold should have similar labels. Thereby, the authors suggest a manifold regularization term to penalize differences in predicted labels between similar instances. Also, to explore the knowledge behind group-level labels, the authors further include a term to constrain the relationship between instance and group labels. The balance between instance similarity and group-level constraints lead to the overall objective function, allowing the model to effectively infer instance-level labels while adhering to group-level information. Max-Matching (Wang *et al.*, 2021c) further consider the non-independent and identically distributed assumption, exploring a novel learning objective that can discover correct pairwise connections between instances and labels and thereby leveraging the relationships among instances in each group to pursue group noise robust learning.

Partial Label Learning is very different from multiple-instance learning and group noise learning, which considers the situation where each training instance is assigned with a set of candidate labels, among which only one of them is the true label. The objective of partial label learning is to leverage the candidate labeling to train a proper classification model that can make the correct prediction for test instances.

Partial label learning considers a practical setup that can be common in domains like image and video recognition, where annotations often come from different sources or annotators, but it is typically unclear which label is correct. Formally, we consider a set of data $\{x_i, Y_i\}$, where $Y_i = \{y_{i1}, \ldots, y_{in_i}\}$ is a set of candidate labels for x_i. The basic assumption for partial label learning is that the true label y_i for x_i is contained within Y_i. Then, the goal of partial label learning is to learn a classifier f that can predicts the true label y for any new instance x.

There are in general two lines of methodologies considered for partial label learning, namely, the average-based and the detection-based methods. For average-based methods, they would like to take all candidate labels with equal contributions. For example, Cour *et al.* (2011) transform the multi-class partial label problem into a set of binary classification problems and tackle the ambiguity within Y_i by measuring the maximum probability that an incorrect label is included in the candidate set alongside the true labels. On the other hand, detection-based methods aim at revealing the true labels among the candidates. For example, Zhang *et al.* (2016) suggests a two-stage method named partial label learning via feature-aware disambiguation (PL-LEAF), which can effectively explore information from the feature space to assist in label disambiguation. Yu and Zhang (2016) directly maximize the margin between the ground-truth label and all other labels, including those within the candidate label space. Yao *et al.* (2020) introduce an entropy minimization regularization term to sharpen the confidence of model predictions and then leverage the average model predictions across different training epochs to form an ensemble label as a proper estimation of the true labels.

Further Discussion. Although group-structured data are prevalent in various real-world scenarios, there are still numerous open questions required to be addressed. In particular, there is still a lack of comprehensive frameworks capable of spanning a wide range of specific problem settings meanwhile also facilitating the effective formalization of these problems. Additionally, there is no unified framework currently available that faciltates an understanding of how to address the noise within group structured data.

In the future, research directions could include considerations of instance-level noise, where some data within groups may not more prone to incorrect assignments than other data. Exploring the connections within and across groups could also be beneficial in enhancing the learning processes for group-structured noise. Furthermore, researchers have the opportunity to broaden the applications of group-structured noise learning and improve the generality of models for the considered research problems.

2.2 Long-tailed Learning

Long-tailed learning tackles the challenge of skewed data distributions, which hinders model performance on underrepresented classes. This issue, common in real-world datasets across domains like visual recognition and healthcare, demands tailored solutions to balance representation quality and enhance generalization. Without loss of generality, in this section, we examine the long-tailed learning in the context of supervised, weakly-supervised, and self-supervised settings, which retrospects several typical paradigms for long-tailed learning in Figure 2.5.

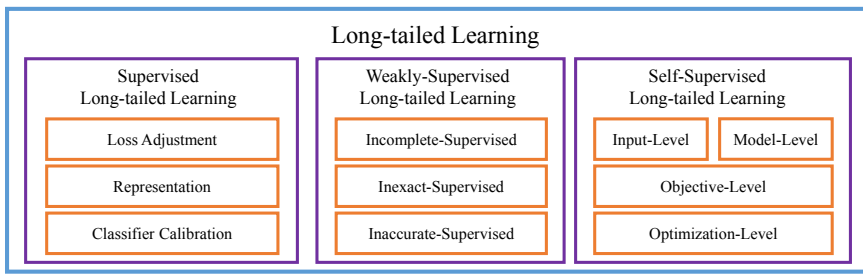

Figure 2.5: The overall framework of long-tailed learning methods.

Generally, supervised long-tailed learning focuses on class rebalancing techniques and classifier adjustments to directly address skewed class distributions during training. Weakly-supervised methods leverage incomplete, inexact, or noisy labels to reduce annotation costs while managing data imbalance. Self-supervised learning emphasizes unsupervised feature extraction, offering greater resilience to label scarcity

and bias. Together, these paradigms demonstrate the evolution of long-tailed learning from explicit supervision to wild relaxation, offering a comprehensive framework for tackling imbalanced datasets.

2.2.1 Supervised Long-tailed Learning

Supervised long-tailed learning deals with the problem of class imbalance by improving loss functions, representation learning, and classifier geometry. This section looks at important studies that try to improve model performance in this area. The main methods in this field are shown in Figure 2.6.

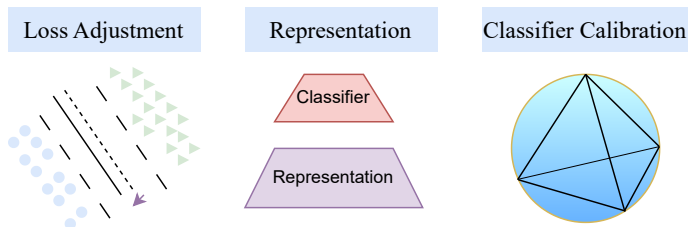

Figure 2.6: The framework of supervised long-tailed learning methods. Methods of loss adjustment adjust the decision boundary, methods of representation decouple the representation structure and classifier, while methods of classifier calibration analyze geometric structure like ETF (Strohmer and Heath Jr, 2003).

Problem Setup. Let x represent the input and y represent the corresponding label. We assume the inputs follow $x \in \mathbb{R}^d$ and come from C distinct classes, i.e., $y \in \{1, \ldots, C\}$. In supervised long-tailed learning, the dataset exhibits significant class imbalance, which is quantified by the imbalance ratio $\text{IR} = \frac{\min_{j \in \mathcal{Y}} |y=j|}{\max_{j \in \mathcal{Y}} |y=j|} \leq 1$, where $|y = j|$ denotes the number of samples in class j. The goal is to train a classifier $f_\theta : \mathbb{R}^d \to \{1, \ldots, C\}$, parameterized by θ, which minimizes the disparity in performance across majority and minority classes. This requires solving the problem of imbalanced data while keeping the overall accuracy. Benchmark datasets like ImageNet-LT, Places-LT, and iNaturalist are often used to test performance.

Loss Function Design. In supervised long-tailed learning, designing loss functions is a key strategy for handling class imbalance. This section looks at three main approaches: analyzing the effective sample number, adjusting decision boundaries, and balancing probabilities.

Methods based on the effective number have become important for handling class imbalance by giving different weights to each class. The Class-Balanced (CB) Loss, introduced by Cui *et al.* (2019), focuses on the decreasing value of extra data in high-density classes. It introduced the idea of the effective number of samples, which is based on the number of samples in each class. This method shows that as the class size increases, extra samples give less new information. Following this idea, the AREA method (Chen *et al.*, 2023d) redefines the effective size from a geometric point of view. It looks at not only the number of samples but also the distribution and relationships among samples in the feature space. It quantifies the effective area spanned by the samples of each class, which may exceed, match, or fall below the actual sample count depending on the sample diversity and overlap. Luo *et al.* (2024) builds on the top of an estimated density ratio for dynamic re-weighting, which generalizes previous constant re-weighting by sample number. The success of these methods shows the importance of considering data overlap and feature diversity when designing re-weighting schemes. It also points to a shift from using only statistical measures to more detailed geometric and probabilistic approaches.

Adjusting decision boundaries is an important strategy to reduce the performance gap between majority and minority classes in long-tailed learning. LDAM, introduced by Cao *et al.* (2019b), improves class-specific margins by giving larger margins to minority classes. These margins are inversely related to the fourth root of the class sample number. This strategy reduces overfitting in tail classes and keeps the performance of head classes. When combined with deferred re-weighting, LDAM balances generalization and accuracy across class distributions. UML (Khan *et al.*, 2019) uses Bayesian uncertainty to change decision boundaries. It increases margins for underrepresented classes with higher uncertainty, which helps improve generalization for rare classes. Additionally, it models sample-level uncertainty using Gaussian distributions, enabling more flexible and context-aware boundary adjustments. LOCE

(Feng *et al.*, 2021) takes a multi-faceted approach by introducing Equilibrium Loss and Memory-augmented Feature Sampling. It increases margin for tail classes based on classification scores, while MFS oversamples minority class features to ensure better representation. RoBal (Wu *et al.*, 2021) integrates margin engineering with a scale-invariant classifier to tackle adversarial robustness and long-tailed distributions. By rebalancing margins during training and adjusting boundaries at inference, RoBal enhances both natural and robust accuracies, challenging in adversarial scenarios.

Many methods address long-tailed recognition from a probabilistic view. They focus on adjusting class probabilities to improve the balance between head and tail classes. These methods share a common insight that the model's prediction tends to be biased by the training data distribution, and thus require mechanisms to calibrate such bias. Logit Adjustment (LA) (Menon *et al.*, 2020) started this approach by adding prior knowledge directly into the prediction process. It adds a correction term based on class frequencies, helping the model adjust for bias in the training data. Distribution Alignment (DisAlign) (Zhang *et al.*, 2021e) looks at how to separate the source label distribution from model predictions. It introduced a framework that can adapt to different target distributions using both post-processing and training-time adjustments. LADE (Hong *et al.*, 2021) builds on these ideas by using bounds from information theory. It offers a method to regularize model predictions during training, making them independent of source label distributions. This helps the model generalize better to different target distributions without retraining. These methods for balancing probabilities have shown strong results in long-tailed recognition tasks and come with solid theoretical support.

Representation and Classifier Geometry. Representation quality has become an important factor in solving the challenges of supervised long-tailed learning. High-quality representations form the foundation for robust and generalizable classifiers, especially under the imbalanced conditions inherent to long-tailed datasets. Conjugated with the representations, the classifier geometry similarly matters along with the representation distribution for supervised long-tailed learning. This section discusses the representation and classifier calibration methods.

Decoupled approaches (Kang *et al.*, 2020; Chu *et al.*, 2020; Desai *et al.*, 2021) have emerged as an effective paradigm for addressing long-tailed recognition challenges. Rather than jointly optimizing feature extractors and classifiers, these methods separate the learning process into two distinct stages. The key insight is that representation learning and classification have different optimal strategies when dealing with imbalanced data distributions. Kang *et al.* (2020) demonstrated that high-quality representations could be learned effectively using instance-balanced sampling, without requiring complex re-sampling strategies. This suggests that the feature extractor can capture general visual characteristics even from imbalanced data. The classification stage can then be optimized separately. Chu *et al.* (2020) introduces a feature-space augmentation strategy that decouples representation learning and classification. By decomposing features into class-specific and class-generic components, this method leverages the transferable knowledge from head classes to generate augmented features for tail classes during training, thus addressing data imbalance at a conceptual level. Desai *et al.* (2021) showed that the decoupling principle was particularly powerful when both entity and predicate distributions were heavily skewed. Their work revealed that maintaining the simple architectures while carefully designing the decoupled training strategy can outperform much more complex models.

Neural collapse, a recently observed phenomenon, describes the convergence of within-class feature means and classifier weights to the vertices of a simplex equiangular tight frame (ETF) during the terminal phase of training. This geometric structure minimizes within-class variability, maximizes between-class separation, and aligns classifier weights with feature means, resulting in a highly symmetric configuration that simplifies classification (Papyan *et al.*, 2020). The ETF (Strohmer and Heath Jr, 2003), as a mathematical construct, ensures optimal equiangular separation among class vectors and has been rigorously defined within the framework of tight frames. Building on this idea, Yang *et al.* (2022b) showed that initializing the classifier as a fixed simplex ETF and training only the backbone can cause neural collapse, even with strong class imbalance. This method helps stabilize feature learning, fix

gradient imbalances, and prevent problems like the merging of minority class weights.

Further Discussion. Supervised long-tailed learning has made great progress in recent years. Many methods address class imbalance by improving loss functions, representation learning, and classifier geometry. However, there are still challenges and open questions that need more study. The methods mentioned above have shown promise, but many are designed for specific problems. Adding techniques from related fields, like meta-learning (Hospedales *et al.*, 2021), transfer learning (Yin *et al.*, 2019), and more statistical methods (Luo *et al.*, 2024), could offer new solutions for long-tailed learning. Also, supervised long-tailed learning benefits from strong theories like Fisher consistency, which connects loss minimization with balanced error reduction (Menon *et al.*, 2020), and fine-grained generalization bounds, which explain class-specific behaviors through data-dependent contraction (Wang *et al.*, 2024f). These tools improve reliability and flexibility, helping guide future research toward clearer and more effective models for complex situations.

2.2.2 Wealy-supervised Long-tailed Learning

Traditionally, long-tailed learning assumes a fully and accurately annotated training dataset. However, in practice, obtaining such complete and high-quality annotations is generally challenging, expensive, and time-consuming, especially for minority classes or groups. Such a constraint is even more prominent in applications where data annotation is highly specialized or involves privacy concerns. Thus, in many applications, we can only obtain imperfect annotations, leading to weakly supervised scenarios. Weak supervision introduces distinct challenges in the context of long-tailed learning because it either masks the imbalance patterns in the data or does not satisfy the assumptions and principles upon which typical supervised long-tailed learning methods require.

Problem Setup. Let $\mathcal{X}$ be the input space and $\mathcal{Y} = \{1, 2, \ldots, C\}$ be the label space, where C is the number of classes. A long-tailed training dataset with weak supervision can be denoted as $\mathcal{D} = \{\mathbf{x}_i, \hat{y}_i, y_i\}_{i=1}^{N}$, where $\hat{y}_i$ is the observed annotation of the sample $\mathbf{x}_i \in \mathcal{X}$, and its

ground truth $y_i \in \mathcal{Y}$ is invisible. Weak supervision means that $\hat{y}_i$ may not match the ground truth y_i for each sample $\mathbf{x}_i$, but it can still provide some useful information compared to having no annotations at all. The sample number N_c of each class $c \in \mathcal{Y}$ in the descending order exhibits a long-tailed distribution. The imbalance ratio is defined as $\text{IR} = \frac{\max_{c\in\mathcal{Y}} N_c}{\min_{c\in\mathcal{Y}} N_c} \gg 1$. For evaluation, a class-balanced test set $\mathcal{D}_{test}$ with clean labels is used. The goal of weakly-supervised long-tailed learning under label noise is to learn a deep model $f : \mathcal{X} \to p(\mathcal{Y})$ on $\mathcal{D}$ that minimizes the error rate on $\mathcal{D}_{test}$. Different forms of weak supervision correspond to different specific problem settings. Generally, the types of weak supervision can be categorized as follows (Zhou, 2017):

- Incomplete supervision: only a subset of the training data is provided with labels, while the remaining data remains unlabeled. That is, the training set $\mathcal{D}$ consists of a labeled subset $\mathcal{D}_l$ and an unlabeled subset $\mathcal{D}_u$, such that $\mathcal{D} = \mathcal{D}_l \cup \mathcal{D}_u$. $\forall(\mathbf{x}, \hat{y}, y) \in \mathcal{D}_l$, $\hat{y} = y$; $\forall(\mathbf{x}, \hat{y}, y) \in \mathcal{D}_u$, $\hat{y} = \emptyset$. This form of supervision is often referred to as semi-supervision in many works (Zhu, 2005; Zhu and Goldberg, 2022).

- Inexact supervision: only coarse-grained labels are provided, which are less precise than desired. A typical scenario is partial label learning, where each sample $\mathbf{x}_i$ is labeled with a coarse set that includes the ground truth rather than the exact ground truth itself, *i.e.*, $\hat{y}_i \in 2^{\mathcal{Y}} \setminus \emptyset$ and $y_i \in \hat{y}_i$ (Hüllermeier and Beringer, 2006; Feng *et al.*, 2020).

- Inaccurate supervision: the supervision is not always the ground truth. A typical scenario is learning with label noise, where certain samples contain incorrect labels. The noise ratio is defined as the mislabeled ratio of training set $\text{NR} = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i \neq y_i)$ (Natarajan *et al.*, 2013; Li *et al.*, 2020b).

As illustrated in Figure 2.7, weakly-supervised learning typically focuses on refining supervision during the training process to make it more comprehensive, precise, and accurate, thereby guiding the model to achieve better performance. However, the imbalanced distribution

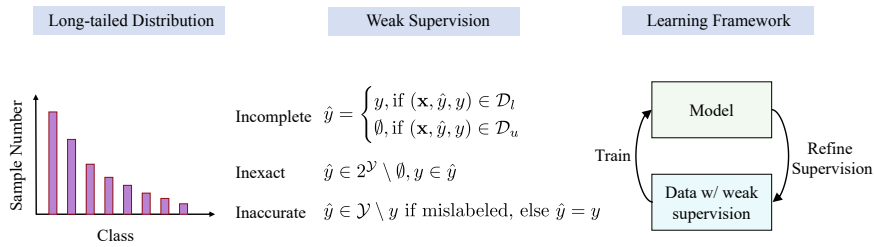

Figure 2.7: The overall framework of weakly-supervised long-tailed learning.

of training data often renders typical supervision refinement methods ineffective. It can also lead to the accumulation of errors and biases during iterative training, significantly harming the model performance. Research on weakly-supervised long-tailed learning primarily focuses on maintaining effective supervision refinement under imbalanced data.

Incomplete-supervised Long-tailed Learning. A common approach for advanced algorithms handling incomplete supervision is to generate pseudo-labels for unlabeled data based on model predictions and iteratively use the pseudo-labeled data for training (Miyato *et al.*, 2019; Gan and Wei, 2024). However, when the training data is imbalanced, the pseudo-labels generated by the model for unlabeled data tend to exhibit even more extreme imbalances, further impairing the model's performance and generalization ability (Kim *et al.*, 2020). DARP (Kim *et al.*, 2020) refines the original, biased pseudo-labels so that their distribution can match the true class distribution of unlabeled data while constraining the refined pseudo-labels to be close to the original ones. Auxiliary Balanced Classifier (ABC) (Lee *et al.*, 2021) is inspired by the observation in supervised long-tailed learning (Kang *et al.*, 2020) that high-quality representations can be learned even when the classifier is biased. ABC trains an auxiliary balanced classifier by resampling a balanced subset while leveraging the high-quality representations learned from the entire dataset. BaCon (Feng *et al.*, 2024) introduced the contrastive learning paradigm in semi-supervised long-tailed learning, while also constraining the feature-label balance to improve both model performance and fairness. BEM (Zheng *et al.*, 2024) firstly explored a balanced data mixing method for semi-supervised

long-tailed learning, considering rebalancing both data quantity and uncertainty.

Inexact-supervised Long-tailed Learning. The state-of-the-art paradigm for handling inexact supervision gradually eliminates ambiguities in coarse annotations based on the model predictions during training, ultimately obtaining fine-grained labels to aid model training. When faced with long-tailed training data, the frequency bias makes the label disambiguation process more challenging, further degrading the model performance (Wang *et al.*, 2022a; Hong *et al.*, 2023). Solar (Wang *et al.*, 2022a) proposes refining the disambiguated labels to match the marginal class prior distribution in imbalanced partial label learning, modeling it as an Optimal Transport problem, and solving it using the Sinkhorn-Knopp algorithm (Cuturi, 2013). Hong *et al.* (2023) suggest that the key challenge in handling imbalanced data under inexact supervision lies in the drastic dynamic changes in model bias caused by the label disambiguation process, which traditional long-tailed learning techniques cannot address. They proposed RECORDS, which dynamically estimates model bias and performs rebalancing through a momentum-updated prototype during the training process. Based on it, Jia *et al.* (2024) proposed a collaborative strategy with a head-tail dual classifier for long-tailed partial label learning, which alleviates the trade-off between head and tail performance through adaptive assignment.

Inaccurate-supervised Long-tailed Learning. The key to handling inaccurate supervision lies in identifying data that is mislabeled or poorly annotated, and then training the model either using a semi-supervised paradigm or by correcting these erroneous labels. Mainstream noise-label learning methods tend to assume that difficult samples are more likely to be mislabeled. However, when the training data is imbalanced, such difficult samples could also be accurately labeled as tail samples. Differentiating between tail samples and mislabeled samples is a key challenge in inaccurate-supervised long-tailed learning. RoLT (Wei *et al.*, 2021) introduced a prototypical noise detection method for long-tailed data that employs a distance-based metric, making it robust to label noise. UCL (Huang *et al.*, 2022) proposed modeling class-specific noise using epistemic uncertainty to identify trustworthy clean samples and refine or discard highly confident true or corrupted labels, while

also incorporating aleatoric uncertainty to prevent noise accumulation. Wei *et al.* (2023a) proposed a Fairness Regularizer (FR), which encourages the model during training to reduce the performance gap between the head and tail groups. TABASCO (Lu *et al.*, 2023) introduced a two-stage, bi-dimensional sample selection method to more effectively distinguish clean samples from noisy ones, especially for tail classes.

Futher Discussion. Existing research primarily focuses on the effects of imbalances in prediction targets. However, as fairness in machine learning becomes increasingly important (Mehrabi *et al.*, 2021; Caton and Haas, 2024), addressing non-prediction-target imbalances, such as attribute or subgroup imbalances, is just as critical. For instance, imbalances in gender, age, or ethnicity in medical diagnosis scenarios highlight this challenge (Chen *et al.*, 2023c). When labels for prediction targets are available but subgroup or attribute annotations are unknown, such settings can be considered a form of weakly-supervised long-tail learning. Addressing potential unknown subgroup or attribute imbalances while training a model for prediction tasks, with the goal of ensuring fairness, is both a challenging and worthwhile problem to explore. Recent studies have begun exploring these contexts. Yang *et al.* (2023b) established a comprehensive benchmark for subgroup imbalance and domain shift, covering fields such as vision, language, and healthcare. SHE (Hong *et al.*, 2024a) proposed using optimal data partition to effectively uncover potential imbalanced subgroups during training and balance predictions across different subgroups.

2.2.3 Self-supervised Long-tailed Learning

Self-supervised learning (SSL) has achieved significant strides in extracting robust and transferable representations from large-scale unannotated datasets in the context of computer vision (He *et al.*, 2020; Henaff, 2020; Chen *et al.*, 2020c), speech understanding (Oord *et al.*, 2018; Schneider *et al.*, 2019) and natural language processing (Devlin, 2018; Brown *et al.*, 2020). Specially, SSL-pretrained features often surpass supervised counterparts, as they generalize well to a variety of downstream tasks and datasets (Chen *et al.*, 2020d; Grill *et al.*, 2020; Zbontar *et al.*, 2021). However, most prominent SSL methods for images are conducted on

well-structured and curated datasets such as ImageNet (Deng *et al.*, 2009). The inherent uniform structure on these datasets is different from real-world long-tailed distributions (Reed, 2001), potentially overlooking the performance disparities of SSL. To address this challenge, a research direction focusing on self-supervised long-tailed learning (Liu *et al.*, 2021c; Jiang *et al.*, 2021; Zhou *et al.*, 2022; Zhou *et al.*, 2023b) has emerged, aiming to mitigate the negative effect of long-tailed data distribution without the guidance of explicit label annotations. In Figure 2.8, we present the core factors in self-supervised long-tailed learning.

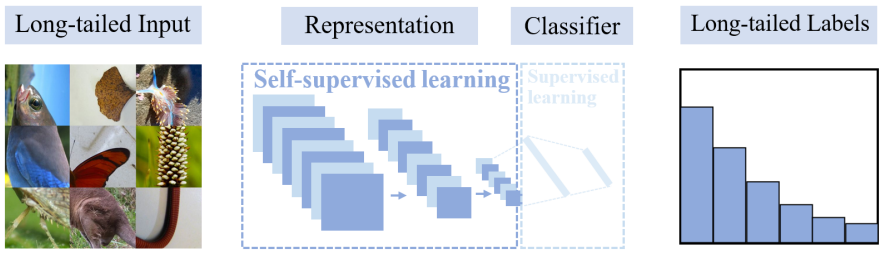

Figure 2.8: Different from supervised and weakly-supervised long-taield learning, which focus on learning a robust classifier, the key goal of self-supervised learning is to enhance long-tailed learning at the representation level.

Problem Setup. Let x represent the input and y represent the corresponding label. We assume the inputs $x \in \mathbb{R}^d$ coming from C distinct classes, *i.e.*, $y \in \{1, \ldots, C\}$. In contrast to supervised learning or weakly-supervised learning settings, where labels y are (partially) available, self-supervised learning observes only the inputs x. The imbalance ratio IR is defined based on the pre-training distribution $\mathcal{P}$ over $\mathbb{R}^d \times [C]$ as $\text{IR} = \frac{\min_{j \in [C]} \mathcal{P}(y=j)}{\max_{j \in [C]} \mathcal{P}(y=j)} \leq 1$. The imbalanced ratio captures the disparity between the minority and majority class probabilities. Self-supervised long-tailed benchmark datasets either use subsampled subsets with varying imbalance ratios such as ImageNet-LT (Liu *et al.*, 2019) and Places-LT (Liu *et al.*, 2019) or large-scale real-world dataset such as iNaturalist (Van Horn *et al.*, 2018). The learning objective of the SSL paradigm is to obtain a feature extractor $f_\phi : \mathbb{R}^d \to \mathbb{R}^m$, parameterized by neural network parameters ϕ, which maps inputs to latent embeddings.

A linear head $g_\theta : \mathbb{R}^m \to \mathbb{R}^C$, is added on top of f_ϕ during pretraining to produce logits, which is often discarded during downstream evaluation. The evaluation phase is commonly be categorized to in-distribution (ID) evaluation and out-of-distribution (OOD) evaluations. ID evaluation measures the performance of feature extractor f_ϕ on a balanced ID test set using linear probing, while OOD evaluation assesses performance on one or multiple downstream target distributions. Here, linear probing involves additionally training a C-way linear classifier on top of f_ϕ using a balanced, i.i.d. sampled dataset from the test distribution, and such evaluation can also be extended to broader tasks, such as detection and segmentation (Zhou *et al.*, 2023b). Existing explorations on self-supervised long-tailed learning can be broadly categorized into four key aspects: input-level, model-level, optimization-level, and objective-level methods, as shown in Table 2.1.

Table 2.1: An overview self-supervised long-tailed learning methods. The *Input*, *Model*, *Optimization*, and *Objective* columns reflect modifications considered at the respective levels. The *description* column provides descriptions of each method.

Method	Input	Model	Optimization	Objective	Description
BCL (Zhou *et al.*, 2022)	✓				Memorization-guided augmentation
COLT (Bai *et al.*, 2023)	✓				Additional OOD data
SDCLR (Jiang *et al.*, 2021)		✓			Model pruning and self-contrast
MoCLR (Tian *et al.*, 2021)		✓			Multi-expert ensemble
RwSAM (Liu *et al.*, 2021c)			✓		Reweighted sharpness regularization
TS (Kukleva *et al.*, 2023)			✓		Temperature schedules
Focal (Lin *et al.*, 2017)				✓	Hard example mining
GH (Zhou *et al.*, 2023b)				✓	Geometric uniform clustering
PMSN (Assran *et al.*, 2023)				✓	Clustering with long-tailed prior

Input-level Methods. These approaches generally seek to introduce additional information to the input data, so that the learned representation can be improved for long-tailed data distribution. From the data augmentation perspective, BCL (Zhou *et al.*, 2022) introduced a memorization-guided data augmentation technique to enhance the learning of tail classes. Specifically, BCL explores the memorization effect of deep neural network on individual sample, using a learning-speed based-proxy in the SSL context to distinguish between head and tail samples. Subsequently, the proxy drives an instance-wise augmentation with distinct information discrepancies for head and tail samples. This

allows enhancement of tail performance while maintaining the head performance. From the perspective of additional training data, COLT (Bai *et al.*, 2023) proposed to leverage out-of-distribution data for tail class rebalancing. Specifically, COLT utilizes a logit-based tailness score to perform head-tail detection and then employs an online sampling strategy to dynamically selects OOD samples from a large external data pool that are close to the tail classes. The model trained with OOD-enriched dataset, applies a distribution-level supervised contrastive loss to improve long-tailed performance without inducing distribution shift.

Model-level Methods. These approaches focus on improving structural design to address long-tailed challenges in SSL paradigm. SDCLR (Jiang *et al.*, 2021) draws inspiration from network pruning to identify samples that are under-represented by the model. Specifically, SDCLR leverages an model augmentation technique by pruning the target model parameters for contrastive learning. The rationale is that tail classes tend to be more sensitive to model pruning, which will exhibit greater prediction differences between pruned and non-pruned models. This self-contrastive pruning thus implicitly rebalances the contrastive loss, placing more emphasis on tail classes. From a model ensemble perspective, MoCLR (Tian *et al.*, 2021) proposed to divide the dataset into fine-grained subgroups, with expert models dedicated to learning each subgroup. Specifically, MoCLR first applies unsupervised clustering using a pretrained base model to split the dataset into smaller subsets. Each expert is then trained from scratch on a specific subset, focusing on semantic-related classes, and the expert knowledge is distilled into a final base model. This approach helps recover local consistency within smaller subsets of large-scale uncurated datasets, improving the model ability to preserve long-tailed information without being dominated by head classes in standard training on the entire dataset.

Optimization-level Methods. This line explores the training schedules and optimization for long-tailed learning in SSL. Motivated by the observation that tail classes are more prone to overfitting, RwSAM (Liu *et al.*, 2021c) introduced a data-dependent regularizer that applies varying penalization to head and tail classes. RwSAM utilizes sharpness-aware minimization (SAM) to penalize the sharpness of loss surface during training, encouraging the model to converge to flatter minima

for better generalization. To further emphasize tail classes, RwSAM employs a reweighting strategy based on kernel density estimation, providing stronger regularization for tail classes to rebalance the model. In contrast, TS (Kukleva *et al.*, 2023) explores the impact of temperature factor in long-tailed contrastive learning. Specifically, TS proposed a dynamic cyclical cosine temperature schedule that facilitates a 'task switching' effecy between instance-wise discrimination and group-wise discrimination, with the former benefiting tail classes and potentially enhancing long-tailed representation learning.

Objective-level Methods. Generally, conventional SSL training typically uses InfoNCE-based loss (Oord *et al.*, 2018), which does not account for class imbalance. The classical hard-example-mining method, Focal loss (Lin *et al.*, 2017) can be adapted to self-supervised long-tailed context with minimal modification, which helps improve representation learning for tail classes as hard examples. In contrast, GH (Zhou *et al.*, 2023b) identifies the limitation of contrastive learning loss from a geometric perspective, and points out that the hidden sample-level uniformity distorts the embedding space. This refers to the excessive expansion of head classes and collapse of tail classes. To mitigate this, GH introduces an optimal geometric structure as a uniformity prior to preserve the global embedding space structure. Then, GH proposes a optimal tranport-based clustering to generate surrogate labels for regularizing samples toward the optimal structure. From a similar perspective, PMSN (Assran *et al.*, 2023) points out that contrastive learning implicitly performs clustering with a uniform prior, which exacerbates long-tailed distribution issues. To address this, PMSN introduces a power-law distribution prior and proposes KL-divergence regularization to align the learned feature clusters with the predefined distribution.

Further Discussion. Long-tailed learning has long been a critical and challenging research problem (Zhang *et al.*, 2023f). However, its application during the pretraining phase remains underexplored. In the era of large foundation models (Dubey *et al.*, 2024) pretrained on massive, uncurated datasets, it is crucial to investigate how long-tailed learning influences large-scale pretraining and how to mitigate head-tail disparity to enhance model performance. Self-supervised long-tailed

learning for images is a starting point, but other areas, such as multimodal (vision-language) long-tailed learning (Parashar *et al.*, 2024) and long-tailed generative learning (Zhang *et al.*, 2024b), need further exploration. In vision-language learning, the differences and interactions between modalities provide new opportunities to better address underrepresented groups. Long-tailed generative learning, on the other hand, poses challenges in understanding how long-tailed distributions behave in diffusion-based or autoregressive models. Additionally, with the increasing size and inaccessibility of pretraining data, detecting underrepresented groups remains an open question.

2.3 Out-of-distribution Learning

Out-of-distribution (OOD) Learning aims to enable machine learning models to tackle samples out of the training distribution. Due to the inherited i.i.d. assumption generically adopted that assumes the testing samples are drawn independently from a distribution identical to the training one, machine learning models are sensitive and easily suffer severe performance degeneration when encountering OOD data.

To mitigate the issue, as shown in Figure 2.9, it has been developed two research approaches by categorizing the OOD data into ungeneralizable and generalizable ones. For the ungeneralizable OOD data, OOD detection has been proposed to detect and separately handle it. As ungeneralizable OOD samples can affect both the testing and training of the model, post hoc and training-time OOD detection are developed to manage OOD data during the respective stages. For the generalizable one, OOD generalization has been proposed. OOD generalization establishes a series of definitions of invariance that enable models to generalize beyond training distribution. Furthermore, the optimization for capturing the invariance raises unique challenges to model training and selection, which requires careful consideration.

2.3.1 Out-of-distribution Detection

Out-of-distribution (OOD) detection refers to the problem of identifying inputs of which the classification models do not have the capacity to

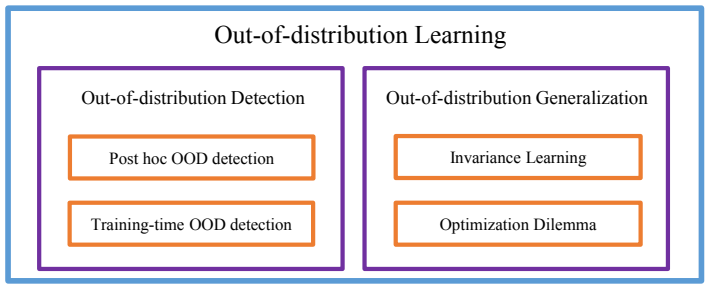

Figure 2.9: The overall framework of out-of-distribution learning methods.

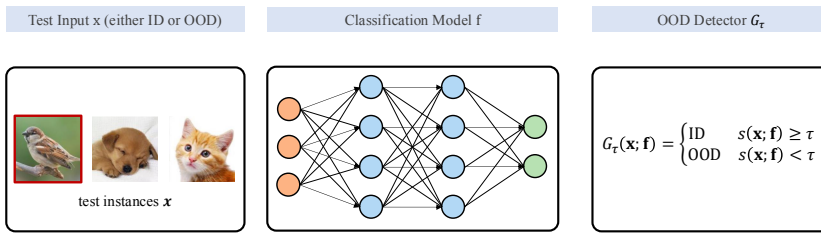

Figure 2.10: The overall framework of OOD detection.

make correct predictions for them. These data typically have the ground truth labels that do not belong to the considered label space of the models. OOD detection is crucial for machine learning models, especially for deep models, as they can produce unreliable or overconfident predictions when presented with OOD inputs. OOD detection is widely used for the field of medical analysis, auto-driving, and economy applications. In Figure 2.10, we present the overall framework of OOD detection.

Problem Setting. Let $\mathcal{X} \subset \mathbb{R}^d$ represent the feature space, while $\mathcal{Y} = \{1, \ldots, C\}$ denotes the label space for in-distribution (ID) data. We define $X_{\text{id}} \in \mathcal{X}$ and $X_{\text{ood}} \in \mathcal{X}$ as the random variables corresponding to ID and OOD data, respectively. The label random variables are represented as $Y_{\text{id}} \in \mathcal{Y}$ for ID data and $Y_{\text{ood}} \notin \mathcal{Y}$ for OOD data. The joint distribution for ID data is denoted by $P_{X_{\text{id}},Y_{\text{id}}}(\mathbf{x}, y)$, while $P_{X_{\text{ood}},Y_{\text{ood}}}$ represents the joint distribution for OOD data. The marginal distributions are given by $P_{X_{\text{id}}}$ for ID data and $P_{X_{\text{ood}}}$ for OOD data.

(1) OOD Score Functions. Let $\mathcal{D}_{\text{ID}}^{\text{Train}} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ be the training dataset consisting of ID samples drawn from the joint distribution

$P_{X_{\text{id}},Y_{\text{id}}}$. Following the framework established by Fang *et al.* (2022b), the goal of OOD detection is to train a detector G using $\mathcal{D}_{\text{ID}}^{\text{Train}}$ such that for any test sample $\mathbf{x}$: 1) if $\mathbf{x}$ is sampled from $P_{X_{\text{id}}}$, then G accurately classifies $\mathbf{x}$ into the correct ID classes; and 2) if $\mathbf{x}$ is drawn from $P_{X_{\text{ood}}}$, then G identifies $\mathbf{x}$ as OOD.

In this context, given a threshold τ, a pre-trained ID model $\mathbf{f}_{\boldsymbol{\theta}}$, and a scoring function S, a sample $\mathbf{x}$ is classified as ID if and only if $S(\mathbf{x};\mathbf{f}_{\boldsymbol{\theta}}) \geq \tau$:

$$\begin{cases} G_\tau(\mathbf{x}) = \text{ID} & \text{, if } S(\mathbf{x};\mathbf{f}_{\boldsymbol{\theta}}) \geq \tau \\ G_\tau(\mathbf{x}) = \text{OOD} & \text{, otherwise} \end{cases} \tag{2.1}$$

The success of OOD detection methods largely hinges on the design of the scoring function S and the model $\mathbf{f}_{\boldsymbol{\theta}}$, ensuring that the scores for OOD samples are consistently lower than those for ID samples.

(2) Outlier Exposure. To improve OOD detection performance, a fine-tuning approach known as *Outlier Exposure* (OE) (Hendrycks *et al.*, 2018) has been introduced. OE incorporates surrogate OOD data $\mathcal{D}_{\text{OUT}} = \{\mathbf{x}_j^s\}_{j=1}^m$ and applies a fine-tuning strategy based on empirical risk minimization, formulated as:

$$\arg\min_{\boldsymbol{\theta}} \frac{1-\alpha}{n} \sum_{i=1}^{n} \ell(\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_i), y_i) + \frac{\alpha}{m} \sum_{j=1}^{m} \ell_{\text{OE}}(\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_j^s)), \tag{2.2}$$

where α is a hyperparameter, ℓ denotes the loss function, and ℓ_{OE} represents the surrogate OOD loss. By leveraging surrogate OOD data, the model can learn to position certain OOD samples in latent embeddings that are distant from all ID classes, which typically enhances the performance of OOD detection.

Post hoc OOD detection. methods are specifically designed to differentiate between ID and OOD data without the need to retrain neural networks or modify their parameters. Some methods just rely on the output of the classifier to achieve OOD detection. For example, MSP (Hendrycks and Gimpel, 2017) assigns scores based on the highest softmax probability across ID categories. Energy (Liu *et al.*, 2020d) applies an energy function to logits for OOD score computation. OpenMax (Bendale and Boult, 2016) enhances this approach by replacing the softmax layer to directly estimate the likelihood of an input belonging

to an unknown class. TempScale (Guo *et al.*, 2017) refines softmax probabilities through temperature adjustment, while ODIN builds on TempScale by introducing input preprocessing to further improve OOD detection.

Other methods focus on features of the classifier, including LogitNorm (Wei *et al.*, 2022), which normalizes logits for better confidence calibration, and GradNorm (Huang *et al.*, 2021a), which computes the Kullback-Leibler (KL) divergence between the softmax probability distribution and a uniform distribution, using gradients from the penultimate layer as the OOD score. ReAct (Sun *et al.*, 2021) modifies feature vectors by applying a threshold based on specified magnitudes, and Energy employs energy-based models for uncertainty estimation. Mahalanobis (Lee *et al.*, 2018) fits class-conditional Gaussian distributions to the penultimate layer features, deriving OOD scores using Mahalanobis distance. ViM (Wang *et al.*, 2022b) augments logits with the norm of feature residuals compared to ID training samples. KNN (Sun *et al.*, 2022a) applies a k-nearest neighbors approach to penultimate layer features. DICE (Sun and Li, 2022) sparsifies the last linear layer before computing logits, while RankFeat (Song *et al.*, 2022b) transforms feature matrices to ensure they have a rank of one. ASH (Djurisic *et al.*, 2023) modifies activations in later layers by simplifying the remaining elements, and SHE (Zhang *et al.*, 2023b) maintains a template representation for each ID category, detecting OOD samples by measuring the distance between an input's representation and its corresponding template.

Training-time OOD detection. Many researchers find that post-hoc detection methods has limited capacity when conducting detection, indicating that conventionally trained models might not be so powerful for this task. It motivates researchers to explore fine-tuning-based methods, directly enhancing the capabilities of models in OOD detection.

Some methods use contrastive learning methods to improve the representations of models in discerning between ID and OOD. For example, contrasting shifted instances (CSI) (Tack *et al.*, 2020) contrasts a given sample with its distributionally shifted augmentations, alongside with a new detection score tailored for the proposed method. Self-supervised outlier detection (SSD) (Sehwag *et al.*, 2021) proposes an

effective data augmentation strategy that can synthesize data that can be viewed as OOD data, and then leverages self-supervised representation learning to capture meaningful features. It also suggests a Mahalanobis distance-based method in identifying OOD data.

Other methods further involve the surrogate OOD data during training. For example, outlier exposure (OE) learns to discern the pattern between ID and OOD data, directly making models learn representations of both ID and OOD samples. However, surrogate OOD data might not be informative enough to characterize the real (unseen) OOD distribution, motivating a series of subsequent works to further improve OE. Posterior Sampling-based Outlier Mining (POEM) (Ming *et al.*, 2022b) assumes that some OOD data are more informative than others, thereby suggesting a dynamic learning framework that balances the exploration of new OOD data and the exploitation of known useful outliers. Katz-Samuels *et al.* (2022) consider the noise situations that use unlabeled data collected from wild, naturally including both ID and OOD samples. To combat this challenging situation, the authors suggest a constrained optimization problem that maximizes OOD detection performance while minimizing mis-classification of ID data. Distributional-Augmented OOD Learning (DAL) (Wang *et al.*, 2023e) considers the situation where the auxiliary OOD data may suffer from the distribution gap over that of the real OOD data. DAL proposes augmenting the auxiliary OOD data by generating a set of candidate distributions within a Wasserstein ball centered around the original auxiliary OOD distribution. Then, the models learn from the worst-case data from the Wasserstein ball to ensure the unifromly well performance within the enlarged OOD distribution set, thereby mitigating OOD distribution gap.

Further Discussion. The emergence of Vision-Language Models (VLMs) has ushered in a new era for OOD detection, specifically through the utilization of pre-trained VLMs for this purpose. This shift has led to a significant rise in the use of textual information for visual OOD detection, yielding impressive results (Fort *et al.*, 2021; Ming *et al.*, 2022a; Wang *et al.*, 2023b). For instance, Fort *et al.* (2021) introduce a method that utilizes the names of potential outlier classes as input for CLIP (Radford *et al.*, 2021b). ZOC (Esmaeilpour *et al.*,

2022) enhances CLIP by incorporating a text-based image description generator, which produces candidates for OOD labels during testing. MCM (Ming *et al.*, 2022a) employs a straightforward approach by using the maximum predicted softmax value as the OOD score, representing an effective post-hoc detection method grounded in vision-language pre-training. CLIPN (Wang *et al.*, 2023b) trains a text encoder to enable CLIP to interpret negative prompts, thereby effectively distinguishing OOD samples based on the similarity differences between two text encoders and a fixed image encoder. Additionally, LSN (Nie *et al.*, 2024) leverages CLIP to create negative classifiers by learning from negative prompts, which helps in identifying images that do not belong to a specified category. NegLabel (Jiang *et al.*, 2024b) proposes a simple yet effective pipeline that involves selecting potential OOD labels from a comprehensive semantic pool, such as WordNet (Fellbaum, 1998), and subsequently utilizing a pre-trained VLM like CLIP to categorize input images into ID and OOD class groups. EOE (Cao *et al.*, 2024) harnesses Large Language Models (LLMs) to generate OOD labels, adapting its prompts to effectively address a variety of tasks.

2.3.2 Out-of-distribution Generalization

Out-of-distribution (OOD) generalization refers to the problem of training a classification model given data from certain environments to generalize well to data from unseen test environments. The underlying data distributions of samples from different environments may contain distribution shifts due to the environmental influence of data collection and processing. Machine learning models can easily fail in OOD generalization due to the violation of the i.i.d. assumption. However, as the distribution shifts are everywhere such as in autopilot systems and scientific discovery, failing to generalize robustly to OOD data may introduce unprecedented risks or fairness issues (Koh *et al.*, 2021). In Figure 2.11, we present one general illustration of the pipeline for OOD generalization.

Problem Setting. OOD generalization considers a standard supervised learning setting, where the data $\mathcal{D} = \{\mathcal{D}^e\}_{e\in\mathcal{E}_{\text{all}}}$ is collected from multiple causally related environments $\mathcal{E}_{\text{all}}$. The data $\mathcal{D}^e = \{\mathbf{x}_i^e, y_i^e\}$ from

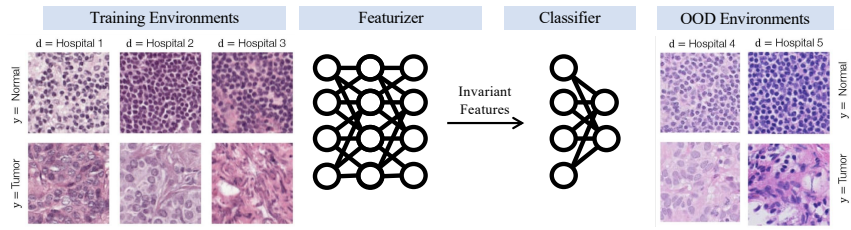

Figure 2.11: Illustration of the pipeline for OOD generalization. Dataset examples are from Koh *et al.* (2021).

a single environment $e \in \mathcal{E}_{\text{all}}$ are drawn independently from an identical distribution P^e (Peters *et al.*, 2016). The objective of OOD generalization is to solve for a predictor $f : \mathcal{X} \to \mathcal{Y}$ such that $\max_{e \in \mathcal{E}_{\text{all}}} \mathcal{L}_e(f)$, where $\mathcal{L}_e$ is the empirical risk (Vapnik, 1991) under environment e, $\mathcal{X}$ and $\mathcal{Y}$ are the input and labeling spaces, respectively. In other words, f needs to generalize well to all (unseen) environments given the access of training environments $\{\mathcal{D}^e\}_{e \in \mathcal{E}_{\text{tr}}}$, The predictor can be decomposed as a feature extractor $\varphi : \mathcal{X} \to \mathcal{Z}$ and a classifier to extract useful features, and a classifier $w : \mathcal{Z} \to \mathcal{Y}$ to make predictions based on the extracted features. For example, φ can be a deep neural network and w can be a simple linear classifier at the last layer (Koh *et al.*, 2021).

Many flavors of invariance for OOD generalization. It is the key to identifying the invariance across different distributions and environments for OOD generalization. Therefore, it has been developed a rich literature of multiple definitions of invariance. **Domain Generalization** establishes the invariance across different domains or environments (Ganin *et al.*, 2016; Sun and Saenko, 2016; Li *et al.*, 2018). However, it has been shown that domain invariant features can not guarantee good OOD generalization performance (Zhao *et al.*, 2019). Moreover, **Distributionally Robust Optimization** requires the models to be robust to mild perturbations onto the training distributions, such that the models are expected to perform well in unseen OOD distributions (Namkoong and Duchi, 2016; Hu *et al.*, 2018; Sagawa *et al.*, 2020).

Recently, the concept of **Causal Invariance** has been adopted to learn invariant representations that capture the direct causal parents of

the target variable (Peters *et al.*, 2016; Rojas-Carulla *et al.*, 2018). Inspired by the Independent Causal Mechanism (ICM) in causality (Peters *et al.*, 2017a), the causal invariance principle considers the generation of the environments are interventions onto the underlying data generation process, and the causal mechanism generating the target variable given its direct parents is independent from the interventions. Therefore, predictions based only on the direct parents of the target variable are invariant to distribution shifts. Arjovsky *et al.* (2019) first implement the causal invariance principle in the deep networks as the framework of Invariant Risk Minimization (IRM), which has inspired a number of invariant learning works (Parascandolo *et al.*, 2021; Mahajan *et al.*, 2021; Wald *et al.*, 2021; Ahuja *et al.*, 2021; Krueger *et al.*, 2021; Shi *et al.*, 2022; Rame *et al.*, 2022a).

The presence of environment or domain labels is crucial to the success of OOD generalization. However, the environment labels may not always be available. Creager *et al.* (2021) and Liu *et al.* (2021d) propose to estimate the environment label by predicting invariance information. Liu *et al.* (2021b), Zhang *et al.* (2022c), and Pezeshki *et al.* (2024) try to infer labels based on the failures of an ERM model. Nevertheless, the inference of environment labels may not be possible without additional inductive biases (Lin *et al.*, 2022). Therefore, Lin *et al.* (2022) and Tan *et al.* (2023) propose to incorporate auxiliary information about the data generation process to identify the invariance.

In addition, the challenge of OOD generalization also emerges in a broader scope. Chen *et al.* (2022), Gui *et al.* (2022), Chen *et al.* (2023e), and Yao *et al.* (2024a) study the OOD generalization on graphs, where the invariant features are considered as critical subgraphs (Miao *et al.*, 2022; Chen *et al.*, 2024c). Gagnon-Audet *et al.* (2023) and Xie *et al.* (2024) study the evolving invariant features along the time.

Optimization challenge in OOD generalization. The trade-off between ERM and OOD generalization is inevitable due to their intrinsic conflicts. Gulrajani and Lopez-Paz (2021) show that most existing OOD algorithms fail to outperform ERM in domain generalization under rigorous and fair comparison. Sagawa *et al.* (2020) and Zhai *et al.* (2023) find that regularization on ERM, or sacrificing ERM performance, is usually needed for achieving satisfactory OOD performance. There is

often a trade-off between in-distribution and OOD performances (Zhao *et al.*, 2022; Sadeghi *et al.*, 2022; Teney *et al.*, 2023). Chen *et al.* (2023g) study and tackle the issue by formulating the OOD generalization as a multi-objective optimization problem. Zhang *et al.* (2022a) and Chen *et al.* (2023f) dive into the optimization challenge from the feature learning perspective and propose to learn rich feature representations to tackle the issue. In addition, Chen *et al.* (2023f) show the limitations of OOD generalization methods in learning useful features than simple ERM, especially in deep neural networks (Rosenfeld *et al.*, 2022; Zhang *et al.*, 2022a). The success of rich feature learning aligns with the empirical success of weight average (Rame *et al.*, 2022b), model ensemble (Arpit *et al.*, 2022; Lin *et al.*, 2024d) and flatness-aware optimization (Cha *et al.*, 2021) in OOD generalization.

Future Discussions. Recently, large pre-trained models, especially large Vision-Language Models (VLMs) have gained huge success in tackling the distribution shifts. The pretraining on an unprecedented scale of real-world data enables large pre-trained models to learn rich world knowledge and to adapt to new environments. In particular, large VLMs demonstrate remarkable performance in OOD generalization across a wide range of vision and multimodal tasks, surpassing conventional ImageNet-trained models by a large margin. It has attracted surging interest from the community to understand the OOD generalization capabilities of large VLMs. Fang *et al.* (2022a) show that the pretraining data distribution has the most significant influence on the generalization capabilities of the large VLMs than other factors such as model architectures or the number of training samples.

Santurkar *et al.* (2023) identify factors such as captioning quality are crucial for the generalization capabilities of large VLMs by comparing to single-modal learning, for which Huang *et al.* (2024d) provide a theoretical explanation. Mayilvahanan *et al.* (2024) show that large VLMs are able to generalize well beyond memorizing matching similar training examples. However, Wang *et al.* (2024b) curate a real-world dataset and show that large VLMs still learn spurious features. Consequently, large VLMs may perform even worse than conventional models trained on ImageNet and cause the hallucination issue when incorporated with large language models. Therefore, it remains an open problem of how to improve the OOD generalization capabilities of large VLMs.

2.4 Adversarial Examples and Defense

Adversarial Examples (AEs) are crafted by applying adversarial attacks to pretrained deep neural networks (Szegedy *et al.*, 2014; Goodfellow *et al.*, 2015). It contains human imperceptive perturbations but AEs can easily obfuscate well-trained classifiers and cause classification accuracy nearly drop to zero drastically. In Figure 2.12, it presents the general paradigm of the generation of AEs through adversarial attack. As society increasingly relies on large-scale models, AEs pose serious safety threats to their reliability. Hence, it is crucial to design robust defense strategies against adversarial attacks to ensure the trustworthiness of machine learning models. This section introduces three predominant adversarial defense frameworks since the discovery of AEs.

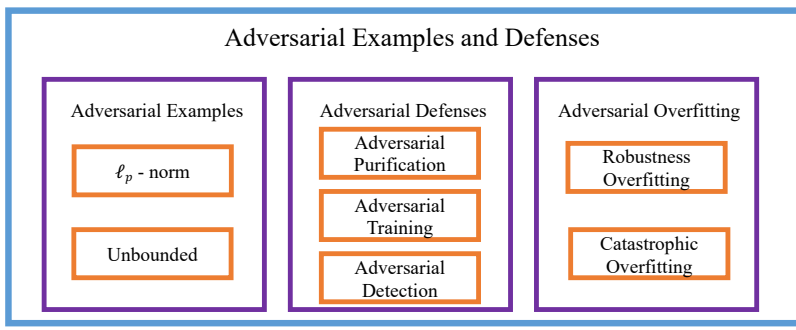

Figure 2.12: Paradigm of adversarial examples and adversarial defense.

2.4.1 Adversarial Purification

Adversarial purification (AP) has recently emerged as an effective adversarial defense framework and a focus on different aspect of the system with other adversarial defense frameworks. AP aims to achieve input level robustness by adapting denoising methods to recover clean data from the adversarial examples. Since pretrained classifiers are robust and accurate to natural data classification, AP methods focus on projecting the adversarial data back to the natural data manifold (Meng and Chen, 2017). Compared to the excellent performance of *Adversarial*

Training (AT) (Madry *et al.*, 2018), AP exhibits strong transferabilities and robustness to unseen attacks because it often designs a off-to-shelf purification module and require no extra modifications to the classifier. In Figure 2.13, we present an overall framework of adversarial purification.

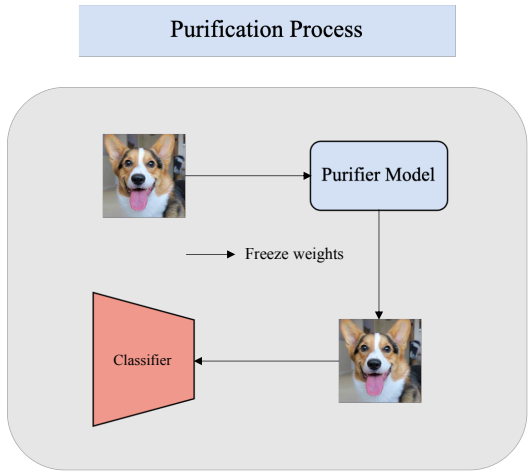

Figure 2.13: The overall framework of adversarial purification.

Problem Setup. Adversarial perturbations can be viewed as imperceptible noises added onto clean data. Hence, AP is formalized as a input-level denoising process to recover clean images from the perturbed images. In specific, the input is preprocessed by a purifier module f_θ before classification:

$$\min_{\phi,\theta} \mathbb{E}_{p_{\text{data}}(\mathbf{x},y)} \left[\max_{\mathbf{x}' \in \mathcal{B}(\mathbf{x})} \mathcal{L}(g_\phi(f_\theta(\mathbf{x}')), y) \right]. \tag{2.3}$$

Here the preprocessor f_θ is often an off-the-shelf generative model that can restore clean images from attacked images. where $\mathcal{L}$ is a loss function, g_ϕ is a classifier, $\mathbf{x}' \in \mathbb{R}^d$ is an adversarial image, y is the true label of $\mathbf{x}$, d is the data dimension, and $\mathcal{B}$ is the maximum allowed perturbation ball, usually within ℓ_∞-norm or ℓ_2-norm.

Based on this formulation, the research studies have developed from two perspectives: (1) improving the robustness by directly incorporating traditional denoising techniques, (i.e., using Auto-encoder or GAN-based

purifier; Liao *et al.*, 2018; Samangouei *et al.*, 2018), and (2) preprocessing the input by utilizing diffusion generative models to denoise adversarial perturbations (i.e., DiffPure; Nie *et al.*, 2022).

Representation-level Denoising Adversarial Purification aims to mitigate adversarial perturbations by either removing them from raw inputs or reconstructing clean latent representations before passing them to the model for prediction. MagNet deploys a detector-reformer module to cleanse adversarial perturbations from adversarial inputs before they reach the network (Meng and Chen, 2017). The detector identifies potential adversarial examples (AEs) by comparing them to the distribution of normal samples. Any examples that remain unidentified are reconstructed by an autoencoder-based reformer, which adjusts the input to align more closely with the data manifold. The autoencoder is trained exclusively on clean, legitimate data with reconstruction objective to ensure it effectively filters out adversarial perturbations while preserving the core characteristics of the input. Similarly, Liao *et al.* (2018) proposes *High-Level Representation Guided Denoiser* (HGD) which adopts a U-Net to purify AEs by aligning their high-level representations with clean inputs to ensure robust feature extraction. Since Generative Adversarial Networks (GANs) demonstrate their ability to generate high-quality images, the concept of DefenseGAN was proposed to enhance the robustness of deep learning models (Samangouei *et al.*, 2018). Given an adversarial input, GAN finds a reconstruct input sample $\mathbf{z}^*$ in its latent space which the process is optimized by the objective:

$$\mathbf{z}^* = \arg\min_{\mathbf{z}} \|\mathbf{x} - G(\mathbf{z})\|_2^2. \tag{2.4}$$

Then $\mathbf{z}^*$ is passed to the generator to generate a clean sample that closes to original data distribution. Differently, Self-supervised Online Adversarial Purification (SOAP) leverages self-supervised learning tasks to purify adversarial perturbations by enforcing robust feature representations (Shi *et al.*, 2021). This approach combines supervised and self-supervised signals, achieving competitive robustness while avoiding computationally expensive adversarial training.

Diffusion-based Adversarial Purification (DBP) leverages the powerful generative ability of diffusion models to recover clean images from the perturbed images. Yoon *et al.* (2021b) first leverages

score-based generative model to recover clean images from adversarial inputs. The training objective of the generative model is Denoising Score Matching (DSM), which aims to estimate the score of original data distribution (Song and Ermon, 2019). Nie *et al.* (2022) proposed DiffPure, the first approach to leverage official diffusion models for adversarial purification. It concludes as two-step purification that involves first adding Gaussian noises through forward diffusion process to input images and then denoising the images through reverse diffusion process. Gaussian noises are larger in scale than adversarial perturbations so that the forward diffusion process can be viewed as gradually covering the perturbations. The paper suggests that using a small diffusion timestep is more suitable for the AP task. This is because adding excessive Gaussian noise during the forward process often results in a loss of semantic information, which degrades the sampling quality of new generated clean data. Also, a novel approach Robust Diffusion Classifier (RDC) directly utilizes diffusion functionality to classify AEs (Chen *et al.*, 2024a). It operates by first maximizing the data likelihood of input through a diffusion process and then classifying the optimized data using the conditional likelihood provided by the diffusion model. This method capitalizes on the inherent properties of diffusion models to approximate data distributions accurately across the input space.

Further Discussion. Currently, DBP methods have shown state-of-the-art performance on adversarial robustness to various adversarial attacks. However, this area lacks reliable evaluations of DBP methods. This includes designing strong adaptive attacks against the stochasticity within diffusion models because the traditional adaptive attack such as BPDA+EOT attack has been shown as weak in recent work (Lee and Kim, 2023). Current strong adaptive attack such as PGD+EOT attack requires iterative optimization steps and each step then contains multiple diffusion function evaluations. Accordingly, evaluating the defense systems often requires extensive time and resources cost. Furthermore, DBP methods exhibit the intrinsic problem of accuracy-robustness trade-off. Specifically, the performance of purifying adversarial images depends on the selection of timestep t within diffusion models. If the Gaussian noise is not enough, then adversarial perturbations cannot be fully removed after the reverse process. Inversely, if the Gaussian

noise is too much, then the purified samples will lose their original semantic meanings (i.e., in this case, it is more like generating new images instead of recovering original images). In the future, research directions could focus on improving the efficiency and trade-off problem of DBP methods and designing reliable robustness evaluations.

2.4.2 Adversarial Detection

Adversarial detection (AD) is one of the most lightweight defense strategies, which focuses on identifying whether an incoming sample is from a clean or adversarial data distribution. AD aims to reject the incoming sample if it is identified as an AE and the key of it is to distinguish the discrepancy between AEs and clean samples. One significant advantage of AD is its compatibility with existing machine learning systems. Specifically, AD can be seamlessly integrated into a machine learning system with only minor modifications. Furthermore, AD-based methods mainly focus on identifying and rejecting AEs, leaving the performance on clean samples nearly unaffected. We present the overall framework of AD in Figure 2.14.

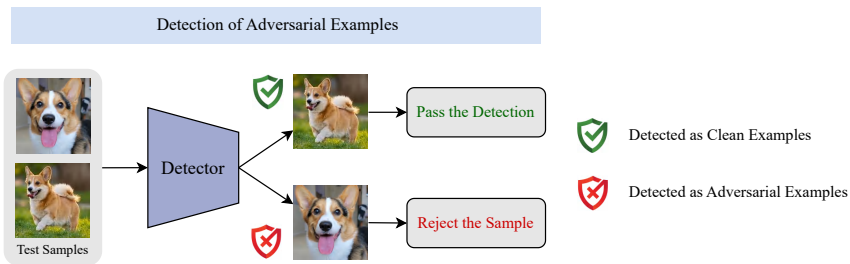

Figure 2.14: The overall framework of adversarial detection.

Problem Setup. Formally, AD can be considered as an binary classification problem. Given a *deep neural network* (DNN) $f : \mathcal{X} \to \mathcal{Y}$, where $\mathcal{X}$ denotes the input space and $\mathcal{Y}$ denotes the label space, AD introduces a separate function $D : \mathcal{X} \to \{0, 1\}$ to determine whether a sample $\mathbf{x}$ is an AE or an clean sample:

$$D(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x} \text{ is an AE,} \\ 0, & \text{if } \mathbf{x} \text{ is a clean sample.} \end{cases}$$

Then, $\mathbf{x}$ will be rejected if $D(\mathbf{x}) = 1$, i.e., it is identified as an AE. Research work in AD can be roughly categorized into three main methods: feature-based adversarial detection, classifier-based adversarial detection and statistical adversarial detection.

Feature-based Adversarial Detection mainly leverages hidden-layer features of DNNs to filter out AEs from test data. For example, Ma *et al.* (2018) propose to use local intrinsic dimensionality of DNN features as a detection metric, which quantifies the local geometric properties around a sample based on the distribution of distances to its neighbours. Lee *et al.* (2018) implement a Mahalanobis distance-based score for identifying AEs. Specifically, their method fits pre-trained DNN features with class-conditional Gaussian distributions under Gaussian discriminant analysis, which enables detection without requiring changes to pre-trained softmax classifiers. Raghuram *et al.* (2021) propose an unsupervised framework to detect AEs, which uses a meta-algorithm to extract intermediate layer representations of DNNs, offering configurable components for detection. In addition, Deng *et al.* (2021) transform the last few layers of a pre-trained DNN into Bayesian layers with pre-trained model weights. Then, it fine-tunes the Bayesian neural network to detect AEs by adding uniform noises to samples.

Classifier-based Adversarial Detection is another prevalent strategy, which involves equipping classifiers with a rejection option. This approach allows the model to not return predictions on uncertain inputs. For example, Stutz *et al.* (2020) introduce a *confidence-calibrated adversarial training* (CCAT) framework, which guides the model to make low-confidence predictions on AEs, thereby determining which samples to reject. Although CCAT can capture some aspects of prediction certainty, it tends to overestimate the certainty, particularly on misclassified samples. To mitigate this issue, Pang *et al.* (2022) introduce the concept of *True Confidence* (T-Con), which is defined as the predicted probability assigned to the true label. T-Con serves as a certainty oracle, indicating a classifier's confidence in its correct prediction. However, since the true label is unknown during inference, T-Con cannot be directly computed. To address this limitation, they further propose *Rectified Confidence* (R-Con), which is derived through a rectified rejection module trained to predict T-Con based on the input

and the classifier's output. R-Con provides an estimate of T-Con, which can effectively separate AEs out.

Statistical Adversarial Detection has delivered increasing insight recently. Previous methods often train a detector for specific classifiers or adversarial attacks, and thus tend to overlook the modeling of data distribution, which can limit their effectiveness against unknown attacks. Unlike other detection-based defense methods, *statistical adversarial detection* (SAD) leverages statistical methods to evaluate the discrepancies between the clean and adversarial distributions. Given the fact that fundamental discrepancies exist between clean and adversarial distributions, SAD offers statistical guarantees against adversarial attacks. Besides, SAD-based methods are effective against adaptive attacks. The philosophy behind this is that, to mislead a SAD-based detector into identifying AEs as clean samples, an adaptive attack must generate samples that can narrow the distributional discrepancy between clean samples and AEs. Thus, this process can indeed align AEs closer with clean samples, making the adaptive attack harder to mislead a well-trained classifier.

One typical example of statistical adversarial detection is Gao *et al.* (2021), which demonstrate that *maximum mean discrepancy* (MMD) (Gretton *et al.*, 2012) is aware of adversarial attacks. Specifically, they replace the Gaussian Kernel with an effective deep kernel with a maximized testing power. Then, they apply wild bootstrap to overcome the issue that AEs may be non-independent. Finally, their proposed MMD statistic can effectively distinguish the discrepancies between AEs and clean samples. Based on this, Zhang *et al.* (2023d) further propose a new statistic called *expected perturbation score* (EPS) that measures the expected score of a sample after multiple perturbations. The philosophy is to perturb samples by injecting various noises, and therefore can extract information from its diverse multi-view observations. Then, an EPS-based MMD is proposed to measure the distributional discrepancy between clean samples and AEs.

Further Discussion. Feature-based and classifier-based adversarial detection often generalize poorly to unseen attacks. On the other hand, statistical adversarial detection can mitigate this issue by consider the distribution information of AEs and clean samples. However,

SAD-based defense methods require processing data in batches during inference. When the batch size is too small or contains a mix of AEs and clean samples, the stability of the detector can be affected. To mitigate this issue, future research could explore more robust statistical methods capable of detecting distributional discrepancies with fewer samples. One key advantage of applying a statistical hypothesis test is its ability to effectively control the false alarm rate. Fang *et al.* (2022b) theoretically demonstrate that for single-instance-based detection to function perfectly, there must be a gap in the support set between *in-distribution* (IID) and *out-of-distribution* (OOD) data. This principle can be extended to adversarial settings. However, such a gap in the support set does not exist in adversarial settings, making perfect single-instance-based detection generally infeasible in adversarial scenarios.

2.4.3 Adversarial Training

Adversarial training (AT) has emerged as a foundational framework to improve the robustness of DNNs against adversarial attacks. Its core idea lies in generating AEs during training and use the generated AEs to train the DNN, forcing the model the learn the underlying distribution of AEs (Madry *et al.*, 2018). By effectively addressing the vulnerabilities of DNNs, AT has been widely applied in many real-world applications such as autonomous driving systems (Shibly *et al.*, 2023), medical image segmentation (Hanif *et al.*, 2023) and anomaly detection (Zhu *et al.*, 2022). We present the overall framework of AT in Figure 2.15.

Problem Setup. Mathematically, AT can be formalized as a min-max optimization problem. Specifically, the inner maximization generates AEs by solving the following constrained optimization problem:

$$\max_{\boldsymbol{\Delta}} \ell(f(\mathbf{x}+\boldsymbol{\Delta}), y), \text{ subject to } \|\boldsymbol{\Delta}\|_p \leq \epsilon, \tag{2.5}$$

where ℓ is a loss function, f is a model, $\mathbf{x} \in \mathbb{R}^d$ is a natural image, y is the true label of $\mathbf{x}$, $\boldsymbol{\Delta} \in [-\epsilon, \epsilon]^d$ is the adversarial perturbation added to $\mathbf{x}$, $\|\cdot\|_p$ is the ℓ_p-norm, d is the data dimension, and ϵ is the maximum allowed perturbation budget. The outer minimization optimizes the model weights to correctly classify generated AEs:

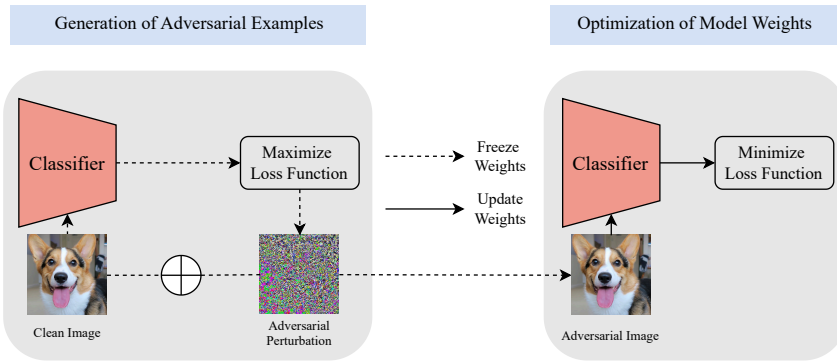

Figure 2.15: The overall framework of adversarial training.

$$\min_{f \in F} \frac{1}{n} \sum_{i=1}^{n} \ell(f(\mathbf{x}_i + \boldsymbol{\Delta}_i^*), y_i), \tag{2.6}$$

where $\tilde{\mathbf{x}}_i = \mathbf{x}_i + \boldsymbol{\Delta}_i^*$ is the most adversarial variant of $\mathbf{x}_i$ within the ϵ-ball centered at $\mathbf{x}_i$, $\boldsymbol{\Delta}_i^* \in [-\epsilon, \epsilon]^d$ is the optimized adversarial perturbation added to $\mathbf{x}_i$ by solving (2.5), y_i is the true label of $\mathbf{x}_i$, ℓ is a loss function, and F is the set of all possible neural network models.

Based on the above formulation, the research studies aim to develop AT from two key perspectives: (1) improving the inner maximization by refining the generation process of AEs (i.e., generation-refined AT), and (2) improving the outer minimization by refining the optimization process of the model (i.e., optimization-refined AT).

Generation-refined Adversarial Training aims to improve the inner maximization in (2.5) (e.g., improve the quality of generated AEs or simplify the generation process). One intuitive approach is to enhance the strength of adversarial attacks. For example, vanilla AT (Madry *et al.*, 2018) propose to iteratively optimize the inner maximization, which outperforms the previously proposed one-step attack (Goodfellow *et al.*, 2015) by a notable margin. However, this will lead to an inevitable increase in computational complexity, making vanilla AT a resource-consuming approach. To mitigate this issue, another line of research focuses on efficient AT methods. For example, Shafahi *et al.* (2019) introduce a method to accelerate the generation of AEs by simultaneously updating both the model parameters and adversarial perturbations

in a single backward pass, eliminating the need for separate gradient computations at each step. Wong *et al.* (2020) propose to integrate random initialization into the AT framework. This approach allows AEs generated by a one-step attack to achieve comparable effectiveness to those generated by iterative attacks, while at the same time, significantly reducing the computational cost.

Besides random initialization, incorporating other perturbation initialization techniques such as learnable initialization (Jia *et al.*, 2022b) and knowledge-driven initialization (Jia *et al.*, 2022a) can also enhance the data diversity of AEs. In addition, another line of research focuses on adjusting the attack intensity. For example, Liu *et al.* (2020a) point out that large perturbation budgets can result in suboptimal perturbation initializations. Based on this, they propose a periodic adversarial scheduling strategy to dynamically adjusts the maximum allowed perturbation budgets during training. More recently, Zhang *et al.* (2024a) observe that different pixels contribute differently to adversarial robustness and standard accuracy. Base on this observation, they propose to pixel-wisely adjust maximum allowed perturbation budgets during the generation of AEs, which aims to guide the model to focus on important pixel regions during training.

Optimization-refined Adversarial Training aims to improve the outer minimization in (2.6) (e.g., improve the design of objective functions). For example, Zhang *et al.* (2019) propose to optimize a regularized surrogate loss function, which captures the trade-off between the adversarial robustness and standard accuracy. Wang *et al.* (2020b) investigate the impact of misclassified samples on the model performance. They discover that misclassified samples can significantly affect the adversarial robustness. Based on this observation, they propose a new loss function to include a distinct differentiation of misclassified samples through regularization. Wu *et al.* (2020) find that there is a positive correlation between the model weight flatness and robust generalization performance. To improve this flatness, they propose to regularize the flatness of weight loss landscape by adding adversarial perturbations to model weights. Another line of research studies focus on reweighting AEs (i.e., assign different weights to different AEs). For example, Ding *et al.* (2020) propose to assign instance-dependent maximum allowed

perturbation budgets ϵ to AEs. Zhang *et al.* (2021d) propose a *geometry-aware instance-reweighted AT* (GAIRAT) framework, which assigns different weights to adversarial loss based on the distance of data points from the decision boundary. However, GAIRAT uses discrete and path-dependent metrics to measure the closeness, which makes it computationally unstable. To mitigate this issue, Wang *et al.* (2021b) propose to use probabilistic margins to reweight AEs due to their continuous and path-independent nature.

Further Discussion. Despite the success of AT in defending against various adversarial attacks, there are still numerous open questions that need to be addressed. For example, improving adversarial robustness will lead to a notable decrease in standard accuracy, affecting the performance of AT-based methods on clean samples.

In the future, research directions could involve developing AT methods that could improve standard accuracy without sacrificing adversarial robustness. For example, aligning the distribution of AEs towards the direction of the distribution of clean samples, while at the same time, maximizing the cross-entropy loss during the generation of AEs might be a possible solution. The intuitive philosophy is to create AEs that are closer to the distribution of clean samples.

2.4.4 Adversarial Overfitting

While *adversarial training* (AT) is widely recognized as the most reliable training paradigm against adversarial attacks, the phenomena of *robust overfitting* (RO) (Rice *et al.*, 2020) and *catastrophoic overfitting* (CO) (Wong *et al.*, 2020) in both single-step and multi-step AT have emerged as critical bottlenecks, fundamentally limiting the continuous progression of model robustness. Consequently, a series of studies have been conducted with the aim of mitigating adversarial overfitting to overcome these inherent performance bottlenecks and further enhance AT's scalability and practicality.

Problem Setup. Distinct from (benign) overfitting in natural training, where over-learning on the training set can still result in good test performance, RO manifests as a continuous degradation of robustness on the test adversarial examples during extended multi-step AT. Be-

sides, unlike conventional data overfitting, CO is characterized by a paradigm overfitting in which the model's robustness against single-step adversarial attacks (training paradigm) abruptly rises to nearly 100%, while its defense against multi-step adversarial attacks (test paradigm) simultaneously collapses from peak to nearly 0%. Moreover, both RO and CO exhibit distinct features of decision boundary distortion. RO leads to significant decision boundary distortion; although the model can correctly classify perturbed training data, it remains vulnerable to adversarial examples generated from test data. CO results in severe decision boundary distortion, achieving flawless classification on perturbations generated by the single-step adversarial attacks, such as FGSM (Goodfellow *et al.*, 2015), but becoming completely vulnerable to adversarial examples generated through the multi-step adversarial attacks, such as PGD (Madry *et al.*, 2018). We illustrate the phenomena of RO and CO in Figure 2.16.

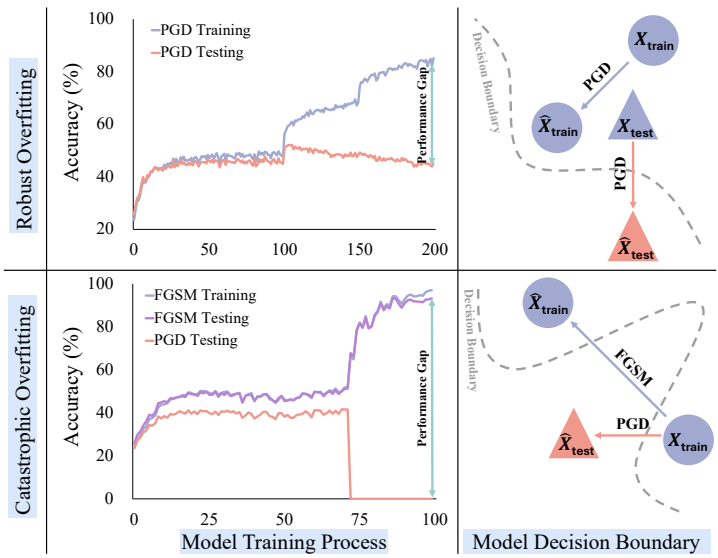

Figure 2.16: The phenomena of robust and catastrophic overfitting.

Robustness Overfitting. Rice *et al.* (2020) initially demonstrates that conventional remedies for natural overfitting have limited effectiveness in addressing RO. To mitigate model overfitting on adversarial examples, Schmidt *et al.* (2018) theoretically and empirically shows that

leveraging larger training datasets can substantially alleviate robust overfitting. Following this direction, Carmon *et al.* (2019) leverages semi-supervised learning with vast amounts of unlabeled data to mitigate RO, as well as circumventes the labor-intensive manual annotation. Recently, Li and Spratling (2023) suggest that simply extending the training dataset through sophisticated data augmentation strategies can effectively eliminate RO. Distinct from the aforementioned approaches, another line of research focuses on understanding the underlying mechanisms of robust overfitting and mitigating it without relying on additional training data. As a pioneering study, Chen *et al.* (2020b) demonstrates that injecting weight smoothing during multi-step adversarial training provides an effective approach to mitigate RO. Following this, Wu *et al.* (2020) identifies a positive correlation between the distorted weight loss landscape and the occurrence of RO and correspondingly proposes incorporating adversarial weight perturbation to flatten the model's landscape under the worst-case scenarios. Yu *et al.* (2022) further reveal that RO is caused by optimizing small-loss adversarial examples, and demonstrated that applying weight perturbation solely to the small-loss subset can reliably prevent its occurrence. From the perspective of minimax optimization, Wang *et al.* (2023f) suggests that improving attack strength can rebalance the minimax game and mitigate RO.

Catastrtophic Overfitting. Wong *et al.* (2020) first identifies the phenomenon of CO, and empirically indicates that existing countermeasures designed to alleviate natural overfitting and RO are entirely ineffective in addressing CO. As a result, Wong *et al.* (2020) proposes using large-magnitude initialization and early stopping to avoid CO, and Jorge Aranda *et al.* (2022) further enhance this approach by employing aggressive initialization and unbounded adversarial perturbations. However, Andriushchenko and Flammarion (2020) demonstrate that the above methods merely delay the onset of CO rather than reliably preventing it, particularly when confronted with more challenging scenarios, such as stronger adversaries. To effectively mitigate CO, Andriushchenko and Flammarion (2020) proposes explicitly maximizing gradient alignment within the perturbation set to avoid nonlinear decision boundaries around adversarial examples and prevent CO. Moreover,

several studies have identified the relationship between distorted decision boundaries and fixed single-step perturbation size, leading to proposals for either instance-dependent perturbation sizes (Huang *et al.*, 2023b) or reduced perturbation magnitudes for successfully misclassified adversarial examples (Kim *et al.*, 2021).

Recently, Lin *et al.* (2023a) observes the existence of abnormal adversarial examples within single-step adversarial training, where their associated loss paradoxically decreases after adding adversarial perturbations. Based on this, Lin *et al.* (2024b) further reveals a vicious cycle between the optimization of abnormal adversarial examples and the extent of model distortion and proposes a regularization term designed to suppress the generation of these abnormal examples. Subsequent research demonstrates that different model layers undergo distinct changes during CO, with earlier layers exhibiting greater sensitivity due to the formation of pseudo-robust shortcuts. Specifically, the model's reliance on pseudo-robust shortcuts for decision-making, while enabling precise defense against single-step adversarial attacks, bypasses genuine robustness learning, ultimately leading to decision boundary distortion and triggering CO. As a solution, Lin *et al.* (2024b) propose layer-aware adversarial weight perturbation, which applies stronger penalties to earlier layers to mitigate the model's stubborn reliance on pseudo-robust shortcuts.

Further Discussion. Despite success in separately addressing natural overfitting, RO, and CO, the solutions for these overfitting types remain isolated from each other. Therefore, developing a unified understanding and providing a universal solution for different types of overfitting represents a promising research direction. As a pioneering effort, from the perspective of memorization effects, Lin *et al.* (2024a) reveals shared over-memorization behavior among these three types of overfitting and proposes a preliminary general solution. Furthermore, although some pilot studies have been conducted (Li and Li, 2023), theoretical explanations for RO and CO remain relatively scarce. This scarcity significantly limits our understanding of their underlying mechanisms. Finally, exploring RO and CO under different ℓ_p-norms, such as ℓ_1 and ℓ_0, emerges as an important avenue for future investigation (Zhong *et al.*, 2023).

3

Trustworthy Private and Secured Learning

The rapid advancement of machine learning technologies has brought unprecedented capabilities to data analysis and decision-making processes, yet it has simultaneously raised critical concerns about privacy and security. Trustworthy private and secured learning has emerged as a crucial research direction that aims to develop learning systems that not only achieve high performance but also provide robust guarantees for data privacy, model security, and result reliability. This field integrates various privacy-preserving techniques, such as differential privacy, federated learning, and secure multi-party computation, to ensure that sensitive information remains protected throughout the learning process. At the core of trustworthy private and secured learning lies the challenge of balancing the trade-offs between model utility, privacy preservation, and security guarantees.

This section constructs a dynamic framework of adversarial evolution, tracing the spiral development of privacy-preserving machine learning through iterative cycles of vulnerability exposure to defense enhancement. Figure 3.1 provides an overall framework of this section. Section 3.1 establishes differential privacy not as an endpoint but as the opening move in this technological arms race. While its noise injection

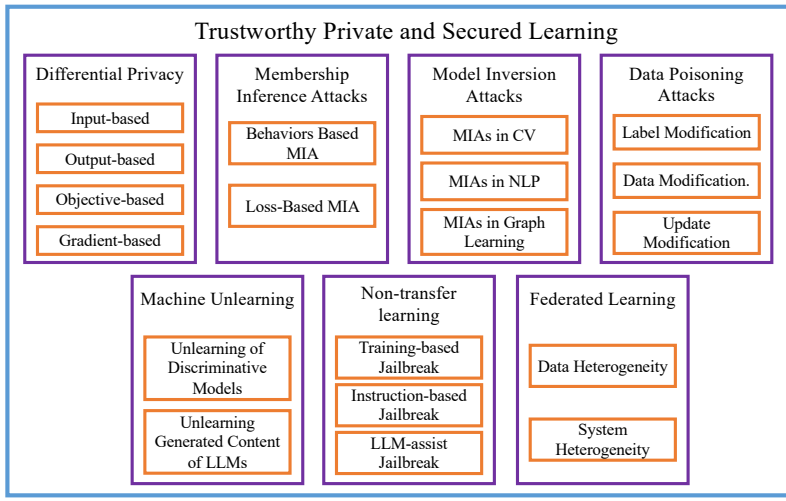

Figure 3.1: The overall framework of trustworthy private and secured learning.

mechanism obscures individual data traces, it inadvertently fuels the rise of membership inference attacks in Section 3.2. This countermeasure exposes the insufficiency of isolated noise mechanisms, forcing differential privacy to evolve from merely output protection to holistic training process safeguards, creating an adaptive defense loop. As surface-level protections mature, Section 3.2 reviews model inversion attacks that mark a tactical escalation: adversaries leverage model memory of training data to reconstruct raw sensitive information from gradients or prediction confidence scores. This shifts privacy risks from "exposure of data participation" to "reconstruction of data content", compelling defenses to hybridize differential privacy with model regularization, balancing noise injection and feature compression. Section 3.4 presents data poisoning attacks transit from passive observers to active saboteurs, corrupting training data to manipulate model behavior. This paradigm shift demands defenses pivot from post-hoc mitigation to preventive hardening, integrating robust optimization to build immune barriers in training pipelines. Section 3.5 introduces machine unlearning then extends defenses across the temporal dimension, which can erase harmful data influence in training and also can realize regulatory mandates like the "right to be forgotten." Section 3.6 explores the protection methods

for the intellectual property of model owners from the perspective of reshaping the generalization abilities. Finally, Section 3.7 discusses federated learning that compounds all prior challenges into a distributed framework. This synthesis propels privacy preservation from fragmented techniques toward systemic, architecture-level solutions.

3.1 Differential Privacy

In an era where data-driven technologies permeate every facet of society, ensuring the privacy and security of individual information has become paramount. Differential privacy emerges as a rigorous mathematical framework that enables the analysis and sharing of data while safeguarding individual identities (Dwork *et al.*, 2006). By introducing carefully calibrated randomness into data outputs, differential privacy ensures that the inclusion or exclusion of a single data point does not significantly affect the overall analysis, thereby protecting personal information. This framework is particularly impactful in machine learning, where it allows models to learn from sensitive datasets while maintaining robust privacy guarantees.

Differential privacy is defined in the following context. Given a database $\mathcal{X}$ of size n, an adversary may query the data via a function$f : X^n \to Y$, with the data being managed by a trusted mechanism M that maps X^n to Y to ensure privacy. Formally, an algorithm M is said to satisfy (ϵ, δ)-differential privacy if, for any two neighboring databases $\mathcal{X}$ and $\mathcal{X}'$ that differ by at most one entry (*i.e.*, $\|\mathcal{X} - \mathcal{X}'\|_0 \leq 1$), and for any subset $S \subseteq Y$, the following inequality holds:

$$\Pr[M(\mathcal{X}) \in S] \leq e^{\epsilon} \Pr[M(\mathcal{X}') \in S] + \delta, \tag{3.1}$$

where ϵ and δ control the privacy guarantees. Specifically, if $\delta = 0$, the mechanism is considered ϵ-pure differentially private. For $\delta > 0$, the mechanism is approximately private, allowing for a small probability $(1 - \delta)$ of violating strict privacy. This definition ensures privacy by comparing the outputs of two "neighboring" databases, requiring the mechanism M to be "stable" over all possible inputs in X^n. In most cases, the "neighboring" property is measured using the Hamming distance (ℓ_0-norm), though other metrics can apply depending on context.

Differential Privacy Mechanisms. In practice, many fundamental mechanisms are used to implement differential privacy by adding well-calibrated noise to data responses. The primary objective of these mechanisms is to protect individual data privacy while minimizing accuracy loss in results. (1) Laplace Mechanism. This mechanism is based on the Laplace distribution, defined as $Lap(b) = \frac{1}{2b} exp(-\frac{|x|}{b})$. For a given function $f : X^n \to \mathbb{R}^k$, the Laplace mechanism adds independently generated noise to each output component of $f(X)$ to achieve privacy:

$$M_\epsilon(x) = f(x) + (Y_1, Y_2, \ldots, Y_k), \tag{3.2}$$

where each Y_i is an independent and identically distributed (i.i.d.) random variable drawn from the Laplace distribution *i.e.*, $Y_i \sim Lap(\Delta_1^{(f)}/\epsilon)$, and $\Delta_p^{(f)} = \max_{\mathcal{X},\mathcal{X}':\|\mathcal{X}-\mathcal{X}'\|_0 \leq 1} \|f(\mathcal{X}) - f(\mathcal{X}')\|_p$ denotes the l_p sensitivity. This mechanism guarantees $(\epsilon, 0)$-differential privacy by ensuring that small changes in the input data produce statistically indistinguishable outputs, thus preserving privacy (Dwork and Roth, 2014). (2) Gaussian Mechanism. The Gaussian mechanism is suitable for functions $f : X^n \to \mathbb{R}^k$ with ℓ_2-sensitivity $\Delta_2^{(f)}$. Like the Laplace mechanism, it introduces noise to each output component of $f(X)$, which is drawn from a Gaussian distribution rather than a Laplace distribution. The Gaussian mechanism provides (ϵ, δ)-differential privacy, which is particularly useful for high-dimensional data queries where the ℓ_2-norm provides better stability than the ℓ_1-norm (Dwork and Roth, 2014).

In high-dimensional settings ($k \gg 1$), the Gaussian mechanism is advantageous as it scales with the square root of the dimension, offering a more controlled privacy-accuracy tradeoff. (3) Exponential Mechanism. For functions with discrete outputs, the Exponential Mechanism is ideal. It is widely applied in scenarios where the goal is to select an output from a finite set while maintaining privacy. Suppose we have a dataset of n individuals $\mathcal{X} \in X^n$, a finite set of possible outputs H, and a scoring function $s : X^n \times H \to \mathbb{R}$ that measures the relevance of each outcome $h \in H$ with respect to the data X. The Exponential Mechanism selects an outcome h with a probability proportional to $\exp(\epsilon s(X, h)/2\Delta_s)$, where Δ_s is the sensitivity of the scoring function s. This mechanism ensures differential privacy by favoring outcomes with higher scores while maintaining privacy protection across possible outcomes.

Currently, the risk of privacy breaches is exposed due to the pervasive application of machine learning technologies in privacy-related applications, which necessitates using privacy-sensitive datasets to train models. Recent research has demonstrated that machine learning models are susceptible to a variety of privacy attacks, including model inversion (Fredrikson *et al.*, 2015) and membership inference attacks (Shokri *et al.*, 2017). These attacks are conducted by adversaries who endeavor to extract sensitive information that data proprietors are unwilling to disclose. Consequently, it is imperative to implement deep learning techniques that prioritize privacy. These techniques are intended to safeguard the sensitive information in training data or models during the learning process. In this context, we investigate the application of differential privacy to machine learning models, which is fundamentally accomplished by introducing perturbations to the training data, models, or intermediate results and outputs. As shown in Figure 3.2, input perturbation, output perturbation, objective perturbation, and gradient perturbation are the four strategies in private machine learning.

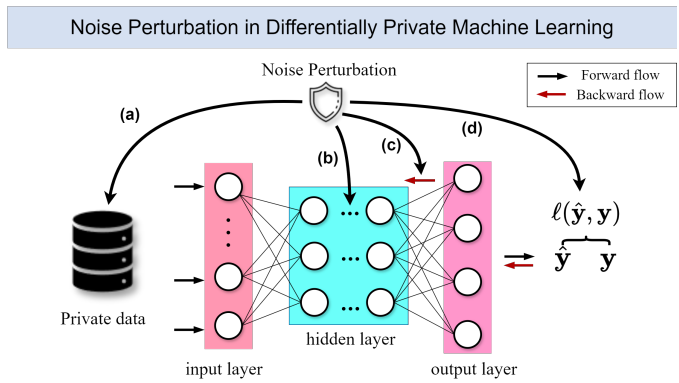

Figure 3.2: Noise perturbation approaches in differentially private machine learning: (a) input perturbation, (b) output perturbation, (c) gradient perturbation, (d) objective perturbation.

Input-based perturbation approaches. Fukuchi *et al.* (2017) proposed the first differential private empirical risk minimization (ERM) framework that was based on input perturbation. This mechanism involves data proprietors introducing noise to their data prior to transmit-

ting it to data collectors. The perturbed data is then utilized to train a public model. Nevertheless, the utility of the model can be substantially influenced by changes in feature values, as it necessitates the extraction of information from the training data.

Output-based perturbation approaches. Lecuyer *et al.* (2019) introduced the earliest output-based perturbation method, which is intended for private regularized empirical risk minimization. This method has been extensively employed in neural networks for training on large-scale datasets by injecting noise into the second-to-last layer of the model to produce a differentially private noisy layer. The privacy-preserving property is maintained in the final layer of the model, as per the post-processing theorem of differential privacy. Subsequently, Phan *et al.* (2019) investigated the sensitivity of models and evaluated the trade-off between privacy preservation and model utility.

Objective-based perturbation approaches. The initial concentration of this approach was on machine learning models, such as logistic regression, as proposed by Chaudhuri and Monteleoni (2008). Subsequently, Kifer *et al.* (2012) offered a more sophisticated and efficient analysis, which enhanced the output perturbation mechanism to ensure more robust privacy guarantees. This was accomplished by reducing the noise and relaxing the differentiability requirement for the regularizer, which thereby broadened its applicability to problems with rigid constraints. In subsequent research, Phan *et al.* (2016) expanded the model to include private deep autoencoders and probabilistic generative models, such as private deep belief networks, which were approximated using Taylor and Chebyshev expansions. In these investigations, the training objective's polynomial form was perturbed. More recent studies, such as Iyengar *et al.* (2019), have proposed an approximation to the minima perturbation method, as earlier works made significant assumptions about the loss function. Under standard assumptions, this procedure is applicable to all objective functions and ensures (ϵ, δ)-DP, regardless of whether the model output is a true minimum of the noisy objective function. This work demonstrates that achieving an approximate minimum of the objective function is sufficient to assure privacy guarantees, thereby making objective perturbation more feasible in random convex optimization settings. Nevertheless, these publications

continue to impose convexity assumptions on the loss function. While working with a discrete domain, Neel *et al.* (2020) relaxed these assumptions, necessitating only that the loss function be bounded. The authors necessitated that the loss function be Lipschitz continuous with regard to its parameters for continuous domains.

Gradient-based perturbation approaches. Gradient perturbation is an additional prevalent approach to differentially private machine learning, which entails the incorporation of noise into gradient descent. Noise gradient descent, which applies stochastic gradient descent (SGD) to convex loss functions and L2-regularized objectives, was introduced by Song *et al.* (2013) in pioneering work. Starting from this, Abadi *et al.* (2016) created the DP-SGD algorithm for the private training of deep models. Gaussian noise is incorporated into the truncated gradients in this method to safeguard the privacy of the data. Additionally, the privacy loss is monitored using the moments accountant, a robust privacy loss accounting method that establishes precise limits that surpass the capabilities of sophisticated composition theorems. The initial dynamic privacy budget allocation strategy was developed by Lee and Kifer (2018) in order to reduce the consumption of the privacy budget while ensuring the performance of the model. A portion of the privacy budget is allocated to calculate the perturbed gradient during each iteration in this strategy, while the remaining portion is used to optimize the step size using a differentially private noise minimization algorithm. This guarantees that the allocation's efficacy is not compromised by the perturbations that are incorporated into the gradient.

Xie *et al.* (2021) observed that the gradient value (and consequently its norm) is inversely proportional to the number of iterations, resulting in variable privacy leakage risks across iterations. The privacy budget is typically distributed equitably among iterations in the majority of DP-SGD implementations. Increasingly precise documentation of gradient values is required as training progresses and gradients tend to decrease. In order to resolve this issue, they suggested an adaptive noise-reducing algorithm for DP-SGD that adaptively allocates a portion of the privacy budget to each iteration. The practical challenge of applying techniques such as DP-SGD to large-scale models, such as neural networks, persists. Yu *et al.* (2021) devised a method to reduce memory costs in order

to make this more feasible. This was accomplished by modifying the representation of gradient vectors/matrices and weights, which led to a correspondingly modified gradient perturbation algorithm.

Further Discussion. (a) Scalability and Efficiency: Examine the computational obstacles linked to the application of differential privacy in extensive machine learning models and explore various strategies to improve efficiency. (b) Examining the continuing research focused on balancing the balance between upholding stringent privacy regulations and ensuring the usability of machine learning models. (c) Regulatory Compliance and Standardization: Analyze the function of differentiated privacy in fulfilling legal obligations for data protection and the initiatives aimed at creating consistent standards across various sectors. (c) Progress in Differential Privacy Techniques: Emphasize novel approaches and instruments aimed at enhancing the efficacy and relevance of differential privacy across diverse machine learning scenarios.

3.2 Membership Inference Attacks

Machine learning, a data-dependent training model, has recently garnered widespread attention due to its high accuracy and capacity to handle complex tasks. More and more trained machine learning models have been used in people's lives or industries. However, the data-driven training process and the heavy use of models inevitably give rise to many data-related issues. Among these, the most commonly discussed problems typically revolve around training data privacy concerns and data copyright infringement issues. In these situations, membership inference attacks(MIAs) is a useful technology to monitor the illegal data used during the training step of a model, but also a good evaluation method to measure the model's security regarding the data. Specifically, MIA aims to identify which data were used as training data (i.e., member data) to train the target model, while data that were not used for training are classified as non-member data. Whether data was used to train the target model is also referred to as membership attribution. The general process of MIA is shown in Figure 3.3. MIA can be roughly divided into two types: one is the more precise but computationally intensive detection method based on model behavior; the other is the

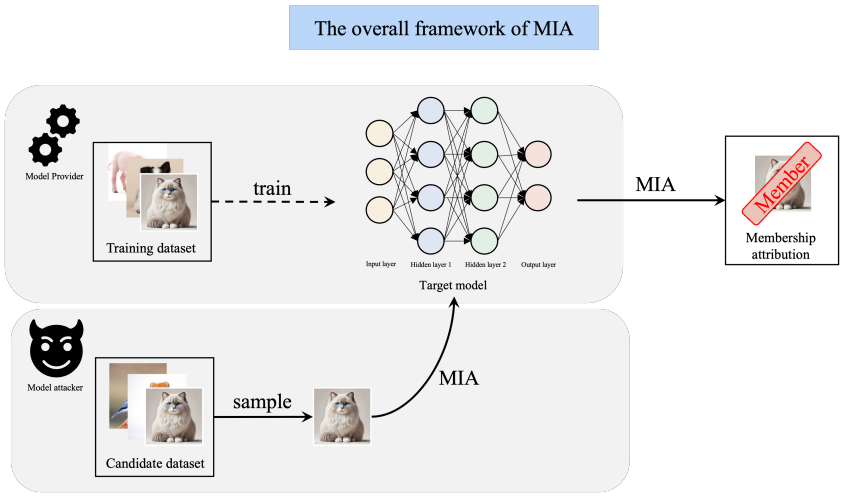

Figure 3.3: The main process of Membership Inference Attack (MIA) involves an attacker determining whether a specific image was part of the training dataset used by the model trainer to train the target model.

less resource-consuming but less accurate detection method based on model output. All in all, MIA is a problem that is often used to deal with model data, but currently has some limitations.

Problem Setup: A basic membership inference attack is conducted under the condition where a trained model and a candidate dataset are provided. The goal is to identify the training data from the candidate dataset. The traditional MIA problem can be defined as follows:

Given a pre-trained model f_θ parameterized by weight θ and candidate dataset $D = \{x_1, x_2, x_3..., x_n\}$. A part of the candidate dataset D_m is the training data for the model f_θ. The remaining part of the dataset D_n is the hold-out set. MIA assesses each x_i and assigns a membership attribute m_i to it. Where $m_i = 1$ if $x_i \sim D_m$; otherwise $m_i = 0$. An MIA algorithm MIA is designed to predict whether or not x_i is in D_m:

$$MIA(x_i, \theta) = \begin{cases} 1, & P(m_i = 1|\theta, x_i) \geq \tau \\ 0, & P(m_i = 1|\theta, x_i) \leq \tau \end{cases} \tag{3.3}$$

where $MIA(x_i, \theta) = 1$ means x_i is identified as member data(comes from D_m), τ is the threshold. The pre-trained model we want to attack is denoted as θ.

MIA based on model behavior differences (behaviors based MIA) is one of the primary principles behind current membership inference attacks (MIA) The key idea is that whether a specific data is part of the training dataset influences the behavior of the target model when the model processes these data (or data similar to it) (Niu *et al.*, 2024).

This type of MIA method leverages these behavioral differences to infer the membership attribution of the target model. Specifically, this method involves randomly sampling multiple subsets from a candidate dataset and training several reference models (shadow models) using these subsets (Liu *et al.*, 2022a). For a given data x in the candidate dataset, this process allows us to obtain a group of models where x is part of the training set and another group of models where x is not included in the training set. Based on these models, the goal is to estimate the parameter probability distribution of models trained on datasets that include the data point x, denoted as $P(\theta|x)$, and the parameter probability distribution of models trained on datasets that do not include x, denoted as $P(\theta|x')$, where x' represents the case where x is not in the training set (Carlini *et al.*, 2022). However, since calculating the parameter distributions directly is challenging, most methods use the loss function $l(f(x), y)$ as a proxy for the model parameters. Here, l represents the loss function of the model when evaluated on data x with the corresponding label y (Carlini *et al.*, 2022).

This method was first proposed by Liu *et al.* (2022a), establishing the foundational principles for behavior-based MIA. However, this method clearly has two significant limitations. First, repeatedly training shadow models is highly resource-intensive, so this approach can only be used to attack models with simple structures handling straightforward tasks. Second, the accuracy of this method is closely tied to the structure of the shadow models; the shadow models need to have a structure similar to the target model to achieve better results.

To address these shortcomings, many improved methods have been proposed. LiRA (Carlini *et al.*, 2022) introduced the idea of treating $P(l(f(x), y)|x)$ as a normal distribution, which allows estimation of $P(l(f(x), y)|x)$ with only a small number of shadow models (at least four). Furthermore, the research pointed out that previous studies used

"accuracy" as the evaluation metric while neglecting the importance of the false positive rate (Carlini *et al.*, 2022). These metrics do not reflect whether the attack can confidently identify any member of the training set. Therefore, the paper proposed that MIA methods should be evaluated based on their true positive rate at low false positive rates (e.g., $\leq 0.1\%$) and found that most previous attacks perform poorly under this criterion (Carlini *et al.*, 2022). Before this, many previous methods showed poor performance on this metric. The study introduced the Likelihood Ratio into the algorithm. By calculating (3.4), the above shortcomings can be addressed,

$$\frac{Pr(\theta|x)}{Pr(\theta|x')}. \tag{3.4}$$

The improvements to the algorithm and evaluation metrics have made this method an important benchmark in the field. Furthermore, to address the issue of poor performance in black-box network attacks, L-leak (Yan *et al.*, 2022) proposed that it is not necessary to use shadow models with the same structure as the target model for membership attribution detection. Instead, logical constraints can be introduced during the training of shadow models (which may not have the same structure as the target model) to ensure that the shadow models exhibit "roughly similar" behavior to the target model at a logical level (Yan *et al.*, 2022) .

In simple terms, the Figure 3.4 illustrates the target model's output structure for a specific image, with the detection result being 2 (75% confidence). The model's logits suggests that the image is most likely 5, and unlikely to be 1 and 7. The trained shadow model must adhere to this basic logits when detecting the image but does not need to match the exact confidence level. This approach removes structural constraints when training shadow models, addressing scenarios where the target model is a black box. It also allows for the use of simpler structures to attack more complex models, thereby reducing resource consumption.

Recently, Gradient-Leaks (Liu *et al.*, 2023a) transformed the process of training multiple full shadow models into constructing a single local ML model (Liu *et al.*, 2023a). By leveraging the differences in observed gradients of the local model, it determines membership information.

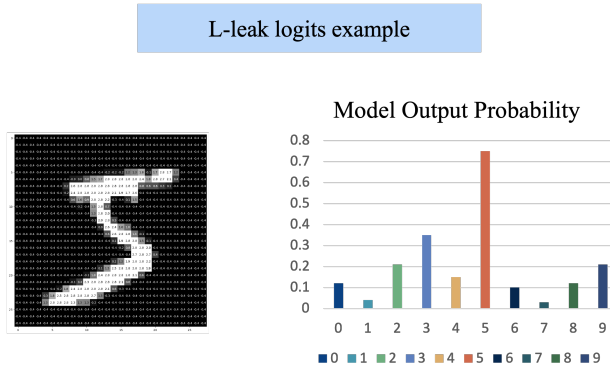

Figure 3.4: The L-Leak method constructs a reference model that mimics the behavior of the target model based on its logits. In this example, the model's logits indicate that the image is most likely to be a 5 and least likely to be a 1 or 7.

Since training a local ML model requires significantly less time compared to a full model, this method can save substantial computational resources.

The latest state-of-the-art method, RMIA (Zarifzadeh *et al.*, 2024), further minimizes the need for shadow models. With this approach, membership inference can be accomplished with the assistance of just one shadow model. Current methods require a large number of shadow models because they need to estimate $P(\theta|x)$ and $P(\theta|x')$. In RMIA, building on LiRA, the problem is transformed into function (3.5),

$$P(\theta \mid x) = \frac{P(x \mid \theta)P(\theta)}{P(x)}. \tag{3.5}$$

In this way, there is no need to directly calculate $P(\theta|x)$. Instead, $P(x)$ can be computed as the empirical mean of $P(x|\theta')$ by sampling shadow models θ' (Zarifzadeh *et al.*, 2024). Using this approach, even with just one shadow model, it is still possible to estimate $P(x)$. The experiments in the paper demonstrate that this method significantly outperforms previous approaches, especially in resource-constrained scenarios.

Output-Based MIA (Loss-Based MIA) is a type of membership inference attack with fewer constraints compared to the behavior-based MIA methods. The basic principle of this approach is to leverage the

tendency of model over-fitting on training data for membership attribution detection. During training, models often over-fit their training data, so when handling a member data, the model typically returns a result with lower loss compared to non-member data. By observing the difference in loss between member and non-member data, membership inference attacks can be performed (Yeom *et al.*, 2018). This type of method is often used to attack models that behavior-based MIA cannot handle, such as generative models. While behavior-based MIA has addressed issues like high resource consumption and the inability to attack black-box models over the years, there are still tasks where behavior-based MIA is not applicable.

For instance, when training generative models, the input is typically random noise, which is usually not recorded after training. This makes it impossible to access inputs consistent with the training phase during MIA attacks on generative models, thus preventing the training of shadow models. Additionally, for models like diffusion models, which require substantial resources during training, even training one additional shadow model during the attack phase incurs unacceptable resource consumption. In such cases, loss-based MIA is more suitable than behavior-based MIA. The use of over-fitting tendency for membership inference was first proposed in Yeom *et al.* (2018). However, this method did not consider the varying difficulty of the model in processing different data, resulting in poor detection performance.

Take image classification models as an example: Some images are inherently harder to classify, while others are easier to determine. For data that are difficult to distinguish, even if it belongs to the target model's training set, the model will exhibit higher losses when processing it. In contrast, for easily distinguishable images, even if they do not belong to the target model's training set, they are still easy to classify. To address this issue, many studies have combined loss-based MIA with behavior-based MIA (Ye *et al.*, 2022; Ye *et al.*, 2023). With the emergence of generative adversarial models (GANs) (Wang *et al.*, 2017), loss-based MIA found its most suitable application scenario. LOGAN (Hayes *et al.*, 2017) was the first to propose membership inference attacks (MIA) targeting GANs. However, this method does not directly attack the GAN itself, but rather its discriminator (Hayes *et al.*, 2017). Since the

discriminator in a GAN is a simple image classification model tasked with distinguishing between real and generated images, its training process does not involve random noise. Therefore, LOGAN adopted the principles of behavior-based MIA to implement its attack.

Subsequently, a more general method, GAN-Leak (Chen *et al.*, 2020a), was proposed, which examines membership attribution through the loss differences when generating target data. GAN-Leak posits that generative models tend to produce better results for member image data than for non-member image data. With its simplicity, good performance, and flexibility in deployment, GAN-Leak established the foundational principles and baseline for loss-based MIA targeting generative models (Duan *et al.*, 2023; Fu *et al.*, 2024; Fu *et al.*, 2023).

With the advent of diffusion models (Croitoru *et al.*, 2023), Duan *et al.* (2023) were the first to propose using the loss differences in intermediate denoising steps of diffusion models to distinguish between member and non-member data. Their method extended the principles of GAN-Leak, suggesting that when a diffusion model attempts to generate member image data, the intermediate noise prediction loss is smaller than that for non-member image data.

Building on this, SecMI (Duan *et al.*, 2023) re-modeled the training steps to more accurately replicate the original model's forward and backward processes, thereby improving detection accuracy. Additionally, other methods have explored adding noise to target images and letting the model reconstruct the images to determine whether the model retains memory of those images, thus verifying their membership attribution (Fu *et al.*, 2024).

Further Discussion. With the development and widespread application of various machine learning technologies, data security has become a critical research issue. However, existing methods still face significant limitations in many scenarios and problems. For instance, can current MIA (Membership Inference Attack) methods effectively detect data usage issues in teacher models when only the distilled student models are accessible? Furthermore, although there has been some research on MIA for large models, these studies have faced considerable skepticism. Many researchers argue that current MIA studies on large models involve unrealistic experimental setups that do not align with the real-world training processes of large models.

3.3 Model Inversion Attacks

Model inversion attacks (M-Inverse[1]) are a type of privacy attack in which adversaries use a trained machine learning model to extract private information from its training data. The goal of M-Inverse is to recreate representative features of the inputs used in model training, as shown in Figure 3.5. M-Inverse, which was first introduced on shallow models by Fredrikson *et al.* (2014), have since evolved to address deep neural networks. M-Inverse uses prior knowledge and specific attack techniques to reconstruct different types of input data. M-Inverse has been studied in a variety of machine learning domains, including computer vision, natural language processing (NLP), and graph learning. In computer vision, adversaries often target classification models with the goal of inferring class-specific private information. M-Inverse in NLP aims to recover sensitive training texts, whereas M-Inverse in graph learning aim to infer the topology or edge connectivity of the graph. The methodology and effectiveness of M-Inverse are shaped by their domain-specific characteristics.

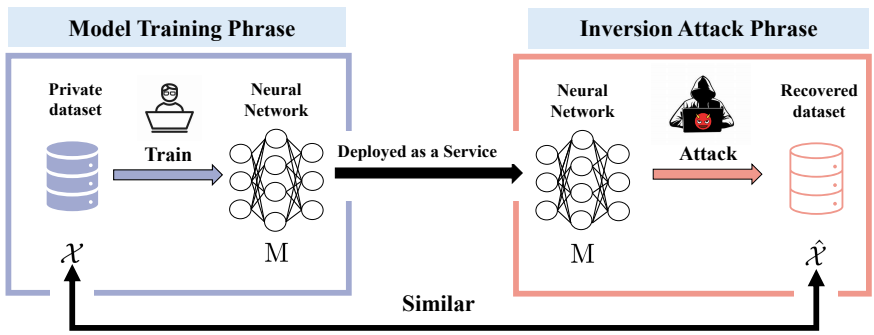

Figure 3.5: General framework of model inversion attack.

Problem Setup. M-Inverse is the process of inverting a well-trained machine learning model to extract private information about its training data. Specifically, given a trained model $\mathrm{M}(\cdot)$ and prior knowledge $\mathcal{K}$,

[1]In the previous section, MIA refers to Membership Inference Attack. To avoid confusion in this work, we adopt M-Inverse for Model Inversion Attack. However, in model inversion papers, MIA can also be used to denote model inversion attack.

the objective of a model inversion attack is to design an algorithm $\mathcal{A}$ that reconstructs as much private information as possible from the original training data $\mathcal{X}$. The algorithm $\mathcal{A}$ typically uses M outputs, such as probabilities or embeddings, to achieve representative and successful reconstructions. In an ideal scenario, a successful attack would yield reconstructed data that closely resembles the training data $\mathcal{X}$, $\hat{\mathcal{X}} = \mathcal{A}(\mathrm{M}, \mathcal{K})$. Adversaries are assumed to know the domain of the private data and can utilize public datasets from the same domain to acquire auxiliary knowledge. Additional auxiliary knowledge depends on the attack setting. In a white-box setting, adversaries have full access to the target model's architecture, parameters, and gradients, allowing them to exploit internal details for data reconstruction. Meanwhile, in a black-box setting, adversaries can only perform input-output queries, observing model outputs without access to the model's internals. A label-only setting is a particular kind of black-box setting in which confidence scores are not available and only the hard label is accessible.

M-Inverse has been studied across various domains, using unique characteristics of each domain to infer private information about the model's training set. We introduce 'M-Inverse' across three specific domains: computer vision, natural language processing, and graph learning. General frameworks for each are illustrated in Figure 3.6. For each domain, we examine how M-Inverse leverages domain-specific knowledge and attack techniques to compromise privacy.

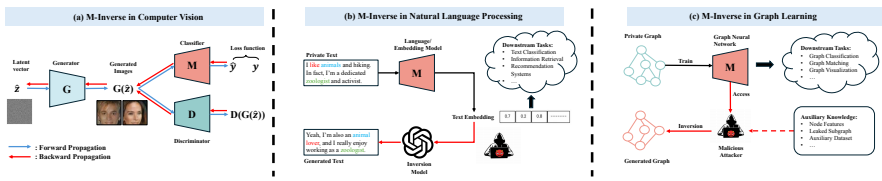

Figure 3.6: General framework of MIAs, with examples across three domains: (a) computer vision, (b) natural language processing, and (c) graph learning.

M-Inverse in Computer Vision. The concept of M-Inverse was first proposed by Fredrikson *et al.* (2014), where they applied M-Inverse on linear regression models. Building on this foundation, Fredrikson *et al.* (2015) applied M-Inverse to shallow networks by proposing a gradient

descent algorithm that exploits the differentiability of these networks. This traditional MIA aimed to reconstruct private data by optimizing a loss function $\mathcal{L}$ to minimize the discrepancy between the target class c and the target model's prediction. Specifically, the objective is to find an input x^* that minimizes $\mathcal{L}(\mathrm{M}, \hat{x}, c)$, where M is the target model, and $\hat{x}$ is the synthetic image. However, this traditional MIA was restricted to shallow networks and grayscale inputs.

To address these limitations above, Zhang *et al.* (2020b) introduced generative M-Inverse, which leverages generative adversarial networks (GANs) to learn prior image characteristics from a data distribution $P(X_{\text{prior}})$. GAN consists of a generator $G : Z \to X_{\text{prior}}$, mapping latent codes $z \in Z$ to synthetic data, and a discriminator $D : X \to \mathbb{R} \in [0, 1]$, which distinguishes between real samples $x \sim P(X_{\text{prior}})$ and generated samples $G(z)$. The generator and discriminator competes with each other in a max-min optimization game, where the generator G aims to maximize the likelihood of the discriminator misclassifying $G(z)$ as real, while the discriminator minimizes its classification error. Under this framework, the MIA objective reformulates the optimization to find a latent code z^* that minimizes $\mathcal{L}(\mathrm{M}, G(z), c)$, ensuring that the generated image $G(z^)$ resembles private images $x \in X_{\text{target}}$ in the feature space. This generative framework significantly enhances M-Inverse' capabilities to reconstruct representative samples on deep neural networks.

More recent works have advanced generative M-Inverse with state-of-the-art techniques that improve attack performance. In the white-box setting, Chen *et al.* (2021b) introduced an inversion-specific GAN that incorporates soft labels produced by the target model, enabling recovery of the target class distribution by optimizing the mean and deviation of the latent space. Wang *et al.* (2021a) further advanced M-Inverse by formulating a variational objective to balance diversity and accuracy. Then, An *et al.* (2022) first introduced StyleGAN (Karras *et al.*, 2019) with distribution clipping to enhance inversion performance.

Generative M-Inverse have also benefited from advanced loss functions as standard cross-entropy loss caused gradient vanishing problems. To overcome this problem and boost MI performance, Struppek *et al.* (2022) investigated M-Inverse in the high-resolution setting by leveraging StyleGAN with Poincaré loss. Besides, they also introduced random transformations and robust result selection, enhancing inversion robust-

ness. Nguyen *et al.* (2023) revisited core objectives by introducing logit maximization as a better loss function and mitigated model overfitting through model augmentation. Building on this, Yuan *et al.* (2023b) used a max-margin loss to optimize latent vectors, decoupling the search space by training a conditional GAN (cGAN) with pseudo-labels generated from the target model. Most recently, Peng *et al.* (2024b) iteratively fine-tuned the generator with pseudo-private data after each attack round, effectively increasing the likelihood of sampling true private data and further refining inversion performance.

In the black-box setting, Yang *et al.* (2019) trained an inversion model using an auxiliary dataset derived from background knowledge and a truncation-based approach. Then, Kahla *et al.* (2022) designed a boundary repulsion algorithm that evaluates the target model's predicted hard labels over a sphere to estimate the update direction. Later, Han *et al.* (2023a) reframed the optimization process in the latent space as a Markov Decision Process (MDP) and applied reinforcement learning to solve it. Nguyen *et al.* (2024) advanced the label-only setting by transferring decision knowledge from the target model to a surrogate model, effectively transforming the label-only scenario into a white-box one. Besides, they used a Target model-assisted ACGAN to further enhance knowledge transfer between target and surrogate models.

M-Inverse in Natural Language Processing. In NLP, M-Inverse is often treated as optimization problems. However, the discrete nature of text makes traditional brute optimization computationally expensive and time-consuming. To address this, Song and Raghunathan (2020) introduced a continuous relaxation, which assigns continuous variables to words and therefore enables gradient-based optimization. Targeting embedding models, they learned a reverse network to map embeddings back to word sets, supported by multi-label classification and multi-set-prediction for evaluation and refinement.

Targeting text classification models, Parikh *et al.* (2022) focused on recovering sensitive personal information, such as addresses or social security numbers and treated M-Inverse as a sentence completion problem. They provided a sentence prefix and used discrete optimization on logits for suffix tokens. Besides, Zhang *et al.* (2022d) leveraged a language generation models like GPT-3 as an attack model. They optimized the

hidden state of the attack model so that it generated text that matched the distribution of the private dataset.

Generative approaches have advanced text inversion techniques. Li *et al.* (2023a) introduced a word-by-word generation method, decoding inputs using target text embeddings as initial token representations. Besides, Morris *et al.* (2023) proposed Vec2Text, combining controlled generation with iterative refinement to align hypothesis embeddings with target embeddings. Extending this, Chen *et al.* (2024b) adapted Vec2Text for multilingual scenarios.

M-Inverse in Graph Learning. Olatunji *et al.* (2023) introduced graph reconstruction attacks by treating feature explanations as auxiliary knowledge. They proposed two approaches: an explanation-only assault based on feature similarity and an explanation-augmentation attack that takes node features into account. Zhou *et al.* (2023a) examined how adjacency matrices can be recovered using GNN latent variables, viewing graph reconstruction as a Markov chain approximation.

To address edge discreteness, Zhang *et al.* (2021f) developed a projected gradient module and a graph auto-encoder to leverage topology, node attributes, and model parameters for edge inference. Shen *et al.* (2022) proposed a model-agnostic graph recovery attack that only relies on node embedding matrix without interacting with the node embedding models. Chanpuriya *et al.* (2021) focused on learning a mapping from embeddings back to graphs, further inferring private information encoded in the embeddings. Then, Liu *et al.* (2023b) extended M-Inverse to both homogeneous and heterogeneous graphs, adapting M-Inverse on different graph types. Additionally, Zhang *et al.* (2022f) explored embedding-based information leakage in graphs, offering theoretical guarantees for M-Inverse to reveal sensitive graph details.

Further Discussion. (a) Cost of M-Inverse on Complex Models: The increasing complexity of models make it challenging for current M-Inverse to successfully invert large models such as transformer-based language models with quite reasonable computational costs. Future studies should focus on designing efficient and cost-effective attack algorithms to reduce high computational costs. (b) Dependency on Prior Knowledge: M-Inverse usually relies strongly on auxiliary data, including leaked datasets or model architecture. Future research needs to

reduce this dependence or develop effective methods that need less prior knowledge. (c) Evaluation metrics and benchmarks: Current metrics sometimes fail to capture the full performance of M-Inverse, particularly in specialized domains such as computer vision. Meaningful comparisons require uniform benchmarks and more varied and domain-relevant measures, including semantic consistency or feature-level similarity.

3.4 Data Poisoning Attacks

With the remarkable advancements in deep learning across various domains, its data-driven and data-hungry nature has heightened models' vulnerability to data poisoning attacks. The fundamental concept of data poisoning attacks is to inject "poisons" i.e., malicious data into the training set, aiming to disrupt model training, degrade model performance, or even implant backdoor within the model. With the growing prevalence of transfer learning (Zhuang *et al.*, 2020) and pretrained models (Devlin *et al.*, 2019), data poisoning attacks have acquired the transferability across different models, which introduces hidden threats and risks to models deployed in real-world scenarios. Additionally, although federated learning (Kairouz *et al.*, 2021) has strengthened privacy protections, the lack of transparency in local client training data has further exacerbated the risk of data poisoning attacks. As deep learning models gain broader broader adoption across diverse fields, including security-sensitive sectors like healthcare (Miotto *et al.*, 2018) and finance (Ozbayoglu *et al.*, 2020), it is crucial to understand the mechanisms and potential impacts of data poisoning attacks to safeguard the trustworthiness and reliability of these models in practical applications. In Figure 3.7, we present the illustration of data poisoning attacks pipeline.

Problem Setup. The initial aim of data poisoning attacks is to undermine the performance of a target model (Biggio *et al.*, 2012). With developments in modern deep learning, data poisoning attacks have evolved to pursue more sophisticated goals, which aim at manipulating specific target samples while minimizing the impact on other samples, thereby making the attack less detectable (Saha *et al.*, 2020). As a data-centric attack, data poisoning requires that attackers access and

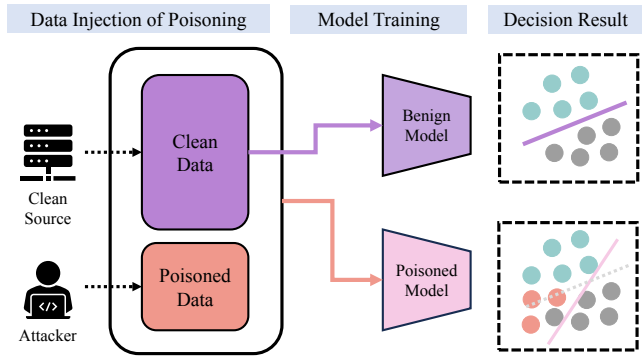

Figure 3.7: Illustration of the pipeline for data poisoning attacks.

modify the training data. This access is relatively easy to obtain in current deep learning landscape, as deep learning models often rely on large volumes of data sourced from publicly available, unverified online web resources (Nelson *et al.*, 2008). Attackers can public malicious data on these web platform, raising the risk that model developers inadvertently include these poisoned data in their training sets. For federated learning, attackers can poison updates by manipulating data on local clients (Tolpegin *et al.*, 2020). In this section, we categorized the methodologies of data poisoning attacks into three types: data poisoning with label modification, data poisoning with data modification, and data poisoning with update modification. Each type corresponds to a distinct attack scenario.

Data Poisoning with Label Modification. Label modification is a type of commonly used data poisoning attack method, in which an attacker manipulates the labels of certain data samples in the training set. Since machine learning models primarily learn their pattern recognition abilities from data-label pairs, label modification can mislead the model into learning incorrect patterns or decision boundaries during training process. In early work, Barreno *et al.* (2006) are the pioneers to first explore the impact of data poisoning attacks on deceiving a classification-based intrusion detection system. Subsequently, Biggio *et al.* (2012) investigate the data poisoning attacks against support vector machines (SVMs). With the advent of the deep learning era,

models have become increasingly vulnerable to data poisoning attacks with label modification. Due to the powerful memorization capabilities of deep neural networks, models can memorize the all patterns of training data including modified labels. For instance, Zhang *et al.* (2021b) conduct extensive experiments to assess the susceptibility of deep neural models to label-flipping attacks, demonstrating that these neural models can overfit on training data and achieve zero training error even when labels are entirely randomized, yet suffer from severely degraded test performance. Considering that different data samples contribute unequally to model learning, Biggio *et al.* (2011) aim to find the most optimal samples for executing the attack. Similarly, Xiao *et al.* (2012) seek to find a combination of label modification to maximize the classification error. Zhao *et al.* (2017) propose an efficient label modification method using an optimization approach to maximize the cosine similarity between the learner's and target model's weight vectors.

Data Poisoning with Data Modification. In comparison to the label space, the data space and distribution are significantly more complex and challenging to estimate, thereby making data modification-based poisoning attacks to be more subtle and harder to detect. In general, there are two types of data modification poisoning methods, i.e., optimization-based data modification and training-based data modification. Specifically, optimization-based methods focus on optimizing a designated objective to generate data samples that maximize the poisoning effect on the target model. For instance, some studies adopt bilevel optimization on linear regression model (Jagielski *et al.*, 2018) and logistic regression model (Demontis *et al.*, 2019) to generate poisoning attacks. Notably, Muñoz-González *et al.* (2017) employ deep neural networks with gradient descent to optimize data samples for targeted class attacks. Huang *et al.* (2020) leverage the meta-learning approaches to approximate bilevel optimization to generate the poisoning samples. Generally, optimization-based methods provide precise control over the generated poisoning attacks, but they are often inefficient and constrained by the limitations of optimization algorithms.

In real-world scenarios, attackers may have no access to the target model, making direct attacks challenging. Training-based data modification introduce an auxiliary model that simulates the behavior of target

model to help indirectly generate poisoning samples. For example, Zhu *et al.* (2019) train a substitute model that approximates the victim model and optimize poison images to form a polytope that encloses the target image within the feature space. This strategy causes the target model to misclassify the target image as the same class as the poison images when it overfits to the poisoned data, achieving high transferability in the attack without requiring access to the victim model. Li *et al.* (2021b) introduce hidden backdoor attacks that bypass human inspection using homograph replacements and model-generated fluent sentences as subtle triggers for natural language models. Additionally, generative models such as auto-encoders (Kingma, 2013) and generative adversarial networks (GANs) (Goodfellow *et al.*, 2014), play a significant role in data poisoning attacks by leveraging their generative abilities to create poisoning samples. For instance, Yang *et al.* (2017) construct a GAN structure in which an autoencoder acts as the generator to produce poisoning samples, while the target model serves as the discriminator to refine them, thereby accelerating the process of generating efficient poisoning attacks. Compared to optimization-based methods, training-based approaches offer greater adaptability and are better suited for scenarios lacking direct access to the target model. However, ensuring the quality and stealth of poisoning data remains a challenge.

Data Poisoning with Update Modification. Although the distributed client-server architecture in federated learning effectively enhances data privacy, it also opens the door to potential data poisoning attacks. Attackers can manipulate compromised clients that train local models on malicious data and send poisoned gradient updates to the server. For instance, Tolpegin *et al.* (2020) demonstrate that even a small proportion of malicious participation using label flipping can substantially reduce the classification performance of the global model. Cao *et al.* (2019a) investigate the effect of the number of attackers and poisoned training samples on attack success rates through distributed label-flipping attacks. Zhang *et al.* (2020a) introduce PoisonGAN, a GAN-based model that crafts the poisoned samples without requiring direct access to the participants' training data. In general, there are several core challenges associated with performing poisoning attacks in

federated learning systems, including the limitation of influence and the necessity to maintain stealth. The aggregation of updates from numerous benign participants dilutes the influence of any single malicious client, which may reduce the attack's effectiveness and make it challenging for the attacker to achieve their intended objectives. In particular, Xue *et al.* (2021b) define a notation called FedInfluence, to quantify the influence of individual clients on federated learning model parameters. When attackers control multiple malicious clients, this scenario is known as sybil attacks (Singh *et al.*, 2006), which amplifies collective influence on the global model.

Generally, the more clients an attacker controls, the higher the success rate of the attack. However, there is a trade-off relationship between influence and stealth: more clients increases influence of updates on the global model but also raises the risk of detection. Therefore, it is crucial to balance both the efficiency and stealth of the attack. For instance, Baruch *et al.* (2019) introduce a concept of perturbation range within which attackers can modify parameters without being detected. Xie *et al.* (2019) propose distributed backdoor attacks, where a global trigger pattern is broken down into distinct local patterns and embedded into the training datasets of multiple adversarial clients. This strategy enhances the stealth of the attack, as the malicious clients resemble benign ones, making detection more difficult. In summary, data poisoning with update modification highlights the inherent vulnerabilities in federated learning, where attackers can exploit the distributed architecture to compromise the global model.

Further Discussion. As data poisoning attacks continue to grow in sophistication and prevalence, it is paramount to develop effective defense mechanisms that can detect and mitigate poisoning attacks without compromising model performance for enhancing the robustness and reliability of deep learning models. Existing countermeasures are largely attack-specific and lack generalization to real-world scenarios with diverse attack methods (Liu *et al.*, 2018). How to design adaptive, scalable, and generalizable defenses that can effectively counter a wide range of poisoning strategies remains an open question. Additionally, in current era of large foundation models, research on poisoning attacks and defenses is still in its early stages (Yao *et al.*, 2024b). Understanding

the intricate attack mechanisms and developing comprehensive defense strategies are critical areas for future work.

3.5 Machine Unlearning

Machine unlearning is an emerging research problem that enable machine learning models to forget specific data points or knowledge while retaining overall performance. This is especially crucial in situations when data privacy and security are critical, such as when sensitive information is no longer required or must be deleted in order to comply with regulations such as GDPR (Shaik *et al.*, 2023) (e.g., the "the right for forgetting"). Since the traditional machine learning models require a post-hoc adjustment regarding the issues with data governance and ethical AI practices, machine unlearning is getting increasing attention and assigned important significance for ensuring trustworthiness. Researchers seek to develop efficient algorithms capable of erasing information from models in an effective and secure manner, guaranteeing the unlearn efficacy while maintaining model integrity. In Figure 3.8, we present the illustration of machine unlearning and two examples.

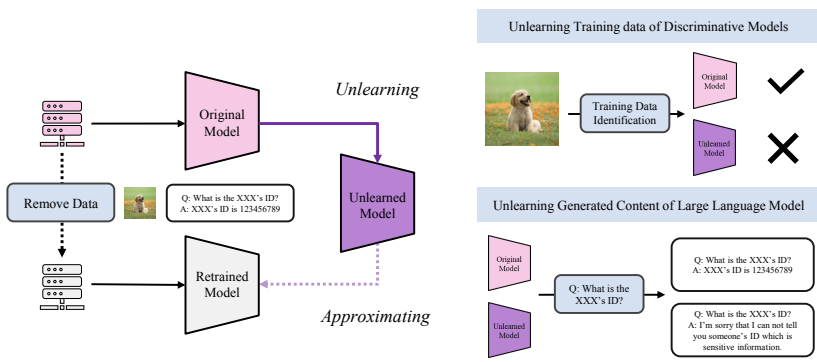

Figure 3.8: Illustration of machine unlearning and two examples regarding unlearning training data in discriminative model or generated content of large language model.

Problem Setup. Given a pre-trained machine learning model, machine unlearning aims to eliminate the influence of training data,

as if the model has never used them during training. Let $\mathcal{X} \subset \mathbb{R}^d$ denote the input space of data and the original training dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ consists of two subsets in machine unlearning, e.g., the forgetting dataset $\mathcal{D}_\mathrm{f}$ and the retaining dataset $\mathcal{D}_\mathrm{r} = \mathcal{D} \backslash \mathcal{D}_\mathrm{f}$. Building upon the model f_{θ^*} with the loss function ℓ, the general target of this research problem is to find an unlearned model θ^*_un, which approximates the behaviors of the model θ^r that retrained on $\mathcal{D}_\mathrm{r}$ from scratch, referring to the exact unlearning. To measure the approximate performance, we can use $\mathcal{R}(\theta^*_\mathrm{un}, \theta^\mathrm{r}, \mathcal{D}_\mathrm{f}, \mathcal{D}_\mathrm{r})$ to indicate a general risk measure for model behaviors (Golatkar *et al.*, 2020), which can be instantiated by comprehensive evaluation metrics regarding its specific application scenarios or original tasks (Jia *et al.*, 2023; Fan *et al.*, 2024). In general, we hope to unlearn the specific knowledge of the pre-trained model but avoid destructive effects on model performance of the retaining part.

The research focusing on machine unlearning can be generally categorized into two aspects. The first is unlearning the training data in discriminative models, which targets the protection of data privacy and gives users the right to forget. The second is unlearning the knowledge learned by generative models like the large language models for forgetting generated content.

Unlearning Training data of Discriminative Models. The conventional machine unlearning exploration mainly focuses on discrimination models (Golatkar *et al.*, 2020; Thudi *et al.*, 2022b; Thudi *et al.*, 2022a; Fan *et al.*, 2024; Chen *et al.*, 2023b; Gandikota *et al.*, 2023; Zhang *et al.*, 2023a), especially for classification tasks. Given the request for training data withdrawal, researchers have developed a series of methods towards approximating the retrained model, which is training excluding the forgetting data. The methodologies can be generally divided into three categories to achieve the unlearning target. The first is the finetuning-based approach (Warnecke *et al.*, 2023), which utilizes the catastrophic forgetting to unlearn the forgetting data via tuning on only the retaining data. The second is adopting the gradient ascent (Thudi *et al.*, 2022a) to conduct active forgetting, which maximizes the loss of forgetting data to reserve the learning process. It can be also achieved by assigning random labels (Jia *et al.*, 2023) or adversarial perturbation (Chen *et al.*, 2023b) to disrupt the learned decision

boundary. Considering the parameter-based similarity definition from a theoretical perspective, there is a third category of the method that achieves effective unlearning by scrubbing the data points using the influence function (Golatkar *et al.*, 2020; Xu *et al.*, 2023).

Unlearning Generated Content of LLMs. Research on unlearning in large language models (LLMs) is still in its nascent stages but has already attracted significant attention, primarily because of its importance in ensuring that public LLMs avoid privacy and copyright issues. Currently, the literature identifies three main directions: gradient ascent (GA), in-context unlearning (ICUL), and task vector (TV).

GA implements unlearning by reversing the model learning process via using gradient ascent (instead of gradient descent) for model updating. Given a set of data $\mathcal{D}_{\mathrm{u}} = \{s_1, \ldots, s_n\}$ targeted to be unlearned, the corresponding unlearning objective can be written as

$$\arg\min_{\boldsymbol{\theta}} \frac{1}{n} \sum_{s \in \mathcal{D}_{\mathrm{u}}} \log p_{\boldsymbol{\theta}}(s), \tag{3.6}$$

where $p_{\boldsymbol{\theta}}(s)$ denotes the probability of generating s for the current model parameterized by $\boldsymbol{\theta}$. It directly reduces the probabilities of generating contents resembling $\mathcal{D}_{\mathrm{u}}$ to approach zero. However, it is worth noting that gradient ascent (GA) carries a remarkable risk of excessive unlearning, where its effectiveness in removing specific information comes at the cost of impairing the general utility, potentially rendering the unlearned models useless. This issue has motivated a series of subsequent studies aimed at refining GA. For instance, Gradient Difference (GD) seeks to regulate the unlearning process used in GA by incorporating a set of retained data, denoted as $\mathcal{D}_{\mathrm{r}}$ and typically sampled from the original training corpus, with a size of m. This method helps balance unlearning with the preservation of overall model performance, following the unlearning objective of

$$\frac{1}{n} \sum_{s \in \mathcal{D}_{\mathrm{u}}} \log p_{\boldsymbol{\theta}}(s) - \frac{1}{m} \sum_{s \in \mathcal{D}_{\mathrm{r}}} \log p_{\boldsymbol{\theta}}(s). \tag{3.7}$$

Also, negative preference optimization (NPO) directly modifies the unlearning objective following

$$\frac{2}{\beta n}\sum_{s\in\mathcal{D}_{\mathrm{u}}}\log\left[1+(\frac{p_{\boldsymbol{\theta}}(s)}{p_{\boldsymbol{\theta}_{\mathrm{ref}}}(s)})^{\beta}\right], \tag{3.8}$$

where β is a hyper-parameter and $\boldsymbol{\theta}_{\mathrm{ref}}$ denotes model parameters before unlearning. It is provable that it is equivalent to GA with instance-wise weighting, following

$$\frac{1}{n}\sum_{s\in\mathcal{D}_{\mathrm{u}}} w_s \log p_{\boldsymbol{\theta}}(s) \text{ with } w_s = \frac{2p_{\boldsymbol{\theta}}(s)^{\beta}}{p_{\boldsymbol{\theta}}(s)^{\beta}+p_{\boldsymbol{\theta}_{\mathrm{ref}}}(s)^{\beta}}, \tag{3.9}$$

where w_s makes NPO converge faster than GA and thereby mitigating excessive unlearning.

Unlike methods based on GA, In-Context Unlearning (ICUL) achieves the removal of specific training data of our interest by manipulating the input context at inference time, without the need for adapting model parameters. In general, it incorporates a series of wrongly labeled corpora into the original input context, along with correctly labeled ones sampled from the training distribution. Since model parameters are intact, ICUL faces less risk of excessive unlearning over GA.

Moreover, task vector (TV) involves further fine-tuning the model on $\mathcal{D}_{\mathrm{u}}$. Therein, we discern between the pre-trained model with parameters $\boldsymbol{\theta}_{\mathrm{o}}$ and that after fine-tuning, namely $\boldsymbol{\theta}_{\mathrm{f}}$. The task vector, defined as $\boldsymbol{\theta}_{\mathrm{f}}-\boldsymbol{\theta}_{\mathrm{o}}$, represents the incremental parameter changes required to strengthen the knowledge related to $\mathcal{D}_{\mathrm{u}}$. Then, we can unlearn $\mathcal{D}_{\mathrm{u}}$ by subtracting the task vector from the original model, resulting in $\boldsymbol{\theta}_{\mathrm{o}}-(\boldsymbol{\theta}_{\mathrm{f}}-\boldsymbol{\theta}_{\mathrm{o}})$ as the resulting unlearned model. TV can also mitigate the risk of excessive unlearning meanwhile can be well adapted to the setup of continous unlearning.

Further Discussion. Although the progress of research in machine unlearning has achieved promising results, further development is still needed toward effective unlearning in practical scenarios and reliable evaluations. From the setting perspective, previous conventional scenarios mainly focus on the all matched scenario in which the forgetting data and the target are matched and all the retaining data is accessible, while sometimes it may not hold since the unlearning target can be different concept from the identified forgetting example (Zhu *et al.*, 2024a; Gandikota *et al.*, 2023); From the methodology perspective, the

intrinsic trade-off between the forgetting and retaining target is still the critical question in conducting complex unlearning on the pre-trained model (Xu *et al.*, 2023; Shaik *et al.*, 2023); From the evaluation perspective, since unlearning has been assigned a more general significance on posthoc adjustment for trustworthiness requirement, the measurement for forgetting or erasing knowledge is differently challenging for different applications or scenarios. As for the foundation models, how to evaluate the knowledge removal for the internal representation remains underexplored in the literature (Wang *et al.*, 2024a).

3.6 Non-transfer Learning

Well-trained deep learning models are the core of Machine-Learning-as-a-Service (MLaaS), which are being provided in a wide range of applications of our daily lives (Oliynyk *et al.*, 2023; Xue *et al.*, 2021a). The training process of deep learning models is always highly cost, requiring massive high-quality annotated data, expensive computation resources, and often takes a long time (e.g., weeks or months). All these lead to the high business value of well-trained deep learning models (Xue *et al.*, 2021a). Thus, how can the model owners properly protect the intellectual property (IP) (Xue *et al.*, 2021a; Zhang *et al.*, 2021c; Guo *et al.*, 2024; Wang *et al.*, 2022c; Wang *et al.*, 2024d) of their models is waiting to be solved.

Recently, non-transfer learning (NTL) (Wang *et al.*, 2022c) was proposed as a novel technology in model IP protection. The IP risks of models trained in the supervised learning (SL) paradigm are mostly owing to its strong generalization ability from the authorized data to any unauthorized data, as shown in Figure 3.9. Such uncontrollable generalization ability can be leveraged by the malicious adverses in an unwanted way (e.g., use the model on unauthorized data or even harmful data), thus leading to IP leakage. Therefore, NTL starts the model IP protection from the perspective of reshaping the generalization abilities of deep learning models.

Problem Setup. According to whether the target domain is known in the training stage, NTL could be subdivided into *target-specified NTL* (restricting the model generalization toward a specific unauthorized domain) and *source-only NTL* (i.e., restricting the generalization toward

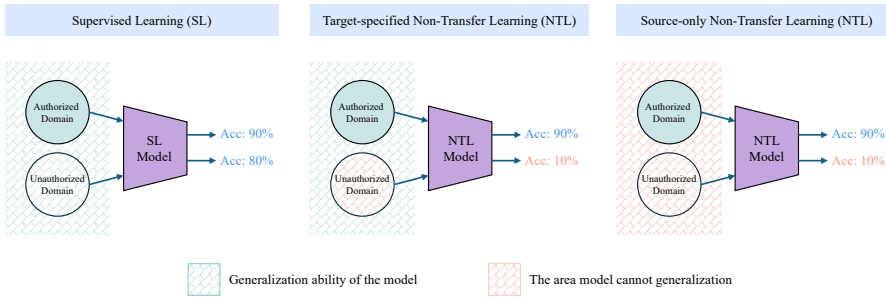

Figure 3.9: Illustration of the objectives of supervised learning (left), target-specified non-transfer learning (middle), and source-only non-transfer learning (right).

all other domains except the authorized domain), as shown in Figure 3.9. We use an image classification task for illustration, as most existing NTL methods aim at classification tasks. Let $\mathcal{X}$ denote the input space and $\mathcal{Y}$ denote the label space. We use $\mathcal{D}_a = \{(x_i, y_i)\}_{i=1}^{N_a}$ and $\mathcal{D}_u = \{(x_i, y_i)\}_{i=1}^{N_u}$ represent the authorized domain and the unauthorized domain, respectively. Note that we mainly consider the NTL problem where the $\mathcal{D}_a$ and $\mathcal{D}_u$ share the same label space. Considering a neural network $f_\theta : \mathcal{X} \to \mathcal{Y}$ with parameters θ, *target-specified NTL* aims to train the f_θ to maximize the risk on the unauthorized domain $\mathcal{D}_u$ and simultaneously minimize the risk on the authorized domain $\mathcal{D}_a$. In addition, for *source-only NTL*, the aim is to train the f_θ to maximize the risk on any possible unauthorized domain $\mathcal{D}_u$ (unknown during the training stage) and minimize the risk on the authorized domain $\mathcal{D}_a$ at the same time.

Target-specified NTL and source-only NTL serve as promising solutions for two types of IP protection techniques: *ownership verification* (Lederer *et al.*, 2023) and *applicability authorization* (Wang *et al.*, 2022c), respectively. After training, an NTL model usually be evaluated by two metrics, including (i) *performance degradation on unauthorized domain*: to which extent the NTL model can degrade the performance on the unauthorized domain; and (ii) *performance maintenance on authorized domain*: whether the NTL model be able to achieve normal performance (i.e., the same level as the SL model) on the authorized domain.

Target-specified NTL and Ownership Verification. To reach the goal of target-specified NTL, a basic framework is to impose a regularization term on the supervised learning (SL) to maximize the unauthorized domain error:

$$\min_{\theta} \Big\{ \mathcal{L}_{\text{ntl}} := \underbrace{\mathbb{E}_{(x,y)\sim\mathcal{D}_a}\left[\mathcal{L}_a(f_\theta(x), y)\right]}_{\mathcal{T}_{\text{auth}}} - \lambda \underbrace{\mathbb{E}_{(x,y)\sim\mathcal{D}_u}\left[\mathcal{L}_u(f_\theta(x), y)\right]}_{\mathcal{T}_{\text{unauth}}} \Big\}, \tag{3.10}$$

where λ is a weight, $\mathcal{L}_a$ and $\mathcal{L}_u$ represent the loss function (e.g., Kullback-Leible divergence) for the authorized and the unauthorized domain, respectively. Intuitively, the general NTL framework can be split into two tasks: (i) an authorized domain learning task $\mathcal{T}_{\text{auth}}$ to maintain the authorized domain performance, and (ii) a non-transferable task $\mathcal{T}_{\text{unauth}}$ to degrade the unauthorized domain performance, which acts as the regularization term.

Existing methods perform the non-transferable regularization $\mathcal{T}_{\text{unauth}}$ either on the feature space or the output space. Wang *et al.* (2022c) maximize both the Kullback-Leibler (KL) divergence between the predictions of unauthorized data and its correct labels, and the Maximum Mean Discrepancy (MMD) between the features from different domains. UNTL (Zeng and Lu, 2022) focuses on the setting of unsupervised unauthorized domain in the text classification task. They train an additional domain classifier to separate features from authorized and unauthorized domains. CUTI-domain (Wang *et al.*, 2023d) combines the image content from the unauthorized domain and the image style from the authorized domain, thus obtaining a middle domain. Then, they maximize the KL divergence between the model predictions on both the middle and unauthorized domains with their labels. H-NTL (Hong *et al.*, 2024d) identifies the problem of fitting spurious-correlation (Zhang *et al.*, 2022e; Lv *et al.*, 2022) in existing NTL methods. They address it by proposing a variational inference framework to disentangle the content and style factors from the authorized and unauthorized data. Then, they let the model features fit content (style) factors with authorized (unauthorized) data as input, thus implementing non-transfer learning.

Target-specified NTL serves as a promising solution of *ownership verification* which is a passive model IP protection strategy for verifying the ownership of a deep learning model (Lederer *et al.*, 2023). Specifically, target-specified NTL implements ownership verification by triggering misclassification on the unauthorized domain (Wang *et al.*, 2022c; Wang *et al.*, 2023d; Hong *et al.*, 2024d; Peng *et al.*, 2024a). Existing methods add some pre-defined shallow triggers (only known by the model owner) on each authorized data and see them as the unauthorized domain. Then, they train the NTL model on these two domains. After training, the NTL model will have a good performance on the authorized domain, but its performance on the unauthorized domain will be poor. In contrast, due to the unconstraint generalization ability, a SL model trained on the authorized domain could have a similar performance on both the authorized data and the unauthorized data with pre-defined triggers. Therefore, by observing the performance difference of a trained model on the data with and without the pre-defined trigger patch, we can verify whether a deep learning model belongs to the model owner, i.e., verification of the ownership.

Source-only NTL and Applicability Authorization. In the source-only NTL setting, the model owners only know the authorized domain, and the purpose is to degrade the model performance on any possible unauthorized domain, i.e., restricting the model generalization abilities inside to the authorized domain. Due to the assumption that only the authorized domain is available during the training stage, existing methods (Wang *et al.*, 2022c; Wang *et al.*, 2023d; Wang *et al.*, 2023a; Hong *et al.*, 2024d; Peng *et al.*, 2024a) take various data augmentation methods on the authorized domain to obtain augmented domains. Then, these augmented domains are seen as the unauthorized domain in (3.10), and thus, existing target-specified NTL methods can be used to solve the source-only NTL problem. Wang *et al.* (2022c) and Wang *et al.* (2023d) leverage generative adversarial network (GAN) (Mirza and Osindero, 2014; Chen *et al.*, 2016) to synthesize fake images following different distribution shifts (distance and directions) from the authorized domain. Hong *et al.* (2024d) use strong image augmentation strategies (Sohn *et al.*, 2020; Cubuk *et al.*, 2020) to obtain the fake unauthorized domain from the authorized domain. DSO (Wang *et al.*, 2023a) performs

a perturbation-based strategy to generate the unauthorized domain distributed in the surroundings of the authorized domain.

Source-only NTL drives the active IP protection strategy: *applicability authorization*. More specifically, applicability authorization intends to lock the model's utility to authorized data, thus preventing their usage on unauthorized data (Wang *et al.*, 2022c). Compared to ownership verification, which can only track the ownership of the model after leaked, applicability authorization can provide active IP protection by totally putting an end to the model utility on unauthorized data.

Further Discussion. Enhancing the robustness of NTL and extending NTL to larger models are two future directions with potential values. *NTL Robustness:* NTL provides promising solutions in the field of model IP protections. However, recent research has identified its vulnerability against well-designed fine-tuning attacks. By using either small parts of authorized data (Hong *et al.*, 2024c) or unauthorized data (Deng *et al.*, 2024), these attacks can re-activate the generalization ability of NTL models to the unauthorized domain. Such findings uncover the risks of NTL-based IP protections in white-box scenarios. Thus, it is significant for future works to advance the robustness of NTL methods against various malicious attacks. *NTL of Large Models:* Existing NTL methods only focus on small-scale models and the applications of IP protection. Compared to small-scale models, the strong abilities of large models (e.g., large language models, LLMs) make it more important to consider their controllable generalization ability (e.g., mitigating LLMs to generate harmful or illegal content). The technology of NTL can be extended to these large models in the future.

3.7 Federated Learning

Federated Learning (FL) is a distributed machine learning paradigm that allows multiple clients (e.g., mobile devices, organizations) to collaboratively train a model without sharing their local data. Rather than centralizing data on a single server, FL leverages the computational power of edge devices, such as smartphones, tablets, and IoT devices, to perform local computations and only share model updates. These updates are aggregated by a central server to improve the global model.

This paradigm enhances data privacy and security since raw data never leaves the local devices, reducing the risk of data breaches and ensuring compliance with data protection regulations. Federated Learning is particularly beneficial in scenarios where data is sensitive, such as healthcare or finance, or when data is distributed across a large number of devices, making traditional centralized training infeasible. By enabling collaborative learning without compromising user privacy, FL opens new avenues for building intelligent systems that respect data ownership and privacy.

Direction 1: Data Heterogeneity. One of the key challenges in Federated Learning is data heterogeneity, which refers to the non-identical and independent distribution (non-IID) of data across different clients. This heterogeneity arises because each client collects data in unique contexts and environments, leading to significant variations in data characteristics such as feature distributions, label distributions, and data quality. This data heterogeneity poses challenges for model training, as traditional machine learning algorithms often assume data to be IID. In Figure 3.10, we present the illustration of the data heterogeneity issue in federated learning.

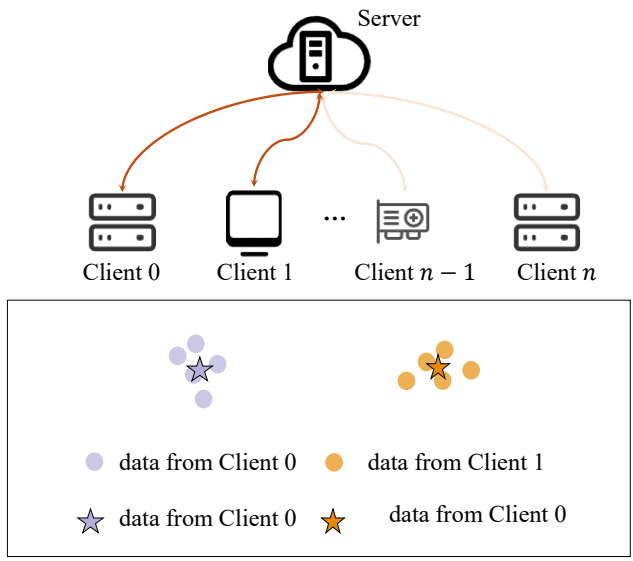

Figure 3.10: Illustration of the data heterogeneity issue in federated learning.

Problem Setup. The Non-IID data problems are in reality. To simulate the data heterogeneity problem, most previous researches adopt two kinds of distribution shifts: Pathological distribution and Dirichlet distribution. For Pathological distribution, most clients will only be assigned with data from a certain number of classes by first sorting the data by its label, then dividing it into shards of same size. As for Dirichlet distribution, it is a probability distribution defined with parameter α, its probability density function (PDF) is given by the following formula:

$$f\left(x_1, \ldots, x_K; \alpha_1, \ldots, \alpha_K\right) = \frac{1}{\mathrm{B}(\boldsymbol{\alpha})} \prod_{i=1}^{K} x_i^{\alpha_i - 1},$$

where $\mathrm{B}(\cdot)$ is the multivariate beta distribution, $\{x_k\}_{k=1}^{k=K}$ is the label indexes. With those two types of distributions to simulate the data heterogeneity in real world, the goal of FL methods are to achieve better performance with minimal communication and computation costs.

To tackle the problem of data heterogeneity, several different kinds of methods have been proposed in previous studies. *Utilizing pubilc datasets.* Creating a small data set that can be shared globally can help mitigate the effect of non-IID by knowledge distillation (Cho *et al.*, 2022) to transfer public data knowledge or correction toward the local model to become less heterogeneous. This data set can originate from a publicly available proxy data source, a separate data set from client data that is not sensitive to privacy, or perhaps a distillation of the raw data following (Wang *et al.*, 2020a).

Regularization on local update. In order to mitigate the bad effect caused by client shift, Li *et al.* (2020d) proposed to add a proximal term in each local objective function so as to make the algorithm be more robust to the heterogeneity across local objectives. The proposed FedProx algorithm empirically improves the performance of federated averaging. Similarly, Li *et al.* (2021a) utilize the similarity between model representations to correct the local training of individual parties by conducting contrastive learning in model-level.

Crafting the objective function. The heterogeneity of client objective functions gives additional importance to the question of how to craft the objective function — it is no-longer clear that treating

all examples equally makes sense. Alternatives include limiting the contributions of the data from any one user.

Personalized FL. If local training is possible, it becomes feasible for each client to have a customized model. This approach can turn the non-IID problem from a bug to a feature, almost literally — since each client has its own model, the client's identity effectively parameterizes the model, rendering some pathological but degenerate non-IID distributions trivial. Smith *et al.* (2018) introduced the MOCHA algorithm for multi-task federated learning allowing every client owns a customized model. In multi-task learning, the result of the training process is one model per task. This directly tackled challenges of communication efficiency, stragglers, and fault tolerance.

Future Discussion. The future of federated learning in the context of data heterogeneity involves several promising directions: *Test-time adaptation FL*: Developing algorithms that dynamically adjust to the heterogeneity of clients at inference time accoring to the test data. For example, FedTHE (Jiang and Lin, 2023) ensemble global model and local model by weighted average, and the adaptive weight is optimized on test data in a unsupervised-style way. *Model Agnostic Techniques*: Techniques such as Model-Agnostic Meta-Learning (MAML) can be used to train models that are agnostic to the specifics of any particular client's data (Park *et al.*, 2021), allowing for quick adaptation and improved performance on heterogeneous data. *Federated Averaging Variants*: Exploring and developing variants of the Federated Averaging (FedAvg) algorithm that are more resilient to data heterogeneity. For example, algorithms that incorporate gradient clipping, adaptive learning rates, or alternative update rules like Sharpness Aware Minimization (SAM) (Qu *et al.*, 2022) can be considered. *Cluster-Based Personalization*: Clients with similar data distributions can be clustered together (Ghosh *et al.*, 2020), and specialized models can be trained for each cluster. This reduces the impact of heterogeneity by focusing on more homogeneous subsets of data.

Direction 2: System Heterogeneity. In the context of federated learning, system heterogeneity refers to the differences in computational resources, communication capabilities, and data distributions across

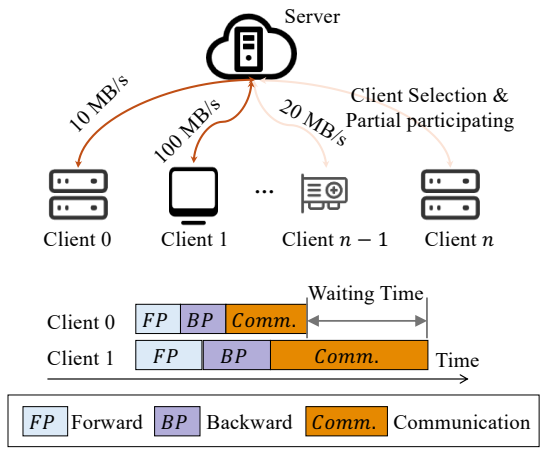

Figure 3.11: Illustration of the system heterogeneity issue in federated learning.

various clients participating in the training process. These differences can significantly impact the efficiency and effectiveness of training a global model. In Figure 3.11, we present the illustration of the system heterogeneity issue in federated learning. Here's a breakdown of how each aspect of system heterogeneity influences training:

Problem Setup. In federated learning, the goal is to train a global model θ by aggregating local models θ_i from $\{1, ..., K\}$. Following attributes are main factors that introduce the system heterogeneity.

- **Computational Heterogeneity:** Clients have varying computational capabilities, which means they differ in processing power, memory, and other hardware resources.
- **Communication Heterogeneity:** Clients experience varying network conditions, resulting in different communication delays.
- **Data Heterogeneity:** Clients possess different data amounts, which leads to variations completion time of local training.

Let T_i^{comp} be the time taken by client i to perform local computations. This time depends on the computational capability of the client. Let T_i^{comm} be the time taken by client i to communicate its local model

updates to the server. This time depends on the network conditions. The total time T_i for client i to complete an iteration is given by:

$$T_i = T_i^{\text{comp}} + T_i^{\text{comm}}. \tag{3.11}$$

Straggler Effect. The straggler problem arises when there is a significant variation in T_i across clients, i.e., $\max(T_1, T_2, \ldots, T_K) - \min(T_1, T_2, \ldots, T_K)$ is large. This leads to delays in aggregating the global model θ (Kairouz *et al.*, 2021; Yang *et al.*, 2022a; Wang and Ji, 2022; Yoon *et al.*, 2021a; Criado *et al.*, 2022).

Asynchronous Federated Learning. To mitigate the straggler effect, asynchronous methods (Wang *et al.*, 2024c; Xu *et al.*, 2021a; Hu *et al.*, 2022a) allow clients to send updates at different times, which the server can incorporate without waiting for all clients. This approach reduces the idle time for faster clients and allows the server to continuously update the global model with the most recent information, thus maintaining momentum in the training process and preventing delays caused by slower clients (Yang *et al.*, 2022a; Wang and Ji, 2022; Tang *et al.*, 2020b; Gao *et al.*, 2022a; Shen *et al.*, 2024).

Client Selection. Selecting a subset of clients based on their availability and resource capabilities can improve efficiency and reduce the impact of heterogeneity (Hu *et al.*, 2022a; Sun *et al.*, 2020). By choosing clients that are more likely to complete their tasks quickly, the server can ensure that updates are received in a timely manner, minimizing the waiting time for stragglers and maintaining a steady flow of information for model updates (Sprague *et al.*, 2019; Lim *et al.*, 2020; Tang *et al.*, 2020b; Gao *et al.*, 2022a; Shen *et al.*, 2024; Nishio and Yonetani, 2019).

Partial Participation. Instead of waiting for all clients to complete their local updates, the server can proceed with aggregating updates from a subset of clients that have completed their tasks, thus reducing waiting time (Anh *et al.*, 2019; Tang *et al.*, 2020b; Hu *et al.*, 2022a; Sun *et al.*, 2020). This approach allows the server to make progress even when some clients are delayed, ensuring that the global model is updated regularly and reducing the impact of any single straggler (Nishio and Yonetani, 2019; Tang *et al.*, 2020b; Gao *et al.*, 2022a; Shen *et al.*, 2024; Li *et al.*, 2020e).

Adaptive Aggregation. Implementing adaptive aggregation techniques that weigh client updates based on their timeliness and reliability can help in reducing the impact of stragglers (Reddi *et al.*, 2020; Tang *et al.*, 2020a). By prioritizing updates from clients that are more consistent and timely, the server can maintain a more accurate and up-to-date global model, while still incorporating valuable information from slower clients when available (Khodak *et al.*, 2019; Gao *et al.*, 2022a; Shen *et al.*, 2024; Sprague *et al.*, 2019; Lim *et al.*, 2020).

Model Compression. Using model compression techniques to reduce the size of updates can decrease communication time, thus helping slower clients to catch up (Frankle and Carbin, 2018; Li *et al.*, 2020a; Khodak *et al.*, 2019; Yu *et al.*, 2020). By minimizing the data that needs to be transmitted, clients with limited bandwidth or slower connections can send their updates more quickly, reducing the overall delay in the training process and mitigating the straggler effect (Amiri *et al.*, 2020; ElKordy and Avestimehr, 2020; Tang *et al.*, 2020b; Tao and Li, 2018; Tang *et al.*, 2024c).

Hierarchical Federated Learning. Organizing clients into a hierarchy where local aggregations are performed before sending updates to the central server can reduce the communication burden and mitigate straggler effects (Kairouz *et al.*, 2021; Tang *et al.*, 2024c). By aggregating updates at intermediate nodes, the amount of data that needs to be sent to the central server is reduced, allowing for faster communication and lessening the impact of slower clients on the overall training process (Liang *et al.*, 2020; Tang *et al.*, 2024a; Tang *et al.*, 2023; Liu *et al.*, 2020b).

Peer-to-Peer Federated Learning. In this kind of approach, clients communicate directly with some clients to share model updates (Tang *et al.*, 2022), rather than relying solely on a central server. This decentralized communication model can enhance the robustness and scalability of federated learning by reducing the server's communication bottleneck and allowing clients to collaboratively improve their models. This method is particularly beneficial in environments where a central server is not feasible or where network connectivity is limited (Tang *et al.*, 2020b; Tang *et al.*, 2024a; Tang *et al.*, 2022; Tang *et al.*, 2020a).

Resource-Aware Scheduling. Implementing scheduling policies that consider the resource constraints of clients to optimize participation and reduce latency. By taking into account the computational and communication capabilities of (Amiri and Gündüz, 2020; Xu *et al.*, 2021b) each client, the server can schedule tasks in a way that maximizes efficiency and minimizes the waiting time for updates, thus reducing the impact of stragglers and ensuring a more balanced and effective training process (Amiri and Gündüz, 2020; Tang *et al.*, 2020b; Tang *et al.*, 2024a; Tang *et al.*, 2022; Ren *et al.*, 2021; Xu *et al.*, 2021b).

Extremely Low Communication Rounds. The one-shot FL (OFL) focuses on minimizing the number of these communication rounds required to achieve a satisfactory model performance (Zhang *et al.*, 2022b; Guha *et al.*, 2019; Li *et al.*, 2020c; Zhou *et al.*, 2020; Dennis *et al.*, 2021). This is crucial because communication can be a significant bottleneck in FL, especially when dealing with a large number of clients or when clients have limited bandwidth (Dai *et al.*, 2024b; Tang *et al.*, 2024b). Some exiting methods propose to use data and ensemble co-boosting to further improve the performance of OFL (Dai *et al.*, 2024b). And recent advanced works improve the OFL performance from the perspective of causal inference (Tang *et al.*, 2024b).

Future Discussion. The future of federated learning in the context of system heterogeneity involves several promising directions: *Adaptive Algorithms*: Developing algorithms that dynamically adjust to the heterogeneity of clients. For example, clients with different computational capabilities and communication delays can have different selection frequency and different local update frequency. *Robust Aggregation*: Designing aggregation methods that are robust to outliers and biased updates due to non-IID data. *Resource-Aware Scheduling*: Implementing scheduling policies that consider the resource constraints of clients to optimize participation and reduce latency. *Modular Design*: Designing FL systems with modular components that can be easily adapted to different scenarios and heterogeneous environments.

In conclusion, addressing system heterogeneity in federated learning is crucial for its scalability and effectiveness. By understanding and tackling the challenges posed by diverse client environments, federated learning can be more widely adopted across various applications.

4

Trustworthy Foundation Models

The creation of foundation models (FM) has ushered in amazing breakthroughs across a variety of fields, which have occurred within the context of the fast-developing landscape of artificial intelligence. On the other hand, in tandem with these accomplishments, considerable questions over the dependability and safety of these models have surfaced. The issues that are presented by jailbreak prompts, watermarking approaches, hallucinations, and causal learning and reasoning are the primary topics that are discussed in this section. This section digs into essential features that support the trustworthiness of foundation models. Each of the sections will investigate the ramifications of these problems, bringing attention to the need to develop solid solutions in order to guarantee that FMs may be implemented into real-world applications in a secure and efficient manner.

The overall framework of this section is illustrated in Figure 4.1, which discusses issues in building the trustworthy foundation models. Section 4.1 ignites the discourse by exposing FMs' susceptibility to jailbreak attacks—malicious prompts that weaponize model flexibility, exploiting semantic ambiguities to bypass ethical safeguards. These attacks reveal a fundamental tension: the very adaptability that empow-

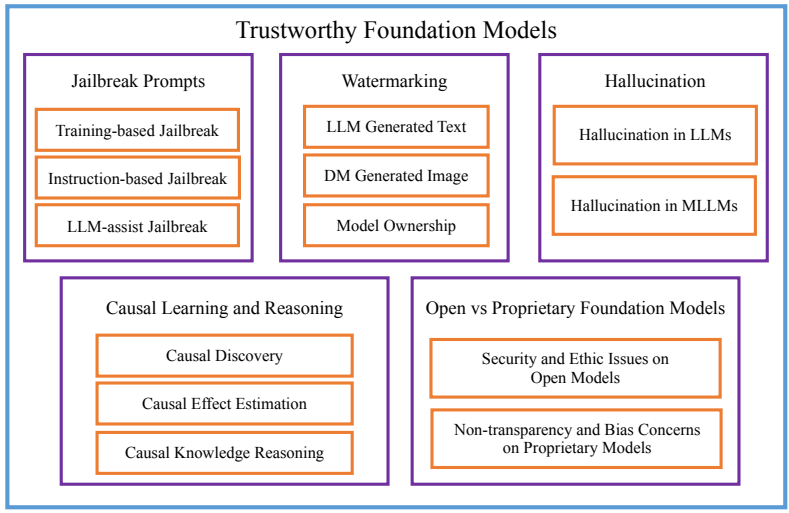

Figure 4.1: The overall framework of trustworthy foundation models.

ers FMs also renders them vulnerable, triggering an urgent quest for defensive countermeasures. Section 4.2 investigate watermarking as a potentially useful method for preventing the harmful use of foundation models, especially with regard to the identification of material created by FMs and the safeguarding of intellectual property. Section 4.3 presents hallucination which exposes a critical gap: safeguarding against external attacks proves insufficient without addressing intrinsic model flaws. The particularly catastrophic in high-stakes domains like healthcare, necessitate a paradigm shift from reactive defenses to architectural redesign. Section 4.4 delivers this evolution through causal learning, reframing reliability as a reasoning challenge. By encoding causal graphs and counterfactual logic into FM architectures, researchers rewire models to mimic human-like deduction, grounding outputs in verifiable causal chains rather than statistical correlations. This transforms FMs from stochastic parrots into accountable reasoners, showing potentials on simultaneously mitigating other unreliable issues. Finally, Section 4.5 discusses the different trustworthy concerns in open and proprietary foundation models regarding their distinct properties.

4.1 Jailbreak Prompts

Notwithstanding the notable accomplishments of Large Language Models (LLMs) in diverse fields, substantial apprehensions persist about their possible misuse despite the introduction of many safety measures. Recent studies (Deng *et al.*, 2023; Zou *et al.*, 2023; Chao *et al.*, 2023; Qi *et al.*, 2023) have demonstrated that LLMs are vulnerable to jailbreak attacks, which can bypass the safety guardrails and trigger the generation of harmful contents, *e.g.*, detailed steps on bomb-making or objectionable information about the minority (Christiano *et al.*, 2017). Jailbreak prompts might cause LLMs to generate improper material, presenting safety issues while using LLMs (Das *et al.*, 2024; Chowdhury *et al.*, 2024; Verma *et al.*, 2024; Cai *et al.*, 2024).

Problem Setting. Shown in Figure 4.2, the primary aim of jailbreak (Deng *et al.*, 2023; Zou *et al.*, 2023; Chao *et al.*, 2023; Feffer *et al.*, 2024) is to develop a prompt that elicits the LLM's production of inappropriate content. In contrast to adversarial jailbreaks necessitating white-box optimization with LLMs for generation (Liu *et al.*, 2024g; Zou *et al.*, 2023), we mainly consider the *training-free* and *black-box* jailbreak, which is more practical. Given a specific prompt P, we expect to induce the response $R_\theta(\mathcal{O})$ from distribution $p_\theta(\cdot|P)$ parameters by LLM θ for objectionable target $\mathcal{O}$ in (4.1).

$$\begin{aligned} &\texttt{Induce } R_\theta(\mathcal{O}) \texttt{ contains objectionable target } \mathcal{O}, \\ &\texttt{where } R_\theta(\mathcal{O}) \sim p_\theta(\cdot|P). \end{aligned} \tag{4.1}$$

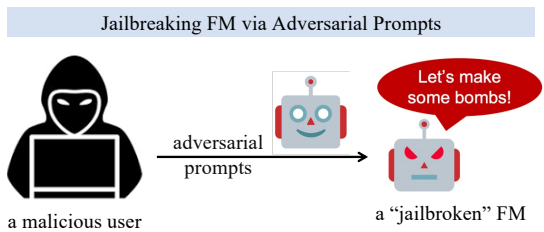

Figure 4.2: A demonstration of jailbreaking an LLM.

Training-based Jailbreak. The pioneering work (Deng *et al.*, 2023) introduces Jailbreaker, an automatic framework designed to explore the

generalization of jailbreaks. This architecture, with a finetuned LLM, illustrates the capability of automatic jailbreak generation for many commercial LLM chatbots. Furthermore, Zou *et al.* (2023) introduce an automated jailbreak technique in a white-box setting explicitly. The follow-up work AutoDAN by Liu *et al.* (2024g) employs a genetic algorithm to automatically generate viable jailbreak prompts derived from existing ones. Qi *et al.* (2024) propose finetuning the LLM with a few adversarial training samples to bypass its safeguards. Likewise, Hong *et al.* (2024b) point out the restricted diversity of test cases in current red teaming reinforcement learning, indicating that finetuned models tend to produce a narrow range of successful test cases once they have been discovered. To tackle this issue, Hong *et al.* (2024b) incorporate novelty rewards and entropy bonuses into the optimization objective, to steer the LLM towards producing a wider variety of harmful responses.

Instruction-based jailbreak. Most safety alignment methodologies concentrate on the natural language perspective, neglecting the influence of non-natural approaches. Yuan *et al.* (2024) introduce CipherChat to encode the attack directives into user-defined ciphers, the regulations of which are explained in the system prompt. Subsequently, the LLM would generate encrypted context containing unsafe information due to the absence of protective alignment on that linguistic domain. Nonetheless, since it specifically delineates a unique language, the model must possess the capability to comprehend and utilize the cipher. Consequently, it may be inappropriate for a relatively small model like Llama2-7B. Carlini *et al.* (2023) focus on the alignment process of LLMs, commonly perceived as a security measure. The research illustrates the capacity to compromise current safeguards by producing adversarial inputs designed for alignment training. Zhao *et al.* (2024b) examine the token distributions of secure LLMs compared to their jailbroken counterparts, emphasizing that the distribution shift takes place in the initial tokens generated rather than in the later ones. Based on this, this study introduces a novel attack vector by restructuring adversarial decoding itself. Anil *et al.* (2024) investigate a category of straightforward long-context attacks on LLMs by prompting with numerous examples of undesirable behavior.

LLM-assisted Jailbreak. Currently, the black-box attack primarily employs supplementary LLM to enhance the initial prompt, which includes adversarial targets like bomb-making requests. PAIR (Chao *et al.*, 2023) produce semantic jailbreaks utilizing solely black-box access to a large language model via several queries. The attacker can systematically interrogate the target LLM to refine and optimize a candidate jailbreak. PromptAttack (Xu *et al.*, 2024c) employ adversarial instruction rewriting techniques to rehearse the initial attack target and to implement heuristic guidance that encourages the LLM to modify the adversarial instructions. Equivalently, Ding *et al.* (2023) apply various techniques for prompt rewriting, *e.g.*, adjusting grammar and altering writing style, rephrasing the attack instructions, and incorporating oversight from LLM to guarantee the consistency of the semantics following modification. Then, the rewritten instructions are strategically integrated into three structured scenarios to prompt the LLM to fill in the blank space. AgentSimith (Gu *et al.*, 2024) identifies a contagious jailbreak, wherein an adversary may compromise a single agent inside a multi-agent system, possibly resulting in the exponential infection of all agents and the emergence of hazardous behaviors. PAP (Zeng *et al.*, 2024a) examines the vulnerability of LLMs to natural and human-like communication in the context of persuasion.

Jailbreak Defense. Robey *et al.* (2023) introduce SmoothLLM by utilizing random permutation techniques multiple times to eradicate the detrimental suffix. Thereafter, the outcome of each permutation would be analyzed independently by the LLM, with the final responses established by majority vote. Dai *et al.* (2024a) present Safe RLHF to disentangle human preferences and mitigate crowd workers' uncertainty regarding the tension. By partitioning the optimization target into reward and cost components, Safe RLHF mitigates the influence of human biases about helpfulness and harmlessness during data annotation, resulting in a more secure aligned LLM. Conversely, Self-reminder (Xie *et al.*, 2023) and In-context Defense (Wei *et al.*, 2023c) are merely founded on manually constructed commands. Zhang *et al.* (2024c) propose PARDEN to instruct the LLM to reiterate its own outputs against jailbreaks. Liu *et al.* (2024h) introduce the Information Bottleneck Protector (IBProtector), which selectively compresses and

modifies prompts. The IBProtector can retain only critical information to ensure that the target LLMs provide the anticipated response. Zeng *et al.* (2024b) propose AutoDefense to process the LLM's generated contents by Multi-Agent system, in order to avert the direct generation of unsafe information by the LLM. Jailbreak prompts are seen as out-of-distribution samples that diverge from the distribution to which the LLM is aligned; hence, Liu *et al.* (2024c) propose Adversarial Tuning to enhance the LLM's defensive mechanisms. This approach involves refining semantic-level adversarial prompts to enhance the model's robustness.

Further Discussion. To further reveal the LLM's vulnerabilities, systematic evaluation of the multi-modal attack scenario is valuable for exploration. This allows us to further explore the psychological properties of LLMs for their safety deployment with inputs and outputs data from different models, like image and speech. In addition, it is difficult to ensure model security across different languages. Addressing these challenges can guarantee trustworthiness when interacting with the LLM in multilingual contexts. To defend against these attacks, strengthening alignment techniques with ethical standards and social norms is critical to guaranteeing the LLMs' safety deployment. The enhanced alignment can help the LLM identify potential harmful instructions that align better with human preference.

4.2 Watermarking

In recent years, foundation models (FMs) have achieved breakthrough advances in text (Brown, 2020) and image generation (Rombach *et al.*, 2022). However, FMs carry risks of malicious exploitation, such as generating fake news and plagiarized content. This poses a significant challenge to their wider adoption: how to detect whether text or images are FM-generated. Watermarking has emerged as a promising technique to address this detection challenge (Liu *et al.*, 2024a; Zhao *et al.*, 2024a). Moreover, since training foundation models requires substantial computational resources, their weights represent valuable intellectual property. Watermarking the models themselves can serve as a means of model provenance to protect the rights of model creators (Zhao *et al.*, 2023c).

In this section, we review watermarking techniques for foundation models, analyze key technical challenges in current approaches, and discuss future research directions. In Figure 4.3, we present an illustration of watermarking on foundation model generated content or FM.

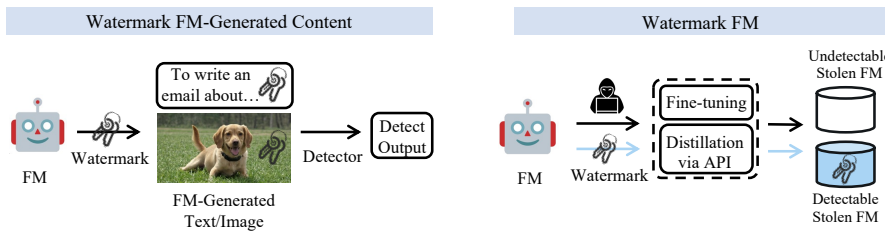

Figure 4.3: Illustration of watermarking on foundation model generated (FM-Generated) content or FM.

Problem Setting. Detecting AI-generated content and safeguarding model intellectual property are the two main functions of watermarking in foundation models. For text, watermarks W are incorporated into text sequences S^W with secure parameters K_D, guaranteeing robust recoverability and semantic integrity even in hostile environments. Binary or multi-bit watermarks are incorporated into generated images I^W while preserving image quality and extractability. Regarding safeguarding intellectual property, identifiers are embedded into model parameters or outputs through model watermarking, making them impervious to changes such as distillation or fine-tuning. Watermarking should balance fidelity, robustness, capacity, security, and efficiency (Uchida *et al.*, 2017). The watermark's robustness ensures that it can resist different removal attacks, while fidelity ensures that the embedding procedure maintains the host object's quality. Security requires strong safeguards to preserve the watermark key, efficiency prioritizes computationally efficient embedding and detection processes, and capacity refers to the system's ability to embed enough information.

Watermarking LLM-Generated Text. Watermarking text generated by LLMs refers to hiding identifiers in the text, safeguarding integrity and ownership while maintaining semantics and readability, which means the identifiers can be detected by a certain method, and the quality of the text is preserved. This technique can be divided into

three categories following their integrated stage of the generation process (Liu *et al.*, 2024a): embedding watermarks during logit generation, during token sampling, and during the training of the language model. Logits-based approaches adjust the probability distribution to prioritize particular token sets based on the model vocabulary. For instance, Kirchenbauer *et al.* (2023) randomly selects a collection of "green" tokens before the process of logit generation and softly promotes their usage frequency during logit sampling. The watermarked text can be effectively detected via z-score analysis, a statistical score representing how far a data point deviates from the mean of a dataset. Unigram (Zhao *et al.*, 2023b) builds upon Kirchenbauer *et al.* (2023) employing a streamlined fixed grouping technique. This method provides shown resilience against text modification and paraphrasing, which can further guarantee better generation quality and precise watermark identification than Kirchenbauer *et al.* (2023). Watermarks during token sampling often mean directing token selection through pseudo-random sequences. Aaronson and Kirchner (2022) incorporate a detectable signal into the produced text to guarantee robustness against token-level modifications.

Likewise, SemStamp (Hou *et al.*, 2023) first defines and partitions the semantic space in the sentence level with locality-sensitive hashing (LSH), and then, generates sentences with multiple times until the sampled sentence is assigned to one partition. Training-based watermarking integrates watermarks directly into the parameters of the model. For example, CoProtector (Sun *et al.*, 2022b) utilizes data poisoning methods, meaning the training dataset has a part of poisoned data. After the training process, a watermark can then be achieved at the parameter level. CoProtector is designed to safeguard open-source code from unwanted training exploitation. Hufu (Xu *et al.*, 2024a) employs input format triggers that leverage the transformer's permutation property to include watermarks in the resulting content. Furthermore, Gu *et al.* (2023) examines the learnability of watermarks in language models, assessing the capability of LLMs to be trained for the direct generation of watermarked text, which carries substantial consequences for the practical use of watermarks. These methodologies are especially appropriate for open-source models, as they inhibit the straightforward elimination of watermarks during post-generation procedures.

Watermarking DM-Generated Image. Similar to watermarking LLM-generated text, watermarking DM-generated images refers to embedding watermarks into images produced by Diffusion Models (DMs). To enable invisible watermarks in generated images for future detection and responsible deployment, Fernandez *et al.* (2023) fine-tunes the latent decoder of the latent diffusion models (LDMs) conditioned on given watermarks. Then, they train a watermark extractor to retrieve watermarks from generated images. More importantly, this fine-tuning method doesn't affect the diffusion process and requires no architectural changes to embed watermarks in all the images LDMs generate. Different from Fernandez *et al.* (2023), which needs additional fine-tuning, Tree-Ring Watermarking (Wen *et al.*, 2023) embed watermarks into the initial noise vector used during the sampling process, leveraging the Fourier space to ensure robustness against various transformations, including crops, flips, and rotations. The embedded watermarks can be retrieved from the initial noise vector via the reverse diffusion process.

Zhao *et al.* (2023e) proposes different strategies for two types of DM, i.e., unconditional/class-conditional DMs and text-to-image DMs. For unconditional/quasi-conditional DM, Zhao *et al.* (2023e) embeds watermarks when training the model from scratch. This is because the model size of this type of DM is generally small and lacks external control, making it difficult to add watermarks through other methods. Regarding text-to-image DMs, they embed watermarks through trigger prompts and associated watermark images. Specifically, Zhao *et al.* (2023e) fine-tunes the pretrained text-to-image DM to establish a relationship between a rare identifier (e.g., "[V]") and a predefined watermark image (e.g., a QR code). Besides focusing on detecting synthetic content, WaDiff (Min *et al.*, 2024) integrates user-specific watermarks into the diffusion model's generation process. WaDiff allows each generated image to carry unique, imperceptible information that can be extracted to identify the user responsible for its creation.

Watermarking Foundation Model. Foundation models need substantial computer resources for training and constitute valuable intellectual property (IP), necessitating their protection against theft. Watermarking foundational models provide strong IP protection against threats like model extraction and unauthorized utilization, primarily

categorized into two types: watermarking for distillation and fine-tuning. Watermarking against distillation involves integrating distinctive, identifiable patterns into a model's outputs to guarantee that distilled models preserve the watermark for identification purposes. GINSEW (Zhao *et al.*, 2023c) integrates covert signals into decoding probabilities throughout the text production process. This method guarantees that the watermark endures despite adversarial assaults, such as synonym randomization, and stays imperceptible in the produced text. During deployment, suspicious models can be recognized by determining whether their outputs contain secret messages, suggesting they were derived from proprietary models. GINSEW demonstrates exceptional robustness to perturbations such as synonym randomization, rendering it notably resilient. Fine-tuning may also be exploited to conceal the origins of illegitimate models.

In response, watermarking techniques to mitigate fine-tuning have been established. Instructional Fingerprinting (Xu *et al.*, 2024b) integrates a confidential private key as an instruction backdoor, eliciting the model to produce designated output upon invocation of the key. This technique guarantees the preservation of the fingerprint through meticulous adjustments while staying undetectable during normal usage. Furthermore, HuRef (Zeng *et al.*, 2023) found that LLM parameters demonstrate directional stability after convergence following pre-training, with parameters preserving their orientation during subsequent fine-tuning phases. Consequent to this discovery, they created a systematic framework for model identification, delineating three distinctive invariants. Empirical evaluations on multiple LLM validate the robustness of this method.

Further Discussion. Despite significant advances in watermarking techniques for both foundation model-generated content and the models themselves, several major challenges remain. Generally, it is difficult to achieve a watermarking strategy that optimally balances fidelity, robustness, capacity, security, and efficiency simultaneously. For watermarking LLM-generated text, a critical challenge lies in balancing watermark strength with text quality. The embedded watermarks must endure numerous potential assaults, including paraphrase and synonym substitution. Moreover, the absence of standardized watermarking and

detection protocols in the community results in watermark detection being effective solely for particular LLMs. In the field of watermarking DM-generated images, resilience to adversarial attacks is especially vital. The watermarking technique must include prevalent image alterations, like compression and cropping. Due to the unpredictable nature of potential attacks, creating a watermarking method that can withstand all conceivable attack vectors presents a considerable problem, necessitating additional research. The mathematical foundations underlying watermarking strategies for foundation models also require further improvement. Furthermore, due to the extensive scale of foundation model parameters, embedding watermarks via fine-tuning methods is excessively costly. This necessitates effective watermarking solutions capable of withstanding unforeseen attacks while maintaining computing feasibility.

4.3 Hallucination

The emergence of LLMs has revolutionized the field of natural language processing (NLP), enabling significant advancements in tasks such as text generation, comprehension, and reasoning. These models, which are trained on extensive datasets, have demonstrated an impressive ability to produce coherent and contextually relevant text. However, one of the critical challenges associated with LLMs is the phenomenon of hallucination—instances where the model generates information that is plausible-sounding but factually incorrect. Hallucinations can manifest in various forms, including the generation of fictitious facts, misattributed quotes, or entirely fabricated narratives. This issue raises concerns about the reliability of LLMs, particularly in applications where accuracy is essential, such as legal documentation, medical advice, and educational content.

As the field has progressed, the development of multimodal large language models (MLLMs) has introduced an additional layer of complexity. MLLMs integrate both textual and visual data, allowing them to perform tasks that require an understanding of the interplay between language and images. For example, MLLMs can generate descriptive captions for images or answer questions based on visual content. How-

ever, this integration also amplifies the potential for hallucinations, as the model must ensure that the generated text accurately reflects the visual information it is associated with. MLLMs can produce hallucinations that stem from cross-modal inconsistencies, where the text does not align with the visual content, leading to misleading or incorrect outputs.

The necessity of addressing hallucinations in both LLMs and MLLMs cannot be overstated. As these models are increasingly deployed in real-world applications, the implications of generating inaccurate or misleading information can be profound. In critical domains such as healthcare, finance, and security, hallucinations can lead to significant consequences, including misdiagnoses, financial losses, or compromised safety. Furthermore, the trustworthiness of AI systems hinges on their ability to provide accurate and reliable information. Therefore, understanding the mechanisms behind hallucinations and developing strategies to mitigate them is essential for ensuring the responsible deployment of LLMs and MLLMs. Ongoing research in this area is crucial to enhance the robustness of these models and to foster greater confidence in their use across various applications.

Problem Setting. Despite the impressive capabilities of LLMs, they continue to encounter various issues in practical applications, with hallucination being one of the most critical. The term "hallucination" has been widely used in the NLP community prior to the emergence of LLMs, typically referring to the generation of nonsensical content or outputs that lack fidelity to the provided source material (Ji *et al.*, 2023). However, we argue that the definition of hallucination has expanded significantly due to the versatility of LLMs. In this context, hallucinations in LLMs can be categorized into three main types (see Figure 4.4):

- **Input-conflicting hallucination**: Occurs when LLMs generate content that deviates from the source input provided by users.
- **Context-conflicting hallucination**: Arises when LLMs produce content that contradicts previously generated information, leading to inconsistencies within the same output.

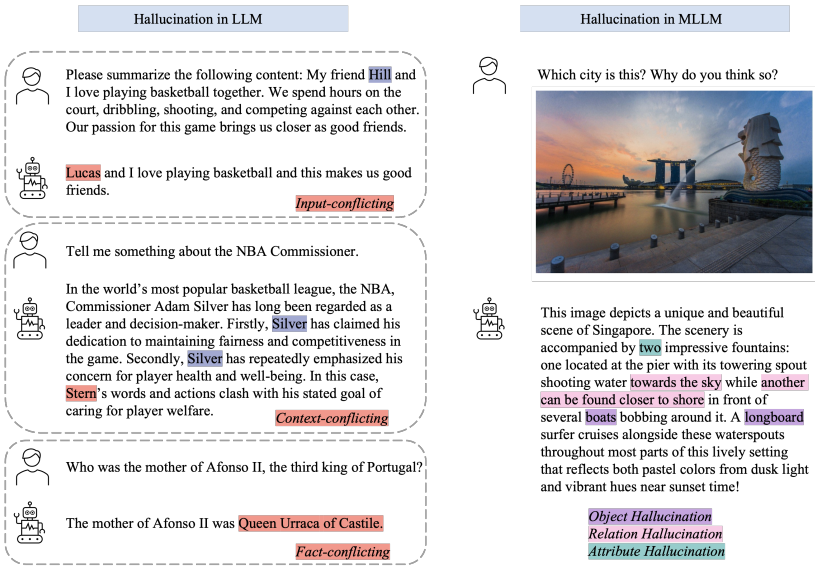

Figure 4.4: Examples of hallucinations in LLM and MLLM.

- **Fact-conflicting hallucination**: Involves the generation of content that is not aligned with established world knowledge.

In the realm of MLLMs, hallucinations present unique challenges that stem from discrepancies between the generated text and the associated visual content. This indicates that existing research on hallucinations in LLMs may not fully address the distinct issues posed by multimodal models, highlighting the need for dedicated studies focused on hallucinations specific to MLLMs.

Hallucinations in MLLMs often arise from cross-modal inconsistencies, where the generated text fails to accurately represent the visual content it is intended to describe (Wang *et al.*, 2023c). Much of the current research has concentrated on object hallucinations, which are critical in both computer vision and multimodal applications. Two prevalent types of failures include the omission of objects that should be visible in the scene and the erroneous inclusion of objects that are not present in the image or misrepresentation of their characteristics.

Object hallucinations in MLLMs can be categorized into three types:

- **Object Category**: MLLMs may incorrectly identify nonexistent object categories or misclassify objects within an image.
- **Object Attribute**: While MLLMs may accurately identify object categories, they can misdescribe the attributes of these objects, including color, shape, or action.
- **Object Relation**: MLLMs might correctly identify objects and their attributes but fail to accurately describe the relationships among them, such as human-object interactions or spatial arrangements.

In summary, understanding and addressing hallucinations in both LLMs and MLLMs is crucial for improving the reliability and accuracy of these models in practical applications.

Hallucination in LLMs. Recent research developments have made major contributions to understanding and minimizing hallucinations in LLMs. These papers examine the life cycle of LLMs. The LLM life cycle consists of four main phases. The first step is pre-training, in which the model collects information from a large dataset and encodes it in its parameters. Subsequently, supervised fine-tuning (SFT) allows the model to develop effective user interface abilities. The model is then refined using reinforcement learning from human feedback (RLHF), which aligns its responses with human preferences. The inference step produces the final result.

Zhang *et al.* (2023g) identify four main origins of hallucinations based on the life cycle. Imperfections in pre-training data might cause gaps in important knowledge or the acquisition of inaccurate information in LLMs, leading to hallucinations (Dziri *et al.*, 2022). Second, throughout the SFT process, there is a possibility for hallucinations to be accidentally induced when LLMs are expected to reply to queries that exceed their knowledge boundaries (Achiam *et al.*, 2023). A faulty alignment method might mislead LLMs and cause hallucinations. The generating approach used during the inference step has possible dangers. Hallucinations may be triggered by accumulating initial mistakes (Zhang *et al.*, 2023c) and random sampling (Lee *et al.*, 2022).

Determining the underlying causes of hallucinations facilitates the development of effective treatment strategies. Various approaches have recently been devised at many levels to mitigate the issue of hallucinations. A systematic approach to reducing inaccuracies and biases in the corpus starts with the careful identification and selection of high-quality pre-training data from credible sources. The emergence of GPT-2 (Radford *et al.*, 2019) has underscored the need to collect internet data with meticulous curation by human experts. The ongoing expansion of pre-training corpora makes the automation of selection and filtering processes more feasible. In Penedo *et al.* (2023), a pre-training dataset is improved by filtering, rule-based optimization, and deduplication of existing web datasets, resulting in higher model performance. Moreover, Li *et al.* (2023b) demonstrate that filtering synthetic data may provide smaller models with comparable capabilities to larger models, underscoring the significance and effectiveness of optimizing pre-training datasets.

During the SFT phase, LLMs often respond to all queries without determining if they surpass their knowledge bounds, which might lead to hallucinations. Integrating honest samples into the SFT dataset is a possible solution (Sun *et al.*, 2024). This technique has disadvantages, such as poor out-of-distribution generalization and a mismatch between human knowledge and huge language models. Schulman (2023) suggests a solution for this problem using RLHF. The idea focuses on creating a different reward function. Receiving these benefits motivates LLMs to critically analyze assumptions, voice doubt, and admit their shortcomings. This technique allows LLMs to explore knowledge boundaries independently, improving out-of-distribution generalization capabilities and minimizing the need for substantial human annotation of LLM knowledge limitations.

Mitigating hallucinations during the inference stage is less expensive and easier to regulate than the previously stated training-phase techniques, leading to more concentrated research efforts. Decoding techniques are critical in the inference process since they have a major impact on the quality of the created material. Decoding techniques are the procedures used to choose tokens from the probability distribution generated by the model. Optimizing decoding algorithms improves the

model's output's accuracy and consistency in a plug-and-play way. For example, Lee *et al.* (2022) shows that nucleus sampling is less successful than greedy decoding in terms of factuality due to the randomness introduced by nucleus sampling to increase variety. They developed a decoding algorithm dubbed factual-nucleus sampling. It aims to balance variety and factuality better by combining the benefits of top-p and greedy decoding techniques.

Furthermore, Burns *et al.* (2022) demonstrate that the activation space of LLMs includes interpretable structures connected to factuality. Based on this idea, Li *et al.* (2024b) propose the Inference-Time Intervention (ITI) approach. This method initially determines a small number of attention heads with good linear probing accuracy for factuality. During the inference phase, ITI modifies activations based on factual directions. This intervention is carried out repeatedly until the whole solution is obtained. This technique helps LLMs to provide more factually correct replies.

Similarly, Chuang *et al.* (2023) examine how to increase factuality in LLM decoding by evaluating the encoding of factual knowledge inside transformer LLMs. Earlier layers collect lower-level features, whereas later layers include more semantic information. As a result, DoLa is designed as an approach that uses contrastive decoding to lessen hallucinations. DoLa enhances the factual accuracy of LLMs while reducing hallucinations by selecting and comparing logits from different levels and highlighting information from higher layers.

RAG (retrieval-augmented generation) is one strategy for decreasing hallucinations during inference. RAG improves LLMs by adding current information or relevant evidence from external knowledge bases or instruments during inference, reducing hallucinations caused by LLMs' intrinsic limitations (Ren *et al.*, 2023). REPLUG (Shi *et al.*, 2023) uses a customizable retrieval model to augment current LLMs during the creation phase, enhancing accuracy and reliability. In the post-processing phase, one typical method is to use an auxiliary fixer to correct hallucinations. One example is RARR (Gao *et al.*, 2022b), which tells an LLM to produce queries from many viewpoints on the material that requires correction. The system uses search engines to gather relevant information, which the LLM-based fixer uses to make the

appropriate changes. Similarly, the Verify-then-Edit algorithm (Zhao *et al.*, 2023a) tries to increase the factual accuracy of forecasts by improving reasoning chains utilizing external information from sites such as Wikipedia. Although retrieval-augmented LLMs aim to reduce hallucinations in LLMs, they may nevertheless produce them (Barnett *et al.*, 2024).

Hallucination in MLLMs. Recent developments in MLLMs have led to the introduction of various techniques aimed at minimizing hallucinations through fine-tuning strategies. One notable method is LRV-Instruction (Liu *et al.*, 2024d), which addresses the shortcomings of existing instruction-tuning datasets that primarily consist of positive samples. This bias often results in models defaulting to affirmative responses, leading to hallucinations. To counter this issue, LRV-Instruction incorporates both positive and negative instructions, categorized into three types: manipulating nonexistent objects, altering attributes of existing objects, and modifying the knowledge conveyed in instructions. This comprehensive approach enhances the robustness of visual instruction tuning by ensuring that models learn to navigate a broader range of scenarios.

Similarly, HalluciDoctor (Yu *et al.*, 2024) refines the instruction-tuning dataset by establishing a hallucination detection pipeline that employs consistency checks across multiple MLLMs to identify and eliminate hallucinated content. Additionally, it utilizes a counterfactual visual instruction generation strategy to balance the dataset, effectively reducing the occurrence of hallucinations. Another significant contribution is EOS Decision (Yue *et al.*, 2024), which focuses on optimizing the end-of-sequence decision-making process. This method posits that hallucinations often arise when models generate details beyond their perceptual limits, particularly concerning objects mentioned later in generated descriptions. By enhancing the EOS decision-making process and implementing a data filtering strategy to remove detrimental training data, this approach aims to ensure that models can conclude sequences effectively without generating misleading information.

On the other hand, several methods aim to mitigate hallucinations without requiring additional training, focusing instead on optimizing the decoding process. HALC (Chen *et al.*, 2024d) emphasizes the critical

importance of selecting optimal visual contexts during the decoding of specific tokens. The research demonstrates that utilizing the best visual contexts can eliminate over 84.5% of hallucinations, underscoring the significance of grounding generated text in accurate visual information. Visual Contrastive Decoding (VCD) (Leng *et al.*, 2024) further seeks to reduce statistical biases and language priors during the decoding phase by contrasting output distributions from original and distorted visual inputs. This method aims to recalibrate the decoding probability distribution, ensuring that the model's responses are more aligned with the actual visual context. Additionally, OPEAR (Huang *et al.*, 2024b) introduces a novel decoding method that incorporates an Over-trust Penalty and a Retrospection-Allocation strategy. This approach addresses the tendency of MLLMs to over-rely on a limited number of summary tokens, which can lead to hallucinations by neglecting relevant visual tokens. By implementing a penalty term during beam-search decoding and employing a rollback strategy to reassess the relevance of previously generated outputs, OPEAR effectively adjusts token selection to minimize hallucinations. Collectively, these methods showcase the potential for optimizing MLLMs through enhanced training and decoding techniques, ultimately improving their reliability and performance in real-world applications.

Further Discussion. Even though a lot of work has been done to understand and treat the hallucinatory issues in LLMs and MLLMs, there are still a lot of issues that need to be addressed. Because they struggle to identify the boundaries of their expertise, LLMs often make untrue assertions with assurance without realizing it. Numerous research have examined this subject, identifying confusing responses using a range of methodologies. For instance, assessing the likelihood of correct responses in multiple-choice scenarios (Kadavath *et al.*, 2022) or examining how LLMs respond differently to the same question in various formulations (Zhao *et al.*, 2023d). However, it has been shown that many of the present detection methods have generalizability problems and are still not sufficiently reliable (Levinstein and Herrmann, 2024). Therefore, whether we can effectively explore the internal knowledge boundaries of LLMs remains open and requires further in-depth research.

Additionally, much of the present research on LLM hallucinations is in English, despite our desire for multilingual LLMs. Multilingual LLMs often face various challenges, particularly when translating in low-resource languages, according to a study on hallucination issues (Guerreiro *et al.*, 2023). As a result, research on lowering the hallucinatory phenomena in multilingual LLMs with respect to low-resource languages is particularly crucial.

Compared to unimodal scenarios, multimodal scenarios in MLLMs provide more complex hallucination problems. Evaluations and studies have shown that current MLLMs are prone to generating responses that are inconsistent with the images presented, including objects with incorrect sorts and characteristics, entities that do not exist, and incorrect semantic correlations (Liu *et al.*, 2024e). This problem arises because current MLLMs inherit the hallucination issues of LLMs, but due to insufficient multimodal alignment, they display more severe hallucinations while addressing multimodal tasks. Furthermore, when several images are shown, MLLMs may misinterpret or omit parts of the visual content and fail to identify the temporal or logical connections between them. Given that existing studies cannot accurately detect and assess hallucinations in multimodal contexts, further research is necessary to identify, understand, and reduce multimodal hallucinations. Additionally, some research has expanded LLMs beyond images to other modalities, including audio (Wu *et al.*, 2023) and video (Maaz *et al.*, 2023). Examining hallucination issues in these new contexts is equally important and valuable.

4.4 Causal Learning and Reasoning

What are the fundamental properties of reasoning ability that are expected to occur in a trustworthy foundation model? This is still an open question. However, one partial answer might be the deep association with the mechanism behind the real world. Although it is debatable whether current foundation models are capable of directly understanding and utilizing causality during their reasoning process, a lot of effort has been made in recent years.

Problem Definition. The meaning of causality has different flavors in different literature. As illustrated in Figure 4.5, the intersection of the Foundation Models with three types of tasks are considered here. The common ground is to understand some realistic mechanism behind real world. *Causal Discovery* is about the structure identification among a set of variables. Given a set of random variables $\mathbf{X} = \{X_1, X_2, \cdots, X_d\}$, this task expects to obtain a directed causal graph $\mathcal{G}$ where each node represents one variables and each edge $X_i \to X_j$ indicate the data-generating process of X_j depends on the value of X_i. *Causal Effect Estimation* is about the estimation the effect of one treatment variable on one target variable people are interested. For a binary treatment $T \in \{0, 1\}$, with notation $Y(0)$ and $Y(1)$ for the potential outcomes, this task expect to estimate the effect of T on Y as $\tau = \mathbb{E}\{Y(1) - Y(0)\}$. The problem is challenging because only one of $Y(0)$ and $Y(1)$ can be observed for an individual. *Causal Knowledge Reasoning* is to properly retrieve and utilize the causal knowledge learned in Foundation Models to answer queries from users.

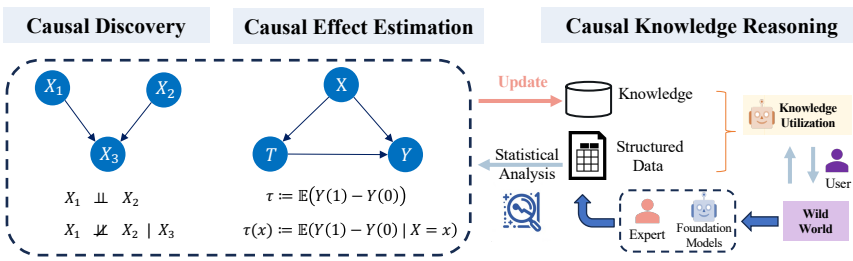

Figure 4.5: Causal learning and reasoning with foundation models.

LLMs for Causal Discovery. Causal discovery is finding causal relations among a set of variables. Classic methods take tabular datasets as input and then output a directed acrylic graph to represent the causal structure among them. Typical methods rely on statistical analysis on their samples (Spirtes *et al.*, 2001; Peters *et al.*, 2017b).

One featured advantage of LLMs is the broad knowledge learned from the pre-training phase. LLMs can utilize its knowledge and serve as a prior to reducing the search space. In addition to tabular data, most of

methods in this line also requires meta data like textual description T_i for each variable X_i. Ban *et al.* (2023b) design a pipeline to generate a set of edges with the understanding from LLMs and then construct a set of ancestral constraints to restrict causal orders. They propose the hard and soft constraint approaches to integrate constraints. An iterative approach (Ban *et al.*, 2023a) is also proposed to update constraints with causal graphs from previous rounds. Vashishtha *et al.* (2023) find that empirically LLMs perform better in deciding causal orders than causal structures. Motivated by this, they combine the LLM-predicted causal orders as a prior with score-based and constraint-based classic causal discovery methods. Li *et al.* (2024c) propose LLM-guided Meta Initialization to integrate knowledge and textual information into score-based causal discovery pipelines for time series. It specifies the initial causal graph before optimization and keeps the identifiability of the original methods. Abdulaal *et al.* (2023) combine the LLMs' reasoning over metadata and the likelihood feedback from data-driven modeling, where an LLM updates a hypothesis over causal structures iteratively with its likelihood verified from numerical data. Le *et al.* (2024) combine metadata and numerical methods in a multi-agent approach, where knowledge retrieval, reasoning, and tools use are conducted by different models.

Instead of searching for causal structures, some works consider different but important aspects relevant to causal discovery. Jiralerspong *et al.* (2024) focus on the efficiency during the pair-wise query for causal knowledge from LLMs. They propose a breadth-first search framework that scales linearly with the number of variables. Jiang *et al.* (2024a) focus on the scenario where users have no sufficient knowledge about causal discovery methods. In this setting, LLMs must understand the users' queries, use causal tools properly, and interpret results. They enhance LLMs' ability by post-training over carefully designed datasets. Liu *et al.* (2024b) focus on the representation of unstructured data like text or images. To extend the scope of traditional causal methods with LLMs, they design the causal representation assistant framework to provably identify high-level factors as a Markov Blanket for a target variable.

LLMs for Causal Effect Estimation. Estimating the causal effect of a treatment T on an interested variable Y is an important task in statistics and economics. Classical methods include randomized

controlled experiments and estimation from observation data with instrumental variables (Peters *et al.*, 2017b; Imbens and Rubin, 2015) .

Lin *et al.* (2023b) consider the estimation of the causal effect of linguistic attributes on human responses. For target domains where data is ready for valid estimation, they established an estimator to transport causal effects between domains with statistical guarantees on uncertainty. Vashishtha *et al.* (2023) propose to use causal orders provided by LLMs to help treatment effect estimation. They show that causal order contains less information than the full causal structure but is easier to acquire and is already sufficient for treatment effect estimation. They show that causal order is good enough to find a valid backdoor set. Then, they propose a method to get causal orders from LLMs. Dhawan *et al.* (2024) aim to construct intensive estimation for causal effect from observational text data without costly data collection. They design a NATURAL family of estimators with built-in LLM conditional distributions. NATURAL estimators have theoretical guarantees of consistency if the true distribution can be accessed through the form of language reports.

Causal Knowledge Reasoning. One important ability of Foundation Models is to properly retrieve and process causal knowledge learned from data during reasoning processes.

It is important to know whether such ability has been learned by LLMs. CORR2CAUSE (Jin *et al.*, 2024) finds LLMs are not learned to infer causal structures based on independence conditions. Kıcıman *et al.* (2023) provided preliminary but a wide range of evaluations over multiple real-world benchmarks under different settings, where LLMs using textual descriptions show competitive performance with numerical methods. A more recent benchmark is CausalBench (Zhou *et al.*, 2024b), a comprehensive benchmark consists of 15 commonly used real-world datasets with a standardized evaluation process. To provide a more thorough evaluation of LLMs' causal reasoning ability, Jin *et al.* (2023) proposed a large benchmark with questions covering the three rungs of the ladder of causation. Zečević *et al.* (2023) proposed to distinguish the ability to *talk* causality from *learn* causality. They empirically justified the hypothesis that current LLMs cannot be solely relied on to provide actual inductive learning, such as causal discovery or causal inference.

Similar conclusions are also found by Cai *et al.* (2023) with a novel experimental setup, where a causal attribution model is used to generate counterfactual data. Chi *et al.* (2024b) distinguished two levels of causal reasoning, where current LLMs can do the shallow one while lacking the ability of the level-2 causal reasoning.

There are methods proposed to enhance the ability further. Causal-CoT (Jin *et al.*, 2023) guides models to follow steps summarized from causal inference tasks. CARE-CA (Ashwani *et al.*, 2024) enhances the causal understanding with knowledge graphs. G^2-Reasoner (Chi *et al.*, 2024b) is a prompting strategy combined with general knowledge from an RAG system. The prompt is designed to steer LLMs to conduct causal reasoning in a goal-driven manner.

Further Discussion. As intensive effort has been put into exploring causal knowledge utilization and its combination with numerical data, new trends are also emerging.

Toward learning and reasoning over wild observations. Rigorous statistical methods are designed for structured data created by human experts. And it is challenging to apply them to text, images, and videos in an unsupervised manner (Schölkopf *et al.*, 2021; Locatello *et al.*, 2019). Liu *et al.* (2024b) integrate the multi-modal knowledge of Foundation models to analyze unstructured data and provably identify representations. Dhawan *et al.* (2024) build a family of causal effect estimators over unstructured textual data with the LLMs' conditional distributions and provide a theoretical guarantee of consistency. Such works bring new opportunities for reliable analysis of real-world data.

Toward knowledge query for real-world applications. The rich knowledge of Foundation models can be acquired by existing methods. Recent works begin to investigate how to utilize those knowledge to downstream tasks reliably and efficiently. Feder *et al.* (2024) consider text classification tasks and conduct interventions on textual data with the LLMs' knowledge of causal structure to eliminate the reliance on spurious features. Jiralerspong *et al.* (2024) reduce the complexity of knowledge query with designed method that scales linearly. Jiang *et al.* (2024a) alleviate the methods' requirement on domain knowledge from inputs and construct a user-friendly pipeline. These work are helpful to make learned causal knowledge to have realistic influence.

4.5 Open vs. Proprietary Foundation Model

Trustworthy issues vary significantly between open and proprietary foundation models. Proprietary models like GPT-4 (Achiam *et al.*, 2023) and Gemini (Team *et al.*, 2024) often rely on the reputation of the developing organization and robust deployment infrastructure to build trust, but their black-box nature can raise concerns about transparency, biases, and accountability. In contrast, open models like LLaMA (Touvron *et al.*, 2023) and Falcon (Almazrouei *et al.*, 2023) foster trust through transparency, allowing users to inspect and modify their architecture and training data, though they may face challenges like misuse or resource constraints. While proprietary models emphasize performance and support, open models prioritize community collaboration and customizability, highlighting the trade-offs between control and transparency in building trust for different use cases.

Problem Background. Regarding their different properties (Chang *et al.*, 2024), foundation models can be classified into two main categories: open and proprietary. Open foundation models are accessible to the public, enabling users to examine, change, and refine them as required. Examples encompass LLaMA, Falcon, and Mistral. Conversely, proprietary models like GPT-4, Gemini, and Claude are closed-source, offering restricted access to their foundational architecture, training data, and weights. The principal distinctions among these paradigms pertain to accessibility, control, security, and reliability. Open models foster transparency, community collaboration, and innovation; yet, they also pose hazards associated with data leakage, adversarial attacks, and detrimental fine-tuning. Conversely, proprietary models emphasize restricted access and protection, however they are plagued by opacity, possible bias, and constraints on public examination.

Security and Ethical Issues of Open Model. Despite their advantages in transparency and adaptability (Chang *et al.*, 2024), open models introduce significant security and ethical risks. First, it can increase the risks of private data leakage (Balloccu *et al.*, 2024). Due to their public accessibility, open models present a heightened risk of data leakage. Malicious entities can obtain sensitive information from these models, either via direct prompt injections or by scrutinizing training datasets (Qiang *et al.*, 2024). Research such as Huang *et al.* (2024a) has

shown that models trained on extensive internet data may unintentionally memorize and reproduce personally identifiable information, trade secrets, or other proprietary information, resulting in significant privacy breaches. Moreover, firms utilizing open models may unintentionally reveal private or confidential information during the fine-tuning process. This risk is particularly alarming in sectors like healthcare, banking, and national security, where data secrecy is critical. Second, open models enable users to customize them using domain-specific datasets, which may induce harmful fine-tuning and model misuse (Huang *et al.*, 2024c). This versatility, however, presents security vulnerabilities, since adversaries may exploit fine-tuning processes to enhance the model for detrimental consequences. Malefactors can create misleading chatbots, prejudiced models, or automated systems proficient at disseminating misinformation. Moreover, fine-tuning on adversarial datasets can circumvent safety measures, resulting in the dissemination of hazardous, unlawful, or unethical content. Recent research indicates that fine-tuning can compromise safety filters, rendering an open architecture more vulnerable to jailbreak attempts. This creates apprehensions regarding the use of open models in contexts where ethical protections are essential.

Non-transparency and Bias Concerns on Proprietary Model. While proprietary models mitigate certain risks associated with open models, they are not without flaws (Kukreja *et al.*, 2024). A significant drawback of proprietary models is their lack of openness. Organizations restrict access to training data, model weights, and fine-tuning methodologies, complicating the auditing of these models for biases or possible dangers (Chang *et al.*, 2024). As a result, systemic biases ingrained in training data continue without external supervision, potentially resulting in unjust or discriminatory outcomes (Hajikhani and Cole, 2024). This opacity erodes user trust, as stakeholders are unable to ascertain the fairness and ethical alignment of these models. Moreover, proprietary models frequently offer restricted elucidations regarding their decision-making methodologies. The black-box characteristic complicates users' comprehension of answer generation (Schwartz *et al.*, 2024), posing significant issues in critical fields including healthcare, law, and finance. In addition, proprietary models are trained on extensive datasets, although the compositions of these datasets remain

undisclosed, prompting questions over biases and fairness. Lack of transparency in data selection precludes the assurance that models are devoid of racial, gender, or ideological biases. Furthermore, entities governing these models may possess motivations to tailor responses to business or political agendas (Kirk *et al.*, 2024), jeopardizing impartiality. Similar to the open models, proprietary models also suffer from hallucination problems, wherein the model produces inaccurate or manufactured responses. Nevertheless, the exclusive character of these models complicates the mitigation of these difficulties. Users frequently struggle to validate the rationale behind results or rectify systematic problems.

Further Discussion. Assessing the reliability of foundation models necessitates a sophisticated strategy that takes into account both open and proprietary systems (Schryen and Kadura, 2009; Kukreja *et al.*, 2024). Although open models enhance openness and research accessibility, they also render users vulnerable to security concerns, including aggressive fine-tuning and data leaking. Proprietary models provide enhanced security control but are plagued by concerns regarding bias, lack of transparency, and susceptibility to convert attacks. A balanced strategy is essential to augment the security of both paradigms: 1) For open models: The implementation of robust safety mechanisms, including differential privacy, adversarial training, and access limitations, can decrease dangers. Transparent governance frameworks, ethical refinement, and community supervision can enhance accountability; 2) For proprietary models: Enhanced openness via external audits, bias assessments, and user accountability measures can augment trustworthiness.

Organizations ought to investigate methods for elucidating model decisions and mitigating biases. Moreover, hybrid models—where proprietary frameworks deliver structured safety protocols while permitting restricted open access for research and auditing—could present a compromise. Promoting interdisciplinary collaboration among AI academics, policymakers, and industry stakeholders is crucial for cultivating a more reliable AI ecosystem. Ultimately, the responsible deployment of foundation models necessitates ongoing assessment, ethical protections, and a dedication to transparency. By mitigating vulnerabilities in both open-source and proprietary models, we can establish a more secure and dependable AI environment that emphasizes justice, security, and ethical principles.

5

Conclusion

In this monograph, we investigate the foundation and trends of trustworthy machine learning from the perspective of data to models. In trustworthy data-centric learning, we discuss robust learning to different data properties by considering label noise, long-tail distributions, out-of-distribution data, and the worst-case scenario, i.e., adversarial examples. In trustworthy private and secured learning, we expand the scope to the model threats of privacy and security, in which we review the foundation methodology like differential privacy, and explore various attacks (e.g., membership inference, model inversion, and data poisoning) and protection methods (e.g., machine unlearning, non-transfer, and federated learning). In trustworthy foundation models, we illustrate the recent critical issues related to the trustworthiness of foundation models, especially for the safety and intellectual property of generated content (e.g., jailbreak prompts and watermarking) and the truthfulness of model reasoning (e.g., hallucination, casual learning and reasoning), and also discuss the different concerns on open and proprietary foundation models. This monograph systematically reviews the problem setting and directions under each sub-area and discusses the potential challenges and future directions. We expect this study will provide a thorough overview and valuable insights in trustworthy machine learning for the community.

References

Aaronson, S. and H. Kirchner. (2022). "Watermarking GPT outputs". URL: https://www.scottaaronson.com/talks/watermark.ppt.

Abadi, M., A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. (2016). "Deep learning with differential privacy". In: *CCS*.

Abdulaal, A., N. Montana-Brown, T. He, A. Ijishakin, I. Drobnjak, D. C. Castro, D. C. Alexander, *et al.* (2023). "Causal Modelling Agents: Causal Graph Discovery through Synergising Metadata-and Data-driven Reasoning". In: *The Twelfth International Conference on Learning Representations*.

Abramson, J., J. Adler, J. Dunger, R. Evans, T. Green, A. Pritzel, O. Ronneberger, L. Willmore, A. J. Ballard, J. Bambrick, *et al.* (2024). "Accurate structure prediction of biomolecular interactions with AlphaFold 3". *Nature*.

Achiam, J., S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, *et al.* (2023). "Gpt-4 technical report". *arXiv preprint arXiv:2303.08774*.

Ahuja, K., E. Caballero, D. Zhang, J.-C. Gagnon-Audet, Y. Bengio, I. Mitliagkas, and I. Rish. (2021). "Invariance Principle Meets Information Bottleneck for Out-of-Distribution Generalization". In: *NeurIPS*.

Almazrouei, E., H. Alobeidli, A. Alshamsi, A. Cappelli, R. Cojocaru, M. Debbah, É. Goffinet, D. Hesslow, J. Launay, Q. Malartic, *et al.* (2023). "The falcon series of open language models". *arXiv preprint arXiv:2311.16867*.

Alowais, S. A., S. S. Alghamdi, N. Alsuhebany, T. Alqahtani, A. I. Alshaya, S. N. Almohareb, A. Aldairem, M. Alrashed, K. Bin Saleh, H. A. Badreldin, *et al.* (2023). "Revolutionizing healthcare: the role of artificial intelligence in clinical practice". *BMC medical education*.

Amiri, M. M., D. Gunduz, S. R. Kulkarni, and H. V. Poor. (2020). "Federated Learning With Quantized Global Model Updates". *arXiv preprint arXiv:2006.10672*.

Amiri, M. M. and D. Gündüz. (2020). "Federated Learning Over Wireless Fading Channels". *IEEE Transactions on Wireless Communications*.

An, S., G. Tao, Q. Xu, Y. Liu, G. Shen, Y. Yao, J. Xu, and X. Zhang. (2022). "Mirror: Model inversion for deep learning network with high fidelity". In: *NDSS*.

Andriushchenko, M. and N. Flammarion. (2020). "Understanding and improving fast adversarial training". In: *NeurIPS*.

Anh, T. T., N. C. Luong, D. Niyato, D. I. Kim, and L.-C. Wang. (2019). "Efficient Training Management for Mobile Crowd-Machine Learning: A Deep Reinforcement Learning Approach". *IEEE Wireless Communications Letters*.

Anil, C., E. Durmus, M. Sharma, J. Benton, S. Kundu, J. Batson, N. Rimsky, M. Tong, J. Mu, D. Ford, *et al.* (2024). "Many-shot jailbreaking". In: *NeurIPS*.

Arjovsky, M., L. Bottou, I. Gulrajani, and D. Lopez-Paz. (2019). "Invariant Risk Minimization". *arXiv preprint arXiv:1907.02893*.

Arpit, D., H. Wang, Y. Zhou, and C. Xiong. (2022). "Ensemble of Averages: Improving Model Selection and Boosting Performance in Domain Generalization". In: *NeurIPS*.

Ashwani, S., K. Hegde, N. R. Mannuru, D. S. Sengar, M. Jindal, K. C. R. Kathala, D. Banga, V. Jain, and A. Chadha. (2024). "Cause and effect: Can large language models truly understand causality?" In: *Proceedings of the AAAI Symposium Series*.

Assran, M., R. Balestriero, Q. Duval, F. Bordes, I. Misra, P. Bojanowski, P. Vincent, M. Rabbat, and N. Ballas. (2023). "The hidden uniform cluster prior in self-supervised learning". In: *ICLR*.

Bai, J., Z. Liu, H. Wang, J. Hao, Y. FENG, H. Chu, and H. Hu. (2023). "On the Effectiveness of Out-of-Distribution Data in Self-Supervised Long-Tail Learning." In: *ICLR*.

Balloccu, S., P. Schmidtová, M. Lango, and O. Dušek. (2024). "Leak, cheat, repeat: Data contamination and evaluation malpractices in closed-source LLMs". *arXiv preprint arXiv:2402.03927*.

Ban, T., L. Chen, D. Lyu, X. Wang, and H. Chen. (2023a). "Causal structure learning supervised by large language model". *arXiv preprint arXiv:2311.11689*.

Ban, T., L. Chen, X. Wang, and H. Chen. (2023b). "From query tools to causal architects: Harnessing large language models for advanced causal discovery from data". *arXiv preprint arXiv:2306.16902*.

Barnett, S., S. Kurniawan, S. Thudumu, Z. Brannelly, and M. Abdelrazek. (2024). "Seven failure points when engineering a retrieval augmented generation system". In: *Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering-Software Engineering for AI*. 194–199.

Barreno, M., B. Nelson, R. Sears, A. D. Joseph, and J. D. Tygar. (2006). "Can machine learning be secure?" In: *ACM CCS*.

Baruch, G., M. Baruch, and Y. Goldberg. (2019). "A little is enough: Circumventing defenses for distributed learning". *NeurIPS*.

Bendale, A. and T. E. Boult. (2016). "Towards open set deep networks". In: *CVPR*.

Betz, J., H. Zheng, A. Liniger, U. Rosolia, P. Karle, M. Behl, V. Krovi, and R. Mangharam. (2022). "Autonomous vehicles on the edge: A survey on autonomous vehicle racing". *IEEE Open Journal of Intelligent Transportation Systems*.

Biggio, B., B. Nelson, P. Laskov, *et al.* (2012). "Poisoning attacks against support vector machines". In: *ICML*.

Biggio, B., B. Nelson, and P. Laskov. (2011). "Support vector machines under adversarial label noise". In: *ACML*.

Brown, T., B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, *et al.* (2020). "Language models are few-shot learners". In: *NeurIPS*.

Brown, T. B. (2020). "Language models are few-shot learners". *arXiv preprint arXiv:2005.14165*.

Burns, C., H. Ye, D. Klein, and J. Steinhardt. (2022). "Discovering latent knowledge in language models without supervision". *arXiv preprint arXiv:2212.03827*.

Cai, H., S. Liu, and R. Song. (2023). "Is Knowledge All Large Language Models Needed for Causal Reasoning?" *arXiv preprint arXiv:2401.00139*.

Cai, H., A. Arunasalam, L. Y. Lin, A. Bianchi, and Z. B. Celik. (2024). "Take a look at it! rethinking how to evaluate language model jailbreak". In: *ACL*.

Cao, C., Z. Zhong, Z. Zhou, Y. Liu, T. Liu, and B. Han. (2024). "Envisioning Outlier Exposure by Large Language Models for Out-of-Distribution Detection". *ICML*.

Cao, D., S. Chang, Z. Lin, G. Liu, and D. Sun. (2019a). "Understanding distributed poisoning attack in federated learning". In: *ICPADS*.

Cao, K., C. Wei, A. Gaidon, N. Arechiga, and T. Ma. (2019b). "Learning imbalanced datasets with label-distribution-aware margin loss". *Advances in neural information processing systems*. 32.

Cao, L. (2022). "Ai in finance: challenges, techniques, and opportunities". *ACM Computing Surveys (CSUR)*.

Carlini, N., S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramer. (2022). "Membership inference attacks from first principles". In: *IEEE SP*.

Carlini, N., M. Nasr, C. A. Choquette-Choo, M. Jagielski, I. Gao, P. W. Koh, D. Ippolito, F. Tramèr, and L. Schmidt. (2023). "Are aligned neural networks adversarially aligned?" In: *NeurIPS*.

Carmon, Y., A. Raghunathan, L. Schmidt, J. C. Duchi, and P. S. Liang. (2019). "Unlabeled data improves adversarial robustness". In: *NeurIPS*.

Caton, S. and C. Haas. (2024). "Fairness in Machine Learning: A Survey". *ACM Comput. Surv.* 56(7): 166:1–166:38.

Cha, J., S. Chun, K. Lee, H.-C. Cho, S. Park, Y. Lee, and S. Park. (2021). "SWAD: Domain Generalization by Seeking Flat Minima". In: *NeurIPS*.

Chang, Y., X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, *et al.* (2024). "A survey on evaluation of large language models". *ACM transactions on intelligent systems and technology*. 15(3): 1–45.

Chanpuriya, S., C. Musco, K. Sotiropoulos, and C. Tsourakakis. (2021). "Deepwalking backwards: from embeddings back to graphs". In: *ICML*.

Chao, P., A. Robey, E. Dobriban, H. Hassani, G. J. Pappas, and E. Wong. (2023). "Jailbreaking black box large language models in twenty queries". *arXiv preprint arXiv:2310.08419*.

Chaudhuri, K. and C. Monteleoni. (2008). "Privacy-preserving logistic regression". In: *NIPS*.

Chen, D., N. Yu, Y. Zhang, and M. Fritz. (2020a). "Gan-leaks: A taxonomy of membership inference attacks against generative models". In: *ACM SIGSAC*.

Chen, H., Y. Dong, Z. Wang, X. Yang, C. Duan, H. Su, and J. Zhu. (2024a). "Robust Classification via a Single Diffusion Model". In: *ICML*.

Chen, I. Y., E. Pierson, S. Rose, S. Joshi, K. Ferryman, and M. Ghassemi. (2021a). "Ethical machine learning in healthcare". *Annual review of biomedical data science*. 4(1): 123–144.

Chen, M., J. Yao, L. Xing, Y. Wang, Y. Zhang, and Y. Wang. (2023a). "Redundancy-adaptive multimodal learning for imperfect data". *arXiv preprint arXiv:2310.14496*.

Chen, M., W. Gao, G. Liu, K. Peng, and C. Wang. (2023b). "Boundary Unlearning". In: *CVPR*.

Chen, R. J., J. J. Wang, D. F. Williamson, T. Y. Chen, J. Lipkova, M. Y. Lu, S. Sahai, and F. Mahmood. (2023c). "Algorithmic fairness in artificial intelligence for medicine and healthcare". *Nature biomedical engineering*. 7(6): 719–742.

Chen, S., M. Kahla, R. Jia, and G. Qi. (2021b). "Knowledge-enriched distributional model inversion attacks". In: *ICCV*.

Chen, T., Z. Zhang, S. Liu, S. Chang, and Z. Wang. (2020b). "Robust overfitting may be mitigated by properly learned smoothening". In: *ICLR*.

Chen, T., S. Kornblith, M. Norouzi, and G. Hinton. (2020c). "A simple framework for contrastive learning of visual representations". In: *ICML*.

Chen, T., S. Kornblith, K. Swersky, M. Norouzi, and G. E. Hinton. (2020d). "Big self-supervised models are strong semi-supervised learners".

Chen, X., Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel. (2016). "Infogan: Interpretable representation learning by information maximizing generative adversarial nets". *NeurIPS*.

Chen, X., Y. Zhou, D. Wu, C. Yang, B. Li, Q. Hu, and W. Wang. (2023d). "Area: adaptive reweighting via effective area for long-tailed classification". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 19277–19287.

Chen, Y., H. C. Lent, and J. Bjerva. (2024b). "Text Embedding Inversion Security for Multilingual Language Models". In: *ACL*.

Chen, Y., Y. Bian, B. Han, and J. Cheng. (2024c). "How Interpretable Are Interpretable Graph Neural Networks?" In: *ICML*.

Chen, Y., Y. Bian, K. Zhou, B. Xie, B. Han, and J. Cheng. (2023e). "Does Invariant Graph Learning via Environment Augmentation Learn Invariance?" In: *NeurIPS*.

Chen, Y., W. Huang, K. Zhou, Y. Bian, B. Han, and J. Cheng. (2023f). "Understanding and Improving Feature Learning for Out-of-Distribution Generalization". In: *NeurIPS*.

Chen, Y., Y. Zhang, Y. Bian, H. Yang, K. Ma, B. Xie, T. Liu, B. Han, and J. Cheng. (2022). "Learning Causally Invariant Representations for Out-of-Distribution Generalization on Graphs". In: *NeurIPS*.

Chen, Y., K. Zhou, Y. Bian, B. Xie, K. Ma, Y. Zhang, H. Yang, B. Han, and J. Cheng. (2023g). "Pareto Invariant Risk Minimization". In: *ICLR*.

Chen, Z., Z. Zhao, H. Luo, H. Yao, B. Li, and J. Zhou. (2024d). "HALC: Object Hallucination Reduction via Adaptive Focal-Contrast Decoding". In: *ICML*.

Cheng, Y., C. Shan, Y. Shen, X. Li, S. Luo, and D. Li. (2024a). "Resurrecting Label Propagation for Graphs with Heterophily and Label Noise". In: *KDD*.

Cheng, Y., C. Shan, Y. Shen, X. Li, S. Luo, and D. Li. (2024b). "Resurrecting Label Propagation for Graphs with Heterophily and Label Noise". In: *KDD*.

Chi, H., H. Li, W. Yang, F. Liu, L. Lan, X. Ren, T. Liu, and B. Han. (2024a). "Unveiling Causal Reasoning in Large Language Models: Reality or Mirage?" In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Chi, H., H. Li, W. Yang, F. Liu, L. Lan, X. Ren, T. Liu, and B. Han. (2024b). "Unveiling Causal Reasoning in Large Language Models: Reality or Mirage?" In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Cho, Y. J., A. Manoel, G. Joshi, R. Sim, and D. Dimitriadis. (2022). "Heterogeneous ensemble knowledge transfer for training large models in federated learning". *arXiv preprint arXiv:2204.12703*.

Chowdhury, A. G., M. M. Islam, V. Kumar, F. H. Shezan, V. Jain, and A. Chadha. (2024). "Breaking down the defenses: A comparative survey of attacks on large language models". *arXiv preprint arXiv:2403.04786*.

Christiano, P. F., J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei. (2017). "Deep reinforcement learning from human preferences". In: *NeurIPS*.

Chu, P., X. Bian, S. Liu, and H. Ling. (2020). "Feature space augmentation for long-tailed data". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16*. Springer. 694–710.

Chuang, C.-Y., R. D. Hjelm, X. Wang, V. Vineet, N. Joshi, A. Torralba, S. Jegelka, and Y. Song. (2022). "Robust contrastive learning against noisy views". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16670–16681.

Chuang, Y.-S., Y. Xie, H. Luo, Y. Kim, J. Glass, and P. He. (2023). "Dola: Decoding by contrasting layers improves factuality in large language models". *arXiv preprint arXiv:2309.03883*.

Cour, T., B. Sapp, and B. Taskar. (2011). "Learning from partial labels". *The Journal of Machine Learning Research*.

Creager, E., J. Jacobsen, and R. S. Zemel. (2021). "Environment Inference for Invariant Learning". In: *ICML*.

Criado, M. F., F. E. Casado, R. Iglesias, C. V. Regueiro, and S. Barro. (2022). "Non-IID data and Continual Learning processes in Federated Learning: A long road ahead". *Information Fusion*.

Croitoru, F.-A., V. Hondru, R. T. Ionescu, and M. Shah. (2023). "Diffusion models in vision: A survey". *IEEE TPAMI*.

Cubuk, E. D., B. Zoph, J. Shlens, and Q. V. Le. (2020). "Randaugment: Practical automated data augmentation with a reduced search space". In: *CVPR workshops*.

Cui, Y., M. Jia, T.-Y. Lin, Y. Song, and S. Belongie. (2019). "Class-balanced loss based on effective number of samples". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 9268–9277.

Cuturi, M. (2013). "Sinkhorn Distances: Lightspeed Computation of Optimal Transport". In: *NeurIPS*.

Dai, E., C. Aggarwal, and S. Wang. (2021). "Nrgnn: Learning a label noise resistant graph neural network on sparsely and noisily labeled graphs". In: *KDD*.

Dai, J., X. Pan, R. Sun, J. Ji, X. Xu, M. Liu, Y. Wang, and Y. Yang. (2024a). "Safe rlhf: Safe reinforcement learning from human feedback". In: *ICLR*.

Dai, R., Y. Zhang, A. Li, T. Liu, X. Yang, and B. Han. (2024b). "Enhancing One-Shot Federated Learning Through Data and Ensemble Co-Boosting". In: *The Twelfth International Conference on Learning Representations*.

Das, B. C., M. H. Amini, and Y. Wu. (2024). "Security and privacy challenges of large language models: A survey". *arXiv preprint arXiv:2402.00888*.

Demontis, A., M. Melis, M. Pintor, M. Jagielski, B. Biggio, A. Oprea, C. Nita-Rotaru, and F. Roli. (2019). "Why do adversarial attacks transfer? explaining transferability of evasion and poisoning attacks". In: *USENIX security*.

Deng, G., Y. Liu, Y. Li, K. Wang, Y. Zhang, Z. Li, H. Wang, T. Zhang, and Y. Liu. (2023). "MasterKey: Automated Jailbreak Across Multiple Large Language Model Chatbots". *arXiv preprint arXiv:2307.08715*.

Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. (2009). "Imagenet: A large-scale hierarchical image database". In: *CVPR*.

Deng, J., S. Pang, Y. Chen, L. Xia, Y. Bai, H. Weng, and W. Xu. (2024). "SOPHON: Non-Fine-Tunable Learning to Restrain Task Transferability For Pre-trained Models". In: *IEEE SP*.

Deng, Z., X. Yang, S. Xu, H. Su, and J. Zhu. (2021). "LiBRe: A Practical Bayesian Approach to Adversarial Detection". In: *CVPR*.

Dennis, D. K., T. Li, and V. Smith. (2021). "Heterogeneity for the win: One-shot federated clustering". In: *International Conference on Machine Learning*.

Desai, A., T.-Y. Wu, S. Tripathi, and N. Vasconcelos. (2021). "Learning of visual relations: The devil is in the tails". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 15404–15413.

Devlin, J. (2018). "Bert: Pre-training of deep bidirectional transformers for language understanding". *arXiv preprint arXiv:1810.04805*.

Devlin, J., M.-W. Chang, K. Lee, and K. Toutanova. (2019). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *NAACL*.

Dhawan, N., L. Cotta, K. Ullrich, R. Krishnan, and C. J. Maddison. (2024). "End-To-End Causal Effect Estimation from Unstructured Natural Language Data". In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Diao, H., Y. Zhang, L. Ma, and H. Lu. (2021). "Similarity reasoning and filtration for image-text matching". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 35. No. 2. 1218–1226.

Ding, G. W., Y. Sharma, K. Y. C. Lui, and R. Huang. (2020). "MMA Training: Direct Input Space Margin Maximization through Adversarial Training". In: *ICLR*.

Ding, P., J. Kuang, D. Ma, X. Cao, Y. Xian, J. Chen, and S. Huang. (2023). "A Wolf in Sheep's Clothing: Generalized Nested Jailbreak Prompts can Fool Large Language Models Easily". In: *NAACL*.

Djurisic, A., N. Bozanic, A. Ashok, and R. Liu. (2023). "Extremely simple activation shaping for out-of-distribution detection". In:

Dong, M. and Y. Kluger. (2023). "Towards understanding and reducing graph structural noise for GNNs". In: *ICML*.

Došilović, F. K., M. Brčić, and N. Hlupić. (2018). "Explainable artificial intelligence: A survey". In: *2018 41st International convention on information and communication technology, electronics and microelectronics (MIPRO)*. IEEE. 0210–0215.

Du, X., T. Bian, Y. Rong, B. Han, T. Liu, T. Xu, W. Huang, Y. Li, and J. Huang. (2021). "Noise-robust graph learning by estimating and leveraging pairwise interactions". *TMLR*.

Duan, J., F. Kong, S. Wang, X. Shi, and K. Xu. (2023). "Are diffusion models vulnerable to membership inference attacks?" In: *ICML*. PMLR.

Dubey, A., A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan, *et al.* (2024). "The llama 3 herd of models". *arXiv*.

Dwork, C., K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor. (2006). "Our data, ourselves: Privacy via distributed noise generation". In: *Advances in Cryptology – EUROCRYPT*.

Dwork, C. and A. Roth. (2014). "The algorithmic foundations of differential privacy". *Foundations and Trends® in Theoretical Computer Science*.

Dziri, N., S. Milton, M. Yu, O. Zaiane, and S. Reddy. (2022). "On the origin of hallucinations in conversational models: Is it the datasets or the models?" *arXiv preprint arXiv:2204.07931*.

ElKordy, A. and A. S. Avestimehr. (2020). "Secure aggregation with heterogeneous quantization in federated learning". *arXiv preprint arxiv:2009.14388*.

Esmaeilpour, S., B. Liu, E. Robertson, and L. Shu. (2022). "Zero-shot out-of-distribution detection based on the pre-trained model clip". In: *AAAI*.

Fan, C., J. Liu, Y. Zhang, D. Wei, E. Wong, and S. Liu. (2024). "SalUn: Empowering Machine Unlearning via Gradient-based Weight Saliency in Both Image Classification and Generation". In: *ICLR*.

Fan, J., Q. Yan, M. Li, G. Qu, and Y. Xiao. (2022). "A survey on data poisoning attacks and defenses". In: *DSC*.

Fang, A., G. Ilharco, M. Wortsman, Y. Wan, V. Shankar, A. Dave, and L. Schmidt. (2022a). "Data Determines Distributional Robustness in Contrastive Language Image Pre-training (CLIP)". In: *ICML*.

Fang, Z., Y. Li, J. Lu, J. Dong, B. Han, and F. Liu. (2022b). "Is out-of-distribution detection learnable?" In:

Feder, A., Y. Wald, C. Shi, S. Saria, and D. Blei. (2024). "Causal-structure driven augmentations for text ood generalization". *Advances in Neural Information Processing Systems*.

Feffer, M., A. Sinha, Z. C. Lipton, and H. Heidari. (2024). "Red-Teaming for Generative AI: Silver Bullet or Security Theater?" In: *AIES*.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Bradford Books. URL: https://mitpress.mit.edu/9780262561167/.

Feng, C., Y. Zhong, and W. Huang. (2021). "Exploring classification equilibrium in long-tailed object detection". In: *Proceedings of the IEEE/CVF International conference on computer vision*. 3417–3426.

Feng, L., J. Lv, B. Han, M. Xu, G. Niu, X. Geng, B. An, and M. Sugiyama. (2020). "Provably Consistent Partial-Label Learning". In: *NeurIPS*.

Feng, Q., L. Xie, S. Fang, and T. Lin. (2024). "BaCon: Boosting Imbalanced Semi-supervised Learning via Balanced Feature-Level Contrastive Learning". In: *AAAI*.

Feng, Z., Z. Zeng, C. Guo, Z. Li, and L. Hu. (2023). "Learning from noisy correspondence with tri-partition for cross-modal matching". *IEEE Transactions on Multimedia*.

Fernandez, P., G. Couairon, H. Jégou, M. Douze, and T. Furon. (2023). "The stable signature: Rooting watermarks in latent diffusion models". In: *ICCV*.

Fort, S., J. Ren, and B. Lakshminarayanan. (2021). "Exploring the Limits of Out-of-Distribution Detection". In: *NeurIPS*.

Frankle, J. and M. Carbin. (2018). "The Lottery Ticket Hypothesis: Training Pruned Neural Networks". *CoRR*.

Fredrikson, M., E. Lantz, S. Jha, S. Lin, D. Page, and T. Ristenpart. (2014). "Privacy in pharmacogenetics: An {End-to-End} case study of personalized warfarin dosing". In: *USENIX Security*.

Fredrikson, M., S. Jha, and T. Ristenpart. (2015). "Model inversion attacks that exploit confidence information and basic countermeasures". In: *CCS*.

Fu, W., H. Wang, C. Gao, G. Liu, Y. Li, and T. Jiang. (2023). "A Probabilistic Fluctuation based Membership Inference Attack for Diffusion Models". *arXiv e-prints*: arXiv–2308.

Fu, X., X. Wang, Q. Li, J. Liu, J. Dai, and J. Han. (2024). "Model Will Tell: Training Membership Inference for Diffusion Models". *arXiv preprint arXiv:2403.08487*.

Fukuchi, K., Q. K. Tran, and J. Sakuma. (2017). "Differentially private empirical risk minimization with input perturbation". In: *Discovery Science*.

Gagnon-Audet, J.-C., K. Ahuja, M. J. D. Bayazi, P. Mousavi, G. Dumas, and I. Rish. (2023). "WOODS: Benchmarks for Out-of-Distribution Generalization in Time Series". *TMLR*.

Gan, K. and T. Wei. (2024). "Erasing the Bias: Fine-Tuning Foundation Models for Semi-Supervised Learning". In: *ICML*.

Gandikota, R., J. Materzynska, J. Fiotto-Kaufman, and D. Bau. (2023). "Erasing concepts from diffusion models". In: *ICCV*.

Ganin, Y., E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. S. Lempitsky. (2016). "Domain-Adversarial Training of Neural Networks". *Journal of Mache Learning Research*.

Gao, D., X. Yao, and Q. Yang. (2022a). "A survey on heterogeneous federated learning". *arXiv preprint arXiv:2210.04505*.

Gao, L., Z. Dai, P. Pasupat, A. Chen, A. T. Chaganty, Y. Fan, V. Y. Zhao, N. Lao, H. Lee, D.-C. Juan, *et al.* (2022b). "Rarr: Researching and revising what language models say, using language models". *arXiv preprint arXiv:2210.08726*.

Gao, R., F. Liu, J. Zhang, B. Han, T. Liu, G. Niu, and M. Sugiyama. (2021). "Maximum Mean Discrepancy Test is Aware of Adversarial Attacks". In: *ICML*.

Ghosh, A., J. Chung, D. Yin, and K. Ramchandran. (2020). "An efficient framework for clustered federated learning". *Advances in Neural Information Processing Systems*.

Golatkar, A., A. Achille, and S. Soatto. (2020). "Eternal sunshine of the spotless net: Selective forgetting in deep networks". In: *CVPR*.

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

Goodfellow, I. J., J. Shlens, and C. Szegedy. (2015). "Explaining and Harnessing Adversarial Examples". In: *ICLR*.

Gretton, A., K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. J. Smola. (2012). "A Kernel Two-Sample Test". *J. Mach. Learn. Res.*

Grill, J.-B., F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, *et al.* (2020). "Bootstrap your own latent-a new approach to self-supervised learning". In: *NeurIPS*.

Gu, C., X. L. Li, P. Liang, and T. Hashimoto. (2023). "On the learnability of watermarks for language models". *arXiv preprint arXiv:2312.04469*.

Gu, X., X. Zheng, T. Pang, C. Du, Q. Liu, Y. Wang, J. Jiang, and M. Lin. (2024). "Agent smith: A single image can jailbreak one million multimodal llm agents exponentially fast". In: *ICML*.

Guerreiro, N. M., D. M. Alves, J. Waldendorf, B. Haddow, A. Birch, P. Colombo, and A. F. Martins. (2023). "Hallucinations in large multilingual translation models". *Transactions of the Association for Computational Linguistics*. 11: 1500–1517.

Guha, N., A. Talwalkar, and V. Smith. (2019). "One-shot federated learning". *arXiv preprint arXiv:1902.11175*.

Gui, S., X. Li, L. Wang, and S. Ji. (2022). "GOOD: A Graph Out-of-Distribution Benchmark". In: *NeurIPS*.

Gulrajani, I. and D. Lopez-Paz. (2021). "In Search of Lost Domain Generalization". In: *ICLR*.

Guo, C., G. Pleiss, Y. Sun, and K. Q. Weinberger. (2017). "On calibration of modern neural networks". In: *ICML*.

Guo, J., Y. Li, L. Wang, S.-T. Xia, H. Huang, C. Liu, and B. Li. (2024). "Domain watermark: Effective and harmless dataset copyright protection is closed at hand". In: *NeurIPS*.

Hajikhani, A. and C. Cole. (2024). "A critical review of large language models: Sensitivity, bias, and the path toward specialized ai". *Quantitative Science Studies*. 5(3): 736–756.

Han, B., J. Yao, G. Niu, M. Zhou, I. Tsang, Y. Zhang, and M. Sugiyama. (2018a). "Masking: A new perspective of noisy supervision". *Advances in neural information processing systems*. 31.

Han, B., Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, and M. Sugiyama. (2018b). "Co-teaching: Robust training of deep neural networks with extremely noisy labels". *Advances in neural information processing systems*. 31.

Han, G., J. Choi, H. Lee, and J. Kim. (2023a). "Reinforcement learning-based black-box model inversion attacks". In: *CVPR*.

Han, H., K. Miao, Q. Zheng, and M. Luo. (2023b). "Noisy correspondence learning with meta similarity correction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7517–7526.

Han, H., Q. Zheng, G. Dai, M. Luo, and J. Wang. (2024). "Learning to Rematch Mismatched Pairs for Robust Cross-Modal Retrieval". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 26679–26688.

Hanif, A., M. Naseer, S. H. Khan, M. Shah, and F. S. Khan. (2023). "Frequency Domain Adversarial Training for Robust Volumetric Medical Segmentation". In: *MICCAI*.

Hayes, J., L. Melis, G. Danezis, and E. De Cristofaro. (2017). "Logan: Membership inference attacks against generative models". *arXiv preprint arXiv:1705.07663*.

He, K., H. Fan, Y. Wu, S. Xie, and R. Girshick. (2020). "Momentum contrast for unsupervised visual representation learning". In: *CVPR*.

Henaff, O. (2020). "Data-efficient image recognition with contrastive predictive coding". In: *ICML*.

Hendrycks, D. and K. Gimpel. (2017). "A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks". In: *ICLR*.

Hendrycks, D., M. Mazeika, and T. Dietterich. (2018). "Deep anomaly detection with outlier exposure". In: *ICLR*.

Hessel, J., A. Holtzman, M. Forbes, R. L. Bras, and Y. Choi. (2021). "Clipscore: A reference-free evaluation metric for image captioning". *arXiv preprint arXiv:2104.08718*.

Holmes, W., K. Porayska-Pomsta, K. Holstein, E. Sutherland, T. Baker, S. B. Shum, O. C. Santos, M. T. Rodrigo, M. Cukurova, I. I. Bittencourt, *et al.* (2022). "Ethics of AI in education: Towards a community-wide framework". *International Journal of Artificial Intelligence in Education*: 1–23.

Hong, F., J. Yao, Y. Lyu, Z. Zhou, I. W. Tsang, Y. Zhang, and Y. Wang. (2024a). "On Harmonizing Implicit Subpopulations". In: *ICLR*.

Hong, F., J. Yao, Z. Zhou, Y. Zhang, and Y. Wang. (2023). "Long-Tailed Partial Label Learning via Dynamic Rebalancing". In: *ICLR*.

Hong, Y., S. Han, K. Choi, S. Seo, B. Kim, and B. Chang. (2021). "Disentangling label distribution for long-tailed visual recognition". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 6626–6636.

Hong, Z.-W., I. Shenfeld, T.-H. Wang, Y.-S. Chuang, A. Pareja, J. R. Glass, A. Srivastava, and P. Agrawal. (2024b). "Curiosity-driven Red-teaming for Large Language Models". In: *ICLR*.

Hong, Z., L. Shen, and T. Liu. (2024c). "Your Transferability Barrier is Fragile: Free-Lunch for Transferring the Non-Transferable Learning". In: *CVPR*.

Hong, Z., Z. Wang, L. Shen, Y. Yao, Z. Huang, S. Chen, C. Yang, M. Gong, and T. Liu. (2024d). "Improving Non-Transferable Representation Learning by Harnessing Content and Style". In: *ICLR*.

Hospedales, T., A. Antoniou, P. Micaelli, and A. Storkey. (2021). "Meta-learning in neural networks: A survey". *IEEE transactions on pattern analysis and machine intelligence*. 44(9): 5149–5169.

Hou, A. B., J. Zhang, T. He, Y. Wang, Y.-S. Chuang, H. Wang, L. Shen, B. Van Durme, D. Khashabi, and Y. Tsvetkov. (2023). "Semstamp: A semantic watermark with paraphrastic robustness for text generation". *arXiv preprint arXiv:2310.03991*.

Hu, C.-H., Z. Chen, and E. Larsson. (2022a). "Scheduling and Aggregation Design for Asynchronous Federated Learning Over Wireless Networks". *IEEE Journal on Selected Areas in Communications*.

Hu, H., Z. Salcic, L. Sun, G. Dobbie, P. S. Yu, and X. Zhang. (2022b). "Membership inference attacks on machine learning: A survey". *ACM CSUR*.

Hu, P., Z. Huang, D. Peng, X. Wang, and X. Peng. (2023). "Cross-modal retrieval with partially mismatched pairs". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 45(8): 9595–9610.

Hu, W., G. Niu, I. Sato, and M. Sugiyama. (2018). "Does Distributionally Robust Supervised Learning Give Robust Classifiers?" In: *ICML*.

Huang, J., D. Yang, and C. Potts. (2024a). "Demystifying verbatim memorization in large language models". *arXiv preprint arXiv:2407.17817*.

Huang, Q., X. Dong, P. Zhang, B. Wang, C. He, J. Wang, D. Lin, W. Zhang, and N. Yu. (2024b). "Opera: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation". In: *CVPR*.

Huang, R., A. Geng, and Y. Li. (2021a). "On the importance of gradients for detecting distributional shifts in the wild". *NeurIPS*. 34.

Huang, R., Y. Long, J. Han, H. Xu, X. Liang, C. Xu, and X. Liang. (2023a). "Nlip: Noise-robust language-image pre-training". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 37. No. 1. 926–934.

Huang, T., S. Hu, F. Ilhan, S. F. Tekin, and L. Liu. (2024c). "Harmful fine-tuning attacks and defenses for large language models: A survey". *arXiv preprint arXiv:2409.18169*.

Huang, W. R., J. Geiping, L. Fowl, G. Taylor, and T. Goldstein. (2020). "Metapoison: Practical general-purpose clean-label data poisoning". *NeurIPS*.

Huang, W., A. Han, Y. Chen, Y. Cao, zhiqiang xu, and T. Suzuki. (2024d). "On the Comparison between Multi-modal and Single-modal Contrastive Learning". In: *NeurIPS*.

Huang, Y., B. Bai, S. Zhao, K. Bai, and F. Wang. (2022). "Uncertainty-Aware Learning against Label Noise on Imbalanced Datasets". In: *AAAI*.

Huang, Z., G. Niu, X. Liu, W. Ding, X. Xiao, H. Wu, and X. Peng. (2021b). "Learning with noisy correspondence for cross-modal matching". *Advances in Neural Information Processing Systems*. 34: 29406–29419.

Huang, Z., M. Yang, X. Xiao, P. Hu, and X. Peng. (2024e). "Noise-robust Vision-language Pre-training with Positive-negative Learning". *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Huang, Z., Y. Fan, C. Liu, W. Zhang, Y. Zhang, M. Salzmann, S. Süsstrunk, and J. Wang. (2023b). "Fast adversarial training with adaptive step size". In: *IEEE Transactions on Image Processing*.

Hüllermeier, E. and J. Beringer. (2006). "Learning from ambiguously labeled examples". *Intell. Data Anal.* 10(5): 419–439.

Imbens, G. W. and D. B. Rubin. (2015). *Causal inference in statistics, social, and biomedical sciences*. Cambridge university press.

Iyengar, R., J. P. Near, D. Song, O. Thakkar, A. Thakurta, and L. Wang. (2019). "Towards practical differentially private convex optimization". In: *SP*.

Jagielski, M., A. Oprea, B. Biggio, C. Liu, C. Nita-Rotaru, and B. Li. (2018). "Manipulating machine learning: Poisoning attacks and countermeasures for regression learning". In: *IEEE symposium on security and privacy (SP)*.

Ji, Z., N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung. (2023). "Survey of hallucination in natural language generation". *ACM Computing Surveys*.

Jia, J., J. Liu, P. Ram, Y. Yao, G. Liu, Y. Liu, P. Sharma, and S. Liu. (2023). "Model Sparsity Can Simplify Machine Unlearning". In: *NeurIPS*.

Jia, X., Y. Zhang, X. Wei, B. Wu, K. Ma, J. Wang, and X. Cao. (2022a). "Prior-Guided Adversarial Initialization for Fast Adversarial Training". In: *ECCV*.

Jia, X., Y. Zhang, B. Wu, K. Ma, J. Wang, and X. Cao. (2022b). "LAS-AT: Adversarial Training with Learnable Attack Strategy". In: *CVPR*.

Jia, Y., X. Peng, R. Wang, and M. Zhang. (2024). "Long-Tailed Partial Label Learning by Head Classifier and Tail Classifier Cooperation". In: *AAAI*. Ed. by M. J. Wooldridge, J. G. Dy, and S. Natarajan.

Jiang, H., L. Ge, Y. Gao, J. Wang, and R. Song. (2024a). "LLM4Causal: Democratized Causal Tools for Everyone via Large Language Model". In: *First Conference on Language Modeling*.

Jiang, L. and T. Lin. (2023). "Test-Time Robust Personalization for Federated Learning".

Jiang, X., F. Liu, Z. Fang, H. Chen, T. Liu, F. Zheng, and B. Han. (2024b). "Negative Label Guided OOD Detection with Pretrained Vision-Language Models". In: *ICLR*.

Jiang, Z., T. Chen, B. J. Mortazavi, and Z. Wang. (2021). "Self-damaging contrastive learning". In: *ICML*.

Jin, Z., Y. Chen, F. Leeb, L. Gresele, O. Kamal, L. Zhiheng, K. Blin, F. G. Adauto, M. Kleiman-Weiner, M. Sachan, *et al.* (2023). "Cladder: Assessing causal reasoning in language models". In: *Thirty-seventh conference on neural information processing systems*.

Jin, Z., J. Liu, L. Zhiheng, S. Poff, M. Sachan, R. Mihalcea, M. T. Diab, and B. Schölkopf. (2024). "Can Large Language Models Infer Causation from Correlation?" In: *The Twelfth International Conference on Learning Representations*.

Jiralerspong, T., X. Chen, Y. More, V. Shah, and Y. Bengio. (2024). "Efficient causal graph discovery using large language models". *arXiv preprint arXiv:2402.01207*.

Jordan, M. I. and T. M. Mitchell. (2015). "Machine learning: Trends, perspectives, and prospects". *Science*.

Jorge Aranda, P. de, A. Bibi, R. Volpi, A. Sanyal, P. Torr, G. Rogez, and P. Dokania. (2022). "Make some noise: Reliable and efficient single-step adversarial training". In: *NeurIPS*.

Kadavath, S., T. Conerly, A. Askell, T. Henighan, D. Drain, E. Perez, N. Schiefer, Z. Hatfield-Dodds, N. DasSarma, E. Tran-Johnson, *et al.* (2022). "Language models (mostly) know what they know". *arXiv preprint arXiv:2207.05221*.

Kahla, M., S. Chen, H. Just, and R. Jia. (2022). "Label-only model inversion attacks via boundary repulsion". In: *CVPR*.

Kairouz, P., H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, *et al.* (2021). "Advances and open problems in federated learning". *Foundations and Trends® in Machine Learning*.

Kang, B., S. Xie, M. Rohrbach, Z. Yan, A. Gordo, J. Feng, and Y. Kalantidis. (2020). "Decoupling Representation and Classifier for Long-Tailed Recognition". In: *ICLR*.

Karras, T., S. Laine, and T. Aila. (2019). "A style-based generator architecture for generative adversarial networks". In: *CVPR*.

Katz-Samuels, J., J. B. Nakhleh, R. Nowak, and Y. Li. (2022). "Training ood detectors in their natural habitats". In: *ICML*.

Khan, S., M. Hayat, S. W. Zamir, J. Shen, and L. Shao. (2019). "Striking the right balance with uncertainty". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 103–112.

Khodak, M., M. Balcan, and A. Talwalkar. (2019). "Adaptive Gradient-Based Meta-Learning Methods". *CoRR*.

Kifer, D., A. Smith, and A. Thakurta. (2012). "Private convex empirical risk minimization and high-dimensional regression". In: *COLT*.

Kim, H., W. Lee, and J. Lee. (2021). "Understanding catastrophic overfitting in single-step adversarial training". In: *AAAI*.

Kim, J., Y. Hur, S. Park, E. Yang, S. J. Hwang, and J. Shin. (2020). "Distribution Aligning Refinery of Pseudo-label for Imbalanced Semi-supervised Learning". In: *NeurIPS*.

Kingma, D. P. (2013). "Auto-encoding variational bayes". *arXiv preprint arXiv:1312.6114*.

Kirchenbauer, J., J. Geiping, Y. Wen, J. Katz, I. Miers, and T. Goldstein. (2023). "A watermark for large language models". In: *ICML*.

Kirk, H. R., B. Vidgen, P. Röttger, and S. A. Hale. (2024). "The benefits, risks and bounds of personalizing the alignment of large language models to individuals". *Nature Machine Intelligence*. 6(4): 383–392.

Kıcıman, E., R. Ness, A. Sharma, and C. Tan. (2023). "Causal reasoning and large language models: Opening a new frontier for causality". *arXiv preprint arXiv:2305.00050*.

Koh, P. W., S. Sagawa, H. Marklund, S. M. Xie, M. Zhang, A. Balsubramani, W. Hu, M. Yasunaga, R. L. Phillips, I. Gao, T. Lee, E. David, I. Stavness, W. Guo, B. Earnshaw, I. Haque, S. M. Beery, J. Leskovec, A. Kundaje, E. Pierson, S. Levine, C. Finn, and P. Liang. (2021). "WILDS: A Benchmark of in-the-Wild Distribution Shifts". In: *ICML*.

Kotzias, D., M. Denil, N. De Freitas, and P. Smyth. (2015). "From group to individual labels using deep features". In: *SIGKDD*.

Krueger, D., E. Caballero, J. Jacobsen, A. Zhang, J. Binas, D. Zhang, R. L. Priol, and A. C. Courville. (2021). "Out-of-Distribution Generalization via Risk Extrapolation (REx)". In: *ICML*.

Kukleva, A., M. Böhle, B. Schiele, H. Kuehne, and C. Rupprecht. (2023). "Temperature Schedules for self-supervised contrastive methods on long-tail data". In: *ICLR*.

Kukreja, S., T. Kumar, A. Purohit, A. Dasgupta, and D. Guha. (2024). "A literature survey on open source large language models". In: *Proceedings of the 2024 7th International Conference on Computers in Management and Business*. 133–143.

Le, H. D., X. Xia, and Z. Chen. (2024). "Multi-Agent Causal Discovery Using Large Language Models". *arXiv preprint arXiv:2407.15073*.

LeCun, Y., Y. Bengio, and G. Hinton. (2015). "Deep learning". *Nature*.

Lecuyer, M., V. Atlidakis, R. Geambasu, D. Hsu, and S. Jana. (2019). "Certified robustness to adversarial examples with differential privacy". In: *SP*.

Lederer, I., R. Mayer, and A. Rauber. (2023). "Identifying appropriate intellectual property protection mechanisms for machine learning models: A systematization of watermarking, fingerprinting, model access, and attacks". *IEEE Transactions on Neural Networks and Learning Systems*.

Lee, H., S. Shin, and H. Kim. (2021). "ABC: Auxiliary Balanced Classifier for Class-imbalanced Semi-supervised Learning". In: *NeurIPS*.

Lee, J. and D. Kifer. (2018). "Concentrated differentially private gradient descent with adaptive per-iteration privacy budget". In: *KDD*.

Lee, K., K. Lee, H. Lee, and J. Shin. (2018). "A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks". In: *NeurIPS*.

Lee, M. and D. Kim. (2023). "Robust Evaluation of Diffusion-Based Adversarial Purification". In: *ICCV*.

Lee, N., W. Ping, P. Xu, M. Patwary, P. N. Fung, M. Shoeybi, and B. Catanzaro. (2022). "Factuality enhanced language models for open-ended text generation". *Advances in Neural Information Processing Systems*. 35: 34586–34599.

Leng, S., H. Zhang, G. Chen, X. Li, S. Lu, C. Miao, and L. Bing. (2024). "Mitigating object hallucinations in large vision-language models through visual contrastive decoding". In: *CVPR*.

Levinstein, B. A. and D. A. Herrmann. (2024). "Still no lie detector for language models: Probing empirical and conceptual roadblocks". *Philosophical Studies*: 1–27.

Li, A., J. Sun, B. Wang, L. Duan, S. Li, Y. Chen, and H. Li. (2020a). "LotteryFL: Personalized and Communication-Efficient Federated Learning with Lottery Ticket Hypothesis on Non-IID Datasets".

Li, B. and Y. Li. (2023). "Towards Understanding Clean Generalization and Robust Overfitting in Adversarial Training". In: *arXiv preprint arXiv:2306.01271*.

Li, D., X. Li, Z. Gan, Q. Li, B. Qu, and J. Wang. (2024a). "Rethinking the impact of noisy labels in graph classification: A utility and privacy perspective". *Neural Networks*.

Li, H., M. Xu, and Y. Song. (2023a). "Sentence Embedding Leaks More Information than You Expect: Generative Embedding Inversion Attack to Recover the Whole Sentence". In: *ACL*.

Li, J., D. Li, C. Xiong, and S. Hoi. (2022a). "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation". In: *International conference on machine learning*. PMLR. 12888–12900.

Li, J., R. Socher, and S. C. H. Hoi. (2020b). "DivideMix: Learning with Noisy Labels as Semi-supervised Learning". In: *ICLR*.

Li, K., O. Patel, F. Viégas, H. Pfister, and M. Wattenberg. (2024b). "Inference-time intervention: Eliciting truthful answers from a language model". *Advances in Neural Information Processing Systems*. 36.

Li, L. and M. W. Spratling. (2023). "Data augmentation alone can improve adversarial training". In: *ICLR*.

Li, P., X. Wang, Z. Zhang, Y. Meng, F. Shen, Y. Li, J. Wang, Y. Li, and W. Zhu. (2024c). "RealTCD: Temporal Causal Discovery from Interventional Data with Large Language Model". *arXiv preprint arXiv:2404.14786v2*.

Li, Q., B. He, and D. Song. (2020c). "Practical One-Shot Federated Learning for Cross-Silo Setting". *arXiv preprint arXiv:2010.01017*.

Li, Q., B. He, and D. Song. (2021a). "Model-Contrastive Federated Learning".

Li, S., H. Liu, T. Dong, B. Z. H. Zhao, M. Xue, H. Zhu, and J. Lu. (2021b). "Hidden backdoors in human-centric language models". In: *ACM SIGSAC*.

Li, T., A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith. (2020d). "Federated Optimization in Heterogeneous Networks".

Li, T., M. Sanjabi, A. Beirami, and V. Smith. (2020e). "Fair Resource Allocation in Federated Learning". In: *International Conference on Learning Representations*.

Li, X., Q. Li, D. Li, H. Qian, and J. Wang. (2024d). "Contrastive learning of graphs under label noise". *Neural Networks*.

Li, Y., X. Tian, M. Gong, Y. Liu, T. Liu, K. Zhang, and D. Tao. (2018). "Deep Domain Generalization via Conditional Invariant Adversarial Networks". In: *ECCV*.

Li, Y., J. Yin, and L. Chen. (2021c). "Unified robust training for graph neural networks against label noise". In: *PAKDD*.

Li, Y., Y. Jiang, Z. Li, and S.-T. Xia. (2022b). "Backdoor learning: A survey". *IEEE TNNLS*.

Li, Y., S. Bubeck, R. Eldan, A. Del Giorno, S. Gunasekar, and Y. T. Lee. (2023b). "Textbooks are all you need ii: phi-1.5 technical report". *arXiv preprint arXiv:2309.05463*.

Liang, P. P., T. Liu, Z. Liu, R. Salakhutdinov, and L. Morency. (2020). "Think Locally, Act Globally: Federated Learning with Local and Global Representations". *CoRR*.

Liao, F., M. Liang, Y. Dong, T. Pang, X. Hu, and J. Zhu. (2018). "Defense against adversarial attacks using high-level representation guided denoiser". In: *CVPR*.

Lim, W. Y. B., N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao. (2020). "Federated Learning in Mobile Edge Networks: A Comprehensive Survey". *IEEE Communications Surveys Tutorials*.

Lin, R., C. Yu, B. Han, and T. Liu. (2024a). "On the Over-Memorization During Natural, Robust and Catastrophic Overfitting". In: *ICLR*.

Lin, R., C. Yu, B. Han, H. Su, and T. Liu. (2024b). "Layer-Aware Analysis of Catastrophic Overfitting: Revealing the Pseudo-Robust Shortcut Dependency". In: *ICML*.

Lin, R., C. Yu, and T. Liu. (2023a). "Eliminating catastrophic overfitting via abnormal adversarial examples regularization". In: *NeurIPS*.

Lin, T.-Y., P. Goyal, R. Girshick, K. He, and P. Dollár. (2017). "Focal loss for dense object detection". In: *ICCV*.

Lin, V., L.-P. Morency, and E. Ben-Michael. (2023b). "Text-Transport: Toward Learning Causal Effects of Natural Language". In: *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Lin, Y., J. Zhang, Z. Huang, J. Liu, Z. Wen, and X. Peng. (2024c). "Multi-granularity correspondence learning from long-term noisy videos". *arXiv preprint arXiv:2401.16702*.

Lin, Y., L. Tan, Y. HAO, H. N. Wong, H. Dong, W. Zhang, Y. Yang, and T. Zhang. (2024d). "Spurious Feature Diversification Improves Out-of-distribution Generalization". In: *ICLR*.

Lin, Y., S. Zhu, L. Tan, and P. Cui. (2022). "ZIN: When and How to Learn Invariance Without Environment Partition?" In: *NeurIPS*.

Liu, A., L. Pan, Y. Lu, J. Li, X. Hu, X. Zhang, L. Wen, I. King, H. Xiong, and P. Yu. (2024a). "A survey of text watermarking in the era of large language models". *ACM Computing Surveys*.

Liu, B., M. Ding, S. Shaham, W. Rahayu, F. Farokhi, and Z. Lin. (2021a). "When machine learning meets privacy: A survey and outlook". *ACM CSUR*.

Liu, C., M. Salzmann, T. Lin, R. Tomioka, and S. Süsstrunk. (2020a). "On the Loss Landscape of Adversarial Training: Identifying Challenges and How to Overcome Them". In: *NeurIPS*.

Liu, C., Y. Chen, T. Liu, M. Gong, J. Cheng, B. Han, and K. Zhang. (2024b). "Discovery of the Hidden World with Large Language Models". In: *Proceedings of the Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Liu, E. Z., B. Haghgoo, A. S. Chen, A. Raghunathan, P. W. Koh, S. Sagawa, P. Liang, and C. Finn. (2021b). "Just Train Twice: Improving Group Robustness without Training Group Information". In: *ICML*.

Liu, F., Z. Xu, and H. Liu. (2024c). "Adversarial tuning: Defending against jailbreak attacks for llms". *arXiv preprint arXiv:2406.06622*.

Liu, F., K. Lin, L. Li, J. Wang, Y. Yacoob, and L. Wang. (2024d). "Mitigating hallucination in large multi-modal models via robust instruction tuning". In: *ICLR*.

Liu, G., T. Xu, R. Zhang, Z. Wang, C. Wang, and L. Liu. (2023a). "Gradient-leaks: Enabling black-box membership inference attacks against machine learning models". *IEEE TIFS*.

Liu, G., J. Wu, and Z.-H. Zhou. (2012). "Key instance detection in multi-instance learning". In: *ACML*.

Liu, H., W. Xue, Y. Chen, D. Chen, X. Zhao, K. Wang, L. Hou, R. Li, and W. Peng. (2024e). "A survey on hallucination in large vision-language models". *arXiv preprint arXiv:2402.00253*.

Liu, H., C. Li, Q. Wu, and Y. J. Lee. (2024f). "Visual instruction tuning". *Advances in neural information processing systems*. 36.

Liu, H., J. Z. HaoChen, A. Gaidon, and T. Ma. (2021c). "Self-supervised Learning is More Robust to Dataset Imbalance". In: *ICLR*.

Liu, J., Z. Hu, P. Cui, B. Li, and Z. Shen. (2021d). "Heterogeneous Risk Minimization". In: *ICML*.

Liu, K., B. Dolan-Gavitt, and S. Garg. (2018). "Fine-pruning: Defending against backdooring attacks on deep neural networks". In: *RAID*.

Liu, L., Y. Wang, G. Liu, K. Peng, and C. Wang. (2022a). "Membership inference attacks against machine learning models via prediction sensitivity". *IEEE TDSC*.

Liu, L., J. Zhang, S. Song, and K. B. Letaief. (2020b). "Client-Edge-Cloud Hierarchical Federated Learning". In: *2020 IEEE International Conference on Communications, ICC 2020, Dublin, Ireland, June 7-11, 2020*.

Liu, R., W. Zhou, J. Zhang, X. Liu, P. Si, and H. Li. (2023b). "Model Inversion Attacks on Homogeneous and Heterogeneous Graph Neural Networks". In: *SecureComm*.

Liu, S., J. Niles-Weed, N. Razavian, and C. Fernandez-Granda. (2020c). "Early-learning regularization prevents memorization of noisy labels". *Advances in neural information processing systems*. 33: 20331–20342.

Liu, S., Z. Zhu, Q. Qu, and C. You. (2022b). "Robust training under label noise by over-parameterization". In: *International Conference on Machine Learning*. PMLR. 14153–14172.

Liu, W., X. Wang, J. D. Owens, and Y. Li. (2020d). "Energy-based Out-of-distribution Detection". In: *NeurIPS*.

Liu, X., N. Xu, M. Chen, and C. Xiao. (2024g). "AutoDAN: Generating Stealthy Jailbreak Prompts on Aligned Large Language Models". In: *ICLR*.

Liu, Z., Z. Wang, L. Xu, J. Wang, L. Song, T. Wang, C. Chen, W. Cheng, and J. Bian. (2024h). "Protecting your llms with information bottleneck". In: *NeurIPS*.

Liu, Z., Z. Miao, X. Zhan, J. Wang, B. Gong, and S. X. Yu. (2019). "Large-scale long-tailed recognition in an open world". In: *CVPR*.

Locatello, F., S. Bauer, M. Lucic, G. Raetsch, S. Gelly, B. Schölkopf, and O. Bachem. (2019). "Challenging common assumptions in the unsupervised learning of disentangled representations". In: *international conference on machine learning*.

Lu, Y., Y. Zhang, B. Han, Y. Cheung, and H. Wang. (2023). "Label-Noise Learning with Intrinsically Long-Tailed Data". In: *ICCV*.

Luo, J., F. Hong, J. Yao, B. Han, Y. Zhang, and Y. Wang. (2024). "Revive Re-weighting in Imbalanced Learning by Density Ratio Estimation". In: *NeurIPS*.

Lv, F., J. Liang, S. Li, B. Zang, C. H. Liu, Z. Wang, and D. Liu. (2022). "Causality inspired representation learning for domain generalization". In: *CVPR*.

Ma, X., B. Li, Y. Wang, S. M. Erfani, S. N. R. Wijewickrema, G. Schoenebeck, D. Song, M. E. Houle, and J. Bailey. (2018). "Characterizing Adversarial Subspaces Using Local Intrinsic Dimensionality". In: *ICLR*.

Ma, X., M. Yang, Y. Li, P. Hu, J. Lv, and X. Peng. (2024). "Cross-modal Retrieval with Noisy Correspondence via Consistency Refining and Mining". *IEEE Transactions on Image Processing*.

Maaz, M., H. Rasheed, S. Khan, and F. S. Khan. (2023). "Video-chatgpt: Towards detailed video understanding via large vision and language models". *arXiv preprint arXiv:2306.05424*.

Madry, A., A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. (2018). "Towards Deep Learning Models Resistant to Adversarial Attacks". In: *ICLR*.

Mahajan, D., S. Tople, and A. Sharma. (2021). "Domain Generalization using Causal Matching". In: *ICML*. Vol. 139.

Mayilvahanan, P., T. Wiedemer, E. Rusak, M. Bethge, and W. Brendel. (2024). "Does CLIP's generalization performance mainly stem from high train-test similarity?" In: *ICLR*.

Mehrabi, N., F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan. (2021). "A survey on bias and fairness in machine learning". *ACM Computing Surveys (CSUR)*. 54(6): 1–35.

Meng, D. and H. Chen. (2017). "Magnet: a two-pronged defense against adversarial examples". In: *ACM SIGSAC*.

Menon, A. K., S. Jayasumana, A. S. Rawat, H. Jain, A. Veit, and S. Kumar. (2020). "Long-tail learning via logit adjustment". *arXiv preprint arXiv:2007.07314*.

Miao, S., M. Liu, and P. Li. (2022). "Interpretable and Generalizable Graph Learning via Stochastic Attention Mechanism". In: *ICML*.

Min, R., S. Li, H. Chen, and M. Cheng. (2024). "A watermark-conditioned diffusion model for ip protection". *arXiv preprint arXiv:2403.10893*.

Ming, Y., Z. Cai, J. Gu, Y. Sun, W. Li, and Y. Li. (2022a). "Delving into Out-of-Distribution Detection with Vision-Language Representations". In: *NeurIPS*.

Ming, Y., Y. Fan, and Y. Li. (2022b). "Poem: Out-of-distribution detection with posterior sampling". In: *ICML*.

Miotto, R., F. Wang, S. Wang, X. Jiang, and J. T. Dudley. (2018). "Deep learning for healthcare: review, opportunities and challenges". *Briefings in bioinformatics*.

Mirza, M. and S. Osindero. (2014). "Conditional generative adversarial nets". *arXiv preprint arXiv:1411.1784*.

Miyato, T., S. Maeda, M. Koyama, and S. Ishii. (2019). "Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning". *IEEE Trans. Pattern Anal. Mach. Intell.* 41(8): 1979–1993.

Morris, J., V. Kuleshov, V. Shmatikov, and A. Rush. (2023). "Text embeddings reveal (almost) as much as text". In: *EMNLP*.

Muñoz-González, L., B. Biggio, A. Demontis, A. Paudice, V. Wongrassamee, E. C. Lupu, and F. Roli. (2017). "Towards poisoning of deep learning algorithms with back-gradient optimization". In: *AISec*.

Murdoch, W. J., C. Singh, K. Kumbier, R. Abbasi-Asl, and B. Yu. (2019). "Definitions, methods, and applications in interpretable machine learning". *Proceedings of the National Academy of Sciences*. 116(44): 22071–22080.

Namkoong, H. and J. C. Duchi. (2016). "Stochastic Gradient Methods for Distributionally Robust Optimization with f-divergences". In: *NeurIPS*.

Natarajan, N., I. S. Dhillon, P. Ravikumar, and A. Tewari. (2013). "Learning with Noisy Labels". In: *NeurIPS*.

Neel, S., A. Roth, G. Vietri, and S. Wu. (2020). "Oracle efficient private non-convex optimization". In: *ICML*.

Nelson, B., M. Barreno, F. J. Chi, A. D. Joseph, B. I. Rubinstein, U. Saini, C. Sutton, J. D. Tygar, and K. Xia. (2008). "Exploiting machine learning to subvert your spam filter." *LEET*.

Nguyen, B., K. Chandrasegaran, M. Abdollahzadeh, and N. Cheung. (2024). "Label-Only Model Inversion Attacks via Knowledge Transfer". In: *NeurIPS*.

Nguyen, N., K. Chandrasegaran, M. Abdollahzadeh, and N. Cheung. (2023). "Re-thinking model inversion attacks against deep neural networks". In: *CVPR*.

Nguyen, T. T., T. T. Huynh, Z. Ren, P. L. Nguyen, A. W.-C. Liew, H. Yin, and Q. V. H. Nguyen. (2022). "A survey of machine unlearning". *arXiv preprint arXiv:2209.02299*.

Nie, J., Y. Zhang, Z. Fang, T. Liu, B. Han, and X. Tian. (2024). "Out-of-Distribution Detection with Negative Prompts". In: *ICLR*.

Nie, W., B. Guo, Y. Huang, C. Xiao, A. Vahdat, and A. Anandkumar. (2022). "Diffusion Models for Adversarial Purification". In: *ICML*.

Nishio, T. and R. Yonetani. (2019). "Client Selection for Federated Learning with Heterogeneous Resources in Mobile Edge". In: *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*.

Niu, J., P. Liu, X. Zhu, K. Shen, Y. Wang, H. Chi, Y. Shen, X. Jiang, J. Ma, and Y. Zhang. (2024). "A survey on membership inference attacks and defenses in Machine Learning". *Journal of Information and Intelligence*.

Niu, S., Y. Liu, J. Wang, and H. Song. (2020). "A decade survey of transfer learning (2010–2020)". *IEEE Transactions on Artificial Intelligence*.

NT, H., C. J. Jin, and T. Murata. (2019). "Learning graph neural networks with noisy labels". In: *ICLR Learning from Limited Labeled Data Workshop*.

Olatunji, I., M. Rathee, T. Funke, and M. Khosla. (2023). "Private graph extraction via feature explanations". *PETS*.

Oliynyk, D., R. Mayer, and A. Rauber. (2023). "I know what you trained last summer: A survey on stealing machine learning models and defences". *ACM Computing Surveys*.

Oord, A. van den, Y. Li, and O. Vinyals. (2018). "Representation learning with contrastive predictive coding". *arXiv preprint arXiv:1807.03748*.

Ozbayoglu, A. M., M. U. Gudelek, and O. B. Sezer. (2020). "Deep learning for financial applications: A survey". *Applied soft computing*.

Pang, T., H. Zhang, D. He, Y. Dong, H. Su, W. Chen, J. Zhu, and T. Liu. (2022). "Two Coupled Rejection Metrics Can Tell Adversarial Examples Apart". In: *CVPR*.

Papyan, V., X. Han, and D. L. Donoho. (2020). "Prevalence of neural collapse during the terminal phase of deep learning training". *Proceedings of the National Academy of Sciences*. 117(40): 24652–24663.

Parascandolo, G., A. Neitz, A. Orvieto, L. Gresele, and B. Schölkopf. (2021). "Learning explanations that are hard to vary". In: *ICLR*.

Parashar, S., Z. Lin, T. Liu, X. Dong, Y. Li, D. Ramanan, J. Caverlee, and S. Kong. (2024). "The Neglected Tails in Vision-Language Models". In: *CVPR*.

Parikh, R., C. Dupuy, and R. Gupta. (2022). "Canary extraction in natural language understanding models". *arXiv preprint arXiv:2203.13920*.

Park, Y., D.-J. Han, D.-Y. Kim, J. Seo, and J. Moon. (2021). "Few-round learning for federated learning". *Advances in Neural Information Processing Systems*.

Penedo, G., Q. Malartic, D. Hesslow, R. Cojocaru, H. Alobeidli, A. Cappelli, B. Pannier, E. Almazrouei, and J. Launay. (2023). "The RefinedWeb dataset for Falcon LLM: Outperforming curated corpora with web data only". *Advances in Neural Information Processing Systems*. 36: 79155–79172.

Peng, B., S. Qu, Y. Wu, T. Zou, L. He, A. Knoll, G. Chen, and C. Jiang. (2024a). "MAP: MAsk-Pruning for Source-Free Model Intellectual Property Protection". In: *CVPR*.

Peng, M. and Q. Zhang. (2019). "Address instance-level label prediction in multiple instance learning". *arXiv preprint arXiv:1905.12226*.

Peng, X., B. Han, F. Liu, T. Liu, and M. Zhou. (2024b). "Pseudo-Private Data Guided Model Inversion Attacks". In: *NeurIPS*.

Pessach, D. and E. Shmueli. (2022). "A review on fairness in machine learning". *ACM Computing Surveys (CSUR)*. 55(3): 1–44.

Peters, J., P. Bühlmann, and N. Meinshausen. (2016). "Causal inference by using invariant prediction: identification and confidence intervals". *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*.

Peters, J., D. Janzing, and B. Schlkopf. (2017a). *Elements of Causal Inference: Foundations and Learning Algorithms*. The MIT Press.

Peters, J., D. Janzing, and B. Schölkopf. (2017b). *Elements of causal inference: foundations and learning algorithms*. The MIT Press.

Pezeshki, M., D. Bouchacourt, M. Ibrahim, N. Ballas, P. Vincent, and D. Lopez-Paz. (2024). "Discovering environments with XRM". In: *ICML*.

Phan, N., M. Vu, Y. Liu, R. Jin, D. Dou, X. Wu, and M. T. Thai. (2019). "Heterogeneous gaussian mechanism: Preserving differential privacy in deep learning with provable robustness". In: *IJCAI*.

Phan, N., Y. Wang, X. Wu, and D. Dou. (2016). "Differential privacy preservation for deep auto-encoders: an application of human behavior prediction". In: *AAAI*.

Qi, X., K. Huang, A. Panda, M. Wang, and P. Mittal. (2023). "Visual adversarial examples jailbreak large language models". *arXiv preprint arXiv:2306.13213*.

Qi, X., Y. Zeng, T. Xie, P.-Y. Chen, R. Jia, P. Mittal, and P. Henderson. (2024). "Fine-tuning Aligned Language Models Compromises Safety, Even When Users Do Not Intend To!" In: *ICLR*.

Qiang, Y., X. Zhou, S. Z. Zade, M. A. Roshani, P. Khanduri, D. Zytko, and D. Zhu. (2024). "Learning to poison large language models during instruction tuning". *arXiv preprint arXiv:2402.13459*.

Qin, Y., D. Peng, X. Peng, X. Wang, and P. Hu. (2022). "Deep evidential learning with noisy correspondence for cross-modal retrieval". In: *Proceedings of the 30th ACM International Conference on Multimedia*. 4948–4956.

Qin, Y., Y. Sun, D. Peng, J. T. Zhou, X. Peng, and P. Hu. (2023). "Cross-modal active complementary learning with self-refining correspondence". *Advances in Neural Information Processing Systems*. 36: 24829–24840.

Qu, Z., X. Li, R. Duan, Y. Liu, B. Tang, and Z. Lu. (2022). "Generalized federated learning via sharpness aware minimization". In: *International conference on machine learning*.

Radford, A., J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.* (2021a). "Learning transferable visual models from natural language supervision". In: *International conference on machine learning*. PMLR. 8748–8763.

Radford, A., J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.* (2021b). "Learning transferable visual models from natural language supervision". In: *ICML*.

Radford, A., J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, *et al.* (2019). "Language models are unsupervised multitask learners". *OpenAI blog*. 1(8): 9.

Raghuram, J., V. Chandrasekaran, S. Jha, and S. Banerjee. (2021). "A General Framework For Detecting Anomalous Inputs to DNN Classifiers". In: *ICML*.

Rame, A., C. Dancette, and M. Cord. (2022a). "Fishr: Invariant Gradient Variances for Out-of-distribution Generalization". In: *ICML*.

Rame, A., M. Kirchmeyer, T. Rahier, A. Rakotomamonjy, patrick gallinari, and M. Cord. (2022b). "Diverse Weight Averaging for Out-of-Distribution Generalization". In: *NeurIPS*.

Rawte, V., A. Sheth, and A. Das. (2023). "A survey of hallucination in large foundation models". *arXiv preprint arXiv:2309.05922*.

Reddi, S., Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konečnỳ, S. Kumar, and H. B. McMahan. (2020). "Adaptive Federated Optimization". *arXiv preprint arXiv:2003.00295*.

Reed, W. J. (2001). "The Pareto, Zipf and other power laws". *Economics letters*.

Ren, J., G. Yu, and G. Ding. (2021). "Accelerating DNN Training in Wireless Federated Edge Learning Systems". *IEEE Journal on Selected Areas in Communications*.

Ren, R., Y. Wang, Y. Qu, W. X. Zhao, J. Liu, H. Tian, H. Wu, J.-R. Wen, and H. Wang. (2023). "Investigating the factual knowledge boundary of large language models with retrieval augmentation". *arXiv preprint arXiv:2307.11019*.

Rice, L., E. Wong, and Z. Kolter. (2020). "Overfitting in adversarially robust deep learning". In: *ICML*.

Robey, A., E. Wong, H. Hassani, and G. J. Pappas. (2023). "Smoothllm: Defending large language models against jailbreaking attacks". *arXiv preprint arXiv:2310.03684*.

Rojas-Carulla, M., B. Schölkopf, R. Turner, and J. Peters. (2018). "Invariant Models for Causal Transfer Learning". *Journal of Machine Learning Research*.

Rombach, R., A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. (2022). "High-resolution image synthesis with latent diffusion models". In: *CVPR*.

Rosenfeld, E., P. Ravikumar, and A. Risteski. (2022). "Domain-Adjusted Regression or: ERM May Already Learn Features Sufficient for Out-of-Distribution Generalization". *arXiv preprint arXiv:2202.06856*.

Sadeghi, B., S. Dehdashtian, and V. Boddeti. (2022). "On Characterizing the Trade-off in Invariant Representation Learning". *TMLR*.

Sagawa, S., P. W. Koh, T. B. Hashimoto, and P. Liang. (2020). "Distributionally Robust Neural Networks". In: *ICLR*.

Saha, A., A. Subramanya, and H. Pirsiavash. (2020). “Hidden trigger backdoor attacks”. In: *AAAI*.

Samangouei, P., M. Kabkab, and R. Chellappa. (2018). “Defense-GAN: Protecting Classifiers Against Adversarial Attacks Using Generative Models”. In: *ICLR*.

Santurkar, S., Y. Dubois, R. Taori, P. Liang, and T. Hashimoto. (2023). “Is a Caption Worth a Thousand Images? A Study on Representation Learning”. In: *ICLR*.

Schmidt, L., S. Santurkar, D. Tsipras, K. Talwar, and A. Madry. (2018). “Adversarially robust generalization requires more data”. In: *NeurIPS*.

Schneider, S., A. Baevski, R. Collobert, and M. Auli. (2019). “wav2vec: Unsupervised pre-training for speech recognition”. *arXiv preprint arXiv:1904.05862*.

Schölkopf, B., F. Locatello, S. Bauer, N. R. Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio. (2021). “Toward causal representation learning”. *Proceedings of the IEEE*.

Schryen, G. and R. Kadura. (2009). “Open source vs. closed source software: towards measuring security”. In: *Proceedings of the 2009 ACM symposium on Applied Computing*. 2016–2023.

Schulman, J. (2023). “Reinforcement learning from human feedback: Progress and challenges”. In: *Berkeley EECS Colloquium. YouTube www. youtube. com/watch*.

Schwartz, I. S., K. E. Link, R. Daneshjou, and N. Cortés-Penfield. (2024). “Black box warning: large language models and the future of infectious diseases consultation”. *Clinical infectious diseases*. 78(4): 860–866.

Sehwag, V., M. Chiang, and P. Mittal. (2021). “SSD: A Unified Framework for Self-Supervised Outlier Detection”. In: *ICLR*.

Shafahi, A., M. Najibi, A. Ghiasi, Z. Xu, J. P. Dickerson, C. Studer, L. S. Davis, G. Taylor, and T. Goldstein. (2019). “Adversarial training for free!” In: *NeurIPS*.

Shaik, T., X. Tao, H. Xie, L. Li, X. Zhu, and Q. Li. (2023). “Exploring the Landscape of Machine Unlearning: A Survey and Taxonomy”. *arXiv preprint arXiv:2305.06360*.

Sharma, P., N. Ding, S. Goodman, and R. Soricut. (2018). "Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2556–2565.

Shen, L., Z. Tang, L. Wu, Y. Zhang, X. Chu, T. Qin, and B. Han. (2024). "Hot Pluggable Federated Learning". In: *International Workshop on Federated Foundation Models in Conjunction with NeurIPS 2024*.

Shen, Y., Y. Han, Z. Zhang, M. Chen, T. Yu, M. Backes, Y. Zhang, and G. Stringhini. (2022). "Finding mnemon: Reviving memories of node embeddings". In: *CCS*.

Shi, C., C. Holtz, and G. Mishne. (2021). "Online Adversarial Purification based on Self-supervised Learning". In: *ICLR*.

Shi, W., S. Min, M. Yasunaga, M. Seo, R. James, M. Lewis, L. Zettlemoyer, and W.-t. Yih. (2023). "Replug: Retrieval-augmented black-box language models". *arXiv preprint arXiv:2301.12652*.

Shi, Y., J. Seely, P. Torr, S. N, A. Hannun, N. Usunier, and G. Synnaeve. (2022). "Gradient Matching for Domain Generalization". In: *ICLR*.

Shibly, K. H., M. D. Hossain, H. Inoue, Y. Taenaka, and Y. Kadobayashi. (2023). "Towards Autonomous Driving Model Resistant to Adversarial Attack". *Appl. Artif. Intell.*

Shokri, R., M. Stronati, C. Song, and V. Shmatikov. (2017). "Membership inference attacks against machine learning models". In: *SP*.

Singh, A., T.-W. Ngan, P. Druschel, and D. Wallach. (2006). "Eclipse Attacks on Overlay Networks: Threats and Defenses". In: *IEEE INFOCOM*.

Smith, V., C.-K. Chiang, M. Sanjabi, and A. Talwalkar. (2018). "Federated Multi-Task Learning".

Sohn, K., D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. A. Raffel, E. D. Cubuk, A. Kurakin, and C.-L. Li. (2020). "Fixmatch: Simplifying semi-supervised learning with consistency and confidence". In: *NeurIPS*.

Song, C. and A. Raghunathan. (2020). "Information leakage in embedding models". In: *CCS*.

Song, H., M. Kim, D. Park, Y. Shin, and J.-G. Lee. (2022a). "Learning from noisy labels with deep neural networks: A survey". *IEEE TNNLS*.

Song, J. and D. Namiot. (2022). "A survey of the implementations of model inversion attacks". In: *DCCN*.

Song, S., K. Chaudhuri, and A. D. Sarwate. (2013). "Stochastic gradient descent with differentially private updates". In: *GlobalSIP*.

Song, Y. and S. Ermon. (2019). "Generative Modeling by Estimating Gradients of the Data Distribution". In: *NeurIPS*.

Song, Y., N. Sebe, and W. Wang. (2022b). "Rankfeat: Rank-1 feature removal for out-of-distribution detection". In:

Spirtes, P., C. Glymour, and R. Scheines. (2001). *Causation, prediction, and search*. MIT press.

Sprague, M. R., A. Jalalirad, M. Scavuzzo, C. Capota, M. Neun, L. Do, and M. Kopp. (2019). "Asynchronous Federated Learning for Geospatial Applications". In: *ECML PKDD 2018 Workshops*.

Strohmer, T. and R. W. Heath Jr. (2003). "Grassmannian frames with applications to coding and communication". *Applied and computational harmonic analysis*. 14(3): 257–275.

Struppek, L., D. Hintersdorf, A. Correia, A. Adler, and K. Kersting. (2022). "Plug and Play Attacks: Towards Robust and Flexible Model Inversion Attacks". In: *ICML*.

Stutz, D., M. Hein, and B. Schiele. (2020). "Confidence-Calibrated Adversarial Training: Generalizing to Unseen Attacks". In: *ICML*.

Sun, B. and K. Saenko. (2016). "Deep CORAL: Correlation Alignment for Deep Domain Adaptation". In: *ECCV*.

Sun, T., X. Zhang, Z. He, P. Li, Q. Cheng, X. Liu, H. Yan, Y. Shao, Q. Tang, S. Zhang, *et al.* (2024). "MOSS: An Open Conversational Large Language Model". *Machine Intelligence Research*: 1–18.

Sun, W., S. Lei, L. Wang, Z. Liu, and Y. Zhang. (2020). "Adaptive Federated Learning and Digital Twin for Industrial Internet of Things". *IEEE Transactions on Industrial Informatics*.

Sun, Y., C. Guo, and Y. Li. (2021). "ReAct: Out-of-distribution Detection With Rectified Activations". In: *NeurIPS*.

Sun, Y. and Y. Li. (2022). "DICE: Leveraging Sparsification for Out-of-Distribution Detection". In: *ECCV*.

Sun, Y., Y. Ming, X. Zhu, and Y. Li. (2022a). "Out-of-distribution Detection with Deep Nearest Neighbors". *ICML*.

Sun, Z., X. Du, F. Song, M. Ni, and L. Li. (2022b). "Coprotector: Protect open-source code against unauthorized training usage with data poisoning". In: *WWW*.

Szegedy, C., W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus. (2014). "Intriguing properties of neural networks". In: *ICLR*.

Tack, J., S. Mo, J. Jeong, and J. Shin. (2020). "CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances". In: *NeurIPS*.

Tan, X., L. Yong, S. Zhu, C. Qu, X. Qiu, X. Yinghui, P. Cui, and Y. Qi. (2023). "Provably Invariant Learning without Domain Information". In: *ICML*.

Tang, Z., X. Chu, R. Y. Ran, S. Lee, S. Shi, Y. Zhang, Y. Wang, A. Q. Liang, S. Avestimehr, and C. He. (2023). "FedML Parrot: A Scalable Federated Learning System via Heterogeneity-aware Scheduling on Sequential and Hierarchical Training". *arXiv preprint arXiv:2303.01778*.

Tang, Z., X. Kang, Y. Yin, X. Pan, Y. Wang, X. He, Q. Wang, R. Zeng, K. Zhao, S. Shi, A. C. Zhou, B. Li, B. He, and X. Chu. (2024a). "FusionLLM: A Decentralized LLM Training System on Geo-distributed GPUs with Adaptive Compression".

Tang, Z., S. Shi, and X. Chu. (2020a). "Communication-efficient decentralized learning with sparsification and adaptive peer selection". In: *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*.

Tang, Z., S. Shi, X. Chu, W. Wang, and B. Li. (2020b). "Communication-efficient distributed deep learning: A comprehensive survey". *arXiv preprint arXiv:2003.06307*.

Tang, Z., S. Shi, B. Li, and X. Chu. (2022). "GossipFL: A Decentralized Federated Learning Framework with Sparsified and Adaptive Communication". *IEEE Transactions on Parallel and Distributed Systems*.

Tang, Z., Y. Zhang, P. Dong, Y.-m. Cheung, A. C. Zhou, B. Han, and X. Chu. (2024b). "FuseFL: One-Shot Federated Learning through the Lens of Causality with Progressive Model Fusion". In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Tang, Z., J. Huang, R. Yan, Y. Wang, Z. Tang, S. Shi, A. C. Zhou, and X. Chu. (2024c). "Bandwidth-Aware and Overlap-Weighted Compression for Communication-Efficient Federated Learning". In: *53rd International Conference on Parallel Processing*.

Tao, Z. and Q. Li. (2018). "eSGD: Communication Efficient Distributed Deep Learning on the Edge". In: *USENIX Workshop on Hot Topics in Edge Computing (HotEdge 18)*.

Team, G., P. Georgiev, V. I. Lei, R. Burnell, L. Bai, A. Gulati, G. Tanzer, D. Vincent, Z. Pan, S. Wang, *et al.* (2024). "Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context". *arXiv preprint arXiv:2403.05530*.

Teney, D., Y. Lin, S. J. Oh, and E. Abbasnejad. (2023). "ID and OOD Performance Are Sometimes Inversely Correlated on Real-world Datasets". In: *NeurIPS*.

Thudi, A., G. Deza, V. Chandrasekaran, and N. Papernot. (2022a). "Unrolling sgd: Understanding factors influencing machine unlearning". In: *IEEE EuroS&P*.

Thudi, A., H. Jia, I. Shumailov, and N. Papernot. (2022b). "On the necessity of auditable algorithmic definitions for machine unlearning". In: *USENIX Security*.

Tian, Y., O. J. Henaff, and A. van den Oord. (2021). "Divide and contrast: Self-supervised learning from uncurated data". In: *ICCV*.

Tolpegin, V., S. Truex, M. E. Gursoy, and L. Liu. (2020). "Data poisoning attacks against federated learning systems". In: *ESORICs*.

Touvron, H., T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, *et al.* (2023). "Llama: Open and efficient foundation language models". *arXiv preprint arXiv:2302.13971*.

Uchida, Y., Y. Nagai, S. Sakazawa, and S. Satoh. (2017). "Embedding watermarks into deep neural networks". In: *ICMR*.

Van Horn, G., O. Mac Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, and S. Belongie. (2018). "The inaturalist species classification and detection dataset". In: *CVPR*.

Vapnik, V. (1991). "Principles of Risk Minimization for Learning Theory". In: *NIPS*.

Vashishtha, A., A. G. Reddy, A. Kumar, S. Bachu, V. N. Balasubramanian, and A. Sharma. (2023). "Causal inference using llm-guided discovery". *arXiv preprint arXiv:2310.15117*.

Verma, A., S. Krishna, S. Gehrmann, M. Seshadri, A. Pradhan, T. Ault, L. Barrett, D. Rabinowitz, J. Doucette, and N. Phan. (2024). "Operationalizing a threat model for red-teaming large language models (llms)". *arXiv preprint arXiv:2407.14937*.

Wald, Y., A. Feder, D. Greenfeld, and U. Shalit. (2021). "On Calibration and Out-of-Domain Generalization". In: *NeurIPS*.

Wang, H., M. Xia, Y. Li, Y. Mao, L. Feng, G. Chen, and J. Zhao. (2022a). "SoLar: Sinkhorn Label Refinery for Imbalanced Partial-Label Learning". In: *NeurIPS*.

Wang, H., Z. Li, L. Feng, and W. Zhang. (2022b). "ViM: Out-Of-Distribution with Virtual-logit Matching". In: *CVPR*.

Wang, H., H. Chi, W. Yang, Z. Lin, M. Geng, L. Lan, J. Zhang, and D. Tao. (2023a). "Domain Specified Optimization for Deployment Authorization". In: *CVPR*.

Wang, H., Y. Li, H. Yao, and X. Li. (2023b). "CLIPN for Zero-Shot OOD Detection: Teaching CLIP to Say No". *ICCV*.

Wang, J., Y. Wang, G. Xu, J. Zhang, Y. Gu, H. Jia, M. Yan, J. Zhang, and J. Sang. (2023c). "An llm-free multi-dimensional benchmark for mllms hallucination evaluation". *arXiv preprint arXiv:2311.07397*.

Wang, K., Y. Fu, K. Li, A. Khisti, R. Zemel, and A. Makhzani. (2021a). "Variational model inversion attacks". In: *NeurIPS*.

Wang, K., C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y. Wang. (2017). "Generative adversarial networks: introduction and outlook". *IEEE/CAA JAS*.

Wang, L., M. Wang, D. Zhang, and H. Fu. (2023d). "Model Barrier: A Compact Un-Transferable Isolation Domain for Model Intellectual Property Protection". In: *CVPR*.

Wang, L., S. Xu, R. Xu, X. Wang, and Q. Zhu. (2022c). "Non-transferable learning: A new approach for model ownership verification and applicability authorization". In: *ICLR*.

Wang, Q., Z. Fang, Y. Zhang, F. Liu, Y. Li, and B. Han. (2023e). "Learning to augment distributions for out-of-distribution detection". *NeurIPS*.

Wang, Q., B. Han, P. Yang, J. Zhu, T. Liu, and M. Sugiyama. (2024a). "Unlearning with Control: Assessing Real-world Utility for Large Language Model Unlearning". *arXiv preprint arXiv:2406.09179*.

Wang, Q., Y. Lin, Y. Chen, L. Schmidt, B. Han, and T. Zhang. (2024b). "A Sober Look at the Robustness of CLIPs to Spurious Features". In: *NeurIPS*.

Wang, Q., F. Liu, B. Han, T. Liu, C. Gong, G. Niu, M. Zhou, and M. Sugiyama. (2021b). "Probabilistic Margins for Instance Reweighting in Adversarial Training". In: *NeurIPS*.

Wang, Q., J. Yao, C. Gong, T. Liu, M. Gong, H. Yang, and B. Han. (2021c). "Learning with group noise". In: *AAAI*.

Wang, S. and M. Ji. (2022). "A Unified Analysis of Federated Learning with Arbitrary Client Participation". In: *Advances in Neural Information Processing Systems*.

Wang, T., J.-Y. Zhu, A. Torralba, and A. A. Efros. (2020a). "Dataset Distillation".

Wang, X., J. Li, X. Kuang, Y.-a. Tan, and J. Li. (2019). "The security of machine learning in an adversarial setting: A survey". *Journal of Parallel and Distributed Computing*.

Wang, Y., L. Li, J. Yang, Z. Lin, and Y. Wang. (2023f). "Balance, imbalance, and rebalance: Understanding robust overfitting from a minimax game perspective". In: *NeurIPS*.

Wang, Y., D. Zou, J. Yi, J. Bailey, X. Ma, and Q. Gu. (2020b). "Improving Adversarial Robustness Requires Revisiting Misclassified Examples". In: *ICLR*.

Wang, Y., Y. Cao, J. Wu, R. Chen, and J. Chen. (2024c). "Tackling the Data Heterogeneity in Asynchronous Federated Learning with Cached Update Calibration". *International Conference on Learning Representations*.

Wang, Z., L. Shen, T. Liu, T. Duan, Y. Zhu, D. Zhan, D. Doermann, and M. Gao. (2024d). "Defending against Data-Free Model Extraction by Distributionally Robust Defensive Training". In: *NeurIPS*.

Wang, Z., D. Sun, S. Zhou, H. Wang, J. Fan, L. Huang, and J. Bu. (2024e). "NoisyGL: A Comprehensive Benchmark for Graph Neural Networks under Label Noise". In: *NeurIPS*.

Wang, Z., Q. Xu, Z. Yang, Y. He, X. Cao, and Q. Huang. (2024f). "A unified generalization analysis of re-weighting and logit-adjustment for imbalanced learning". *Advances in Neural Information Processing Systems*. 36.

Warnecke, A., L. Pirch, C. Wressnegger, and K. Rieck. (2023). "Machine unlearning of features and labels". *NDSS*.

Wei, H., R. Xie, H. Cheng, L. Feng, B. An, and Y. Li. (2022). "Mitigating neural network overconfidence with logit normalization". In: *ICML*.

Wei, J., Z. Zhu, G. Niu, T. Liu, S. Liu, M. Sugiyama, and Y. Liu. (2023a). "Fairness Improves Learning from Noisily Labeled Long-Tailed Data". arXiv: 2303.12291.

Wei, T., J. Shi, W. Tu, and Y. Li. (2021). "Robust Long-Tailed Learning under Label Noise". arXiv: 2108.11569.

Wei, X., X. Gong, Y. Zhan, B. Du, Y. Luo, and W. Hu. (2023b). "Clnode: Curriculum learning for node classification". In: *WSDM*.

Wei, Z., Y. Wang, A. Li, Y. Mo, and Y. Wang. (2023c). "Jailbreak and guard aligned language models with only few in-context demonstrations". *arXiv preprint arXiv:2310.06387*.

Wen, Y., J. Kirchenbauer, J. Geiping, and T. Goldstein. (2023). "Tree-ring watermarks: Fingerprints for diffusion images that are invisible and robust". *arXiv preprint arXiv:2305.20030*.

Wong, E., L. Rice, and J. Z. Kolter. (2020). "Fast is better than free: Revisiting adversarial training". In: *ICLR*.

Wu, D., S. Xia, and Y. Wang. (2020). "Adversarial Weight Perturbation Helps Robust Generalization". In: *NeurIPS*.

Wu, J., Y. Gaur, Z. Chen, L. Zhou, Y. Zhu, T. Wang, J. Li, S. Liu, B. Ren, L. Liu, *et al.* (2023). "On decoder-only architecture for speech-to-text and large language model integration". In: *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE. 1–8.

Wu, J., R. Hu, D. Li, Z. Huang, L. Ren, and Y. Zang. (2024). "Robust Heterophilic Graph Learning against Label Noise for Anomaly Detection". In: *IJCAI*.

Wu, T., Z. Liu, Q. Huang, Y. Wang, and D. Lin. (2021). "Adversarial robustness under long-tailed distribution". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8659–8668.

Xiao, H., H. Xiao, and C. Eckert. (2012). "Adversarial label flips attack on support vector machines". In: *ECAI*.

Xie, B., Y. Chen, J. Wang, K. Zhou, B. Han, W. Meng, and J. Cheng. (2024). "Enhancing Evolving Domain Generalization through Dynamic Latent Representations". In: *AAAI*.

Xie, C., K. Huang, P.-Y. Chen, and B. Li. (2019). "Dba: Distributed backdoor attacks against federated learning". In: *ICLR*.

Xie, Y., J. Yi, J. Shao, J. Curl, L. Lyu, Q. Chen, X. Xie, and F. Wu. (2023). "Defending ChatGPT against jailbreak attack via self-reminders". *Nature Machine Intelligence*.

Xie, Y., P. Li, C. Wu, and Q. Wu. (2021). "Differential privacy stochastic gradient descent with adaptive privacy budget allocation". In: *ICCECE*.

Xu, C., Y. Qu, Y. Xiang, and L. Gao. (2021a). "Asynchronous Federated Learning on Heterogeneous Devices: A Survey". *Computer Science Review*.

Xu, H., T. Zhu, L. Zhang, W. Zhou, and P. S. Yu. (2023). "Machine unlearning: A survey". *ACM Computing Surveys*.

Xu, H., L. Xiang, X. Ma, B. Yang, and B. Li. (2024a). "Hufu: A Modality-Agnositc Watermarking System for Pre-Trained Transformers via Permutation Equivariance". *arXiv preprint arXiv:2403.05842*.

Xu, J., F. Wang, M. D. Ma, P. W. Koh, C. Xiao, and M. Chen. (2024b). "Instructional fingerprinting of large language models". *arXiv preprint arXiv:2401.12255*.

Xu, J., H. Wang, and L. Chen. (2021b). "Bandwidth Allocation for Multiple Federated Learning Services in Wireless Edge Networks". *CoRR*.

Xu, X., K. Kong, N. Liu, L. Cui, D. Wang, J. Zhang, and M. Kankanhalli. (2024c). "An LLM can Fool Itself: A Prompt-Based Adversarial Attack". In: *ICLR*.

Xue, M., Y. Zhang, J. Wang, and W. Liu. (2021a). "Intellectual property protection for deep learning models: Taxonomy, methods, attacks, and evaluations". *IEEE Transactions on Artificial Intelligence*.

Xue, Y., C. Niu, Z. Zheng, S. Tang, C. Lyu, F. Wu, and G. Chen. (2021b). "Toward understanding the influence of individual clients in federated learning". In: *AAAI*.

Yan, H., S. Li, Y. Wang, Y. Zhang, K. Sharif, H. Hu, and Y. Li. (2022). "Membership inference attacks against deep learning models via logits distribution". *IEEE TDSC*.

Yang, C., Q. Wu, H. Li, and Y. Chen. (2017). "Generative poisoning attack method against neural networks". *arXiv preprint arXiv:1703.01340*.

Yang, H., X. Zhang, P. Khanduri, and J. Liu. (2022a). "Anarchic Federated Learning". In: *Proceedings of the 39th International Conference on Machine Learning*.

Yang, J., K. Zhou, Y. Li, and Z. Liu. (2024). "Generalized out-of-distribution detection: A survey". *IJCV*.

Yang, S., Z. Xu, K. Wang, Y. You, H. Yao, T. Liu, and M. Xu. (2023a). "Bicro: Noisy correspondence rectification for multi-modality data via bi-directional cross-modal similarity consistency". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 19883–19892.

Yang, Y., S. Chen, X. Li, L. Xie, Z. Lin, and D. Tao. (2022b). "Inducing neural collapse in imbalanced learning: Do we really need a learnable classifier at the end of deep neural network?" *Advances in neural information processing systems*. 35: 37991–38002.

Yang, Y., H. Zhang, D. Katabi, and M. Ghassemi. (2023b). "Change is Hard: A Closer Look at Subpopulation Shift". In: *ICML*.

Yang, Z., E. Chang, and Z. Liang. (2019). "Adversarial neural network inversion via auxiliary knowledge alignment". *arXiv preprint arXiv:1902.08552*.

Yao, T., Y. Chen, Z. Chen, K. Hu, Z. Shen, and K. Zhang. (2024a). "Empowering Graph Invariance Learning with Deep Spurious Infomax". In: *ICML*.

Yao, Y., J. Deng, X. Chen, C. Gong, J. Wu, and J. Yang. (2020). "Deep discriminative CNN with temporal ensembling for ambiguously-labeled image classification". In: *AAAI*.

Yao, Y., J. Duan, K. Xu, Y. Cai, Z. Sun, and Y. Zhang. (2024b). "A survey on large language model (llm) security and privacy: The good, the bad, and the ugly". *High-Confidence Computing*.

Ye, J., A. Borovykh, S. Hayou, and R. Shokri. (2023). "Leave-one-out distinguishability in machine learning". *arXiv preprint arXiv:2309.17310*.

Ye, J., A. Maddi, S. K. Murakonda, V. Bindschaedler, and R. Shokri. (2022). "Enhanced membership inference attacks against machine learning models". In: *ACM SIGSAC*.

Yeom, S., I. Giacomelli, M. Fredrikson, and S. Jha. (2018). "Privacy risk in machine learning: Analyzing the connection to overfitting". In: *IEEE CSF*.

Yi, S., Y. Liu, Z. Sun, T. Cong, X. He, J. Song, K. Xu, and Q. Li. (2024). "Jailbreak attacks and defenses against large language models: A survey". *arXiv preprint arXiv:2407.04295*.

Yin, X., X. Yu, K. Sohn, X. Liu, and M. Chandraker. (2019). "Feature transfer learning for face recognition with under-represented data". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 5704–5713.

Yoon, J., W. Jeong, G. Lee, E. Yang, and S. J. Hwang. (2021a). "Federated Continual Learning with Weighted Inter-client Transfer". In: *Proceedings of the 38th International Conference on Machine Learning*.

Yoon, J., S. J. Hwang, and J. Lee. (2021b). "Adversarial Purification with Score-based Generative Models". In: *ICML*.

Yu, C., B. Han, L. Shen, J. Yu, C. Gong, M. Gong, and T. Liu. (2022). "Understanding robust overfitting of adversarial training and beyond". In: *ICML*.

Yu, D., H. Zhang, W. Chen, J. Yin, and T.-Y. Liu. (2021). "Large scale private learning via low-rank reparametrization". In: *ICML*.

Yu, F. and M.-L. Zhang. (2016). "Maximum margin partial label learning". In: *ACML*.

Yu, Q., J. Li, L. Wei, L. Pang, W. Ye, B. Qin, S. Tang, Q. Tian, and Y. Zhuang. (2024). "Hallucidoctor: Mitigating hallucinatory toxicity in visual instruction data". In: *CVPR*.

Yu, T., E. Bagdasaryan, and V. Shmatikov. (2020). "Salvaging federated learning by local adaptation". *arXiv preprint arXiv:2002.04758*.

Yuan, J., X. Luo, Y. Qin, Y. Zhao, W. Ju, and M. Zhang. (2023a). "Learning on graphs under label noise". In: *ICASSP*.

Yuan, X., K. Chen, J. Zhang, W. Zhang, N. Yu, and Y. Zhang. (2023b). "Pseudo label-guided model inversion attack via conditional generative adversarial network". In: *AAAI*.

Yuan, Y., W. Jiao, W. Wang, J.-t. Huang, P. He, S. Shi, and Z. Tu. (2024). "Gpt-4 is too smart to be safe: Stealthy chat with llms via cipher". In: *ICLR*.

Yue, Z., L. Zhang, and Q. Jin. (2024). "Less is more: Mitigating multimodal hallucination from an eos decision perspective". *arXiv preprint arXiv:2402.14545*.

Zarifzadeh, S., P. Liu, and R. Shokri. (2024). "Low-Cost High-Power Membership Inference Attacks". In: *ICML*.

Zbontar, J., L. Jing, I. Misra, Y. LeCun, and S. Deny. (2021). "Barlow twins: Self-supervised learning via redundancy reduction". In: *ICML*.

Zečević, M., M. Willig, D. S. Dhami, and K. Kersting. (2023). "Causal parrots: Large language models may talk causality but are not causal". *arXiv preprint arXiv:2308.13067*.

Zeng, B., L. Wang, Y. Hu, Y. Xu, C. Zhou, X. Wang, Y. Yu, and Z. Lin. (2023). "Huref: Human-readable fingerprint for large language models". In: *NeurIPS*.

Zeng, G. and W. Lu. (2022). "Unsupervised Non-transferable Text Classification". In: *EMNLP*.

Zeng, Y., H. Lin, J. Zhang, D. Yang, R. Jia, and W. Shi. (2024a). "How johnny can persuade llms to jailbreak them: Rethinking persuasion to challenge ai safety by humanizing llms". In: *ACL*.

Zeng, Y., Y. Wu, X. Zhang, H. Wang, and Q. Wu. (2024b). "Autodefense: Multi-agent llm defense against jailbreak attacks". *arXiv preprint arXiv:2403.04783*.

Zhai, R., C. Dan, J. Z. Kolter, and P. K. Ravikumar. (2023). "Understanding Why Generalized Reweighting Does Not Improve Over ERM". In: *ICLR*.

Zhang, C., Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao. (2021a). "A survey on federated learning". *Knowledge-Based Systems*.

Zhang, C., S. Bengio, M. Hardt, B. Recht, and O. Vinyals. (2021b). "Understanding deep learning (still) requires rethinking generalization". *Communications of the ACM*.

Zhang, E., K. Wang, X. Xu, Z. Wang, and H. Shi. (2023a). "Forget-Me-Not: Learning to Forget in Text-to-Image Diffusion Models". *arXiv preprint arXiv:2211.08332*.

Zhang, H., Y. Yu, J. Jiao, E. P. Xing, L. E. Ghaoui, and M. I. Jordan. (2019). "Theoretically Principled Trade-off between Robustness and Accuracy". In: *ICML*.

Zhang, J., F. Liu, D. Zhou, J. Zhang, and T. Liu. (2024a). "Improving Accuracy-robustness Trade-off via Pixel Reweighted Adversarial Training". In: *ICML*.

Zhang, J., B. Chen, X. Cheng, H. T. T. Binh, and S. Yu. (2020a). "PoisonGAN: Generative poisoning attacks against federated learning in edge computing systems". *IEEE Internet of Things Journal*.

Zhang, J., D. Lopez-Paz, and L. Bottou. (2022a). "Rich Feature Construction for the Optimization-Generalization Dilemma". In:

Zhang, J., C. Chen, B. Li, L. Lyu, S. Wu, S. Ding, C. Shen, and C. Wu. (2022b). "Dense: Data-free one-shot federated learning". *Advances in Neural Information Processing Systems*.

Zhang, J., D. Chen, J. Liao, W. Zhang, H. Feng, G. Hua, and N. Yu. (2021c). "Deep model intellectual property protection via deep watermarking". *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zhang, J., J. Zhu, G. Niu, B. Han, M. Sugiyama, and M. S. Kankanhalli. (2021d). "Geometry-aware Instance-reweighted Adversarial Training". In: *ICLR*.

Zhang, J., Q. Fu, X. Chen, L. Du, Z. Li, G. Wang, S. Han, D. Zhang, *et al.* (2023b). "Out-of-distribution detection based on in-distribution data patterns memorization with modern hopfield energy". In: *ICLR*.

Zhang, M., N. S. Sohoni, H. R. Zhang, C. Finn, and C. Ré. (2022c). "Correct-N-Contrast: a Contrastive Approach for Improving Robustness to Spurious Correlations". In: *ICML*.

Zhang, M.-L., B.-B. Zhou, and X.-Y. Liu. (2016). "Partial label learning via feature-aware disambiguation". In: *SIGKDD*.

Zhang, M., O. Press, W. Merrill, A. Liu, and N. A. Smith. (2023c). "How language model hallucinations can snowball". *arXiv preprint arXiv:2305.13534*.

Zhang, R., S. Hidano, and F. Koushanfar. (2022d). "Text Revealer: Private Text Reconstruction via Model Inversion Attacks against Transformers". *arXiv preprint arXiv:2209.10505*.

Zhang, S., F. Liu, J. Yang, Y. Yang, C. Li, B. Han, and M. Tan. (2023d). "Detecting Adversarial Data by Probing Multiple Perturbations Using Expected Perturbation Score". In: *ICML*.

Zhang, S., Z. Li, S. Yan, X. He, and J. Sun. (2021e). "Distribution alignment: A unified framework for long-tail visual recognition". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2361–2370.

Zhang, T., H. Zheng, J. Yao, X. Wang, M. Zhou, Y. Zhang, and Y. Wang. (2024b). "Long-tailed diffusion models with oriented calibration". In: *ICLR*.

Zhang, Y., R. Jia, H. Pei, W. Wang, B. Li, and D. Song. (2020b). "The secret revealer: Generative model-inversion attacks against deep neural networks". In: *CVPR*.

Zhang, Y., B. Kang, B. Hooi, S. Yan, and J. Feng. (2023e). "Deep long-tailed learning: A survey". *IEEE TPAMI*.

Zhang, Y., B. Kang, B. Hooi, S. Yan, and J. Feng. (2023f). "Deep long-tailed learning: A survey". *TPAMI*.

Zhang, Y., M. Gong, T. Liu, G. Niu, X. Tian, B. Han, B. Schölkopf, and K. Zhang. (2022e). "CausalAdv: Adversarial Robustness through the Lens of Causality". In: *ICLR*.

Zhang, Y., Y. Li, L. Cui, D. Cai, L. Liu, T. Fu, X. Huang, E. Zhao, Y. Zhang, Y. Chen, *et al.* (2023g). "Siren's song in the AI ocean: a survey on hallucination in large language models". *arXiv preprint arXiv:2309.01219*.

Zhang, Z., M. Chen, M. Backes, Y. Shen, and Y. Zhang. (2022f). "Inference attacks against graph neural networks". In: *USENIX Security*.

Zhang, Z., Q. Liu, Z. Huang, H. Wang, C. Lu, C. Liu, and E. Chen. (2021f). "Graphmi: Extracting private graph data from graph neural networks". In: *IJCAI*.

Zhang, Z., H. Luo, L. Zhu, G. Lu, and H. T. Shen. (2022g). "Modality-invariant asymmetric networks for cross-modal hashing". *IEEE Transactions on Knowledge and Data Engineering*. 35(5): 5091–5104.

Zhang, Z., Q. Zhang, and J. Foerster. (2024c). "PARDEN, Can You Repeat That? Defending against Jailbreaks via Repetition". *arXiv preprint arXiv:2405.07932*.

Zhao, H., R. T. des Combes, K. Zhang, and G. J. Gordon. (2019). "On Learning Invariant Representations for Domain Adaptation". In: *ICML*.

Zhao, H., C. Dan, B. Aragam, T. S. Jaakkola, G. J. Gordon, and P. Ravikumar. (2022). "Fundamental Limits and Tradeoffs in Invariant Representation Learning". *Journal of Machine Learning Research*.

Zhao, M., B. An, W. Gao, and T. Zhang. (2017). "Efficient label contamination attacks against black-box learning models." In: *IJCAI*.

Zhao, R., X. Li, S. Joty, C. Qin, and L. Bing. (2023a). "Verify-and-Edit: A Knowledge-Enhanced Chain-of-Thought Framework". In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 5823–5840.

Zhao, X., P. Ananth, L. Li, and Y.-X. Wang. (2023b). "Provable robust watermarking for ai-generated text". *arXiv preprint arXiv:2306.17439*.

Zhao, X., Y.-X. Wang, and L. Li. (2023c). "Protecting language generation models via invisible watermarking". In: *ICML*.

Zhao, X., Y.-X. Wang, and L. Li. (2024a). "Watermarking for Large Language Model". *Tutorials of ACL*.

Zhao, X., X. Yang, T. Pang, C. Du, L. Li, Y.-X. Wang, and W. Y. Wang. (2024b). "Weak-to-strong jailbreaking on large language models". *arXiv preprint arXiv:2401.17256*.

Zhao, Y., L. Yan, W. Sun, G. Xing, C. Meng, S. Wang, Z. Cheng, Z. Ren, and D. Yin. (2023d). "Knowing what llms do not know: A simple yet effective self-detection method". *arXiv preprint arXiv:2310.17918*.

Zhao, Y., T. Pang, C. Du, X. Yang, N.-M. Cheung, and M. Lin. (2023e). "A recipe for watermarking diffusion models". *arXiv preprint arXiv:2303.10137*.

Zhao, Z., M. Chen, T. Dai, J. Yao, B. Han, Y. Zhang, and Y. Wang. (2024c). "Mitigating Noisy Correspondence by Geometrical Structure Consistency Learning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 27381–27390.

Zhen, L., P. Hu, X. Wang, and D. Peng. (2019). "Deep supervised cross-modal retrieval". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 10394–10403.

Zheng, H., L. Zhou, H. Li, J. Su, X. Wei, and X. Xu. (2024). "BEM: Balanced and Entropy-Based Mix for Long-Tailed Semi-Supervised Learning". In: *CVPR*.

Zhong, X., Y. HUANG, and C. Liu. (2023). "Towards Efficient Training and Evaluation of Robust Models against l_0 Bounded Adversarial Perturbations". In: *ICML*.

Zhou, C., Q. Li, C. Li, J. Yu, Y. Liu, G. Wang, K. Zhang, C. Ji, Q. Yan, L. He, *et al.* (2024a). "A comprehensive survey on pretrained foundation models: A history from bert to chatgpt". *International Journal of Machine Learning and Cybernetics*.

Zhou, Y., G. Pu, X. Ma, X. Li, and D. Wu. (2020). "Distilled one-shot federated learning". *arXiv preprint arXiv:2009.07999*.

Zhou, Y., X. Wu, B. Huang, J. Wu, L. Feng, and K. C. Tan. (2024b). "CausalBench: A Comprehensive Benchmark for Causal Learning Capability of Large Language Models". *arXiv preprint arXiv:2404.06349*.

Zhou, Z., C. Zhou, X. Li, J. Yao, Q. Yao, and B. Han. (2023a). "On Strengthening and Defending Graph Reconstruction Attack with Markov Chain Approximation". In: *ICML*.

Zhou, Z.-H. (2017). "A brief introduction to weakly supervised learning". *National Science Review*. 5(1): 44–53.

Zhou, Z., J. Yao, F. Hong, Y. Zhang, B. Han, and Y. Wang. (2023b). "Combating Representation Learning Disparity with Geometric Harmonization".

Zhou, Z., J. Yao, Y.-F. Wang, B. Han, and Y. Zhang. (2022). "Contrastive Learning with Boosted Memorization". In: *ICML*.

Zhu, C., W. R. Huang, H. Li, G. Taylor, C. Studer, and T. Goldstein. (2019). "Transferable clean-label poisoning attacks on deep neural nets". In: *ICML*.

Zhu, H., S. Liu, and F. Jiang. (2022). "Adversarial training of LSTM-ED based anomaly detection for complex time-series in cyber-physical-social systems". *Pattern Recognit. Lett.*

Zhu, J., B. Han, J. Yao, J. Xu, G. Niu, and M. Sugiyama. (2024a). "Decoupling the Class Label and the Target Concept in Machine Unlearning". *arXiv preprint arXiv:2406.08288*.

Zhu, X. and A. B. Goldberg. (2022). *Introduction to semi-supervised learning*. Springer Nature.

Zhu, X. J. (2005). "Semi-supervised learning literature survey".

Zhu, Y., L. Feng, Z. Deng, Y. Chen, R. Amor, and M. Witbrock. (2024b). "Robust Node Classification on Graph Data with Graph and Label Noise". In: *AAAI*.

Zhuang, F., Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He. (2020). "A comprehensive survey on transfer learning". *Proceedings of the IEEE*.

Zou, A., Z. Wang, J. Z. Kolter, and M. Fredrikson. (2023). "Universal and transferable adversarial attacks on aligned language models". *arXiv preprint arXiv:2307.15043*.