Towards Understanding and Practices of Ethical Artificial Intelligence

A Dissertation

Presented to

the Faculty of the School of Engineering and Applied Science

University of Virginia

In Partial Fulfillment

of the requirements for the Degree

Doctor of Philosophy (Computer Science)

by

Jianfeng Chi

August 2022

Abstract

The successes of machine learning (ML) and artificial intelligence (AI) models encourage their widespread deployments in high-stakes domains – from public transportation to social decision-making such as autonomous driving, criminal justice, and company hiring. Such widespread deployments call for assessing and addressing the ethical concerns of AI systems.

The thesis aims to develop practical techniques and theoretical understanding for building ethical AI systems. We divide the thesis into two parts. The first part of the thesis focuses on automatic information extraction using natural language processing (NLP) from policy documents. Policy documents are natural language documents about how different stakeholders (e.g., users and ML services providers) in internet services agree on how the services providers commit to the ethical usage of users' data. Specifically, we develop NLP techniques and benchmarks for privacy policies, a type of policy document describing the practices of using, sharing, and protecting users' data. Such developed NLP techniques could be extended to other natural language law documents describing ethical AI principles and help improve mutual trust among different parties.

The second part of the thesis focuses on the theoretical understanding and development of algorithmic interventions for ethical artificial intelligence. In particular, we study the fairness problems for various machine learning tasks, such as classification, regression, and sequential decision-making: (1) we provide bias mitigation techniques for text classification using contrastive representation learning; (2) we provide the theoretical understanding and mitigation techniques for accuracy disparity problem in regression; (3) we propose a fairness notion that requires long-term equality on expected utility for different demographic groups for sequential decision-making and develop methods to achieve the proposed fairness notion. In addition, we also study adversarial representation learning, a technique that has been widely used for algorithmic fairness, and its implications for information obfuscation.

We hope the research presented in the thesis will facilitate the practices of building ethical machine learning systems and help increase the understanding and trust among stakeholders towards the machine learning systems.

Approval Sheet

This dissertation is submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy (Computer Science)

Jianfeng Chi

This dissertation has been read and approved by the Examining Committee:

Yuan Tian, Advisor

Aidong Zhang, Chairperson

Yixin Sun, Committee member

Cong Shen, Committee member

Han Zhao, Committee member

Accepted for the School of Engineering and Applied Science:

Engineering Dean, Dean, School of Engineering and Applied Science

August 2022

To my wife and my family.

Acknowledgements

I am fortunate to have Yuan Tian as my Ph.D. advisor. She has supported me in all ways, encouraged me to explore the research directions I am interested in, and provided me with the freedom to pursue exciting research problems. Her kindness and supportiveness guided me through many difficult times during my Ph.D. study.

I am very grateful to Aidong Zhang, Yixin Sun, Cong Shen, and Han Zhao for serving as my thesis committee members. In particular, Han has been a great mentor: he helped me with research and life, giving me his insightful advice and thoughts. My Ph.D. research would not have been as productive without his help.

I also want to thank my collaborators and co-authors (in alphabetical order): Wasi Uddin Ahmad, Kai-Wei Chang, Xinyi Dai, David Evans, Geoffrey J Gordon, Tu Le, Sihang Liu, Thomas Norton, Emmanuel Owusu, Md Rizwan Parvez, William Shand, Jian Shen, Fnu Suya, Patrick Tague, Yizhou Wei, Xuwang Yin, Tong Yu, and Yaodong Yu. During my internship at Amazon and Facebook (Meta), I was lucky to work with Baris Coskun, Luca Melis, Wei Ding, Eric Ma, Ruben Sipos, and Ryan Gehring.

Last but most importantly, I would like to thank my parents for their unconditional support, love, and sacrifices during my whole life. Special thanks go to my caring and lovely wife for becoming my lifelong companion, bringing peace, love, and happiness to my world, and delivering a baby boy during the journey. Without you, it would be impossible to go through the hardships and difficult times in the last five years.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Recently, the increasing computing capabilities and larger datasets available for model training together have advanced the successes of ML and AI models in many areas, such as computer vision [3, 4], natural language processing [5, 6], and speech recognition [7]. The progress in machine learning and artificial intelligence also has led to automatic decision-making in many high-stakes domains, including but not limited to autonomous driving [8], company hiring [9], lending and loan management [10], and medical analysis [11].

Meanwhile, ML/AI models trained with users' data come with the risks of violating ethical norms in privacy [12–14] and fairness [15, 16]. As a result, concerns about ethical AI grow, attracting research attention to building ethical artificial intelligence systems. For example, [17] showed that it is possible to extract "secrets" such as social security numbers or passwords in machine learning models, and [18] demonstrated that ML models (e.g., word embeddings) could amplify biases present in training data.

However, it poses several challenges to the goal of building ethical AI systems. On the one hand, different parties (e.g., users, machine learning service providers, and other third parties such as law enforcement agencies) need to reach a consensus about the ethical principles in artificial intelligence before any AI/ML practitioners put them into practice. This process requires all parties to understand the laws and principles related to ethical AI for the tasks of interest [19–23]. On the other hand, enforcing ethical norms such as fairness or privacy might cause undesirable effects (e.g., accuracy loss) on the machine learning models [24–26]. Thus, it is imperative to (1) understand the expectations and needs of ethical AI systems among different stakeholders and (2) provide formal analysis and algorithmic intervention to satisfy ethical requirements such as fairness and privacy.

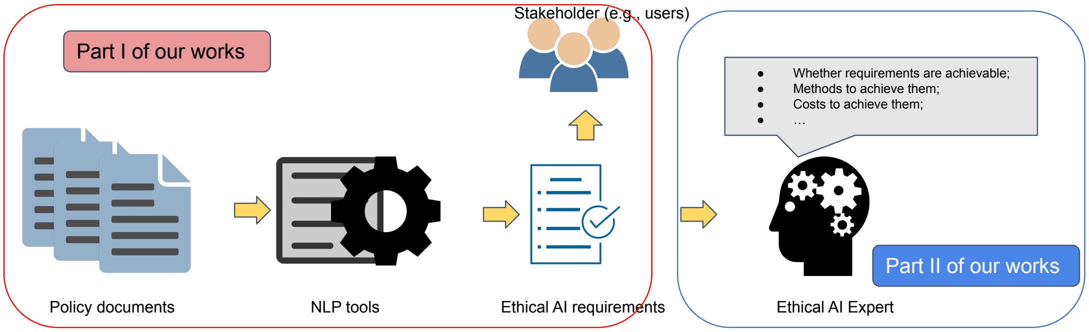


Figure 1.1: Connections between two parts of our works. The first part focuses on automatic information for natural language policy documents, and the second part provides theoretical understanding and algorithmic intervention for Ethical AI. The extracted Ethical AI requirements in part I could help ML/AI practitioners (e.g., ethical AI experts) better understand the needs and expectations of Ethical AI among the stakeholders and build a better ethical AI system. Note that though the privacy policies we use in the first part of the thesis are one type of policy documents most available online, the underlying developed NLP techniques could be naturally extended to other policy documents (e.g., fairness-related legal documents).

1.1 Thesis Contributions

The thesis aims to develop practical techniques and theoretical understanding for building ethical AI systems. We divide the thesis into two parts. The first part of the thesis focuses on fine-grained and structured information extraction using natural language processing (NLP) from policy documents. Policy documents are natural language documents about how different stakeholders (e.g., users and ML services providers) in internet services (e.g., commercial AI systems) agree on how the services providers commit to the ethical usage of users' data. We develop NLP techniques and benchmarks for privacy policies, a type of policy document describing the practices of using, sharing, and protecting users' data [27, 28]. Specifically, we develop (1) a QA corpus and system to facilitate fine-grained information extraction and (2) A NLU (intent classification and slot filling) corpus and system to facilitate structured information extraction. Such developed NLP techniques could be extended to other natural language law documents describing ethical AI principles and help improve mutual trust among different parties.

The second part of the thesis focuses on the theoretical understanding and development of algorithmic interventions for ethical artificial intelligence. In particular, we study the fairness problems for various machine learning tasks, such as classification, regression, and sequential decision-making: (1) we provide bias mitigation techniques for text classification using contrastive representation learning [29]; (2) we provide the theoretical understanding and mitigation techniques for accuracy disparity problem in regression [30]; (3) we propose a fairness notion that requires long-term equality on expected utility for different demographic groups for sequential decision-making [31], and develop methods to achieve the proposed fairness notion. In addition, we also study adversarial representation learning, a technique that has been widely used for algorithmic fairness, and its implications for information obfuscation [32]. Figure 1.1 illustrates the connections between two parts of our works.

1.2 Thesis Structure

The remainder of this dissertation is organized as follows. In Chapter 2, we provide the background of the dissertation, including natural language processing for policy documents, algorithmic fairness, and privacy-preserving machine learning. Next, we present the first line of works in Chapter 3 and Chapter 4, and the second line of works in Chapter 5, Chapter 6, Chapter 7, and Chapter 8. Finally, we conclude the dissertation in Chapter 9

Chapter 2

Background

This section reviews the most related literature, including natural language processing for policy documents, privacy-preserving machine learning, and algorithmic fairness.

2.1 Natural Language Processing for Policy Documents

Privacy policy documents describe how an entity collects, maintains, uses, and shares users' information. In large technology companies, users' data are usually used in machine learning services in both training and inference phases. Thus, users need to read the privacy policies of the websites they visit or the mobile applications they use and know about their data practices that are pertinent to them. However, prior works suggested that people do not read privacy policies because they are long complicated [33], and confusing [34]. Hence, giving users quick access to the information they seek from long and verbose policy documents can help them better understand their options and rights.

The recent development of natural language processing (NLP) techniques facilitates automatic privacy policy analysis. In literature, information extraction from policy documents is formulated as text classification [35–38], text alignment [39,40], named-entity recognition [41,42], and question answering (QA) [37,43,44]. However, those works fail to provide fine-grained annotations or information that users might need. In Chapter 3 and Chapter 4, we will present our works using NLP to extract fine-grained and structured information from privacy policies and compare them with the related works in the literature in detail.

Since privacy policies is a type of legal documents, the developed NLP could be extended to other natural language law documents describing other ethical AI principles, in the applications such as legal research, electronic discovery, contract review, document automation, and legal advice [45–49].

2.2 Algorithmic Fairness

The section covers standard definitions of fairness in machine learning, methods for fair machine learning, and fairness research in natural language processing.

Definitions of Fairness In the context of decision-making, fairness is the *absence of any prejudice or favoritism toward an individual or a group based on their inherent or acquired characteristics* [50]. The definitions of fairness can be broadly categorized into two types: *group fairness* and *individual fairness* [51]. Group fairness requires that the performances of ML models among different groups are (approximately) equal, while individual fairness requires that similar individuals are treated similarly. There are many proposed notions for group fairness (e.g., demographic parity [51], equalized odds [52], and others [53]) and individual fairness (e.g., fairness through awareness [51], counterfactual fairness [54], etc.). Next, we will present the most standard and pertinent fairness notions in binary classification, where the protected attribute is also binary.

Definition 2.2.1 (Demographic Parity)**.** A predictor $\hat{Y}$ satisfies demographic parity if $\Pr(\hat{Y}|A = 0) = \Pr(\hat{Y}|A = 1)$.

Demographic parity is also known as *statistical parity*, and it requires the likelihoods of any outcomes should be the same among different demographic groups.

Definition 2.2.2 (Equalized Odds)**.** A binary predictor $\hat{Y}$ satisfies equalized odds with respect to protected attribute A and label Y if $\hat{Y}$ and A are independent conditional on Y:

$$\Pr(\hat{Y} = 1 \mid A = 0, Y = y) = P(\hat{Y} = 1 \mid A = 1, Y = y), \quad y \in \{0, 1\}.$$

Equalized odds requires true positive rates and false positive rates among different demographic groups should be the same. A weaker notion of equalized odds is *equalized opportunity*, which only requires true positive rates among different demographic groups should be the same.

Definition 2.2.3 (Accuracy Parity)**.** A binary predictor $\hat{Y}$ satisfies accuracy parity with respect to protected attribute A if

$$\Pr(\hat{Y} = Y \mid A = 0) = P(\hat{Y} = Y \mid A = 1).$$

Methods for Fair Machine Learning There are various methods proposed to satisfy one or more fairness definitions in different areas of machine learning. These methods target different parts of the model development life cycle:

- **Pre-processing methods** target transforming the data before inputting it into the model to remove the underlying discrimination and biases. These include data re-weighting and re-sampling [55], Edit the labels or features in the training data [55, 56], learning (probabilistic) transformations that edits the features and labels [57].

- **In-processing methods** aim to train a fair predictor. It usually involves modifying the loss function to achieve our two goals simultaneously: the predictor should be fair and accurate. Standard in-processing techniques include adversarial training [58], learning fair representations [59], and adding constraints and regularizations to the loss function [60, 61].

- **Post-processing methods** adjust the outputs of a predictor and leave the underlying predictor and training data unchanged. Standard techniques involve modifying predicted outcomes and prediction thresholds in a group-specific manner [52, 53, 62, 63]

Fairness in NLP Systems Unintended social biases in NLP models have been identified in word/sentence embedding [18, 64] and applications such as coreference resolution [65], language modeling [66], machine translation [67]. Compared to *allocational harms* caused by machine learning models (e.g., classifier) in high-stake domains, the *representational harms* encoded in NLP systems are more difficult to formalize [68]. Researchers proposed various bias mitigation techniques [69] depending on the types of representational bias (e.g., stereotyping).

2.3 Privacy-Preserving Machine Learning

Machine learning models are vulnerable to a variety of privacy attacks: membership inference attacks [12, 70, 71], attribute inference (model inversion) attacks [13, 14, 72], model stealing attacks [73, 74], and others [75, 76]. Next, we review the literature on privacy-preserving mechanisms for machine learning. Note that many related techniques and mechanisms are devoted to providing privacy-preserving machine learning. We focus on the most widely-adopted and pertinent ones, including differential privacy, homomorphic encryption, secure multi-party computation, and information-theoretic obfuscation.

Differential Privacy Since ML models can leak sensitive information about their training data, it is important that ML models do not memorize information about any examples in the training set. Differential privacy (DP) [77] provides such formal guarantees to ensure that.

Definition 2.3.1 ((ϵ, δ)-Differential Privacy)**.** A randomized mechanism $M : \mathfrak{D} \to \mathfrak{R}$ satisfies (ϵ, δ)-differential privacy if for any two adjacent inputs $d, d' \in \mathfrak{D}$ and for any subset of outputs $S \subseteq \mathfrak{R}$ it holds that

$$\Pr[M(d) \in S] \leq \exp(\epsilon) \Pr[M(d') \in S] + \delta.$$

The definition of DP requires that the output distributions of the two adjacent inputs are similar with high probability. In the context of machine learning, differentially private ML algorithms should output similar models when training on the training datasets that only differ in any one example, which forces models not to memorize any particular training example. Several techniques [78, 79] have been proposed to train differentially private ML models.

Homomorphic Encryption Homomorphic Encryption (HE) [80] allows machine learning services providers to compute over users' encrypted data without decrypting it, and the users can decrypt the encrypted results. Hence, HE has been proposed for ML models inference [81–84] at the cost of significant computation overheads, prohibiting HE from being deployed in production systems [85, 86].

Secure Multi-Party Computation Secure multi-party computation (SMC) techniques allow multiple parties jointly to perform computations and receive the resulting output while keeping their inputs secret. SMC techniques have been used for ML model training [87–91] and inference [85, 92–94], and it has also been used collectively with HE [95–98]. Compared to HE, SMC is less computationally expensive at the cost of additional communications overhead and assumptions of the proportions of malicious coordinating parties during computation.

Information-Theoretic Obfuscation Assuming the data distribution for ML models and the tasks of interest, a line of works [99–107] focuses on obfuscating sensitive information while maximizing the task accuracy via (adversarial) representation learning. This line of works explicitly defines the sensitive information (e.g., sensitive attributes) they want to obfuscate and tries to degrade any excessive sensitive information from the input data that is redundant for the main inference task. In Chapter 8, we will give formal characterizations of trade-offs and guarantees using adversarial representation learning for information obfuscation and their connections to algorithmic fairness.

Part I

Information Extraction from Policy Documents

Chapter 3

Question Answering for Policy Documents

3.1 Introduction

Security and privacy policy documents describe how an entity collects, maintains, uses, and shares users' information. Users need to read the privacy policies of the websites they visit or the mobile applications they use and know about their privacy practices that are pertinent to them. However, prior works suggested that people do not read privacy policies because they are long and complicated [33], and confusing [34]. Hence, giving users access to a question answering system to search for answers from long and verbose policy documents can help them better understand their rights.

In recent years, we have witnessed noteworthy progress in developing question answering (QA) systems with a colossal effort to benchmark high-quality, large-scale datasets for a few application domains (e.g., Wikipedia, news articles). However, annotating large-scale QA datasets for domains such as security and privacy is challenging as it requires expert annotators (e.g., law students). Due to the difficulty of annotating policy documents at scale, the only available QA dataset is PrivacyQA [108] on privacy policies for 35 mobile applications.

An essential characteristic of policy documents is that they are well structured as they are written by following guidelines set by the policymakers. Besides, due to the homogeneous nature of different

Table 3.1: Question-answer pairs that we collect from OPP-115 [1] dataset. The evidence spans are highlighted in color and they are used to form the question-answer pairs.

Website: Amazon.com
Information You Give Us: We receive and store any information you enter on our Web site or give us in any other way. Click here to see ... Question. How do you collect my information? information you enter on our Web site
Promotional Offers: Sometimes we send offers to selected groups of Amazon.com customers on behalf of other businesses. When we do this, we do not give that business your name and address. If you do not want to receive such offers, ... Question. Is my information shared with others? we do not give that business your name and address

Table 3.2: Comparison of PolicyQA and PrivacyQA.

	PolicyQA (This work)	PrivacyQA
Source	Website privacy policies	Mobile application privacy policies
# Policies	115	35
# Questions	714	1,750
# Annotations	25,017	3,500
Question annotator	Domain experts	Mechanical Turkers
Form of QA	Reading comprehension	Sentence selection
Answer type	A sequence of words	A list of sentences

entities (e.g., Amazon, eBay), their privacy policies have a similar structure. Therefore, we can exploit the document structure (meta data) to form examples from existing corpora. In this chapter, we present PolicyQA, a reading comprehension style question answering dataset with 25,017 question-passage-answer triples associated with text segments from privacy policy documents. PolicyQA consists of 714 questions on 115 website privacy policies and is curated from an existing corpus, OPP-115 [1]. Table 3.1 presents a couple of examples from PolicyQA.

In contrast to PrivacyQA [108] that focuses on extracting long text spans from policy documents, we argue that highlighting a shorter text span in the document facilitates the users to zoom into the policy and identify the target information quickly. To enable QA models to provide such short answers, PolicyQA provides examples with an average answer length of 13.5 words (in comparison, the PrivacyQA benchmark has examples with an average answer length of 139.6 words). We present a comparison between PrivacyQA and PolicyQA in Table 3.2.

We present two strong neural baseline models trained on PolicyQA and perform a thorough analysis to shed light on the advantages and challenges offered by the proposed dataset. The data and the

Table 3.3: Sample span annotations from OPP-115 associated with a segment of *Amazon.com* privacy policy.

“Practice”: First Party Collection/Use “Attribute”: Purpose “value”: “Additional service/feature” “startIndexInSegment”: 360 “endIndexInSegment”: 387 “selectedText”: “responding to your requests”
“Practice”: Third Party Sharing/Collection “Attribute”: Third Party Entity “value”: “Unnamed third party” “startIndexInSegment”: 573 “endIndexInSegment”: 596 “selectedText”: “Third-Party Advertisers”

implemented baseline models are made publicly available.[1]

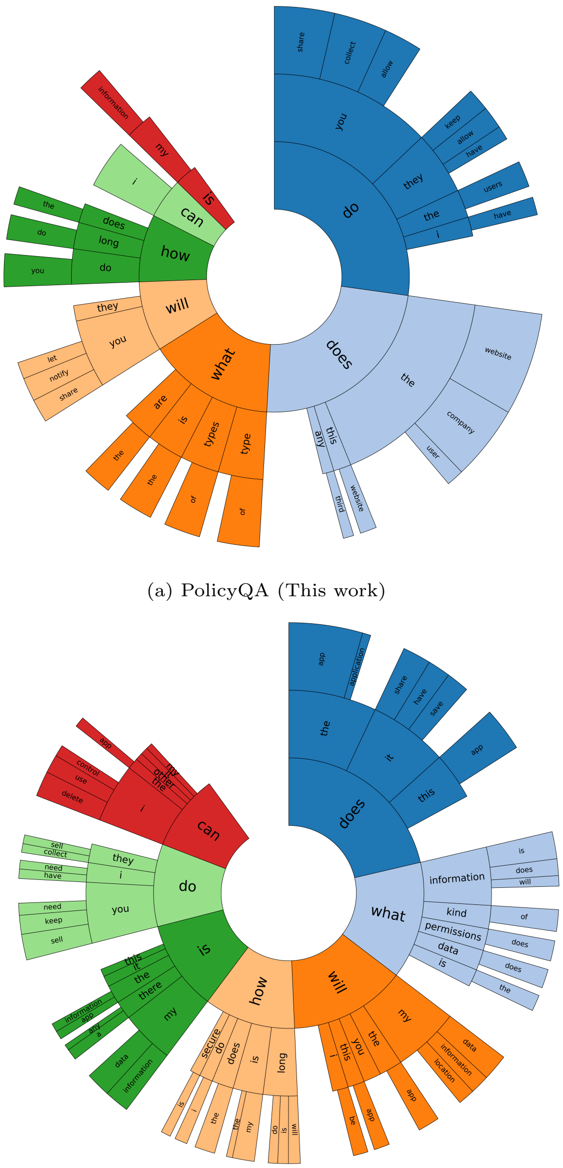
3.2 Dataset

PolicyQA consists question-passage-answer triples, curated from OPP-115 [1]. OPP-115 is a corpus of 115 website privacy policies (3,792 segments), manually annotated by skilled annotators following the annotation schemes predefined by domain experts. The annotation schemes are composed of 10 data practice categories (e.g., *First Party Collection/Use*, *Third Party Sharing/Collection*, *User Choice/Control* etc.). The data practices are further categorized into a set of practice attributes (e.g., *Personal Information Type*, *Purpose*, *User Type* etc.). Each practice attribute is associated with a predefined set of values. In the Appendix (in Table 3.9), we list all the attributes under the *First Party Collection/Use* category.

In total, OPP-115 contains 23,000 data practices, 128,000 practice attributes, and 103,000 annotated text spans. Each text span belongs to a policy segment, and OPP-115 provides its character-level start and end indices. We provide an example in Table 3.3. We use the annotated spans, corresponding policy segments, and the associated {*Practice, Attribute, Value*} triples to form PolicyQA examples. We exclude the spans with practices labeled as “Other” and the values labeled as “Unspecified”. Next, we describe the question annotation process.

Question annotations. Two skilled annotators manually annotate the questions. During annotation, the annotators are provided with the triple {Practice, Attribute, Value}, and the associated text span. For example, given the triple {*First Party Collection/Use*, *Personal Information Type*, *Contact*} and

[1] `https://github.com/wasiahmad/PolicyQA`



(a) PolicyQA (This work)

(b) PrivacyQA

Figure 3.1: Distribution of trigram prefixes of questions in (a) PolicyQA and (b) PrivacyQA.

the associated text span "*name, address, telephone number, email address*", the annotators created questions, such as, (1) *What type of contact information does the company collect?*, (2) *Will you use my contact information?*, etc.

For a specific triple, the process is repeated for 5-10 randomly chosen samples to form a list of questions. We randomly assign a question from this list to the examples associated with the triple that were not chosen during the sampling process. In total, we considered 258 unique triples and created 714 individual questions. In Table 3.4, we provide an example question for each practice

Table 3.4: OPP-115 categories of the questions in the PolicyQA dataset.

Privacy Practice	Proportion	Example Question From PolicyQA
First Party Collection/Use	44.4 %	Why do you collect my data?
Third Party Sharing/Collection	34.1 %	Do they share my information with others?
Data Security	2.2 %	Do you use encryption to secure my data?
Data Retention	1.7 %	How long they will keep my data?
User Access, Edit and Deletion	3.1 %	Will you let me access and edit my data?
User Choice/Control	11.0 %	What use of information does the user choice apply to?
Policy Change	1.9 %	How does the website notify about policy changes?
International and Specific Audiences	1.5 %	What is the company's policy towards children?
Do Not Track	0.1 %	Do they honor the user's do not track preference?

Table 3.5: Statistics of the PolicyQA dataset.

Dataset	Train	Valid	Test
# Examples	17,056	3,809	4,152
# Policies	75	20	20
# Questions	693	568	600
# Passages	2,137	574	497
Avg. question length	11.2	11.2	11.2
Avg. passage length	106.0	96.6	119.1
Avg. answer length	13.3	12.8	14.1

category. Also, we compare the distribution of questions' trigram prefixes in PolicyQA (Figure 3.1a) with PrivacyQA (Figure 3.1b). It is important to note that, PolicyQA questions are written in a generic fashion to become applicable for text spans, associated with the same practice categories. Therefore, PolicyQA questions are less diverse than PrivacyQA questions.

We split OPP-115 into 75/20/20 policies to form training, validation, and test examples, respectively.

3.3 Experiment

In this section, we evaluate two neural question answering (QA) models on PolicyQA and present the findings from our analysis.

Baselines. PolicyQA frames the QA task as predicting the answer span that exists in the given policy segment. Hence, we consider two existing neural approaches from literature as baselines for PolicyQA. The first model is **BiDAF** [109] that uses a bi-directional attention flow mechanism to extract the evidence spans. The second baseline is based on **BERT** [110] with two linear classifiers to predict the boundary of the evidence, as suggested in the original work.

Fine-tuning	SQuAD Pre-training	Valid		Test	
		EM	F1	EM	F1
BiDAF					
✗	✗	25.1	52.3	22.0	48.0
✗	✓	26.7	53.7	23.3	49.5
✓	✗	27.9	57.2	24.4	52.8
BERT-base					
✗	✗	30.5	59.4	28.1	55.6
✗	✓	30.5	60.2	28.0	56.2
✓	✗	**32.8**	60.9	28.6	**56.6**
✓	✓	32.7	**61.2**	**29.5**	**56.6**

Table 3.6: Performance of baselines on PolicyQA. The bold face values indicate the best performances.

Implementation. PolicyQA has a similar setting as SQuAD [111]. Therefore, we pre-train the QA models using their default settings on the SQuAD dataset. Besides, we consider leveraging unlabeled privacy policies in fine-tuning the models, as noted below.

- **Fine-tuning.** We train word embeddings using *fastText* [112] based on a corpus of 130,000 privacy policies (137M words) collected from apps on the Google Play Store.[2] These word embeddings are used as fixed word representations in BiDAF while training on PolicyQA. Similarly, to adapt BERT to the privacy domain, we first fine-tune BERT using masked language modeling [110] based on the privacy policies and then train on PolicyQA.

- **No fine-tuning.** In this setting, we use the publicly available *fastText* [112] embeddings with BiDAF, and the BERT model is not fine-tuned on those privacy policies.

We adopt the default model architecture and optimization setup for the baseline methods. We detail the hyper-parameters in Appendix (in Table 3.10).

Evaluation. Following [111], we use *exact match* (EM) and *F1 score* to evaluate the model's accuracy.

BERT Size	Valid		Test	
	EM	F1	EM	F1
Tiny	21.0	47.1	15.5	39.9
Mini	26.5	55.2	22.8	49.8
Small	28.4	57.2	24.6	52.3
Medium	**31.1**	59.1	25.2	53.5
Base	30.5	**59.4**	**28.1**	**55.6**

Table 3.7: Performance of different sized QA models.

[2] We thank the authors of [37] for sharing the 130,000 privacy policies.

Table 3.8: Test performance breakdown of BERT-base model for privacy practice categories, sorted by the average answer length as indicated by |ans|.

	|ans|	EM	F1
Third Party Sharing/Collection	9.3	35.0	60.2
First Party Collection/Use	10.1	28.3	55.7
Data Retention	10.6	29.1	55.9
User Choice/Control	11.0	24.3	53.2
User Access, Edit and Deletion	12.2	21.6	51.5
Policy Change	14.6	43.4	67.7
Do Not Track	30.9	37.5	69.2
Data Security	34.6	24.4	54.3
Intl. and Specific Audiences	52.8	5.3	43.1

3.3.1 Results and Analysis

The experimental results are presented in Table 3.6. Overall, the BERT-base methods outperform the BiDAF models by 6.1% and 7.6% in terms of EM and F1 score (on the test split), respectively.

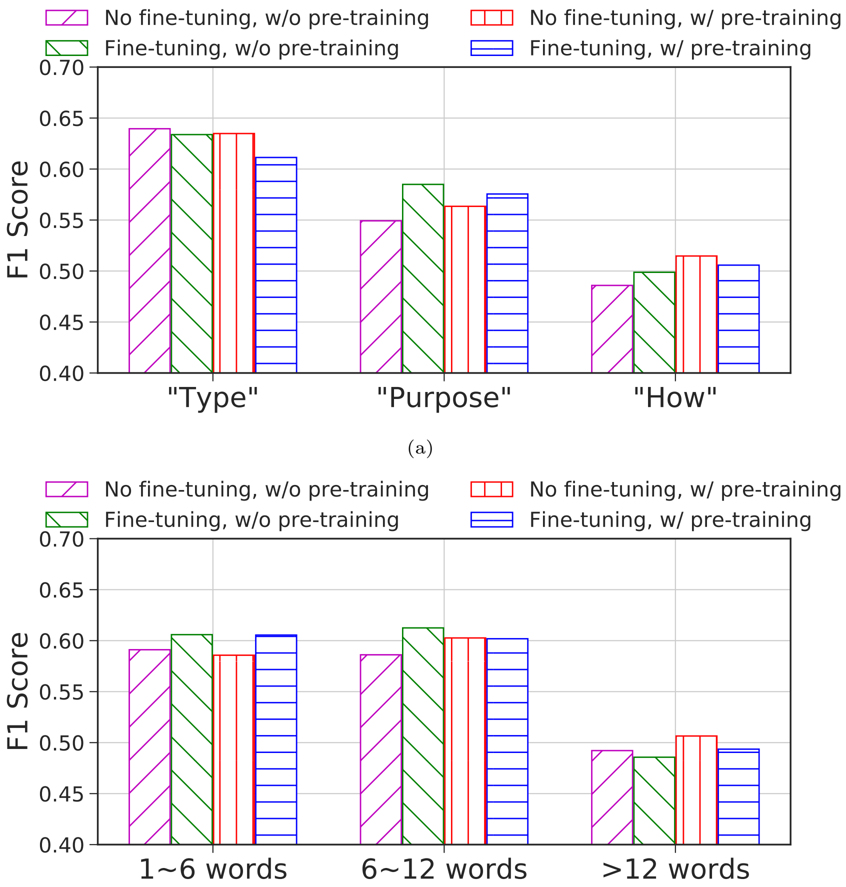
Impact of fine-tuning. Table 3.6 demonstrates that the fine-tuning step improves the downstream task performance. For example, BERT-base performance is improved by 0.5% and 1.0% EM and F1 score, respectively, on the test split. This result encourages to train/fine-tune BERT on a larger collection of security and privacy documents.

Impact of SQuAD pre-training. Given a small number of training examples, it is challenging to train deep neural models. Hence, we pre-train the extractive QA models on SQuAD [111] and then fine-tune on PolicyQA. The additional pre-training step improves performance. For example, in *no fine-tuning* setting, BiDAF, and BERT-base improve the performance by 1.5% and 0.6% F1 score, respectively (on the test split).

Impact of model size. We experiment with different sized BERT models [113] and the results in Table 3.7 shows that the performance improves with increased model size. The results also indicate that PolicyQA is a challenging dataset, and hence, a larger model performs better.

Analysis. We breakdown the test performance of the BERT-base method to examine the model performance across practice categories. The results are presented in Table 3.8. We see the model performs comparably on the three most frequent categories (comprise 89.5% of the total examples).

We further analyze the performance on questions associated with (1) the top three frequent attributes for the two most frequent practice categories, and (2) different answer lengths. The results are presented in Figure 3.2a and 3.2b. Our findings are (1) shorter evidence spans (e.g., evidence spans



(a)

(b)

Figure 3.2: BERT-base model's performance on (a) the three most frequent attributes of "First Party Collection/Use" and "Third Party Sharing/Collection" practice categories, and (b) questions with different answer lengths.

for *Personal Information Type* questions) are easier to extract than longer spans; and (2) SQuAD pre-training helps more in extracting shorter evidence spans. Leveraging diverse extractive QA resources may reduce the length bias and boost the QA performance on privacy policies.

3.4 Related Work

The *Usable Privacy Project* [114] has made several attempts to automate the analysis of privacy policies [1, 38]. Noteworthy works include identification of policy segments commenting on specific data practices [115], extraction of opt-out choices, and their provisions in policy text [36, 116], and others [117, 118]. [119] used a keyword-based technique to compare online privacy policies. Natural language processing (NLP) techniques such as text alignment [40, 120], text classification [1, 37, 38] and question answering [37, 108, 121] has been studied in prior works to facilitate privacy policy analysis.

Among the question answering (QA) methods, [37] framed the task as retrieving the most relevant policy segments as an answer, while [108] presented a dataset and models to answer questions with a list of sentences. In comparison to the prior QA approaches, we encourage developing QA systems capable of providing precise answers by using PolicyQA.

3.5 Conclusion

This chapter presents PolicyQA, a reading comprehension style question answering (QA) dataset. PolicyQA can contribute to the development of QA systems in the security and privacy domain that have a sizeable real-word impact. We evaluate two strong neural baseline methods on PolicyQA and provide thorough ablation analysis to reveal important considerations that affect answer span prediction. In our future work, we want to explore how transfer learning can benefit question answering in the security and privacy domain.

3.6 Appendix of Chapter 3

Table 3.9: The attributes and their values for the *First Party Collection/Use* data practice category. We do not consider the data practices associated with "Unspecified" values.

Attribute	Values
Does/Does Not	Does; Does Not
Collection Mode	Explicit; Implicit; Unspecified
Action First-Party	Collect on website; Collect in mobile app; Collect on mobile website; Track user on other websites; Collect from user on other websites; Receive from other parts of company/affiliates; Receive from other service/third-party (unnamed); Receive from other service/third-party (named); Other; Unspecified
Identifiability	Identifiable; Aggregated or anonymized; Other; Unspecified
Personal Information Type	Financial; Health; Contact; Location; Demographic; Personal identifier; User online activities; User profile; Social media data; IP address and device IDs; Cookies and tracking elements; Computer information; Survey data; Generic personal information; Other; Unspecified
Purpose	Basic service/feature; Additional service/feature; Advertising; Marketing; Analytics/Research; Personalization/Customization; Service Operation and Security; Legal requirement; Merger/Acquisition; Other; Unspecified
User Type	User without account; User with account; Other; Unspecified
Choice Type	Dont use service/feature; Opt-in; Opt-out link; Opt-out via contacting company; First-party privacy controls; Third-party privacy controls; Browser/device privacy controls; Other; Unspecified
Choice Scope	Collection; Use; Both; Unspecified

Table 3.10: Hyper-parameters used in our experiments.

Model	Hyper-parameter	Value	Model	Hyper-parameter	Value
BiDAF	dimension	300	BERT	d_{model}	768
	rnn_type	LSTM		num_heads	12
	num_layers	1		num_layers	12
	hidden_size	300		d_{ff}	3072
	dropout	0.2		dropout	0.2
	optimizer	Adam		optimizer	BertAdam
	learning rate	0.001		learning rate	0.00003
	batch size	16		batch size	16
	epoch	15		epoch	5

Table 3.11: Examples questions from PolicyQA for the "Action First-Party" attribute under the *First Party Collection/Use* data practice category.

Value	Example Question From PolicyQA
Collect on website	Do you collect my information on your website?
Collect in mobile app	Will you collect my data if I use your phone app?
Collect on mobile website	How do you collect data when I use my mobile?
Track user on other websites	Do they track users' activities on other websites?
Collect from user on other websites	Does the website collect my info on other websites?
Receive from other parts of company/affiliates	Do you collect my information from your affiliates?
Receive from other service/third-party (unnamed)	Does the website obtain my data from others?
Receive from other service/third-party (named)	Who provides you my data?
Other	How do you receive data from users?

Chapter 4

Intent Classification and Slot Filling for Policy Documents

4.1 Introduction

Privacy policies inform users about how a service provider collects, uses, and maintains the users' information. The service providers collect the users' data via their websites or mobile applications and analyze them for various purposes. The users' data often contain sensitive information; therefore, the users must know how their information will be used, maintained, and protected from unauthorized and unlawful use. Privacy policies are meant to explain all these use cases in detail. This makes privacy policies often very long, complicated, and confusing [33, 34]. As a result, users do not tend to read privacy policies [122–124], leading to undesirable consequences. For example, users might not be aware of their data being sold to third-party advertisers even if they have given their consent to the service providers to use their services in return. Therefore, automating information extraction from verbose privacy policies can help users understand their rights and make informed decisions.

In recent years, we have seen substantial efforts to utilize natural language processing (NLP) techniques to automate privacy policy analysis. In literature, information extraction from policy documents is formulated as text classification [1, 37, 38], text alignment [40, 120], and question answering (QA) [27, 37, 108, 121]. Although these approaches effectively identify the sentences or segments in a policy document relevant to a privacy practice, they lack in extracting fine-grained

Table 4.1: Annotation examples from PolicyIE Corpus. Best viewed in color.

[We]$_{\text{Data Collector: First Party Entity}}$ may also [use]$_{\text{Action}}$ or display [your]$_{\text{Data Provider: user}}$ [username]$_{\text{Data Collected: User Online Activities/Profiles}}$ and [icon or profile photo]$_{\text{Data Collected: User Online Activities/Profiles}}$ on [marketing purpose or press releases]$_{\text{Purpose: Advertising/Marketing}}$. *Privacy Practice.* Data Collection/Usage
[We]$_{\text{Data Sharer: First Party Entity}}$ do [not]$_{\text{Polarity: Negation}}$ [sell]$_{\text{Action}}$ [your]$_{\text{Data Provider: user}}$ [personal information]$_{\text{Data Shared: General Data}}$ to [third parties]$_{\text{Data Receiver: Third Party Entity}}$. *Privacy Practice.* Data Sharing/Disclosure

structured information. As shown in the first example in Table 4.1, the privacy practice label "Data Collection/Usage" informs the user how, why, and what types of user information will be collected by the service provider. The policy also specifies that users' "username" and "icon or profile photo" will be used for "marketing purposes". This informs the user precisely what and why the service provider will use users' information.

The challenge in training models to extract fine-grained information is the lack of labeled examples. Annotating privacy policy documents is expensive as they can be thousands of words long and requires domain experts (e.g., law students). Therefore, prior works annotate privacy policies at the sentence level, without further utilizing the constituent text spans to convey specific information. Sentences written in a policy document explain privacy practices, which we refer to as *intent classification* and identifying the constituent text spans that share further specific information as *slot filling*. Table 4.1 shows a couple of examples. This formulation of information extraction lifts users' burden to comprehend relevant segments in a policy document and identify the details, such as how and why users' data are collected and shared with others.

To facilitate fine-grained information extraction, we present PolicyIE, an English corpus consisting of 5,250 intent and 11,788 slot annotations over 31 privacy policies of websites and mobile applications. We perform experiments using sequence tagging and sequence-to-sequence (Seq2Seq) learning models to jointly model intent classification and slot filling. The results show that both modeling approaches perform comparably in intent classification, while Seq2Seq models outperform the sequence tagging models in slot filling by a large margin. We conduct a thorough error analysis and categorize the errors into seven types. We observe that sequence tagging approaches miss more slots while Seq2Seq models predict more spurious slots. We further discuss the error cases by considering other factors to help guide future work. We release the code and data to facilitate research.[1]

[1] https://github.com/wasiahmad/PolicyIE

4.2 Construction of PolicyIE Corpus

4.2.1 Privacy Policies Selection

The scope of privacy policies primarily depends on how service providers function. For example, service providers primarily relying on mobile applications (e.g., Viber, Whatsapp) or websites and applications (e.g., Amazon, Walmart) have different privacy practices detailed in their privacy policies. In PolicyIE, we want to achieve broad coverage across privacy practices exercised by the service providers such that the corpus can serve a wide variety of use cases. Therefore, we go through the following steps to select the policy documents.

Initial Collection [40] introduced a corpus of 1,010 privacy policies of the top websites ranked on `Alexa.com`. We crawled those websites' privacy policies in November 2019 since the released privacy policies are outdated. For mobile application privacy policies, we scrape application information from Google Play Store using `play-scraper` public API[2] and crawl their privacy policy. We ended up with 7,500 mobile applications' privacy policies.

Filtering First, we filter out the privacy policies written in a non-English language and the mobile applications' privacy policies with the app review rating of less than 4.5. Then we filter out privacy policies that are too short (< 2,500 words) or too long (> 6,000 words). Finally, we randomly select 200 websites and mobile application privacy policies each (400 documents in total).[3]

Post-processing We ask a domain expert (working in the security and privacy domain for more than three years) to examine the selected 400 privacy policies. The goal for the examination is to ensure the policy documents cover the four privacy practices: (1) *Data Collection/Usage*, (2) *Data Sharing/Disclosure*, (3) *Data Storage/Retention*, and (4) *Data Security/Protection*. These four practices cover how a service provider processes users' data in general and are included in the General Data Protection Regulation (GDPR). Finally, we shortlist 50 policy documents for annotation, 25 in each category (websites and mobile applications).

4.2.2 Data Annotation

Annotation Schema To annotate sentences in a policy document, we consider the first four privacy practices from the annotation schema suggested by [1]. Therefore, we perform sentence

[2] `https://github.com/danieliu/play-scraper`

[3] We ensure the mobile applications span different application categories on the Play Store.

categorization under five *intent classes* that are described below.

(1) *Data Collection/Usage*: What, why and how user information is collected;

(2) *Data Sharing/Disclosure*: What, why and how user information is shared with or collected by third parties;

(3) *Data Storage/Retention*: How long and where user information will be stored;

(4) *Data Security/Protection*: Protection measures for user information;

(5) *Other*: Other privacy practices that do not fall into the above four categories.

Apart from annotating sentences with privacy practices, we aim to identify the text spans in sentences that explain specific details about the practices. For example, in the sentence *"we collect personal information in order to provide users with a personalized experience"*, the underlined text span conveys the purpose of data collection. In our annotation schema, we refer to the identification of such text spans as *slot filling*. There are 18 slot labels in our annotation schema (provided in Appendix). We group the slots into two categories: type-I and type-II based on their role in privacy practices. While the type-I slots include participants of privacy practices, such as *Data Provider*, *Data Receiver*, type-II slots include purposes, conditions that characterize more details of privacy practices. Note that type-I and type-II slots may overlap, e.g., in the previous example, the underlined text span is the *purpose* of data collection, and the span "user" is the *Data Provider* (whose data is collected). In general, type-II slots are longer (consisting of more words) and less frequent than type-I slots.

In total, there are 14 type-I and 4 type-II slots in our annotation schema. These slots are associated with a list of attributes, e.g., *Data Collected* and *Data Shared* have the attributes *Contact Data*, *Location Data*, *Demographic Data*, etc. Table 4.1 illustrates a couple of examples. We detail the slots and their attributes in the Appendix.

Annotation Procedure General crowdworkers such as Amazon Mechanical Turkers are not suitable to annotate policy documents as it requires specialized domain knowledge [33, 34]. We hire two law students to perform the annotation. We use the web-based annotation tool, BRAT [125] to conduct the annotation. We write a detailed annotation guideline and pretest them through multiple rounds of pilot studies. The guideline is further updated with notes to resolve complex or

Table 4.2: Statistics of the PolicyIE Corpus.

Dataset	Train	Test
# Policies	25	6
# Sentences	4,209	1,041
# Type-I slots	7,327	1,704
# Type-II slots	2,263	494
Avg. sentence length	23.73	26.62
Avg. # type-I slot / sent.	4.48	4.75
Avg. # type-II slot / sent.	1.38	1.38
Avg. type-I slot length	2.01	2.15
Avg. type-II slot length	8.70	10.70

Table 4.3: An example of input / output used to train the two types of models on PolicyIE. For brevity, we replaced part of label strings with symbols: DP.U, DC.FPE, DC.UOAP, P.AM represents Data-Provider.User, Data-Collector.First-Party-Entity, Data-Collected.User-Online-Activities-Profiles, and Purpose.Advertising-Marketing.

Joint intent and slot tagging
Input: [CLS] We may also use or display your username and icon or profile photo on marketing purpose or press releases .
Type-I slot tagging output Data-Collection-Usage B-DC.FPE O O B-Action O O B-DP.U B-DC.UOAP O B-DC.UOAP I-DC.UOAP I-DC.UOAP I-DC.UOAP O O O O O O O
Type-II slot tagging output Data-Collection-Usage O O O O O O O O O O O O O O B-P.AM I-P.AM I-P.AM I-P.AM I-P.AM O
Sequence-to-sequence (Seq2Seq) learning
Input: We may also use or display your username and icon or profile photo on marketing purpose or press releases .
Output: [IN:Data-Collection-Usage [SL:DC.FPE *We*] [SL:Action *use*] [SL:DP.U *your*] [SL:DC.UOAP *username*] [SL:DC.UOAP *icon or profile photo*] [SL:P.AM *marketing purpose or press releases*]]

corner cases during the annotation process. The annotation process is closely monitored by a domain expert and a legal scholar and is granted IRB exempt by the Institutional Review Board (IRB). The annotators are presented with one segment from a policy document at a time and asked to perform annotation following the guideline. We manually segment the policy documents such that a segment discusses similar issues to reduce ambiguity at the annotator end. The annotators worked 10 weeks, with an average of 10 hours per week, and completed annotations for 31 policy documents. Each annotator is paid $15 per hour.

Post-editing and Quality Control We compute an inter-annotator agreement for each annotated segment of policy documents using Krippendorff's Alpha (α_K) [126]. The annotators are asked to discuss their annotations and re-annotate those sections with token-level α_K falling below 0.75. An α_K value within the range of 0.67 to 0.8 is allowed for tentative conclusions [127, 128]. After the re-annotation process, we calculate the agreement for the two categories of slots individually. The inter-annotator agreement is 0.87 and 0.84 for type-I and type-II slots, respectively. Then the adjudicators discuss and finalize the annotations. The adjudication process involves one of the annotators, the legal scholar, and the domain expert.

Data Statistics & Format Table 4.2 presents the statistics of the PolicyIE corpus. The corpus consists of 15 and 16 privacy policies of websites and mobile applications, respectively. We release the annotated policy documents split into sentences.[4] Each sentence is associated with an intent label, and the constituent words are associated with a slot label (following the BIO tagging scheme).

4.3 Model & Setup

PolicyIE provides annotations of privacy practices and corresponding text spans in privacy policies. We refer to privacy practice prediction for a sentence as *intent classification* and identifying the text spans as *slot filling*. We present two alternative approaches; the first approach jointly models intent classification and slot tagging [130], and the second modeling approach casts the problem as a sequence-to-sequence learning task [131, 132].

Table 4.4: Test set performance of the sequence tagging models on PolicyIE corpus. We individually train and evaluate the models on intent classification and type-I and type-II slots tagging and report average intent F1 score.

Model	# param (in millions)	Intent F1	Type-I		Type-II	
			Slot F1	EM	Slot F1	EM
Human	-	96.5	84.3	56.6	62.3	55.6
Embedding	1.7	$50.9_{\pm 27.3}$	$19.1_{\pm 0.3}$	$0.8_{\pm 0.3}$	$0.0_{\pm 0.0}$	$0.0_{\pm 0.0}$
BiLSTM	8	$75.9_{\pm 1.1}$	$40.8_{\pm 0.9}$	$7.6_{\pm 0.9}$	$3.9_{\pm 3.0}$	$10.0_{\pm 2.7}$
Transformer	34.8	$80.1_{\pm 0.6}$	$41.0_{\pm 3.5}$	$6.5_{\pm 2.8}$	$3.5_{\pm 1.0}$	$13.1_{\pm 2.4}$
BERT	110	$\mathbf{84.7}_{\pm 0.7}$	$55.5_{\pm 1.1}$	$17.0_{\pm 1.1}$	$29.6_{\pm 2.4}$	$24.2_{\pm 4.2}$
RoBERTa	124	$84.5_{\pm 0.7}$	$54.2_{\pm 1.9}$	$14.3_{\pm 2.4}$	$29.8_{\pm 1.7}$	$24.8_{\pm 1.4}$
Embedding w/ CRF	1.7	$67.9_{\pm 0.6}$	$26.0_{\pm 1.5}$	$1.20_{\pm 0.3}$	$5.7_{\pm 4.6}$	$3.1_{\pm 0.6}$
BiLSTM w/ CRF	8	$76.7_{\pm 1.4}$	$45.1_{\pm 1.2}$	$9.2_{\pm 0.9}$	$26.8_{\pm 2.2}$	$18.1_{\pm 2.0}$
Transformer w/ CRF	34.8	$77.9_{\pm 2.7}$	$43.7_{\pm 2.3}$	$8.9_{\pm 3.0}$	$5.7_{\pm 0.9}$	$11.0_{\pm 2.1}$
BERT w/ CRF	110	$82.1_{\pm 2.0}$	$56.0_{\pm 0.8}$	$\mathbf{19.2}_{\pm 1.1}$	$31.7_{\pm 1.9}$	$19.7_{\pm 2.6}$
RoBERTa w/ CRF	124	$83.3_{\pm 1.6}$	$\mathbf{57.0}_{\pm 0.6}$	$18.2_{\pm 1.2}$	$\mathbf{34.5}_{\pm 1.3}$	$\mathbf{27.7}_{\pm 3.9}$

4.3.1 Sequence Tagging

Following [130], given a sentence $s = w_1, \dots, w_l$ from a privacy policy document D, a special token (w_0 = [CLS]) is prepended to form the input sequence that is fed to an encoder. The encoder produces contextual representations of the input tokens $h_0, h_1, \dots, h_l$ where h_0 and $h_1, \dots, h_l$ are

[4] We split the policy documents into sentences using UDPipe [129].

fed to separate softmax classifiers to predict the target intent and slot labels.

$$\begin{aligned} y^i &= \text{softmax}(W_i^T h_0 + b_i), \\ y_n^s &= \text{softmax}(W_s^T h_n + b_s), n \in 1, \ldots l, \end{aligned}$$

where $W_i \in R^{d \times I}, W_s \in R^{d \times S}, b_r \in R^I$ and $b_i \in R^I, b_s \in R^S$ are parameters, and I, S are the total number of intent and slot types, respectively. The sequence tagging model (composed of an encoder and a classifier) learns to maximize the following conditional probability to perform intent classification and slot filling jointly.

$$P(y^i, y^s | s) = p(y^i | s) \prod_{n=1}^{l} p(y_n^s | s).$$

We train the models end-to-end by minimizing the cross-entropy loss. Table 4.3 shows an example of input and output to train the joint intent and slot tagging models. Since type-I and type-II slots have different characteristics as discussed in Section 4.2.2 and overlap, we train two separate sequential tagging models for type-I and type-II slots to keep the baseline models simple.[5] We use BiLSTM [135, 136], Transformer [137], BERT [137], and RoBERTa [138] as encoder to form the sequence tagging models.

Besides, we consider an embedding based baseline where the input word embeddings are fed to the softmax classifiers. The special token (w_0 = [CLS]) embedding is formed by applying average pooling over the input word embeddings. We train WordPiece embeddings with a 30,000 token vocabulary [110] using *fastText* [112] based on a corpus of 130,000 privacy policies collected from apps on the Google Play Store [37]. We use the hidden state corresponding to the first WordPiece of a token to predict the target slot labels.

Conditional Random Field (CRF) helps structure prediction tasks, such as semantic role labeling [139] and named entity recognition [140]. Therefore, we model slot labeling jointly using a conditional random field (CRF) [141] (only interactions between two successive labels are considered). We refer the readers to [142] for details.

[5] Span enumeration based techniques [133, 134] can be utilized to perform tagging both types of slots jointly, and we leave this as future work.

4.3.2 Sequence-to-Sequence Learning

Recent works in semantic parsing [131, 132, 143] formulate the task as sequence-to-sequence (Seq2Seq) learning. Taking this as a motivation, we investigate the scope of Seq2Seq learning for joint intent classification and slot filling for privacy policy sentences. In Table 4.3, we show an example of encoder input and decoder output used in Seq2Seq learning. We form the target sequences by following the template: [IN:LABEL [SL:LABEL $w_1, \ldots, w_m$] . . .]. During inference, we use greedy decoding and parse the decoded sequence to extract intent and slot labels. Note that we only consider text spans in the decoded sequences that are surrounded by "[]"; the rest are discarded. Since our proposed PolicyIE corpus consists of a few thousand examples, instead of training Seq2Seq models from scratch, we fine-tune pre-trained models as the baselines. Specifically, we consider five state-of-the-art models: MiniLM [144], UniLM [145], UniLMv2 [146], MASS [147], and BART [148].

Table 4.5: Test set performance of the Seq2Seq models on PolicyIE corpus.

Model	# param (in millions)	Intent F1	Type-I		Type-II	
			Slot F1	EM	Slot F1	EM
Human	-	96.5	84.3	56.6	62.3	55.6
MiniLM	33	$83.9_{\pm 0.3}$	$52.4_{\pm 1.5}$	$19.8_{\pm 1.6}$	$40.4_{\pm 0.4}$	$27.9_{\pm 1.6}$
UniLM	110	$83.6_{\pm 0.5}$	$58.2_{\pm 0.7}$	$28.6_{\pm 1.2}$	$53.5_{\pm 1.4}$	$\mathbf{35.4}_{\pm 1.9}$
UniLMv2	110	$\mathbf{84.7}_{\pm 0.5}$	$61.4_{\pm 0.9}$	$\mathbf{29.9}_{\pm 1.2}$	$53.5_{\pm 1.5}$	$33.5_{\pm 1.5}$
MASS	123	$81.8_{\pm 1.2}$	$54.1_{\pm 2.5}$	$21.3_{\pm 2.0}$	$44.9_{\pm 1.2}$	$25.3_{\pm 1.3}$
BART	140	$83.3_{\pm 1.1}$	$53.6_{\pm 1.7}$	$10.6_{\pm 1.7}$	$52.4_{\pm 2.7}$	$27.5_{\pm 2.2}$
	400	$83.6_{\pm 1.3}$	$\mathbf{63.7}_{\pm 1.3}$	$23.0_{\pm 1.3}$	$\mathbf{55.2}_{\pm 1.0}$	$31.6_{\pm 2.0}$

4.3.3 Setup

Implementation We use the implementation of BERT and RoBERTa from `transformers` API [149]. For the Seq2Seq learning baselines, we use their public implementations.[6][7][8] We *train* BiLSTM, Transformer baseline models and *fine-tune* all the other baselines for 20 epochs and choose the best checkpoint based on validation performance. From 4,209 training examples, we use 4,000 examples for training (∼95%) and 209 examples for validation (∼5%). We tune the learning rate in [1e-3, 5e-4, 1e-4, 5e-5, 1e-5] and set the batch size to 16 in all our experiments (to fit in one GeForce GTX 1080 GPU with 11gb memory). We train (or fine-tune) all the models five times with different seeds and report average performances.

[6] `https://github.com/microsoft/unilm`
[7] `https://github.com/microsoft/MASS`
[8] `https://github.com/pytorch/fairseq/tree/master/examples/bart`

Evaluation Metrics To evaluate the baseline approaches, we compute the F1 score for intent classification and slot filling tasks.[9] We also compute an exact match (EM) accuracy (if the predicted intent matches the reference intent and slot F1 = 1.0).

Human Performance is computed by considering each annotator's annotations as predictions and the adjudicated annotations as the reference. The final score is an average across all annotators.

4.4 Experiment Results & Analysis

We aim to address the following questions.

1. How do the two modeling approaches perform on our proposed dataset (Section 4.4.1)?
2. How do they perform on different intent and slot types (Section 4.4.2)?
3. What type of errors do the best performing models make (Section 4.4.3)?

4.4.1 Main Results

Sequence Tagging The overall performances of the sequence tagging models are presented in Table 4.4. The pre-trained models, BERT and RoBERTa, outperform other baselines by a large margin. Using conditional random field (CRF), the models boost the slot tagging performance with a slight degradation in intent classification performance. For example, RoBERTa + CRF model improves over RoBERTa by 2.8% and 3.9% in terms of type-I slot F1 and EM with a 0.5% drop in intent F1 score. The results indicate that predicting type-II slots is difficult compared to type-I slots as they differ in length (type-I slots are mostly phrases, while type-II slots are clauses) and are less frequent in the training examples. However, the EM accuracy for type-I slots is lower than type-II slots due to more type-I slots ($\sim$4.75) than type-II slots ($\sim$1.38) on average per sentence. Note that if models fail to predict one of the slots, EM will be zero.

Seq2Seq Learning Seq2Seq models predict the intent and slots by generating the labels and spans following a template. Then we extract the intent and slot labels from the generated sequences. The experiment results are presented in Table 4.5. To our surprise, we observe that all the models perform well in predicting intent and slot labels. The best performing model is BART (according to

[9] We use a micro average for intent classification.

Table 4.6: Test performance of the RoBERTa and BART model for each intent type.

Intent labels	Intent F1	Slot F1	
		Type-I	Type-II
RoBERTa			
Data Collection	$74.1_{\pm 1.1}$	$59.8_{\pm 0.8}$	$28.9_{\pm 2.7}$
Data Sharing	$67.2_{\pm 2.0}$	$53.6_{\pm 5.7}$	$34.4_{\pm 3.4}$
Data Storage	$61.7_{\pm 3.6}$	$40.1_{\pm 3.7}$	$31.6_{\pm 3.1}$
Data Security	$68.9_{\pm 2.9}$	$53.9_{\pm 4.9}$	$21.9_{\pm 2.5}$
BART			
Data Collection	$73.5_{\pm 2.3}$	$67.0_{\pm 4.2}$	$56.2_{\pm 2.8}$
Data Sharing	$70.4_{\pm 2.7}$	$61.2_{\pm 1.6}$	$53.5_{\pm 3.4}$
Data Storage	$63.1_{\pm 4.7}$	$56.2_{\pm 8.2}$	$64.9_{\pm 2.5}$
Data Security	$67.2_{\pm 3.9}$	$66.0_{\pm 2.2}$	$32.8_{\pm 1.3}$

slot F1 score) with 400 million parameters, outperforming its smaller variant by 10.1% and 2.8% in terms of slot F1 for type-I and type-II slots, respectively.

Sequence Tagging vs. Seq2Seq Learning It is evident from the experiment results that Seq2Seq models outperform the sequence tagging models in slot filling by a large margin, while in intent classification, they are competitive. However, both the modeling approaches perform poorly in predicting all the slots in a sentence correctly, resulting in a lower EM score. One interesting factor is, the Seq2Seq models significantly outperform sequence tagging models in predicting type-II slots. Note that type-II slots are longer and less frequent, and we suspect conditional text generation helps Seq2Seq models predict them accurately. In comparison, we suspect that due to fewer labeled examples of type-II slots, the sequence tagging models perform poorly on that category (as noted before, we train the sequence tagging models for the type-I and type-II slots individually).

Next, we break down RoBERTa (w/ CRF) and BART's performances, the best performing models in their respective model categories, followed by an error analysis to shed light on the error types.

4.4.2 Performance Breakdown

Intent Classification In the PolicyIE corpus, 38% of the sentences fall into the first four categories: Data Collection, Data Sharing, Data Storage, Data Security, and the remaining belong to the Other category. Therefore, we investigate how much the models are confused in predicting the accurate intent label. We provide the confusion matrix of the models in Appendix. Due to an imbalanced distribution of labels, BART makes many incorrect predictions. We notice that BART is confused most between *Data Collection* and *Data Storage* labels. Our manual analysis reveals that BART is confused between slot labels {"Data Collector", "Data Holder"} and {"Data Retained", "Data

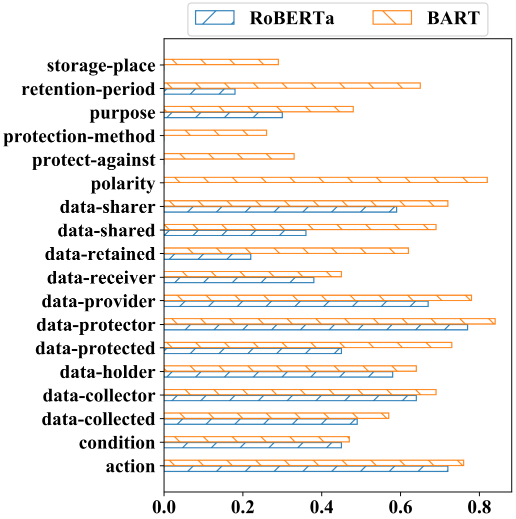


Figure 4.1: Test set performance (Recall score) on PolicyIE for the eighteen slot types.

Collected"} as they are often associated with the same text span. We suspect this leads to BART's confusion. Table 4.6 presents the performance breakdown across intent labels.

Slot Filling We breakdown the models' performances in slot filling under two settings. First, Table 4.6 shows slot filling performance under different intent categories. Among the four classes, the models perform worst on slots associated with the "Data Security" intent class as PolicyIE has the lowest amount of annotations for that intent category. Second, we demonstrate the models' performances on different slot types in Figure 4.1. RoBERTa's recall score for "polarity", "protect-against", "protection-method" and "storage-place" slot types is zero. This is because these slot types have the lowest amount of training examples in PolicyIE. On the other hand, BART achieves a higher recall score, specially for the "polarity" label as their corresponding spans are short.

We also study the models' performances on slots of different lengths. The results show that BART outperforms RoBERTa by a larger margin on longer slots (see Figure 4.2), corroborating our hypothesis that conditional text generation results in more accurate predictions for longer spans.

4.4.3 Error Analysis

We analyze the incorrect intent and slot predictions by RoBERTa and BART. We categorize the errors into seven types. Note that a predicted slot is considered correct if its' label and span both match (*exact match*) one of the references. We characterize the error types as follows.

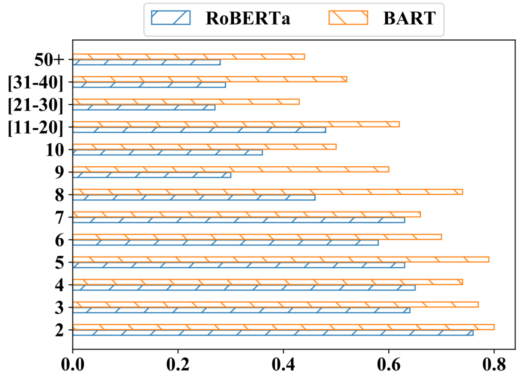


Figure 4.2: Test set performance (Recall score) on PolicyIE for slots with different length.

Table 4.7: Three examples showing different error types appeared in BART's predictions. + and − indicates the reference and predicted sequences, respectively. Best viewed in color.

+ [IN:data-collection-usage [SL:data-provider.third-party-entity *third parties*] [SL:action collect] [SL:data-provider.user *your*] [SL:data-collected.data-general *information*] [SL:data-collector.first-party-entity *us*]]

− [IN:data-sharing-disclosure [SL:data-receiver.third-party-entity *third parties*] [SL:action *share*] [SL:data-provider.user *your*] [SL:data-shared.data-general *information*] [SL:data-sharer.first-party-entity *us*] [SL:condition *where applicable*] [SL:condition *based on their own privacy policies*]]

Error types: Wrong Intent (WI), Wrong Label (WL), Wrong Slot (WS), Spurious Slot (SS)

+ [. . . [SL:data-provider.third-party-entity *third parties*] [SL:condition *it is allowed by applicable law or according to your agreement with third parties*]]

− [. . . [SL:condition *allowed by applicable law or according to your agreement with third parties*]]

Error types: Wrong Boundary (WB), Missing Slot (MS)

+ [. . . [SL:data-receiver.third-party-entity *social media and other similar platforms*] . . .]

− [. . . [SL:data-receiver.third-party-entity *social media*] [SL:data-receiver.third-party-entity *other similar platforms*] . . .]

Error types: Wrong Split (WSp)

Table 4.8: Counts for each error type on the test set of PolicyIE using RoBERTa and BART models.

Error	RoBERTa	BART
Wrong Intent	161	178
Spurious Slot	472	723
Missing Slot	867	517
Wrong Boundary	130	160
Wrong Slot	103	143
Wrong Split	32	27
Wrong Label	18	19
Total Slots	2,198	2,198
Correct Prediction	1,064	1,361
Total Errors	1,622	1,589
Total Predictions	2,686	2,950

1. **Wrong Intent (WI)**: The predicted intent label does not match the reference intent label.

2. **Missing Slot (MS)**: None of the predicted slots *exactly* match a reference slot.

3. **Spurious Slot (SS)**: Label of a predicted slot does not match any of the references.

4. **Wrong Split (WSp)**: Two or more predicted slot spans with the same label could be merged to match one of the reference slots. A merged span and a reference span may *only* differ in punctuations or stopwords (e.g., and).

5. **Wrong Boundary (WB)**: A predicted slot span is a sub-string of the reference span or vice versa. The slot label must exactly match.

6. **Wrong Label (WL)**: A predicted slot span matches a reference, but the label does not.

7. **Wrong Slot (WS)**: All other types of errors fall into this category.

We provide one example of each error type in Table 4.7. In Table 4.8, we present the counts for each error type made by RoBERTa and BART models. The two most frequent error types are SS and MS. While BART makes more SS errors, RoBERTa suffers from MS errors. While both the models are similar in terms of total errors, BART makes more correct predictions resulting in a higher Recall score, as discussed before. One possible way to reduce SS errors is by penalizing more on wrong slot label prediction than slot span. On the other hand, reducing MS errors is more challenging as many missing slots have fewer annotations than others. We provide more qualitative examples in Appendix (see Table 4.11 and 4.12).

In the error analysis, we exclude the test examples (sentences) with the intent label “Other” and no slots. Out of 1,041 test instances in PolicyIE, there are 682 instances with the intent label “Other”. We analyze RoBERTa and BART’s predictions on those examples separately to check if the models predict slots as we consider them as spurious slots. While RoBERTa meets our expectation of performing highly accurate (correct prediction for 621 out of 682), BART also correctly predicts 594 out of 682 by precisely generating “[IN:Other]”. Overall the error analysis aligns with our anticipation that the Seq2Seq modeling technique has promise and should be further explored in future works.

4.5 Related Work

Automated Privacy Policy Analysis Automating privacy policy analysis has drawn researchers' attention as it enables the users to know their rights and act accordingly. Therefore, significant research efforts have been devoted to understanding privacy policies. Earlier approaches [150] designed rule-based pattern matching techniques to extract specific types of information. Under the *Usable Privacy Project* [114], several works have been done [1, 36, 38, 115–118, 151, 152]. Notable works leveraging NLP techniques include text alignment [40, 120], text classification [1, 37, 38], and question answering (QA) [27, 37, 108, 121]. [41] is the most closest to our work that used named entity recognition (NER) modeling technique to extract third party entities mentioned in policy documents.

Our proposed PolicyIE corpus is distinct from the previous privacy policies benchmarks: OPP-115 [1] uses a hierarchical annotation scheme to annotate text segments with a set of data practices and it has been used for multi-label classification [1,37] and question answering [27,37,153]; PrivacyQA [108] frame the QA task as identifying a list of relevant sentences from policy documents. Recently, [42] created a dataset by tagging documents from OPP-115 for privacy practices and uses NER models to extract them. In contrast, PolicyIE is developed by following semantic parsing benchmarks, and we model the task following the NLP literature.

Intent Classification and Slot Filling Voice assistants and chat-bots frame the task of natural language understanding via classifying intents and filling slots given user utterances. Several benchmarks have been proposed in literature covering several domains, and languages [132, 154–159]. Our proposed PolicyIE corpus is a new addition to the literature within the security and privacy domain. PolicyIE enables us to build conversational solutions that users can interact with and learn about privacy policies.

4.6 Conclusion

This chapter aims to stimulate research on automating information extraction from privacy policies and reconcile it with users' understanding of their rights. We present PolicyIE, an intent classification and slot filling benchmark on privacy policies with two alternative neural approaches as baselines. We perform a thorough error analysis to shed light on the limitations of the two baseline approaches.

We hope this contribution would call for research efforts in the specialized privacy domain from both privacy and NLP communities.

4.7 Appendix of Chapter 4

Table 4.9: Slots and their associated attributes. “None” indicates there are no attributes for the those slots.

Type-I slots	Attributes
Action	None
Data Provider	(1) User (2) Third party entity
Data Collector	(1) First party entity
Data Collected	(1) General Data (2) Aggregated/Non-identifiable data (3) Contact data (4) Financial data (5) Location data (6) Demographic data (7) Cookies, web beacons and other technologies (8) Computer/Device data (9) User online activities/profiles (10) Other data
Data Sharer	(1) First party entity
Data Shared	(1) General Data (2) Aggregated/Non-identifiable data (3) Contact data (4) Financial data (5) Location data (6) Demographic data (7) Cookies, web beacons and other technologies (8) Computer/Device data (9) User online activities/profiles (10) Other data
Data Receiver	(1) Third party entity
Data Holder	(1) First party entity (2) Third party entity
Data Retained	(1) General Data (2) Aggregated/Non-identifiable data (3) Contact data (4) Financial data (5) Location data (6) Demographic data (7) Cookies, web beacons and other technologies (8) Computer/Device data (9) User online activities/profiles (10) Other data
Storage Place	None
Retention Period	None
Data Protector	(1) First party entity (2) Third party entity
Data Protected	(1) General Data (2) Aggregated/Non-identifiable data (3) Contact data (4) Financial data (5) Location data (6) Demographic data (7) Cookies, web beacons and other technologies (8) Computer/Device data (9) User online activities/profiles (10) Other data
Protect Against	Security threat
Type-II slots	Attributes
Purpose	(1) Basic service/feature (2) Advertising/Marketing (3) Legal requirement (4) Service operation and security (5) Personalization/customization (6) Analytics/research (7) Communications (8 Merge/Acquisition (9) Other purpose
Condition	None
Polarity	(1) Negation
Protection Method	(1) General safeguard method (2) User authentication (3) Access limitation (5) Encryptions (6) Other protection method

Table 4.10: Privacy practices and the associated slots with their distributions. "X / Y" indicates there are X instances in the train set and Y instances in the test set.

Privacy Practices	Data Collection/Usage	Data Sharing/Disclosure	Data Storage/Retention	Data Security/Protection
Type-I slots				
Action	750 / 169	344 / 70	198 / 57	102 / 31
Data Provider	784 / 172	247 / 54	139 / 44	65 / 20
Data Collector	653 / 151	-	-	-
Data Collected	1833 / 361	-	-	-
Data Sharer	-	288 / 54	-	-
Data Shared	-	541 / 110	-	-
Data Receiver	-	456 / 115	-	-
Data Holder	-	-	192 / 59	-
Data Retained	-	-	291 / 119	-
Storage Place	-	-	70 / 21	-
Retention Period	-	-	101 / 17	-
Data Protector	-	-	-	105 / 31
Data Protected	-	-	-	119 / 34
Protect Against	-	-	-	49 / 15
Type-II slots				
Purpose	894 / 193	327 / 65	168 / 40	5 / 0
Condition	337 / 81	154 / 26	81 / 25	43 / 7
Polarity	50 / 15	21 / 1	22 / 1	18 / 5
Protection Method	-	-	-	143 / 35
# of slots	5301 / 1142	2378 / 495	1262 / 383	649 / 178
# of sequences	919 / 186	380 / 83	232 / 61	103 / 29

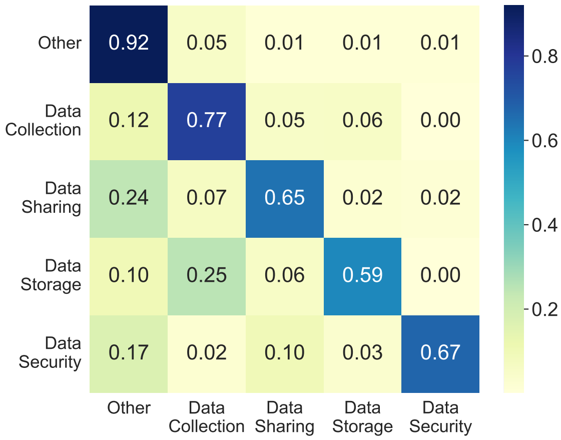


Figure 4.3: Confusion matrix for intent classification using the RoBERTa model.

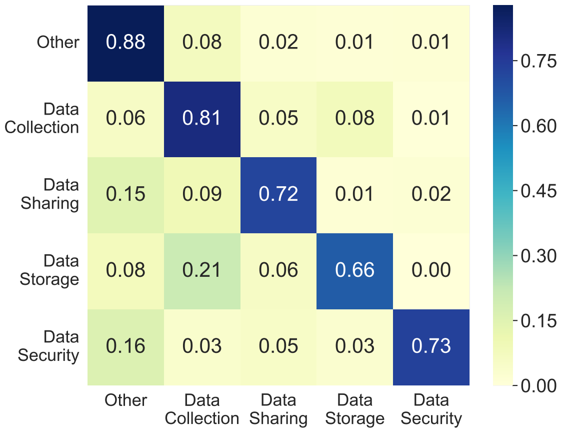


Figure 4.4: Confusion matrix for intent classification using the BART model.

Table 4.11: Sample RoBERTa and BART predictions of Type-I slots. (✓) and (✗) indicates correct and incorrect predictions, respectively. Precision (P) and recall (R) score is reported for each example in the left column.

		Label	Text
Ground truth		data-holder.first-party-entity	We
		action	keep
		data-retained.data-general	records
		retention-period.retention-period	a period of no more than 6 years
RoBERTa (P:1.0, R: 0.75)	✓	data-holder.first-party-entity	We
	✓	action	keep
	✓	retention-period.retention-period	a period of no more than 6 years
BART (P:1.0, R: 1.0)	✓	data-holder.first-party-entity	We
	✓	action	keep
	✓	data-retained.data-general	records
	✓	retention-period.retention-period	a period of no more than 6 years
Ground truth		data-collector.first-party-entity	We
		action	access
		data-collected.data-general	information
RoBERTa (P:0.0, R: 0.0)	✗	data-sharer.first-party-entity	We
	✗	data-shared.data-general	information
BART (P:0.0, R: 0.0)	✗	data-sharer.first-party-entity	We
	✗	action	disclose
	✗	data-shared.data-general	information
Ground truth		data-sharer.first-party-entity	Marco Polo
		data-receiver.third-party-entity	third party
		data-shared.data-general	Personal Information
		data-provider.user	users
		action	transferred
RoBERTa (P:0.6, R: 0.6)	✗	data-receiver.third-party-entity	Marco
	✗	data-sharer.first-party-entity	our
	✓	data-receiver.third-party-entity	third party
	✓	data-shared.data-general	Personal Information
	✓	action	transferred
BART (P:0.83, R: 1.0)	✓	data-sharer.first-party-entity	Marco Polo
	✓	data-receiver.third-party-entity	third party
	✓	data-shared.data-general	Personal Information
	✗	data-sharer.first-party-entity	us
	✓	data-provider.user	users
	✓	action	transferred
Ground truth		data-sharer.first-party-entity	We
		data-receiver.third-party-entity	third parties
		action	provide
		data-shared.data-general	information
RoBERTa (P:1.0, R: 1.0)	✓	data-sharer.first-party-entity	We
	✓	data-receiver.third-party-entity	third parties
	✓	action	provide
	✓	data-shared.data-general	information
BART (P:0.25, R: 0.25)	✗	data-collector.first-party-entity	We
	✗	data-provider.third-party-entity	third parties
	✓	action	provide
	✗	data-collected.data-general	information

Table 4.12: Sample RoBERTa and BART predictions of Type-II slots. (✓) and (✗) indicates correct and incorrect predictions, respectively. Precision (P) and recall (R) score is reported for each example in the left column.

Ground truth		[Label] condition [Text] you use our product and service or view the content provided by us
RoBERTa (P:1.0, R: 1.0)	✓	[Label] condition [Text] you use our product and service or view the content provided by us
BART (P:1.0, R: 1.0)	✓	[Label] condition [Text] you use our product and service or view the content provided by us
Ground truth		[Label] purpose.other [Text] their own purposes
		[Label] purpose.advertising-marketing [Text] inform advertising related services provided to other clients
RoBERTa (P:0.0, R: 0.0)	✗	[Label] None [Text] None
BART (P:1.0, R: 1.0)	✓	[Label] purpose.other [Text] their own purposes
	✓	[Label] purpose.advertising-marketing [Text] inform advertising related services provided to other clients
Ground truth		[Label] purpose.personalization-customization [Text] provide more tailored services and user experiences
		[Label] purpose.basic-service-feature [Text] remembering your account identity
		[Label] purpose.service-operation-and-security [Text] analyzing your account 's security
		[Label] purpose.analytics-research [Text] analyzing your usage of our product and service
		[Label] purpose.advertising-marketing [Text] advertisement optimization (helping us to provide you with more targeted advertisements instead of general advertisements based on your information)
RoBERTa (P:0.17, R: 0.2)	✗	[Label] purpose.basic-service-feature [Text] provide
	✗	[Label] purpose.other [Text] purposes
	✗	[Label] purpose.analytics-research [Text] remembering your account identity
	✗	[Label] purpose.analytics-research [Text] analyzing your account 's security
	✓	[Label] purpose.analytics-research [Text] analyzing your usage of our product and service
	✗	[Label] purpose.advertising-marketing [Text] advertisement optimization
BART (P:0.43, R: 0.6)	✓	[Label] purpose.personalization-customization [Text] provide more tailored services and user experiences
	✗	[Label] purpose.service-operation-and-security [Text] remembering your account identity
	✓	[Label] purpose.service-operation-and-security [Text] analyzing your account 's security
	✓	[Label] purpose.analytics-research [Text] analyzing your usage of our product and service
	✗	[Label] purpose.advertising-marketing [Text] advertisement optimization
	✗	[Label] purpose.advertising-marketing [Text] provide you with more targeted advertisements instead of general advertisements
	✗	[Label] purpose.advertising-marketing [Text] based on your information

Part II

Understanding and Methods of Ethical AI

Chapter 5

Conditional Supervised Contrastive Learning for Fair Text Classification

5.1 Introduction

Recent progress in natural language processing (NLP) has led to its increasing use in various domains such as machine translation, virtual assistants, and social media monitoring. However, studies have demonstrated societal bias in existing NLP models [18,64,66,160–162]. In one major NLP application, text classification, bias is referred as the performance disparity of the trained classifiers over different demographic groups such as gender and ethnicity [163]. Such bias poses potential risks: for example, if toxicity classification models in online social media platforms show disparate performance in different social groups, they will lead to increased silencing of under-served groups [69,164].

Meanwhile, an increasing line of work in contrastive learning (CL) has led to significant advances in representation learning [165–171]. The general idea of contrastive learning in these works is to learn representations such that similar examples stay close to each other while dissimilar ones are far apart. Inspired by those works, recent works [172–174] also propose to leverage contrastive learning to learn fair representations in classification. However, these works either lack theoretical justifications for the proposed approaches or simply adopt *demographic parity* [51] as the fairness criterion, which eliminates the perfect classifier in the common scenario when the *base rates* differ among demographic groups [52, 175].

In this chapter, we aim to mitigate bias in text classification models via contrastive learning. In particular, we adopt the fairness notion, *equalized odds* (EO) [52], which asks for equal true positive rates (TPRs) and false positive rates (FPRs) across different demographic groups [176]. Based on information-theoretic concepts, we bridge the problem of learning fair representations with equalized odds constraint with contrastive learning objectives. We then propose an algorithm, called *conditional supervised contrastive learning*, to learn fair text classifiers.

Empirically, we conduct experiments on two text classification datasets (e.g., toxic comment classification and biography classification) to show the proposed methods (1) can flexibly tune the trade-offs between main task performance and the fairness constraint; (2) achieve the best trade-offs between main task performance and equalized odds compared to the existing bias mitigation approaches in text classification; (3) are stable to different hyperparameter settings, such as data augmentations, temperatures, and batch sizes. To the best of our knowledge, our work is the first to both theoretically and empirically study how to ensure the EO constraint via contrastive learning in text classification.

5.2 Background

We use $X \in \mathcal{X}$ and $Y \in \mathcal{Y}$ to denote the random variables for the input text and the categorical label for the main task, respectively. Furthermore, $A \in \mathcal{A}$ is the sensitive attribute (protected group) associated with the input text X (e.g., the gender information in the occupation classification task). The corresponding lowercase letters denote the instantiation of the random variables. Given a text encoder $f : \mathcal{X} \to \mathcal{Z}$ (e.g., BERT [177]) and a classifier $g : \mathcal{Z} \to \mathcal{Y}$, we first transform the input text X into latent representation Z via f, and Z is used to give a prediction $\hat{Y}$ via g (i.e., $X \xrightarrow{f} Z \xrightarrow{g} \hat{Y}$).

In the context of contrastive learning, data augmentation strategies have been widely adopted. Let $\mathcal{T}$ be a set of data augmentations and X' be the augmented input given the data augmentation $t(\cdot)$: $X' = t(X),\ t \sim \mathcal{T}$, where we assume that the augmentation t is sampled uniformly at random from $\mathcal{T}$. Similarly, we have $X' \xrightarrow{f} Z' \xrightarrow{g} \hat{Y}'$. Let H denote the entropy and I denote the mutual

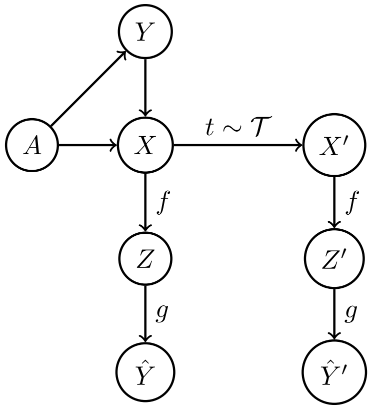


Figure 5.1: Graphical model of the dependencies between input variables and outputs. Note that we only assume there is a joint distribution over X, Y, and A from which the data are sampled, so the figure only shows one case of the dependencies over X, Y, and A.

information, e.g., $H(Z \mid Z', Y)$ is the conditional entropy of Z given Z' and Y, and $I(Z'; Z \mid Y)$ is the conditional mutual information of Z' and Z given Y. Due to the space limit, we refer readers to [178] for more background knowledge of the related notions (entropy and mutual information) in information theory.

We assume there is a joint distribution over X, Y, and A from which the data are sampled. Figure 5.1 shows the graphical model of the dependencies between input variables and outputs. We also assume that the sensitive attribute A is available only during model training, but it is not available during the testing phase. As a result, any post-processing methods that leverage sensitive attributes for bias mitigation during the testing phase are not feasible in our setting. We use equalized odds—a fairness criterion for classification problems—in this chapter.

Definition 5.2.1 (Equalized Odds [52])**.** A model satisfies equalized odds if $\hat{Y} \perp A \mid Y$.

At a high-level, EO asks the model prediction to be independent of the sensitive attribute conditioned on the task label. If a model perfectly satisfies equalized odds, the differences of true positive rates and false positive rates across demographic groups will be 0. Equivalently, it also implies $I(\hat{Y}; A \mid Y) = 0$. As a real-world example to motivate the use of EO as a notion of fairness, consider online comment toxicity classification. In this case, false positive cases (benign text comments marked as toxic) can be seen as unintentional censoring, and false negative cases (toxic text comments marked as benign) might result in debates and discomforts [179].

In contrast to another well-known group fairness definition, i.e., demographic parity, EO does not require positive prediction rates to be the same across different demographic groups, which could possibly severely downgrade the model performance when the sensitive attribute is correlated to the task label [52, 175].

5.3 Our Method

In this section, we first theoretically connect learning fair representations with contrastive learning (Sec. 5.3.1). In particular, we first show that learning fair representations for equalized odds requires the minimization of $I(Z'; Z \mid Y)$ and the simultaneous maximization of $I(Z'; Z \mid A, Y)$. To this end, we provide an upper bound of $I(Z'; Z \mid Y)$ and a lower bound of $I(Z'; Z \mid A, Y)$ to relax the original objective and then establish a relationship between the bounds and the (conditional) supervised contrastive learning objectives. Finally, inspired by our theoretical analysis, we design two practical methods for learning fair representations (Sec. 5.3.2). Due to the space limit, we defer all detailed proofs to Appendix 5.7.1.

5.3.1 Connections between Contrastive Learning and Learning Fair Representations

In order to learn a model (text encoder followed by classifier) to satisfy equalized odds, we aim to learn a latent representation Z such that $Z \perp A \mid Y$. From an information-theoretic perspective, it suffices to minimize the conditional mutual information $I(Z; A \mid Y)$ to ensure EO due to the celebrated data-processing inequality. We identify a connection between contrastive learning and learning fair representations when the representations enjoy certain benign structures. Next, we formally state the assumptions to characterize such a structure.

Assumption 5.3.1. Let Z and Z' be the corresponding features from X and X', respectively. We assume that there exists a positive constant $\epsilon > 0$, such that $H(Z \mid Z', Y) \leq \epsilon$.

At a high-level, Assumption 5.3.1 says that the learned features from the contrastive learning procedure is well conditionally aligned [180]. Specifically, given the label of a feature and its corresponding augmented feature, it is relatively easy to infer the corresponding positive pair used in the contrastive learning procedure. Note that the conditional entropy could be understood as the minimum inference error from this perspective [181]. Now, under Assumption 5.3.1, we provide the following lemma to characterize the relationship between Z, Z', A, and Y in terms of (conditional) mutual information.

Lemma 5.3.1. Under Assumption 5.3.1, given a set of data augmentations $\mathcal{T}$, let X' be the augmented input data where $X' = t(X),\ t \sim \mathcal{T}$. Assuming the following Markov chains $X \xrightarrow{f} Z \xrightarrow{g} \hat{Y}$ and $X' \xrightarrow{f} Z' \xrightarrow{g} \hat{Y}'$ hold, we have

$$I(Z'; Z \mid Y) - I(Z'; Z \mid A, Y) - \epsilon \leq I(Z; A \mid Y) \leq I(Z'; Z \mid Y) - I(Z'; Z \mid A, Y) + \epsilon.$$

Lemma 5.3.1 indicates that we can minimize $I(Z; A \mid Y)$ via (1) minimizing $I(Z'; Z \mid Y)$ and (2) maximizing $I(Z'; Z \mid A, Y)$. In what follows, we will present an upper (lower) bound to minimize $I(Z'; Z \mid Y)$ (maximize $I(Z'; Z \mid A, Y)$) and connect the bounds with contrastive learning objectives. We first provide an upper bound of $I(Z'; Z \mid Y)$.

Proposition 5.3.1. Given the assumptions in Lemma 5.3.1, we have

$$I(Z'; Z \mid Y) \leq -\mathbb{E}_{p(y)}\big[\mathbb{E}_{p(z'|y)}[\mathbb{E}_{p(z|y)}[\log p(z' \mid z, y)]]\big].$$

In order to better interpret the second term in Proposition 5.3.1, we define a similarity function $s(z', z; y)$ between z' and z for each y and assume $s(z', z; y) \propto p(z' \mid z, y)$ (i.e., the more similar z' and z are in the latent space given task label y, the more likely z' is generated by z via data augmentation)[1]. With this assumption, the upper bound provided in Proposition 5.3.1 implies that $I(Z'; Z \mid Y)$ can be minimized by encouraging similarity between any latent representations given the same task label, which is consistent with the goal of supervised contrastive loss [170]. Formally, given a batch of augmented examples $(x_i, y_i, a_i)_{i=1}^{2N}$ with size $2N$, where the last half examples of the batch are the augmented views of the first half and they share the same task labels (as well as the same sensitive attributes), i.e., $x_{i+N} = t(x_i)$ for $i \in [N]$ and $t \sim \mathcal{T}$. Let N_{y_i} be the total number of examples in the batch that have the same task label as y_i, then supervised contrastive loss takes the following form:

$$L_{\text{sup}} = \sum_{i=1}^{2N} \frac{1}{N_{y_i} - 1} \sum_{j=1}^{2N} \mathbf{1}_{i \neq j, y_i = y_j} \log(\ell_{ij}), \tag{5.1}$$

[1] In the remainder of the paper, we let $s(\cdot, \cdot) = s(\cdot, \cdot; y),\ \forall\ Y = y$ for the ease of practical implementations.

and ℓ_{ij} is defined as

$$\ell_{ij} = \frac{\exp\big(f(x_i)^\top f(x_j)/\tau\big)}{\sum_{k=1}^{2N} \mathbf{1}_{i\neq k} \exp\big(f(x_i)^\top f(x_k)/\tau\big)},$$

where τ is the temperature parameter, $\mathbf{1}_{i\neq k} = \mathbf{1}\{i \neq k\}$ and $\mathbf{1}\{\cdot\}$ is the indicator function, and the similarity function is $s\big(f(x_i), f(x_j)\big) = \exp\big(f(x_i)^\top f(x_j)/\tau\big)$. Supervised contrastive loss L_{sup} aims to encourage similarity between different examples with the same task label and discourage the ones having different labels. Thus, we minimize L_{sup} to approximately minimize $I(Z'; Z \mid Y)$. Next, we provide a lower bound of $I(Z'; Z \mid A, Y)$ for the maximization of $I(Z'; Z \mid A, Y)$.

Proposition 5.3.2. Given the assumptions in Lemma 5.3.1, define conditional supervised InfoNCE as CS-InfoNCE, i.e.,

$$\underbrace{\sup_s \mathbb{E}_{p(a,y)}\Big[\mathbb{E}_{p(z_i', z_i \mid a,y)^{\otimes N}}\big[\log \frac{\exp(s(z_i', z_i))}{\frac{1}{N}\sum_{j=1}^{N} \exp(s(z_i', z_j))}\big]\Big]}_{\text{CS-InfoNCE}},$$

where $p(\cdot)^{\otimes N}$ denotes the probability distributions of N independent examples and $s(\cdot,\cdot)$ is any similarity function that measure the similarity of z_i' and z_i. Then, we have

$$\text{CS-InfoNCE} \leq I(Z'; Z \mid A, Y).$$

Proposition 5.3.2 indicates the maximization of CS-InfoNCE leads to the maximization of $I(Z'; Z \mid A, Y)$. Given the examples that share the same task label and sensitive attribute, CS-InfoNCE encourages the similarity between different views of the same examples while discouraging others. Note that all positive and negative examples w.r.t. the anchoring example share the same task labels and sensitive attributes. Given the same batch of examples $(x_i, y_i, a_i)_{i=1}^{2N}$ with size $2N$, we can formulate the contrastive objective as

$$L_{\text{CS-InfoNCE}} = \sum_{i=1}^{2N} \frac{1}{N_{a_i,y_i} - 1} \log(\ell_i), \tag{5.2}$$

and ℓ_i is defined as

$$\ell_i = \frac{\exp\big(f(x_i)f(x_i')/\tau\big)}{\sum_{k=1}^{2N} \mathbf{1}_{i\neq k, a_i=a_k, y_i=y_k} \exp\big(f(x_i)f(x_k)/\tau\big)},$$

where N_{a_i,y_i} is the total number of examples in the batch that have the same task label and sensitive attribute as y_i and a_i, and x_i and x_i' are the different views of the same example.

Interpretation of L_{sup} and $L_{\text{CS-InfoNCE}}$ in learning fair representations. In learning fair representations, the role of L_{sup} is to learn aligned and uniform representations [180] for each task label, while the role of $L_{\text{CS-InfoNCE}}$ is to mix different examples that share the same sensitive attribute within each task label. In an ideal case where $L_{\text{sup}} = 0$, each data point that shares the same task label in the latent representation collapse to a single point, and the perfect representations are learned. In this case, $L_{\text{CS-InfoNCE}} = 0$ as well. However, this is not always the case in practice and $L_{\text{CS-InfoNCE}}$ is used to "mix" the examples in the same sensitive attribute within the same task label in the latent space. In Appendix 5.7.11, we provide T-SNE visualization [182] of the text embeddings using different training objectives to help better understand our methods.

5.3.2 Practical Implementations

The existing contrastive representation learning approaches fall into two categories: two-stage methods [170, 183] and one-stage methods [184, 185]. Two-stage methods first pretrain the encoder in the first stage using the contrastive objective, then fix the encoder, and fine-tune the classifier using cross-entropy (CE) loss in the second stage. One-stage methods train both encoder and classifier using CE loss and contrastive loss end-to-end. Following the previous settings, we also implement our methods in these two ways. For the two-stage CL method, we first pretrain the text encoder using the following loss function in the first stage:

$$L_{\text{sup}} + \lambda \cdot L_{\text{CS-InfoNCE}}, \tag{5.3}$$

then we fix the pretrained encoder, and fine-tune classifier using CE loss. Note that $\lambda \geq 0$ controls the intensity of $L_{\text{CS-InfoNCE}}$. For the one-stage CL method, similar to [184], we formulate the loss function as:

$$(1 - \gamma) \cdot L_{\text{CE}} + \gamma \cdot L_{\text{sup}} + \lambda \cdot L_{\text{CS-InfoNCE}}, \tag{5.4}$$

where $\gamma \in [0, 1]$ controls the relative weight of L_{sup} compared to L_{CE}. The major advantage of our approach is that it can be directly substituted into existing NLP pipelines that use the "pretrain-and-finetune" paradigm popularized by large language models such as BERT. NLP practitioners can swap the fair CL finetuner into these pipelines to boost model fairness at low cost, with robust behavior against hyperparameter choices (see Sec. 5.4.2). Whereas large language models made it simple to build models with high performance, fair CL makes it simple to build models with high performance and fairness.

5.4 Experiments

In this section, we conduct experiments to investigate the following research questions:

RQ 1. How can we control the trade-offs between model classification performance and fairness via conditional supervised contrastive learning?

RQ 2. How do conditional supervised contrastive learning methods perform in terms of trade-offs between model performance and fairness compared to other in-processing bias mitigation methods in text classification?

RQ 3. Is conditional supervised contrastive learning sensitive to hyperparameter changes?

5.4.1 Experimental Setup

Datasets. We perform experiments using the following two datasets (see Appendix 5.7.5 for more details of the datasets and the data prepossessing pipelines):

- `Jigsaw-toxicity`[2] is a dataset for online comment toxicity classification. The main task of the dataset is to determine if the online comment is toxic, and we use "race and ethnicity" as the sensitive attribute (e.g., whether "black" identity is mentioned in the comment text or not).
- `Biasbios` [186] is a dataset for occupation classification. The main task of the dataset is to determine the peoples' occupations given their biographies. The sensitive attribute is binary gender (i.e., male and female).

Evaluation Metrics. We evaluate our model based on model classification performance and EO fairness. We use the F1 score for model performance and True Positive Equality Difference + False Positive Equality Difference [164] for EO fairness:

$$\Delta_{\mathrm{TPR}} = \sum_a |\mathrm{TPR}_a - \mathrm{TPR}_{\mathrm{overall}}|,$$

$$\Delta_{\mathrm{FPR}} = \sum_a |\mathrm{FPR}_a - \mathrm{FPR}_{\mathrm{overall}}|,$$

[2] The dataset is available at `https://www.kaggle.com/competitions/jigsaw-unintended-bias-in-toxicity-classification`.

where Δ_{TPR} (Δ_{FPR}) is the true positive rate (false negative rate) for sensitive attribute a and $\text{TPR}_{\text{overall}}$ ($\text{FPR}_{\text{overall}}$) is the overall true positive rate (false negative rate). Following [187], we define equalized odds gap $\Delta_{\text{EO}} = \Delta_{\text{TPR}} + \Delta_{\text{FPR}}$, since equalized odds aligns with $\Delta_{\text{TPR}} + \Delta_{\text{FPR}}$, and when it is satisfied, $\Delta_{\text{TPR}} = \Delta_{\text{FPR}} = 0$ [188]. Note that when $|\mathcal{Y}| > 2$, Δ_{EO} will be summed over each value in $\mathcal{Y}$ since TPR and FPR are defined over each class (i.e., $\Delta_{\text{EO}} = \sum_y \Delta_{\text{EO}}^y$).

Implementations and Baselines. In our experiments, we use BERT [177] (`bert-base-uncased` as the text encoder followed by a two-layer MLP as the classifier)[3]. As suggested by previous works [170, 171], the performance of contrastive learning is closely related to the choice of the following hyperparameters: (1) temperature, (2) (pre-training) batch size, and (3) data augmentation strategy. Thus, we conduct a grid-based hyperparameter search for temperature $\tau \in \{0.1, 0.5, 1.0, 2.0\}$, (pre-training) batch size $\text{bsz} \in \{32, 64, 128, 256\}$, and data augmentation strategy $t \in$ {EDA, back translation, CLM insert, CLM substitute} (see Appendix 5.7.5 for the detailed description of different augmentation strategies) for both two-stage CL and one-stage CL. We also conduct grid search of $\gamma \in \{0.1, 0.3, 0.7, 0.9\}$ in Eq. (5.4) for one-stage CL. In Appendix 5.7.5, we provide the remaining hyperparameter details (e.g., learning rate, training epochs, optimizer). Since it is not feasible to train large language models with large batch sizes via contrastive objectives given limited GPU memory, we use the gradient cache technique [189] to adapt our implementations to limited GPU memory settings.

We compare our methods with the following baselines, which have been empirically demonstrated effective for bias mitigation in text classification:

(1) Adversarial training [190]: Following the encoder + classifier setting, adversarial training leverages a discriminator to learn latent representations oblivious to the sensitive attribute. Note that the original adversarial training method is tailored for demographic parity and it is well known that demographic parity and equal odds in given different base rates [191, 192]. To this end, we use the conditional learning techniques [176, 193] to adapt adversarial training for equalized odds.

(2) Adversarial training with diverse adversaries (diverse adversaries) [194]: Adversarial training with diverse adversaries improves adversarial training by using an ensemble of discriminators and encourages the discriminator to learn orthogonal representations. Similar to adversarial training, we also apply the conditional learning techniques for learning the adversarial discriminators.

[3] We use the huggingface transformer implementation: `https://github.com/huggingface/transformers`.

(3) Iterative null-space projection (INLP) [195]: Given a pretrained text encoder (we use CE loss to pretrain the text encoder and drop the prediction head using the validation set), INLP learns a linear guarding layer on top of the pretrained text encoder to filter the sensitive information and fine-tune the classifier given the pretrained text encoder and INLP. INLP learns the linear guarding layer by projecting the parameter matrices of linear classifiers (e.g., SVM) to their null spaces iteratively. The training data of linear classifiers are the latent representations of input texts and sensitive attributes. In order to tailor INLP for equaled odds, [195] learns the linear classifier given the data from the same class each round.

We also use CE training (CE) as a baseline. Except for INLP, all methods we test in our experiments train the text encoder (e.g., BERT) directly, while INLP is a post-hoc debiasing method given a text encoder. In a sense, INLP is orthogonal to other methods since it tries to remove group-specific information after we learn the representations, while other methods learn the fair representations directly. We run each experiment with five different seeds and report the mean and standard deviation values for each evaluation metric.

5.4.2 Results and Analysis

RQ 1. In order to control the trade-offs between model classification performance and EO fairness, we vary the values of λ in Eq. (5.3) and Eq. (5.4). Figure 5.2 shows the classification performance and EO fairness of one-stage and two-stage CL when λ changes. Overall, as λ increases, the equalized odds gaps shrink at the cost of model classification performance. Compared to one-stage CL, two-stage CL achieves more flexible trade-offs in general. Given the same range of λ, the change of equalized odds gaps in two-stage CL is more significant than in one-stage CL. At the same time, the corresponding model classification performances are comparable or remain better.

RQ 2. We study the trade-offs between model performance and EO fairness of our proposed methods compared to the baselines. Figure 5.3 displays the performance and fairness of these methods under different hyperparameter settings for the `jigsaw` and `biasbios` datasets (trade-off parameters for all methods are described in more detail in Appendix 5.7.5).

Among all methods, we find that two-stage CL and INLP achieve the best performance and fairness trade-offs. In the `biasbios` dataset, two-stage CL and INLP achieve similar performance and fairness trade-offs, and two-stage CL achieves more consistent results (i.e., lower variance). In the `jigsaw` dataset, two-stage CL achieves more flexible performance and fairness trade-offs as it reaches the

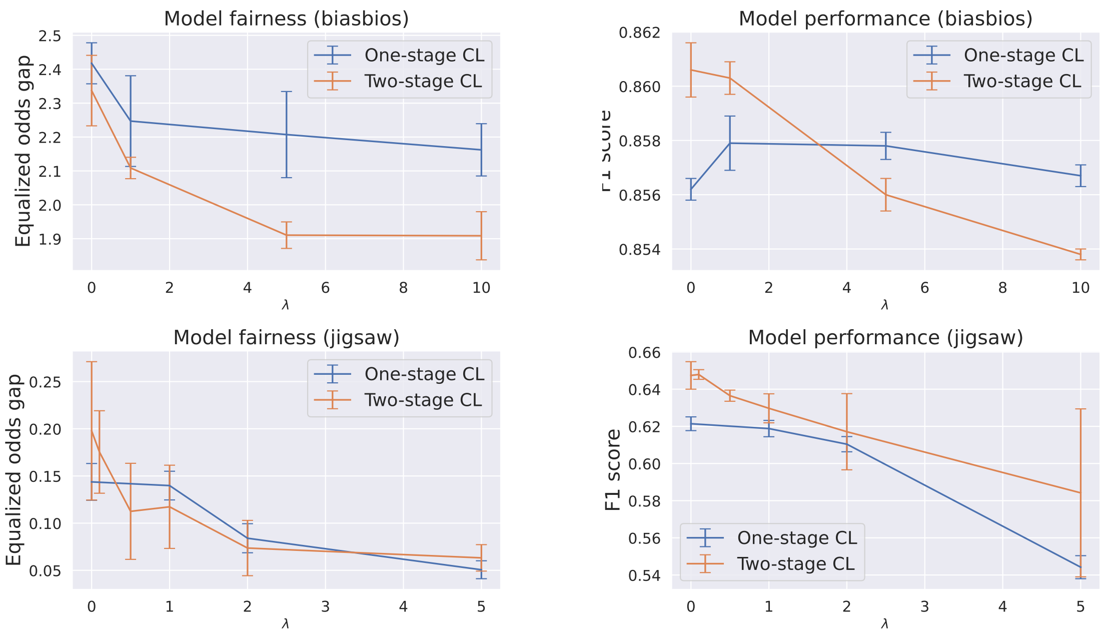


Figure 5.2: Classification performance and EO fairness of one-stage and two-stage CL when λ changes. The equalized odds gaps shrink at the cost of model classification performance as λ increases.

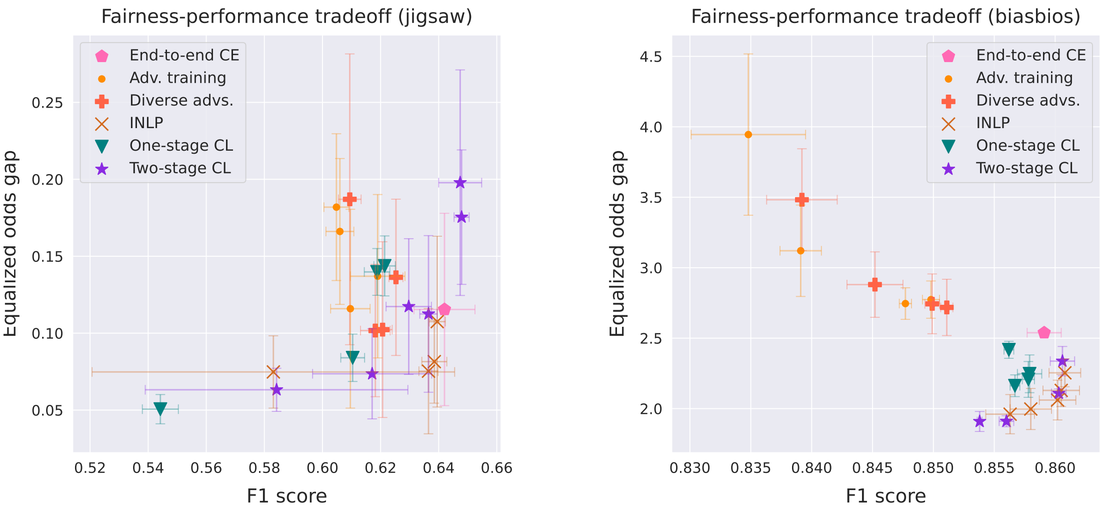


Figure 5.3: Classification performance and EO fairness and of our proposed methods compared against the baselines. Two-stage CL and INLP achieves the best performance and fairness trade-offs in general, and two-stage CL typically achieves more consistent results with lower variance.

highest model performance. Besides, when F1 scores are around 0.58, two-stage CL also achieves more consistent results and a lower EO gap. Meanwhile, when F1 scores are between 0.62∼0.64, INLP performs better. We note that the effectiveness of INLP highly depends on the pretrained encoder for INLP (see Appendix 5.7.12 for the effects of different pre-training strategies for the text encoder in INLP), and a slight change in the text encoder could lead to a significant difference in the results, while CL-based methods target training the text encoder directly to ensure EO fairness and we demonstrate they are stable under hyperparameter changes (see RQ 3 below).

In comparison, the adversarial-training-based methods are relatively more unstable and consistently perform worse than CL-based methods and INLP, especially in the `biasbios` dataset. Furthermore, both adversarial-training-based methods and INLP introduce additional model components (e.g., adversarial networks in adversarial-training-based methods and linear guarding layer in INLP) during training or inference, which complicates the actual implementation of the whole pipeline. In contrast, CL-based methods are well-suited to pre-training and fine-tuning paradigms in NLP applications.

RQ 3. We have shown that two-stage CL performs better than one-stage CL in RQ 1 and RQ 2. Thus, we choose two-stage CL to see if it is sensitive to key hyperparameter changes. As mentioned above, the performance of contrastive learning is closely related to temperature, (pre-training) batch size, and data augmentation strategy. Thus, we study whether the performance of two-stage CL is sensitive to these hyperparameters.

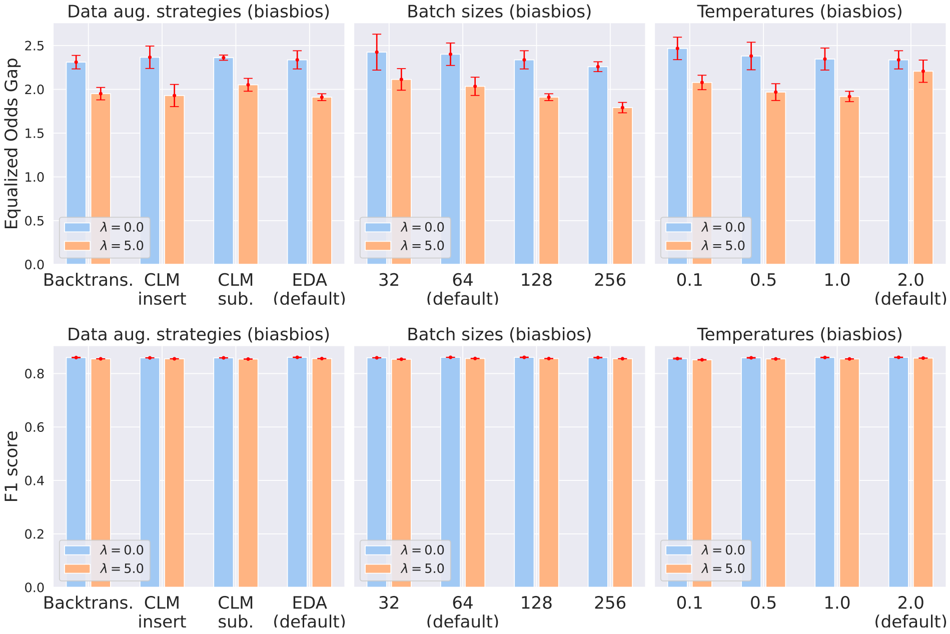


Figure 5.4: Sensitivity analysis of two-stage CL to key hyperparameter changes in (`biasbios`). "Default" in the X-axis indicates the default hyperparamter settings used in RQ 1 and RQ 2.

Figure 5.4 shows model performance and EO fairness of two-stage CL under different hyperparameter settings when $\lambda \in \{0.0, 5.0\}$ in the `biasbios` dataset (Figure 5.9 for the `jigsaw` in Appendix 5.7.10).

We see that two-stage CL are stable under a wide range of parameter settings: The equalized odds gaps are consistently decreasing when $\lambda = 5.0$ and the F1 scores are relatively high.

5.5 Related Work

Unintended social biases in NLP models have been identified in word/sentence embedding [18,64] and applications such as coreference resolution [196], language modeling [66], machine translation [197], and text classification [192,198].

In the literature, there are some recent works that aim to learn fair representations via contrastive learning [172–174,199]. Among these works, [199] propose contrastive objectives to learn debiased sentence embeddings that minimizes the correlation between embedded sentences and bias words. In classification tasks, [173,174] propose contrastive objectives to remove sensitive information; [172] also propose similar a contrastive objective to achieve a similar goal. According to [173], all those proposed contrastive objectives target for demographic parity in principle.

Our theoretical results involve key notions (e.g., entropy and mutual information) in information theory [200,201]. Information-theoretic-based methods have been used for representation learning for NLP applications. For example, [201] also proposed a variational upper bound of mutual information to learn disentangled textual representations for fair classification and style transfer.

Compared to the previous work, our work uses equalized odds as the fairness criterion. To the best of our knowledge, our work is the first to connect the problem of learning fair representations with contrastive learning to ensure the EO constraint and explore its effectiveness for bias mitigation in text classification in large language models (e.g., BERT).

5.6 Conclusion

In this chapter, we theoretically and empirically study how to leverage contrastive learning for fair text classification. Inspired by our theoretical results, we propose conditional supervised contrastive objectives to learn aligned and uniform representations while mixing the representation of different examples that share the same sensitive attribute for every task label. We conduct experiments to demonstrate the effectiveness of our algorithms in learning fair representations for text classification and show that our methods are stable in different hyperparameter settings. In the future, we plan to extend our algorithms to the settings of intersectional bias [202,203].

5.7 Appendix of Chapter 5

5.7.1 Omitted Proofs

5.7.2 Proof of Lemma 5.3.1

Proof. By the definition of conditional mutual information:

$$\begin{aligned}
&I(Z';Z \mid Y) - I(Z';Z \mid A,Y) \\
&= \big(H(Z \mid Y) - H(Z \mid Z',Y)\big) - \big(H(Z \mid A,Y) - H(Z \mid Z',A,Y)\big) \\
&= \big(H(Z \mid Y) - H(Z \mid A,Y)\big) + \big(H(Z \mid Z',A,Y) - H(Z \mid Z',Y)\big) \\
&= I(Z;A \mid Y) + \big(H(Z \mid Z',A,Y) - H(Z \mid Z',Y)\big) \\
&\leq I(Z;A \mid Y) + H(Z \mid Z',A,Y) \\
&\leq I(Z;A \mid Y) + H(Z \mid Z',Y) \\
&\leq I(Z;A \mid Y) + \epsilon
\end{aligned}$$

Next, we prove the opposite side,

$$\begin{aligned}
&I(Z';Z \mid Y) - I(Z';Z \mid A,Y) \\
&= \big(H(Z \mid Y) - H(Z \mid Z',Y)\big) - \big(H(Z \mid A,Y) - H(Z \mid Z',A,Y)\big) \\
&= \big(H(Z \mid Y) - H(Z \mid A,Y)\big) + \big(H(Z \mid Z',A,Y) - H(Z \mid Z',Y)\big) \\
&= I(Z;A \mid Y) + \big(H(Z \mid Z',A,Y) - H(Z \mid Z',Y)\big) \\
&\geq I(Z;A \mid Y) - H(Z \mid Z',Y) \\
&\geq I(Z;A \mid Y) - \epsilon,
\end{aligned}$$

which completes the proof. □

5.7.3 Proof of Proposition 5.3.1

Proof.

$$\begin{aligned} I(Z';Z \mid Y) = & -H(Z' \mid Z,Y) + H(Z' \mid Y) \\ = & \, \mathbb{E}_{p(y)}\big[\mathbb{E}_{p(z',z|y)}[\log p(z' \mid z,y)] - \mathbb{E}_{p(z'|y)}[\log p(z' \mid y)]\big] \\ = & \, \mathbb{E}_{p(y)}\big[\mathbb{E}_{p(z',z|y)}[\log p(z' \mid z,y)] - \mathbb{E}_{p(z'|y)}[\log \mathbb{E}_{p(z|y)}[p(z' \mid z,y)]]\big] \\ \leq & \, \mathbb{E}_{p(y)}\big[\mathbb{E}_{p(z',z|y)}[\log p(z' \mid z,y)] - \mathbb{E}_{p(z'|y)}[\mathbb{E}_{p(z|y)}[\log p(z' \mid z,y)]]\big] \\ \leq & \, -\mathbb{E}_{p(y)}\big[\mathbb{E}_{p(z'|y)}[\mathbb{E}_{p(z|y)}[\log p(z' \mid z,y)]]\big] \end{aligned}$$

where the second lines follow the marginal of a joint distribution can be expressed as the expectation of the corresponding conditional distribution, and the third line follows Jensen's Inequality.

□

5.7.4 Proof of Proposition 5.3.2

The proof of Proposition 5.3.2 is dependent on the lemmas show in Figure 5.5 as well as Proposition 5.7.1. We present these lemmas and their proofs before turning to the proof of Proposition 5.3.2.

Proof of Lemma 5.7.1

Proof. We first get the second-order functional derivative of the objective: $-\exp(s(z',z)) \cdot d\mathcal{Q}$, which is negative and it implies there is a supreme value for the objective. Next, we set the first-order functional derivative of the objective to be zero:

$$d\mathcal{P} - \exp(s(z',z)) \cdot d\mathcal{Q} = 0.$$

Reorganizing the equation above we get the optimal similarity function $s^*(z',z) = \log(\frac{d\mathcal{P}}{d\mathcal{Q}})$. Plugging it into the original objective, we have

$$\mathbb{E}_{\mathcal{P}}[s^*(z',z)] - \mathbb{E}_{\mathcal{Q}}[\exp(s^*(z',z))] + 1 = \mathbb{E}_{\mathcal{P}}[\log(\frac{d\mathcal{P}}{d\mathcal{Q}})] = D_{\mathrm{KL}}(\mathcal{P}\|\mathcal{Q}).$$

□

Proof of Lemma 5.7.2

Lemma 5.7.1 ([204])**.** Let $\mathcal{Z}$ be the sample space for Z' and Z, $s : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}$ be any function, and $\mathcal{P}$ and $\mathcal{Q}$ be the probability measures over $\mathcal{Z} \times \mathcal{Z}$. We have

$$D_{\mathrm{KL}}(\mathcal{P} \| \mathcal{Q}) = \sup_s \mathbb{E}_{(z',z)\sim\mathcal{P}}[s(z',z)] - \mathbb{E}_{(z',z)\sim\mathcal{Q}}[\exp(s(z',z))] + 1$$

Lemma 5.7.2 (Four-variable variant of Lemma 5.7.1)**.** Let $\mathcal{Z}$ be the sample space for Z' and Z, $\mathcal{Y}$ be the sample space for Y, $\mathcal{A}$ be the sample space for A, $s : \mathcal{Z} \times \mathcal{Z} \times \mathcal{Y} \times \mathcal{A} \to \mathbb{R}$ be any function, and $\mathcal{P}$ and $\mathcal{Q}$ be the probability measures over $\mathcal{Z} \times \mathcal{Z} \times \mathcal{Y} \times \mathcal{A}$. We have

$$D_{\mathrm{KL}}(\mathcal{P} \| \mathcal{Q}) = \sup_s \mathbb{E}_{(z',z,y,a)\sim\mathcal{P}}[s(z',z,y,a)] - \mathbb{E}_{(z',z,y,a)\sim\mathcal{Q}}[\exp(s(z',z,y,a))] + 1$$

Lemma 5.7.3. $\sup_s \mathbb{E}_{(z',z_1)\sim\mathcal{P},(z',z_{2:N})\sim\mathcal{Q}^{\otimes N-1}}\Big[\log \frac{\exp(s(z',z_1))}{\frac{1}{N}\sum_{j=1}^{N}\exp(s(z',z_j))}\Big] \le D_{\mathrm{KL}}(\mathcal{P}\|\mathcal{Q})$

Lemma 5.7.4.

$$\begin{aligned} & D_{\mathrm{KL}}(P_{Z',Z} \parallel \mathbb{E}_{P_{A,Y}}[P_{Z'|A,Y}P_{Z|A,Y}]) \\ = & \sup_s \mathbb{E}_{(z',z)\sim P_{Z',Z}}[s(z',z)] - \mathbb{E}_{(z',z)\sim \mathbb{E}_{P_{A,Y}}[P_{Z'|A,Y}P_{Z|A,Y}]}[\exp(s(z',z))] + 1. \end{aligned}$$

Lemma 5.7.5.

$$\begin{aligned} & I(Z';Z \mid A,Y) \\ = & D_{\mathrm{KL}}(P_{Z',Z,A,Y} \parallel P_{A,Y}P_{Z'|A,Y}P_{Z|A,Y}) \\ = & \sup_s \mathbb{E}_{(z',z,a,y)\sim P_{Z',Z,A,Y}}[s(z',z,a,y)] \\ & \qquad - \mathbb{E}_{(z',z,a,y)\sim P_{A,Y}P_{Z'|A,Y}P_{Z|A,Y}}[\exp(s(z',z,a,y))] + 1. \end{aligned}$$

Figure 5.5: Lemmas required for the proof of Proposition 5.3.2.

Proof. The proof technique is identical to the proof of Lemma 5.7.1 and the only difference is that the similarity function takes four variables as input. □

Proof of Lemma 5.7.3 See Figure 5.6.

Proof of Lemma 5.7.4

Proof. We use Lemma 5.7.1 and substitute $\mathcal{P}$ and $\mathcal{Q}$ with $P_{Z',Z}$ and $\mathbb{E}_{P_{A,Y}}[P_{Z'|A,Y}P_{Z|A,Y}$, respectively. □

Proof of Lemma 5.7.5

Proof. We use Lemma 5.7.2 and substitute $\mathcal{P}$ and $\mathcal{Q}$ with $P_{Z',Z,A,Y}$ and $P_{A,Y}P_{Z'|A,Y}P_{Z|A,Y}$, respectively. □

Proof.

$$
\begin{aligned}
D_{\mathrm{KL}}(\mathcal{P}\|\mathcal{Q}) &= \mathbb{E}_{(z',z_{2:N})\sim\mathcal{Q}^{\otimes N-1}}\Big[D_{\mathrm{KL}}(\mathcal{P}\|\mathcal{Q})\Big] \\
&\geq \mathbb{E}_{(z',z_{2:N})\sim\mathcal{Q}^{\otimes N-1}}\Big[\mathbb{E}_{\mathcal{P}}[\log \frac{\exp(s^*(z',z))}{\frac{1}{N}\sum_{j=1}^N \exp(s(z',z_j))}] - \mathbb{E}_{\mathcal{Q}}[\frac{\exp(s^*(z',z))}{\frac{1}{N}\sum_{j=1}^N \exp(s(z',z_j))}] + 1\Big] \\
&= \mathbb{E}_{(z',z_{2:N})\sim\mathcal{Q}^{\otimes N-1}}\Big[\mathbb{E}_{\mathcal{P}}[\log \frac{\exp(s^*(z',z))}{\frac{1}{N}\sum_{j=1}^N \exp(s(z',z_j))}] - 1 + 1\Big] \\
&= \mathbb{E}_{(z',z_1)\sim\mathcal{P},(z',z_{2:N})\sim\mathcal{Q}^{\otimes N-1}}\big[\log \frac{\exp(s(z',z_1))}{\frac{1}{N}\sum_{j=1}^N \exp(s(z',z_j))}\big],
\end{aligned}
$$

where the first line follows the fact that $D_{\mathrm{KL}}(\mathcal{P}\|\mathcal{Q})$ is a constant, the second line follows Lemma 5.7.1, the third line follows the fact that (z', z_1) and $(z', z_{2:N})$ are interchangeable when sampling from $\mathcal{Q}$. Thus, for all similarity function s, we have

$$
\sup_s \mathbb{E}_{(z',z_1)\sim\mathcal{P},(z',z_{2:N})\sim\mathcal{Q}^{\otimes N-1}}\big[\log \frac{\exp(s(z',z_1))}{\frac{1}{N}\sum_{j=1}^N \exp(s(z',z_j))}\big] \leq D_{\mathrm{KL}}(\mathcal{P}\|\mathcal{Q})
$$

□

Figure 5.6: Proof of Lemma 5.7.3

Proposition 5.7.1.

$$
D_{\mathrm{KL}}(P_{Z',Z} \parallel \mathbb{E}_{P_{A,Y}}[P_{Z'|A,Y}P_{Z|A,Y}]) \leq I(Z';Z \mid A,Y)
$$

Proof of Proposition 5.7.1 See Figure 5.7.

Proof of Proposition 5.3.2 See Figure 5.8.

5.7.5 Experimental Details

5.7.6 Data prepossessing pipelines

Jigsaw Our first dataset, which we refer to as `jigsaw`, is a corpus of comments from an online forum associated with a toxicity rating. `jigsaw`'s main task is binary classification: given a "toxicity" score in the range $[0, 1]$ that has been assigned to each comment, we determine whether the "toxicity" score is greater or equal to 0.5. Each comment is also annotated with some "identity" labels, indicating whether some identities belonging to specific demographic groups are mentioned in the comment. We focus on the identity labels related to "race or ethnicity" and binaries the identity labels into black and non-black. Note that there are a multitude of other sensitive attributes in the *Jigsaw-Toxicity*

Proof. We have

$$\begin{aligned}
&D_{\mathrm{KL}}(P_{Z',Z} \parallel \mathbb{E}_{P_{A,Y}}[P_{Z'|A,Y}P_{Z|A,Y}]) \\
=\ &\sup_s \mathbb{E}_{(z',z)\sim P_{Z',Z}}[s(z',z)] - \mathbb{E}_{(z',z)\sim \mathbb{E}_{P_{A,Y}}[P_{Z'|A,Y}P_{Z|A,Y}]}[\exp(s(z',z))] + 1 \\
=\ &\sup_s \mathbb{E}_{(z',z,a,y)\sim P_{Z',Z,A,Y}}[s(z',z)] - \mathbb{E}_{(z',z,a,y)\sim P_{A,Y}P_{Z'|A,Y}P_{Z|A,Y}}[\exp(s(z',z))] + 1,
\end{aligned}$$

where the first equation follows Lemma 5.7.4. Let $s^*(z',z)$ be the function when the supreme value is achieved and let $\hat{s}^*(z',z,a,y) = s^*(z',z), \forall\ (a,y) \in P_{A,Y}$, and we have

$$\begin{aligned}
&D_{\mathrm{KL}}(P_{Z',Z} \parallel \mathbb{E}_{P_{A,Y}}[P_{Z'|A,Y}P_{Z|A,Y}]) \\
=\ &\sup_s \mathbb{E}_{(z',z,a,y)\sim P_{Z',Z,A,Y}}[s(z',z)] - \mathbb{E}_{(z',z,a,y)\sim P_{A,Y}P_{Z'|A,Y}P_{Z|A,Y}}[\exp(s(z',z))] + 1 \\
=\ &\mathbb{E}_{(z',z,a,y)\sim P_{Z',Z,A,Y}}[\hat{s}^*(z',z,a,y)] - \mathbb{E}_{(z',z,a,y)\sim P_{A,Y}P_{Z'|A,Y}P_{Z|A,Y}}[\exp(\hat{s}^*(z',z,a,y)] + 1 \\
\le\ &\sup_{\hat{s}} \mathbb{E}_{(z',z,a,y)\sim P_{Z',Z,A,Y}}[\hat{s}(z',z,a,y)] - \mathbb{E}_{(z',z,a,y)\sim P_{A,Y}P_{Z'|A,Y}P_{Z|A,Y}}[\exp(\hat{s}(z',z,a,y)] + 1 \\
=\ &I(Z';Z \mid A,Y),
\end{aligned}$$

where the last equation follows Lemma 5.7.5. □

Figure 5.7: Proof of Proposition 5.7.1.

Proof. Define two probability measures $\mathcal{P} = P_{Z',Z}$ and $\mathcal{Q} = \mathbb{E}_{P_{A,Y}}[P_{Z'|A,Y}P_{Z|A,Y}]$, we have

$$\begin{aligned}
&\mathbb{E}_{p(a,y)}\Big[\mathbb{E}_{p(z_i',z_i|a,y)^{\otimes N}}\big[\log \frac{\exp(s(z_i',z_i))}{\frac{1}{N}\sum_{j=1}^N \exp(s(z_i',z_j))}\big]\Big] \\
=\ &\mathbb{E}_{(z',z_1)\sim\mathcal{P},(z',z_{2:N})\sim\mathcal{Q}^{\otimes N-1}}\big[\log \frac{\exp(s(z',z_1))}{\frac{1}{N}\sum_{j=1}^N \exp(s(z',z_j))}\big] \\
\le\ &D_{\mathrm{KL}}(\mathcal{P}\|\mathcal{Q}) \\
=\ &D_{\mathrm{KL}}(P_{Z',Z} \parallel \mathbb{E}_{P_{A,Y}}[P_{Z'|A,Y}P_{Z|A,Y}]) \\
\le\ &I(Z';Z \mid A,Y).
\end{aligned}$$

where the second equation follows Lemma 5.7.3 and the last equation follows Proposition 5.7.1. □

Figure 5.8: Proof of Proposition 5.3.2.

dataset and we constrain the scope of our study to the "race" attributes present in text classification datasets. We follow [205] to perform the train/val/test splits. The data with "race or ethnicity" identity labels are split into training, validation, and test sets summarized in Table 5.1.

Bias-in-Bios To measure model fairness and performance in the multi-class classification setting, we use the professional biographies dataset of [186], which we refer to as the `biasbios` dataset. The data consist of nearly 400,000 online biographies collected from the Common Crawl corpus. These

Table 5.1: Summary of training, validation, and test splits for the `jigsaw` dataset.

Data split	**Samples**	**Protected attribute** (NB = non-black, B = black)	**Task label average**
Training	25,954 (60.5%)	61.9% (NB), 38.1% (B)	0.2822
Validation	4,390 (10.2%)	62.4% (NB), 37.6% (B)	0.2897
Test	12,562 (29.3%)	61.2% (NB), 38.8% (B)	0.2873
Total	42,906	61.7% (NB), 38.3% (B)	0.2844

Table 5.2: Summary of the training, validation, and test splits of the `biasbios` dataset.

Data split	**Samples**	**Protected attribute** (F = female, M = male)
Training	255,710 (65.0%)	46.0% (F), 54.0% (M)
Validation	39,369 (10.0%)	47.8% (F), 52.2% (M)
Test	98,344 (25.0%)	46.5% (F), 53.5% (M)
Total	393,423	46.3% (F), 53.7% (M)

biographies are annotated with one of the 28 professions to which their subject belongs. The data are mapped to a binary gender based on the occurrence of gendered pronouns and are scrubbed to exclude the authors' names and pronouns. It is worth noting that mapping gender to binary labels is a strong simplified assumption to map data to a demographic label cleanly; it ignores people who do not identify as female or male as well as the complexity of gender identity more generally. We refer readers to the original work [186] for further discussion of these issues. For our experiments, we attempt to predict the profession as our task label while protecting against the gender attribute. We replicate the splits of `biasbios` used by [195], which are summarized in Table 5.2.

5.7.7 Detailed Implementations and Hyperparameter Settings

In this section, we provide more detail on our implementations and give the hyperparameter we use in our experiments. We first give detail on how we tune the performance and fairness trade-offs for each method.

- **One-stage / Two-stage CL:** for one- and two-stage CL, once we determine the best classification performance by conducting a grid search on temperature, (pre-training) batch size, and data augmentation strategies (as well as γ in one-stage CL) , we only tune the parameter λ described in Sec. 5.3.2, which affects the trade-offs between supervised contrastive loss L_{sup} and the conditional supervised InfoNCE loss $L_{\text{CS-InfoNCE}}$.

- **Diverse adversarial training:** Following [194], we use an ensemble of three adversarial discriminators and use the same adversarial network architecture. There are two hyperparameters of interest: λ_{diff} and λ_{adv}. λ_{diff} is a difference loss hyperparameter that encourages discriminators to learn orthogonal representations. λ_{adv} affects the tradeoff in the model between task performance and learning a hidden representation independent of the protected attribute. We first do a grid search on $\lambda_{diff} = \{0, 100, 1000, 5000\}$ and vary the values of λ_{adv} to the determine the best hyperparameter configurations.

- **Adversarial training:** The implementation is nearly identical to diverse adversarial training, except that there is just one adversarial discriminators.
- **INLP:** Following [195], we use the weight of the a SVM classifier as the parameters of the linear guarding layer and follow the same hyperparameters in training the linear guarding layer. The trade-off hyperparameter that we tune for INLP is N_{clf}, which is the number of classifiers trained by INLP (i.e., the number of rounds).

Table 5.3 contains the trade-off hyperparameters used for our experiments on the `jigsaw` dataset, while Table 5.4 summarizes the hyperparameter choices for the `biasbios` dataset. The remaining hyperparamters for all methods are listed in Table 5.5.

Table 5.3: Hyperparameters tested for RQ2 (Figure 5.3) for the `jigsaw` dataset.

Method	**Hyperparameters tested**
Adversarial training	$\lambda_{adv} \in \{0.1, 0.5, 1, 2\}$
Diverse adversarial training ($N_{adv} = 3$, $\lambda_{diff} = 100$)	$\lambda_{adv} \in \{0.1, 0.5, 1, 2\}$
INLP	$N_{clf} \in \{20, 50, 80, 100, 150\}$
One-stage CL	$\lambda \in \{0, 1, 2, 5\}$
Two-stage CL	$\lambda \in \{0, 0.1, 0.5, 1, 2, 5\}$

Table 5.4: Hyperparameters tested for RQ2 (Figure 5.3) for the `biasbios` dataset.

Method	**Hyperparameters tested**
Adversarial training	$\lambda_{adv} \in \{0.1, 0.2, 0.5, 1\}$
Diverse adversarial training ($\lambda_{diff} = 5000$)	$\lambda_{adv} \in \{0.1, 0.2, 0.5, 1\}$
INLP	$N_{clf} \in \{20, 50, 100, 300, 400\}$
One-stage CL	$\lambda \in \{0, 1, 5, 10\}$
Two-stage CL	$\lambda \in \{0, 1, 5, 10\}$

Table 5.5: Additional hyperparameters used for experiments.

Hyperparameter	(jigsaw)	(biasbios)
Batch size	32	32
Learning rate	2e-5	2e-5
Epochs	10	7
Optimizer	Adam	Adam

5.7.8 Data Augmentation Strategies

In this section, we provide a description of the data augmentation strategies used in CL-based methods[4].

- Easy data augmentation (EDA) [206]: EDA consists of four simple operations: synonym replacement, random insertion, random swap, and random deletion. Following the suggestions provided by the original paper, we choose the augmentation ratio to be 0.1 and create four augmented examples per example.

- Back translation [207]: It first translates the input example to another language and back to English. We use the machine translation model `wmt19-en-de` in our experiment.

- Word replacement using contextual language model (CLM insert) [208]: It replaces words based on a language model that leverages contextual word embeddings to find the most similar word for augmentation. We use the RoBERTa-base language model and choose the augmentation rate of 0.1.

- Word insertion using contextual language model (CLM insert) [208]: It inserts words based on a language model that leverages contextual word embeddings to find the most similar word for augmentation. We use the RoBERTa-base language model and choose the augmentation rate of 0.1.

5.7.9 Additional Experimental Results

5.7.10 More Comments for CL-based Methods

Our method achieves highly consistent results with respect to fairness and performance compared to the baseline methods. Figure 5.9 visualize hows model performance and EO fairness of two-stage CL under different hyperparameter settings when $\lambda = \{0.0, 5.0\}$ in the `jigsaw` dataset.

[4] We use the implementations of data augmentation at: `https://github.com/makcedward/nlpaug`.

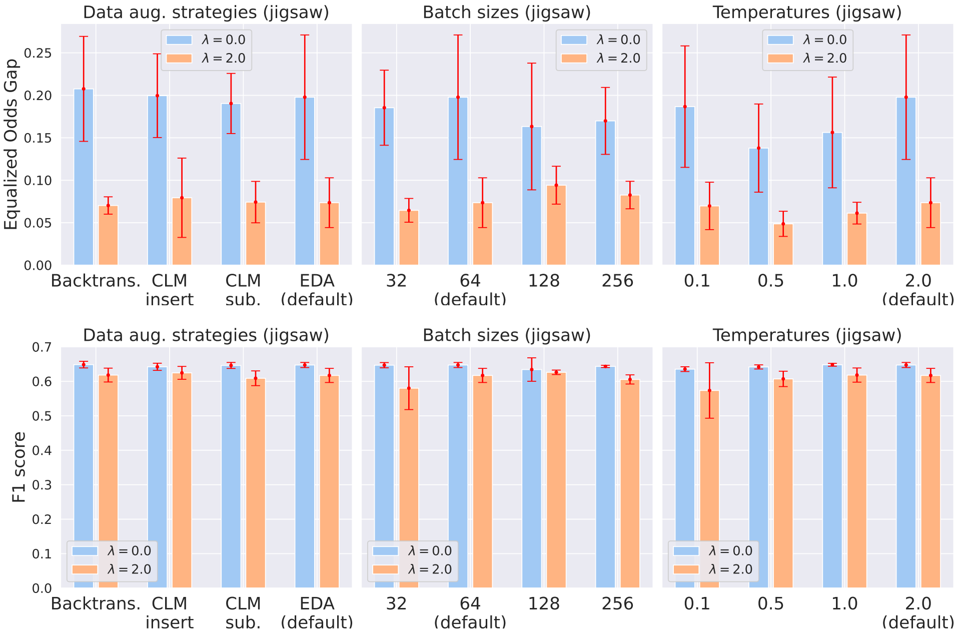


Figure 5.9: Sensitivity analysis of two-stage CL to key hyperparameter changes in (`Jigsaw`).

5.7.11 Visualization of the BERT Embeddings using Different Objectives

In Figure 5.10, we show the T-SNE visualization [182] of text embeddings learned with different training objectives. We can see that both CE-trained embeddings and CL-trained embeddings capture the class information well (points with the same markers forms their own clusters). However, points that share the sensitive attributes within the same class are more likely to form small clusters. When we introduce $L_{\text{CS-InfoNCE}}$, those points tend to “mix” together.

5.7.12 How Different Pretrained Text Encoders affects the Performance of INLP?

To provide the clearest comparison between our proposed methods and the baselines, we used the best settings for the baseline methods that we were able to attain. Nonetheless, we observed that the performance of INLP was highly sensitive to the encoder training settings, which could be an important practical consideration for practitioners selecting between different ways of improving model fairness. Figure 5.11 compares the performance and fairness of INLP using different encoder pre-training strategies. We see that in both datasets, the classification and fairness performance of INLP changes drastically even with the same values of trade-off parameter. Even training with the same objectives (CE loss), the text encoders obtained in different epochs after convergence greatly

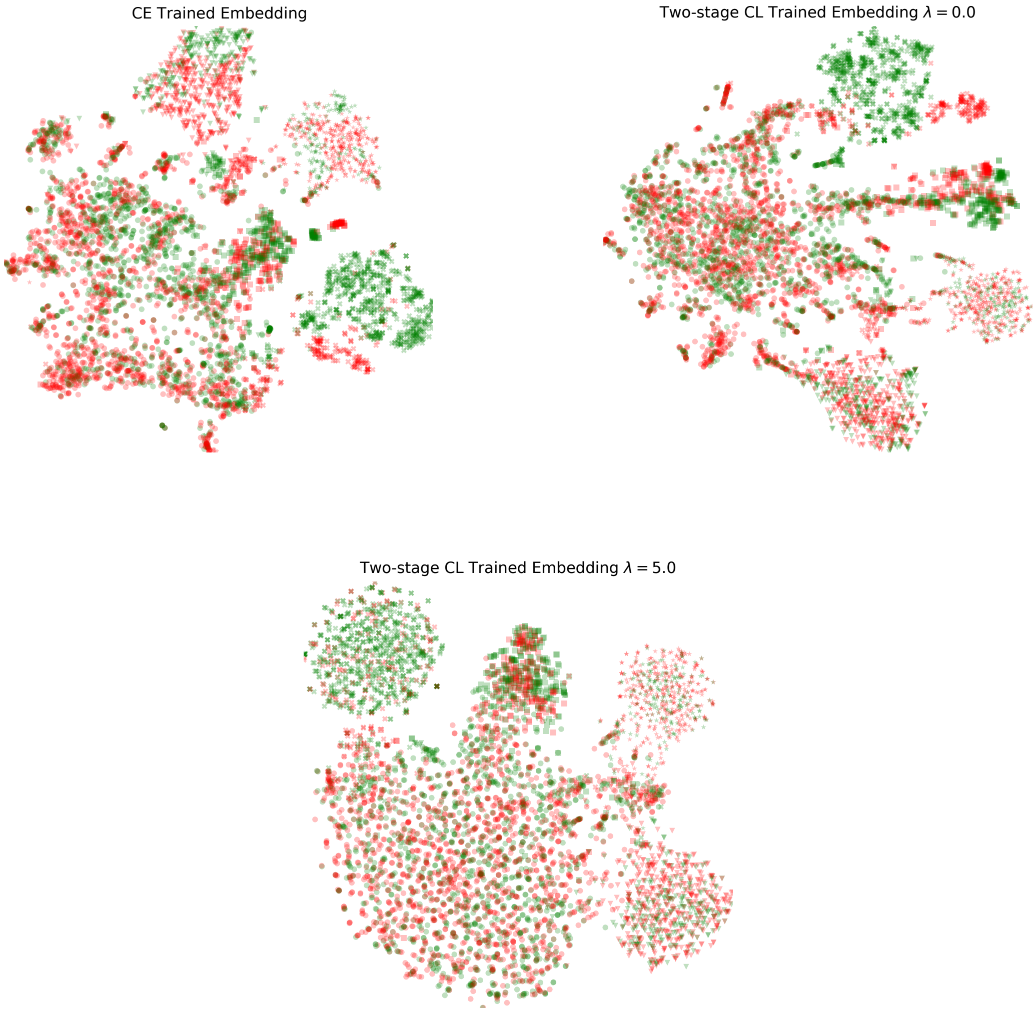


Figure 5.10: T-SNE visualization of text embeddings using different training objectives (zoom in for better visualization) in the `biasbios` dataset. Different colors indicate different sensitive attribute (e.g., red for male and green for female) and different markers indicate different classes. Both CE-trained embeddings and CL-trained embedding capture the class information well (points with the same markers forms their own clusters). However, points that share the sensitive attributes within the same class are more likely to form small clusters. When we introduce $L_{\text{CS-InfoNCE}}$, those points tend to "mix" together.

affect its performance. For example, the CE-trained encoder obtained in the last epoch of training nearly shows no effects on bias mitigation. If we do not train the text encoder using our datasets and directly use the parameters of the `bert-base-uncased` (this is the experimental setting of the previous work [195]), the model performances drastically decrease of the training iterations of INLP increase. Lastly, INLP also does not perform well when using supervised contrastive loss to train the text encoder. In comparison, our methods are more robust to hyperparameter changes.

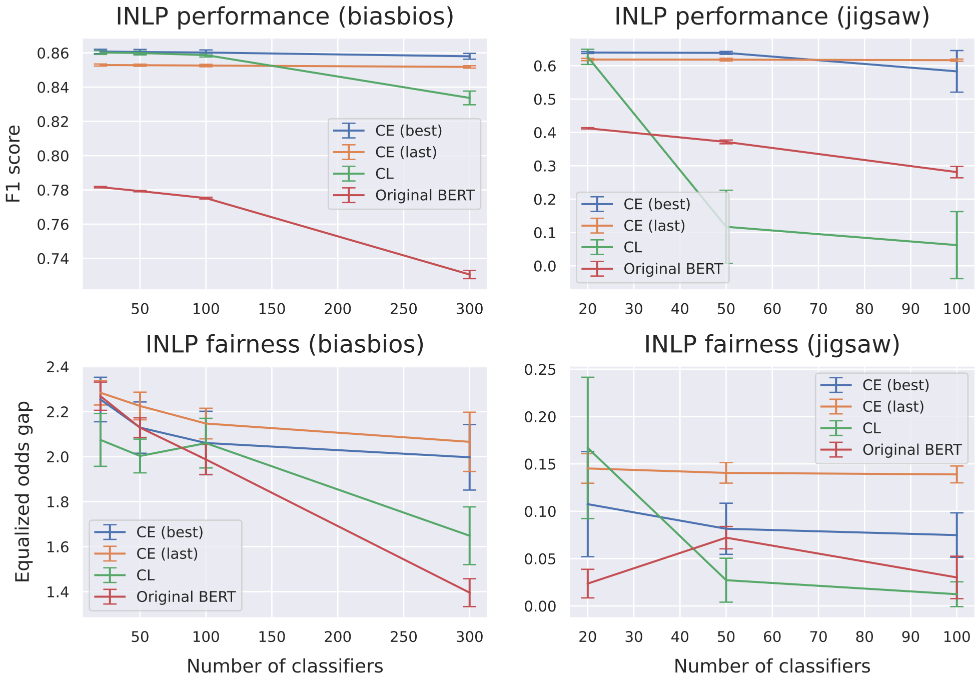


Figure 5.11: Comparison of INLP performance and fairness under different pretrained encoders. CE (best) indicates that we train the text encoder using CE loss and save the encoder that achieve the best validation loss for INLP. CE (last) indicates that we train the text encoder using CE loss and save the encoder in the last epoch for INLP. CL indicates that we train the text encoder using supervised contrastive loss. Original BERt indicates that we use the `bert-based-uncased` as the text encoder.

Chapter 6

Understanding and Mitigating Accuracy Disparity in Regression

6.1 Introduction

Recent progress in machine learning has led to its widespread use in many high-stakes domains, such as criminal justice, healthcare, student loan approval, and hiring. Meanwhile, it has also been widely observed that accuracy disparity could occur inadvertently under various scenarios in practice [15]. For example, errors are inclined to occur for individuals of certain underrepresented demographic groups [209]. In other cases, [210] showed that notable accuracy disparity exists across different racial and gender demographic subgroups on several real-world image classification systems. Moreover, [24] found out that a differentially private model even exacerbates such accuracy disparity. Such accuracy disparity across demographic subgroups not only raises concerns in high-stake applications but also can be utilized by malicious parties to cause information leakage [32, 211].

Despite the ample needs of accuracy parity, most prior work limits its scope to studying the problem in binary classification settings [52, 212–214]. Compared to the accuracy disparity problem in classification settings, accuracy disparity[1] in regression is a more challenging but less studied problem, due to the fact that many existing algorithmic techniques designed for classification cannot be extended in a straightforward way when the target variable is continuous [176]. In a seminal

[1] Technically, accuracy disparity refers to (squared) error difference in our paper. We would like to use accuracy disparity throughout our paper since it is a more commonly used term in fairness problems.

work, [215] analyzed the impact of data collection on accuracy disparity in general learning models. They provided a descriptive analysis of such parity gaps and advocated for collecting more training examples and introducing more predictive variables. While such a suggestion is feasible in applications where data collection and labeling is cheap, it is not applicable in domains where it is time-consuming, expensive, or even infeasible to collect more data, e.g., in autonomous driving, education, etc.

Our Contributions In this chapter, we provide a prescriptive analysis of accuracy disparity and aim at providing algorithmic interventions to reduce the disparity gap between different demographic subgroups in the regression setting. To start with, we first formally characterize why accuracy disparity appears in regression problems by depicting the feasible region of the underlying group-wise errors. Next, we derive an error decomposition theorem that decomposes the accuracy disparity into the distance between marginal label distributions and the distance between conditional representations. We also provide a lower bound on the joint error across groups. Based on these results, we illustrate why regression models aiming to minimize the global loss will inevitably lead to accuracy disparity if the marginal label distributions or conditional representations differ across groups. See Figure 6.1 for illustration.

Motivated by the error decomposition theorem, we propose two algorithms to reduce accuracy disparity via joint distribution alignment with the total variation distance and the Wasserstein distance, respectively. Furthermore, we analyze the game-theoretic optima of the objective functions and illustrate the principle of our algorithms from a game-theoretic perspective. To corroborate the effectiveness of our proposed algorithms in reducing accuracy disparity, we conduct experiments on five benchmark datasets. Experimental results suggest that our proposed algorithms help to mitigate accuracy disparity while maintaining the predictive power of the regression models. We believe our theoretical results contribute to the understanding of why accuracy disparity occurs in machine learning models, and the proposed algorithms provides an alternative for intervention in real-world scenarios where accuracy parity is desired but collecting more data/features is time-consuming or infeasible.

6.2 Preliminaries

Notation We use $\mathcal{X} \subseteq \mathbb{R}^d$ and $\mathcal{Y} \subseteq \mathbb{R}$ to denote the input and output space. We use X and Y to denote random variables which take values in $\mathcal{X}$ and $\mathcal{Y}$, respectively. Lower case letters $\mathbf{x}$ and y denote the instantiation of X and Y. We use $H(X)$ to denote the Shannon entropy of random

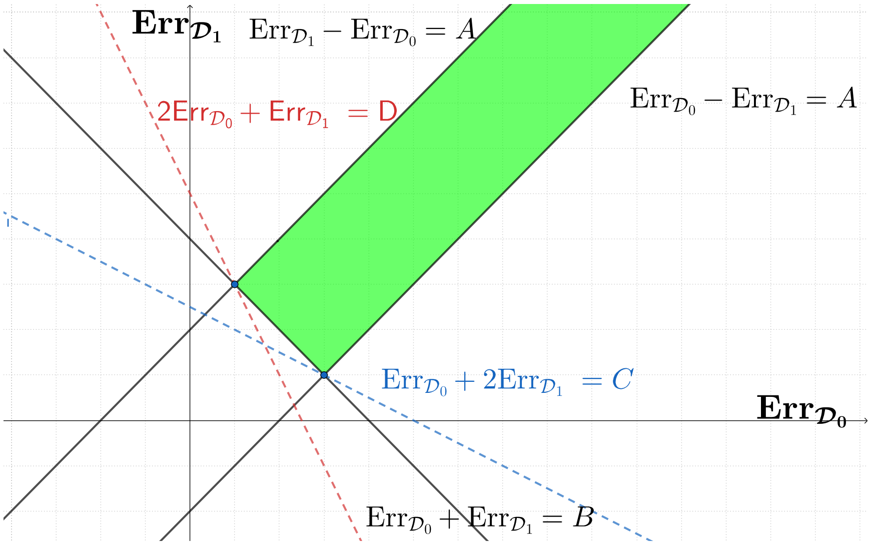

$\mathbf{Err}_{\mathcal{D}_1}$
$\mathrm{Err}_{\mathcal{D}_1} - \mathrm{Err}_{\mathcal{D}_0} = A$
$2\mathrm{Err}_{\mathcal{D}_0} + \mathrm{Err}_{\mathcal{D}_1} = D$
$\mathrm{Err}_{\mathcal{D}_0} - \mathrm{Err}_{\mathcal{D}_1} = A$
$\mathrm{Err}_{\mathcal{D}_0} + 2\mathrm{Err}_{\mathcal{D}_1} = C$
$\mathbf{Err}_{\mathcal{D}_0}$
$\mathrm{Err}_{\mathcal{D}_0} + \mathrm{Err}_{\mathcal{D}_1} = B$

Figure 6.1: Geometric interpretation of accuracy disparity in regression. The green area corresponds to the feasible region of $\mathrm{Err}_{\mathcal{D}_0}$ and $\mathrm{Err}_{\mathcal{D}_1}$ under the hypothesis class $\mathcal{H}$. For any optimal hypothesis h which is solely designed to minimize the overall error, the best the hypothesis h can do is to intersect with one of the two bottom vertices of the green area, leading to accuracy disparity if the width of the feasible region is nonzero. See section 6.3.1 for more details.

variable X, $H(X \mid Y)$ to denote the conditional entropy of X given Y, and $I(X;Y)$ to denote the mutual information between X and Y. To simplify the presentation, we use $A \in \{0,1\}$ as the sensitive attribute, e.g., gender, race, etc. Let $\mathcal{H}$ be the hypothesis class of regression models. In other words, for $h \in \mathcal{H}$, $h : \mathcal{X} \to \mathcal{Y}$ is a predictor. Note that even if the predictor does not explicitly take the sensitive attribute A as an input variable, the prediction can still be biased due to the correlations with other input variables. We study the stochastic setting where there is a joint distribution $\mathcal{D}$ over X, Y and A from which the data are sampled. For $a \in \{0,1\}$ and $y \in \mathbb{R}$, we use $\mathcal{D}_a$ to denote the conditional distribution of $\mathcal{D}$ given $A = a$ and $\mathcal{D}^y$ to denote the conditional distribution of $\mathcal{D}$ given $Y = y$. For an event E, $\mathcal{D}(E)$ denotes the probability of E under $\mathcal{D}$. Given a feature transformation function $g : \mathcal{X} \to \mathcal{Z}$ that maps instances from the input space $\mathcal{X}$ to feature space $\mathcal{Z}$, we define $g_\sharp\mathcal{D} := \mathcal{D} \circ g^{-1}$ to be the induced (pushforward) distribution of $\mathcal{D}$ under g, i.e., for any event $E' \subseteq \mathcal{Z}$, $g_\sharp\mathcal{D}(E') := \mathcal{D}(\{x \in \mathcal{X} \mid g(x) \in E'\})$. We define $(\cdot)_+$ to be $\max\{\cdot, 0\}$.

For regression problems, given a joint distribution $\mathcal{D}$, the error of a predictor h under $\mathcal{D}$ is defined as $\mathrm{Err}_{\mathcal{D}}(h) := \mathbb{E}_{\mathcal{D}}[(Y - h(X))^2]$. To make the notation more compact, we may drop the subscript $\mathcal{D}$ when it is clear from the context. Furthermore, we also use $\mathrm{MSE}_{\mathcal{D}}(\widehat{Y},\ Y)$ to denote the mean squared loss between the predicted variable $\widehat{Y} = h(X)$ and the true label Y over the joint distribution $\mathcal{D}$. Similarly, we also use $\mathrm{CE}_{\mathcal{D}}(A \parallel \widehat{A})$ to denote the cross-entropy loss between the predicted variable $\widehat{A}$ and the true label A over the joint distribution $\mathcal{D}$. Throughout the paper, we make the following standard boundedness assumption:

Assumption 6.2.1. There exists $M > 0$, such that for any hypothesis $\mathcal{H} \ni h : \mathcal{X} \to \mathcal{Y}$, $\|h\|_\infty \leq M$ and $|Y| \leq M$.

Problem Setup Our goal is to learn a regression model that is fair in the sense that the errors of the regressor are approximately equal across the groups given by the sensitive attribute A. We assume that the sensitive attribute A is only available to the learner during the training phase and is not visible during the inference phase. We would like to point out that there are many other different and important definitions of fairness [216] even in the sub-category of group fairness, and our discussion is by no means comprehensive. For example, two frequently used definitions of fairness in the literature are the so-called statistical parity [51] and equalized odds [52]. Nevertheless, throughout this paper we mainly focus accuracy parity as our fairness notion, due to the fact that machine learning systems have been shown to exhibit substantial accuracy disparities between different demographic subgroups [15, 209, 210]. This observation has already brought huge public attention (e.g., see New York Times, The Verge, and Insurance Journal) and calls for machine learning systems that (at least approximately) satisfy accuracy parity. For example, in a healthcare spending prediction system, stakeholders do not want the prediction error gaps to be too large among different demographic subgroups. Formally, accuracy parity is defined as follows:

Definition 6.2.1. Given a joint distribution $\mathcal{D}$, a predictor h satisfies accuracy parity if $\mathrm{Err}_{\mathcal{D}_0}(h) = \mathrm{Err}_{\mathcal{D}_1}(h)$.

In practice the exact equality of accuracy between two groups is often hard to ensure, so we define *error gap* to measure how well the model satisfies accuracy parity:

Definition 6.2.2. Given a joint distribution $\mathcal{D}$, the error gap of a hypothesis h is $\Delta_{\mathrm{Err}}(h) := |\mathrm{Err}_{\mathcal{D}_0}(h) - \mathrm{Err}_{\mathcal{D}_1}(h)|$.

By definition, if a model satisfies accuracy parity, $\Delta_{\mathrm{Err}}(h)$ will be zero. Next we introduce two distance metrics that will be used in our theoretical analysis and algorithm design:

- Total variation distance: it measures the largest possible difference between the probabilities that the two probability distributions can assign to the same event E. We use $d_{\mathrm{TV}}(\mathcal{P}, \mathcal{Q})$ to denote the total variation:

$$d_{\mathrm{TV}}(\mathcal{P}, \mathcal{Q}) := \sup_{E} |\mathcal{P}(E) - \mathcal{Q}(E)|.$$

- Wasserstein distance: the Wasserstein distance between two probability distributions is

$$W_1(\mathcal{P}, \mathcal{Q}) = \sup_{f \in \{f : \|f\|_L \leq 1\}} \left| \int_{\Omega} f d\mathcal{P} - \int_{\Omega} f d\mathcal{Q} \right|,$$

where $\|f\|_L$ is the Lipschitz semi-norm of a real-valued function of f and Ω is the sample space over which two probability distributions $\mathcal{P}$ and $\mathcal{Q}$ are defined. By the Kantorovich-Rubinstein duality theorem [217], we recover the primal form of the Wasserstein distance, defined as

$$W_1(\mathcal{P}, \mathcal{Q}) := \inf_{\gamma \in \Gamma(\mathcal{P}, \mathcal{Q})} \int d(X, Y) \, \mathrm{d}\gamma(X, Y),$$

where $\Gamma(\mathcal{P}, \mathcal{Q})$ denotes the collection of all couplings of $\mathcal{P}$ and $\mathcal{Q}$, and X and Y denote the random variables with law $\mathcal{P}$ and $\mathcal{Q}$ respectively. Throughout this paper we use L_1 distance for $d(\cdot, \cdot)$, but extensions to other distances, e.g., L_2 distance, is straightforward.

6.3 Main Results

In this section, we first characterize why accuracy disparity arises in regression models. More specifically, given a hypothesis $h \in \mathcal{H}$, we first prove a lower bound of joint errors. Then, we provide an error decomposition theorem which upper bounds the accuracy disparity and decompose it into the distance between marginal label distributions and the distance between conditional representations. Based on these results, we give a geometric interpretation to visualize the feasible region of $\mathrm{Err}_{\mathcal{D}_0}$ and $\mathrm{Err}_{\mathcal{D}_1}$ and illustrate how error gap arises when learning a hypothesis h that minimizes the global square error. Motivated by the error decomposition theorem, we propose two algorithms to reduce accuracy disparity, connect the game-theoretic optima of the objective functions in our algorithms with our theorems, and describe the practical implementations of the algorithms. Due to the space limit, we defer all the detailed proofs to the appendix.

6.3.1 Bounds on Conditional Errors and Accuracy Disparity Gap

Before we provide the prescriptive analysis of the accuracy disparity problem in regression, it is natural to ask whether accuracy parity is achievable in the first place. Hence, we first provide a sufficient condition to achieve accuracy parity in regression.

Proposition 6.3.1. Assume both $\mathbb{E}_{\mathcal{D}_a}[Y]$ and $\mathbb{E}_{\mathcal{D}_a}[Y^2]$ are equivalent for any $A = a$, then using a constant predictor ensures accuracy parity in regression.

Proposition 6.3.1 states if the first two order moments of marginal label distributions are equal across different groups, then using a constant predictor leads to accuracy parity in regression. Proposition 6.3.1 is a relaxation of our proposed error decomposition theorem (Theorem 6.3.2) which requires the total variation distance between group-wise marginal label distributions to be zero. However, the condition rarely holds in real-world scenarios and it does not provide any insights to algorithm design. Next we provide more in-depth analysis to understand why accuracy disparity appears in regression models and provide algorithm interventions to mitigate the problem.

When we learn a predictor, the prediction function induces $X \xrightarrow{h} \widehat{Y}$, where $\widehat{Y}$ is the predicted target variable given by hypothesis h. Hence for any distribution $\mathcal{D}_0$ ($\mathcal{D}_1$) of X, the predictor also induces a distribution $h_\sharp\mathcal{D}_0$ ($h_\sharp\mathcal{D}_1$) of $\widehat{Y}$. Recall that the Wasserstein distance is metric, hence the following chain of triangle inequalities holds:

$$W_1(\mathcal{D}_0(Y), \mathcal{D}_1(Y)) \le W_1(\mathcal{D}_0(Y), h_\sharp\mathcal{D}_0) + W_1(h_\sharp\mathcal{D}_0, h_\sharp\mathcal{D}_1) + W_1(h_\sharp\mathcal{D}_1, \mathcal{D}_1(Y))$$

Intuitively, $W_1(\mathcal{D}_a(Y), h_\sharp\mathcal{D}_a)$ measures the distance between the true marginal label distribution and the predicted one when $A = a$. This distance is related to the prediction error of function h conditioned on $A = a$:

Lemma 6.3.1. Let $\widehat{Y} = h(X)$, then for $a \in \{0, 1\}$, $W_1(\mathcal{D}_a(Y), h_\sharp\mathcal{D}_a) \le \sqrt{\mathrm{Err}_{\mathcal{D}_a}(h)}$.

Now we can get the following theorem that characterizes the lower bound of joint error on different groups:

Theorem 6.3.1. Let $\widehat{Y} = h(X)$ be the predicted variable, then $\mathrm{Err}_{\mathcal{D}_0}(h) + \mathrm{Err}_{\mathcal{D}_1}(h) \ge \frac{1}{2}\left[\left(W_1(\mathcal{D}_0(Y), \mathcal{D}_1(Y)) - W_1(h_\sharp\mathcal{D}_0, h_\sharp\mathcal{D}_1)\right)_+\right]^2$.

In Theorem 6.3.1, we see that if the difference between marginal label distributions across groups is large, then statistical parity could potentially lead to a large joint error. Moreover, Theorem 6.3.1 could be extended to give a lower bound on the joint error incurred by h as well:

Corollary 6.3.1. Let $\widehat{Y} = h(X)$ and $\alpha = \mathcal{D}(A = 0) \in [0, 1]$, we have $\mathrm{Err}_{\mathcal{D}}(h) \geq \frac{1}{2}\min\{\alpha, 1-\alpha\} \cdot \left[\left(W_1(\mathcal{D}_0(Y), \mathcal{D}_1(Y)) - W_1(h_\sharp\mathcal{D}_0, h_\sharp\mathcal{D}_1)\right)_+\right]^2$.

Now we upper bound the error gap. We first relate the error gap to marginal label distributions and the predicted distributions conditioned on $Y = y$:

Theorem 6.3.2. If Assumption 6.2.1 holds, then for $\forall h \in \mathcal{H}$, let $\widehat{Y} = h(X)$, the following inequality holds:

$$\Delta_{\mathrm{Err}}(h) \leq 8M^2 d_{\mathrm{TV}}(\mathcal{D}_0(Y), \mathcal{D}_1(Y)) + 3M\min\{\mathbb{E}_{\mathcal{D}_0}[|\mathbb{E}_{\mathcal{D}_0^y}[\widehat{Y}] - \mathbb{E}_{\mathcal{D}_1^y}[\widehat{Y}]|], \mathbb{E}_{\mathcal{D}_1}[|\mathbb{E}_{\mathcal{D}_0^y}[\widehat{Y}] - \mathbb{E}_{\mathcal{D}_1^y}[\widehat{Y}]|]\}.$$

Remark We see that the error gap is upper bounded by two terms: the distance between marginal label distributions and the discrepancy between conditional predicted distributions across groups. Given a dataset, the distance between marginal label distributions is a constant since the marginal label distributions are fixed. For the second term, if we can minimize the discrepancy of the conditional predicted distribution across groups, we then have a model that is free of accuracy disparity when the marginal label distributions are well aligned.

Geometric Interpretation By Theorem 6.3.1 and Theorem 6.3.2, we can visually illustrate how accuracy disparity arises given data distribution and the learned hypothesis that aims to minimize the global square error. In Figure 6.1, given the hypothesis class $\mathcal{H}$, we use the line $\mathrm{Err}_{\mathcal{D}_0} + \mathrm{Err}_{\mathcal{D}_1} = B$ to denote the lower bound in Theorem 6.3.1 and the two lines $|\mathrm{Err}_{\mathcal{D}_0} - \mathrm{Err}_{\mathcal{D}_1}| = A$ to denote the upper bound in Theorem 6.3.2. These three lines form a feasible region (the green area) of $\mathrm{Err}_{\mathcal{D}_0}$ and $\mathrm{Err}_{\mathcal{D}_1}$ under the hypothesis class $\mathcal{H}$. For any optimal hypothesis h which is solely designed to minimize the overall error, the best the hypothesis h can do is to intersect with one of the two bottom vertices. For example, the hypotheses (the red dotted line and the blue dotted line) trying to minimize overall error intersect with the two vertices of the region to achieve the smallest $\mathrm{Err}_{\mathcal{D}_0}$-intercept ($\mathrm{Err}_{\mathcal{D}_1}$-intercept), due to the imbalance between these two groups. However, since these two vertices are not on the diagonal of the feasible region, there is no guarantee that the hypothesis can satisfy accuracy parity ($\mathrm{Err}_{\mathcal{D}_0} = \mathrm{Err}_{\mathcal{D}_1}$), unless we can shrink the width of green area to zero.

6.3.2 Algorithm Design

Inspired by Theorem 6.3.2, we can mitigate the error gap by aligning the group distributions via minimizing the distance of the conditional distributions across groups. However, it is intractable to do so explicitly in regression problems since Y can take infinite values on $\mathbb{R}$. Next we will present two algorithms to approximately solve the problem through adversarial representation learning.

Given a Markov chain $X \xrightarrow{g} Z \xrightarrow{h} \widehat{Y}$, we are interested in learning group-invariant conditional representations so that the discrepancy between the induced conditional distributions $\mathcal{D}_0^Y(Z = g(X))$ and $\mathcal{D}_1^Y(Z = g(X))$ is minimized. In this case, the second term of the upper bound in Theorem 6.3.2 is minimized. However, it is in general not feasible since Y is a continuous random variable. Instead, we propose to learn the representations of Z to minimize the discrepancy between the joint distributions $\mathcal{D}_0(Z = g(X), Y)$ and $\mathcal{D}_1(Z = g(X), Y)$. Next, we will show the distances between conditional predicted distributions $\mathcal{D}_0^Y(Z = g(X))$ and $\mathcal{D}_1^Y(Z = g(X))$ are minimized when we minimize the joint distributions $\mathcal{D}_0(Z = g(X), Y)$ and $\mathcal{D}_1(Z = g(X), Y)$ in Theorem 6.3.3 and Theorem 6.3.4.

To proceed, we first consider using the total variation distance to measure the distance between two distributions. In particular, we can choose to learn a binary discriminator $f : Z \times Y \longrightarrow \widehat{A}$ that achieves minimum binary classification error on discriminating between points sampled from two distributions. In practice, we use the cross-entropy loss as a convex surrogate loss. Formally, we are going to consider the following minimax game between g and f:

$$\min_{f \in \mathcal{F}} \max_{g} \quad \mathrm{CE}_{\mathcal{D}}(A \parallel f(g(X), Y)) \tag{6.1}$$

Interestingly, for the above equation, the optimal feature transformation g corresponds to the one that induces invariant conditional feature distributions.

Theorem 6.3.3. Consider the minimax game in (6.1). The equilibrium (g^*, f^*) of the game is attained when 1). $Z = g^*(X)$ is independent of A conditioned on Y; 2). $f^*(Z, Y) = \mathcal{D}(A = 1 \mid Y, Z)$.

Since in the equilibrium of the game Z is independent of A conditioned on Y, the optimal $f^*(Z, Y)$ could also be equivalently written as $f^*(Z, Y) = \mathcal{D}(A = 1 \mid Y)$, i.e., the only useful information for the discriminator in the equilibrium is through the external information Y. In Theorem 6.3.3, the minimum cross-entropy loss that the discriminator (the equilibrium of the game) can achieve is

$H(A \mid Z, Y)$ (see Proposition 6.7.1 in Appendix 6.7.1). For any feature transform g, by the basic property of conditional entropy, we have:

$$\min_{f\in\mathcal{F}} \mathrm{CE}_{\mathcal{D}}(A \parallel f(g(X), Y)) = H(A \mid Z, Y) = H(A \mid Y) - I(A; Z \mid Y).$$

We know that $H(A \mid Y)$ is a constant given the data distribution. The maximization of g in (6.1) is equivalent to the minimization of $\min_{Z=g(X)}\ I(A; Z \mid Y)$, and it follows that the optimal strategy for the transformation g is the one that induces conditionally invariant features, e.g., $I(A; Z \mid Y) = 0$. Formally, we arrive at the following minimax problem:

$$\min_{h,g}\max_{f\in\mathcal{F}}\ \mathrm{MSE}_{\mathcal{D}}(h(g(X)),\ Y) - \lambda\cdot\mathrm{CE}_{\mathcal{D}}(A \parallel f(g(X), Y))$$

In the above formulation, the first term corresponds to the minimization of prediction loss of the target task and the second term is the loss incurred by the adversary f. As a whole, the minimax optimization problem expresses a trade-off (controlled by the hyper-parameter $\lambda > 0$) between accuracy and accuracy disparity through the representation learning function g.

Wasserstein Variant Similarly, if we choose to align joint distributions via minimizing Wasserstein distance, the following theorem holds.

Theorem 6.3.4. Let the optimal feature transformation $g^* := \arg\min_g W_1(\mathcal{D}_0(g(X), Y), \mathcal{D}_1(g(X), Y))$, then $\mathcal{D}_0^Y(Z = g^*(X)) = \mathcal{D}_1^Y(Z = g^*(X))$ almost surely.

One notable advantage of using the Wasserstein distance instead of the TV distance is that, the Wasserstein distance is a continuous functional of both the feature map g as well as the discriminator f [218]. Furthermore, if both g and f are continuous functions of their corresponding model parameters, which is the case for models we are going to use in experiments, the objective function will be continuous in both model parameters. This property of the Wasserstein distance makes it more favorable from an optimization perspective. Using the dual formulation, equivalently, we can learn a Lipschitz function $f : Z \times Y \to \mathbb{R}$ as a witness function:

$$\min_{h,g,Z_0\sim g_\sharp\mathcal{D}_0,Z_1\sim g_\sharp\mathcal{D}_1}\ \max_{f:\|f\|_L\le 1} \mathrm{MSE}_{\mathcal{D}}(h(g(X)),\ Y) + \lambda\cdot\big|f(Z_0, Y) - f(Z_1, Y)\big|.$$

Game-Theoretic Interpretation We provide a game-theoretic interpretation of our algorithms in Figure 6.2 to make our algorithms easier to follow.

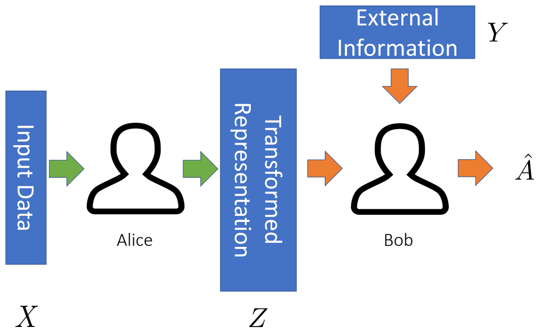
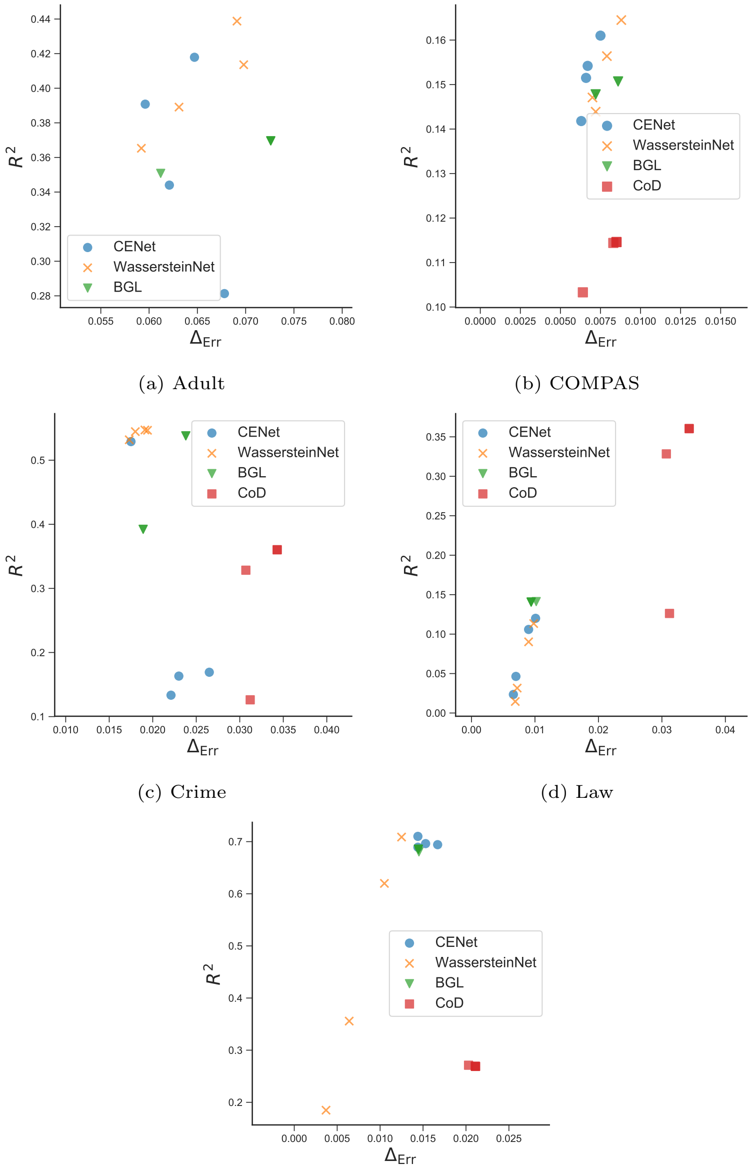


Figure 6.2: The game-theoretic illustration of our algorithms. Bob's goal is to guess the group membership A of each feature Z sent by Alice with the corresponding labels Y as the external information, while Alice's goal is to find a transformation from X to Z to confuse Bob.

As illustrated in Figure 6.2, consider Alice (encoder) and Bob (discriminator) participate a two-player game: upon receiving a set of inputs X, Alice applies a transformation to the inputs to generate the corresponding features Z and then sends them to Bob. Besides the features sent by Alice, Bob also has access to the external information Y, which corresponds to the corresponding labels for the set of features sent by Alice. Once having both the features Z and the corresponding labels Y from external resources, Bob's goal is to guess the group membership A of each feature sent by Alice, and to maximize his correctness as much as possible. On the other hand, Alice's goal is to compete with Bob, i.e., to find a transformation to confuse Bob as much as she can. Different from the traditional game without external information, here due to the external information Y Bob has access to, Alice cannot hope to fully fool Bob, since Bob can gain some insights about the group membership A of features from the external label information anyway. Nevertheless, Theorem 6.3.3 and Theorem 6.3.4 both state that when Bob uses a binary discriminator or a Wasstertein discriminator to learn A, the best Alice could do is to to learn a transformation g so that the transformed representation Z is insensitive to the values of A conditioned on any values of $Y = y$.

6.4 Experiments

Inspired by our theoretical results that decompose accuracy disparity into the distance between marginal label distributions and the distance between conditional representations, we propose two algorithms to mitigate it. In this section, we conduct experiments to evaluate the effectiveness of our proposed algorithms in reducing the accuracy disparity.



(a) Adult

(b) COMPAS

(c) Crime

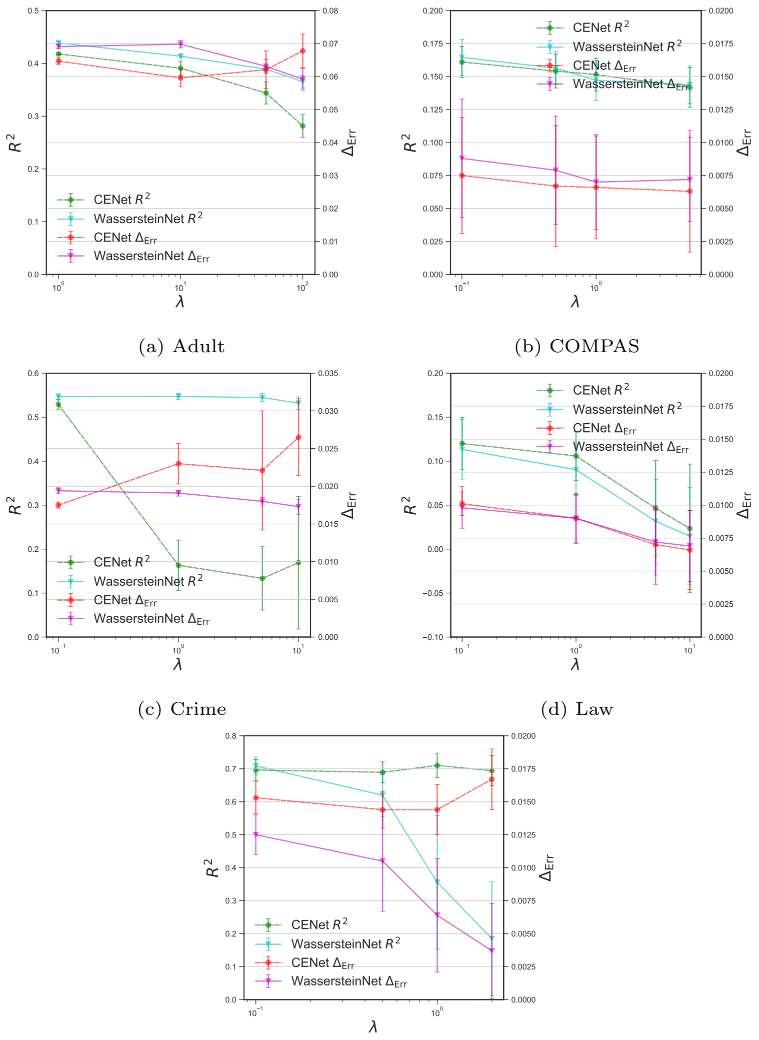
(d) Law

(e) Insurance

Figure 6.3: Overall results: R^2 regression scores and error gaps of different methods in five datasets. Our goal is to achieve high R^2 scores with small error gap values (i.e., the most desirable points are located in the upper-left corner). Our proposed methods achieve the best trade-offs in Adult, COMPAS, Crime and Insurance datasets.

6.4.1 Experimental Setup

Datasets We conduct experiments on five benchmark datasets: the Adult dataset [219], COMPAS dataset [220], Communities and Crime dataset [219], Law School dataset [221] and Medical Insurance Cost dataset [222]. All datasets contain binary sensitive attributes (e.g., male/female, white/non-white). We refer readers to Appendix 6.7.2 for detailed descriptions of the datasets and the data pre-processing pipelines. Note that although the Adult and COMPAS datasets are for binary classification



(a) Adult

(b) COMPAS

(c) Crime

(d) Law

(e) Insurance

Figure 6.4: R^2 regression scores and error gaps when λ changes in CENet and WassersteinNet. The general trend is that with the increase of λ, the error gap values and R^2 scores gradually decrease, except the cases where λ increases in CENet in Adult, Crime and Insurance dataset. The exceptions are caused by the instability of the training processes of CENet [2].

tasks, recent evidences [223–225] suggest that square loss achieves comparable performance with cross-entropy loss and hinge loss. In this regard, we take them as regression tasks with two distinctive ordinal values.

Methods We term the proposed algorithms CENet and WassersteinNet for our two proposed algorithms respectively and implement them using Pytorch [226].[2] To the best of our knowledge, no previous study aims to minimize accuracy disparity in regression using representation learning. However, there are other similar fairness notions and mitigation techniques proposed for regression

[2] Our code is publicly available at:
`https://github.com/JFChi/Understanding-and-Mitigating-Accuracy-Disparity-in-Regression`

and we add them as our baselines: (1) Bounded group loss (BGL) [227], which asks for the prediction errors for any groups to remain below a predefined level ϵ; (2) Coefficient of determination (CoD) [228], which asks for the coefficient of determination between the sensitive attributes and the predictions to remain below a predefined level ϵ.

For each dataset, we perform controlled experiments by fixing the regression model architectures to be the same. We train the regression models via minimizing mean squared loss. Among all methods, we vary the trade-off parameter (i.e., λ in CENet and WassersteinNet and ϵ in BGL and CoD) and report and the corresponding R^2 scores and the error gap values. For each experiment, we average the results for ten different random seeds. Note that CoD cannot be implemented on the Adult dataset since the size of the Adult dataset is large and the QCQP optimization algorithm to solve CoD needs a quadratic memory usage of the dataset size. We refer readers to Appendix 6.7.2 for detailed hyper-parameter settings in our experiments and Appendix 6.7.3 for additional experimental results.

6.4.2 Results and Analysis

The overall results are visualized in Figure 6.3. The following summarizes our observations and analyses: (1) Our proposed methods WassersteinNet and CENet are most effective in reducing the error gap values in all datasets compared to the baselines. Our proposed methods also achieve the best trade-offs in Adult, COMPAS, Crime and Insurance datasets: with the similar error gap values (R^2 scores), our methods achieve the highest R^2 scores (lowest error gap values). In the Law dataset, the error gap values decrease with high utility losses in our proposed methods due the significant trade-offs between the predictive power of the regressors and accuracy parity. We suspect this is because the feature noise distribution in one group differs significantly than the others in the Law dataset. (2) Among our proposed methods, WassersteinNet are more effective in reducing the error gap values while CENet might fail to decrease the error gaps in Adult, Crime and Insurance datasets and might even cause non-negligible reductions in the predictive performance of the regressors in Adult and Crime datasets. The reason behind it is that the minimax optimization in the training of CENet could lead to an unstable training process under the presence of a noisy approximation to the optimal discriminator [2]. We will provide more analysis in Figure 6.4 next. (3) Compared to our proposed methods, BGL and CoD can also decrease error gaps to a certain extent. This is because: (i) BGL aims to keep errors remaining relatively low in each group, which helps to reduce accuracy disparity; (ii) CoD aims to reduce the correlation between the sensitive attributes and the predictions (or the inputs) in the feature space, which might somehow reduce the dependency between the distributions of these two variables.

We further analyze how the trade-off parameter λ in the objective functions affect the performance of our methods. Figure 6.4 shows R^2 regression scores and error gaps when λ changes in CENet and WassersteinNet. We see the general trend is that with the increase of the trade-off parameter λ, the error gap values and R^2 scores gradually decrease. Plus, the increase of λ generally leads to the instability of training processes with larger variances of both R^2 scores and error gap values. In Adult, Crime and Insurance datasets, WassersteinNet is more effective in mitigating accuracy disparity when λ increases, while CENet fails to decrease the error gap values and might suffer from significant accuracy loss. The failure to decrease the error gap values with significant accuracy loss and variance indicates the estimation of total variation in minimax optimization for CENet could lead to a highly unstable training process [2].

6.5 Related Work

Algorithmic Fairness In the literature, two main notions of fairness, i.e., *group fairness* and *individual fairness*, has been widely studied [51, 52, 56, 59, 212, 229–231]. In particular, [215] analyzed the impact of data collection on discrimination (e.g., false positive rate, false negative rate, and zero-one loss) from the perspectives of bias-variance-noise decomposition, and they suggested collecting more training examples and collect additional variables to reduce discrimination. [232] argued that the loss difference among different groups is determined by the amount of latent (unobservable) feature noise and the difference between means, variances, and sizes of the groups with an assumption that there are a latent random feature and a noise feature that are involved in the generation of the observable features. [233] further found out that spurious features from inputs can hurt accuracy and affect groups disproportionately. [213] proposed an error decomposition theorem which upper bounds accuracy disparity in the classification setting by three terms: the sum of group-wise noise, the distance of marginal input distributions across groups and the discrepancy of group-wise optimal decision functions. However, their error decomposition theorem does not lead to any mitigation approaches in classification: minimizing the distance of marginal input distributions across groups does not necessarily mitigate accuracy disparity since it could possibly exacerbate the noise term and the discrepancy of group-wise optimal decision functions in the meantime. Besides, the optimal group-wise decision functions are unknown and intractable to approximate in the feature spaces, which also adds to the difficulty of applying their upper bound directly. In comparison, our work only assumes that there is a joint distribution where all variables are sampled and precisely characterizes disparate predictive accuracy in regression in terms of the distance between marginal label distributions and the distance between conditional representations. Inspired by our theoretical results, we also propose practical algorithms to mitigate the problem when collecting more data becomes infeasible.

Fair Regression A series of works focus on fairness under the regression problems [228, 234–238]. To the best of our knowledge, no previous study aimed to minimize accuracy disparity in regression from representation learning. However, there are different fairness notions and techniques proposed for regression: [227] proposed fair regression with bounded group loss (i.e., it asks that the prediction error for any protected group remains below some pre-defined level) and used exponentiated-gradient approach to satisfy BGL. [228] aimed to reduce the coefficient of determination between the sensitive attributes between the predictions to some pre-defined level and used an off-the-shelf convex optimizer to solve the problem. [239] used the Hirschfeld-Gebelein-Rényi Maximum Correlation Coefficient to generalize fairness measurement to continuous variables and ensured equalized odds (demographic parity) constraint by minimizing the χ^2 divergence between the predicted variable and the sensitive variable (conditioned on target variable). [240] considered regression problems in health care spending and proposed five fairness criteria (e.g., covariance constraint, net compensation penalization, etc.) in the healthcare domain. [241] proposed pairwise fairness notions (e.g., pairwise equal opportunity requires each pair from two arbitrary different groups to be equally-likely to be ranked correctly) for ranking and regression models. [242] studied the regression problem with demographic parity constraint and showed the optimal fair predictor is achieved in the Wasserstein barycenter of group distributions. In contrast, we source out the root of accuracy disparity in regression through the lens of information theory and reduce it via distributional alignment using TV distance and Wasserstein distance in the minimax games.

Fair Representation A line of works focus on building algorithmic fair decision making systems using adversarial techniques to learn fair representations [176, 243, 244]. The main idea behind is to learn a good representation of the data so that the data owner can maximize the accuracy while removing the information related to the sensitive attribute. [245] proposed a generalized framework to learn adversarially fair and transferable representations and suggests using the label information in the adversary to learn equalized odds or equal opportunity representations in the classification setting. Apart from adversarial representation, recent work also proposed to use distance metrics, e.g., the maximum mean discrepancy [246] and the Wasserstein distance [214] to remove group-related information. Prior to this work, it is not clear aligning conditional distributions via adversarial representation learning could lead to (approximate) accuracy parity. Our analysis is the first work to connect accuracy parity and (conditional) distributional alignment in regression and we also provide algorithm interventions to mitigate the problem where it is challenging to align conditional distributions in regression problems.

6.6 Conclusion

In this chapter, we theoretically and empirically study accuracy disparity in regression problems. Specifically, we prove an information-theoretic lower bound on the joint error and a complementary upper bound on the error gap across groups to depict the feasible region of group-wise errors. Our theoretical results indicate that accuracy disparity occurs inevitably due to the marginal label distributions differ across groups. To reduce such disparity, we further propose to achieve accuracy parity by learning conditional group-invariant representations using statistical distances. The game-theoretic optima of the objective functions in our proposed methods are achieved when the accuracy disparity is minimized. Our empirical results on five benchmark datasets demonstrate that our proposed algorithms help to reduce accuracy disparity effectively. We believe our results take an important step towards better understanding accuracy disparity in machine learning models.

6.7 Appendix of Chapter 6

6.7.1 Missing Proofs

Proposition 6.3.1. Assume both $\mathbb{E}_{\mathcal{D}_a}[Y]$ and $\mathbb{E}_{\mathcal{D}_a}[Y^2]$ are equivalent for any $A = a$, then using a constant predictor ensures accuracy parity in regression.

Proof. For $a \in \{0,1\}$, we have

$$\begin{aligned}
&\mathrm{Err}_{\mathcal{D}_a}(h)\\
&= \mathbb{E}_{\mathcal{D}_a}[(h(X) - Y)^2]\\
&= \mathbb{E}_{\mathcal{D}_a}[(h(X) - \mathbb{E}_{\mathcal{D}_a}(Y) + \mathbb{E}_{\mathcal{D}_a}(Y) - Y)^2]\\
&= \mathbb{E}_{\mathcal{D}_a}[(h(X) - \mathbb{E}_{\mathcal{D}_a}(Y))^2] + \mathbb{E}_{\mathcal{D}_a}[(Y - \mathbb{E}_{\mathcal{D}_a}(Y))^2] - 2\,\mathbb{E}_{\mathcal{D}_a}[(h(X) - \mathbb{E}_{\mathcal{D}_a}(Y))(Y - \mathbb{E}_{\mathcal{D}_a}(Y))].
\end{aligned}$$

It is easy to see the first two terms are equal across different groups since $\mathbb{E}_{\mathcal{D}_a}[Y]$, $\mathbb{E}_{\mathcal{D}_a}[Y^2]$ and $h(X)$ are the same across different groups. For the third term, we have

$$\begin{aligned}
&\mathbb{E}_{\mathcal{D}_a}[(h(X) - \mathbb{E}_{\mathcal{D}_a}(Y))(Y - \mathbb{E}_{\mathcal{D}_a}(Y))] \\
&= \mathbb{E}_{\mathcal{D}_a(X)}[\mathbb{E}_{\mathcal{D}_a(Y|X)}[(h(X) - \mathbb{E}_{\mathcal{D}_a}(Y))(Y - \mathbb{E}_{\mathcal{D}_a}(Y)) \mid X]] \\
&= \mathbb{E}_{\mathcal{D}_a(X)}[(h(X) - \mathbb{E}_{\mathcal{D}_a}[Y \mid X])(\mathbb{E}_{\mathcal{D}_a}[Y \mid X]) - \mathbb{E}_{\mathcal{D}_a}[Y \mid X])] \\
&= 0.
\end{aligned}$$

Thus, the errors across different groups made by the constant predictor are the same if $\mathbb{E}_{\mathcal{D}_a}[Y]$ and $\mathbb{E}_{\mathcal{D}_a}[Y^2]$ are equivalent across different groups. □

Lemma 6.3.1. Let $\widehat{Y} = h(X)$, then for $a \in \{0, 1\}$, $W_1(\mathcal{D}_a(Y), h_\sharp \mathcal{D}_a) \leq \sqrt{\mathrm{Err}_{\mathcal{D}_a}(h)}$.

Proof. The prediction error conditioned on $a \in \{0, 1\}$ is

$$\begin{aligned}
\mathrm{Err}_{\mathcal{D}_a}(h) = &\ \mathbb{E}[\big(Y - h(X)\big)^2 | A = a] \\
\geq &\ \mathbb{E}^2[|Y - h(X)| | A = a] \\
\geq &\ \Big(\inf_{\Gamma(\mathcal{D}_a(Y), \mathcal{D}_a(h(X)))} \mathbb{E}[|Y - h(X)|]\Big)^2 \\
= &\ W_1^2(\mathcal{D}_a(Y), h_\sharp \mathcal{D}_a).
\end{aligned}$$

Taking square root at both sides then completes the proof. □

Theorem 6.3.1. Let $\widehat{Y} = h(X)$ be the predicted variable, then $\mathrm{Err}_{\mathcal{D}_0}(h) + \mathrm{Err}_{\mathcal{D}_1}(h) \geq \frac{1}{2}\big[\big(W_1(\mathcal{D}_0(Y), \mathcal{D}_1(Y)) - W_1(h_\sharp \mathcal{D}_0, h_\sharp \mathcal{D}_1)\big)_+\big]^2$.

Proof. Since $W_1(\cdot, \cdot)$ is a distance metric, the result follows immediately the triangle inequality and Lemma 6.3.1:

$$W_1(\mathcal{D}_0(Y), \mathcal{D}_1(Y)) \leq \sqrt{\mathrm{Err}_{\mathcal{D}_0}(h)} + W_1(h_\sharp \mathcal{D}_0, h_\sharp \mathcal{D}_1) + \sqrt{\mathrm{Err}_{\mathcal{D}_1}(h)}.$$

Rearrange the equation above and by AM-GM inequality, we have

$$W_1(\mathcal{D}_0(Y), \mathcal{D}_1(Y)) - W_1(h_\sharp \mathcal{D}_0, h_\sharp \mathcal{D}_1) \leq \sqrt{\mathrm{Err}_{\mathcal{D}_0}(h)} + \sqrt{\mathrm{Err}_{\mathcal{D}_1}(h)} \leq \sqrt{2(\mathrm{Err}_{\mathcal{D}_0}(h) + \mathrm{Err}_{\mathcal{D}_1}(h))}.$$

Taking square at both sides then completes the proof. □

Corollary 6.3.1. Let $\widehat{Y} = h(X)$ and $\alpha = \mathcal{D}(A = 0) \in [0, 1]$, we have $\mathrm{Err}_{\mathcal{D}}(h) \geq \frac{1}{2}\min\{\alpha, 1-\alpha\} \cdot \big[\big(W_1(\mathcal{D}_0(Y), \mathcal{D}_1(Y)) - W_1(h_\sharp\mathcal{D}_0, h_\sharp\mathcal{D}_1)\big)_+\big]^2$.

Proof. The joint error is

$$\begin{aligned} & \mathrm{Err}_{\mathcal{D}}(h) \\ = {} & \alpha\,\mathrm{Err}_{\mathcal{D}_0}(h) + (1-\alpha)\,\mathrm{Err}_{\mathcal{D}_1}(h) \\ \geq {} & \min\{\alpha, 1-\alpha\}\big(\mathrm{Err}_{\mathcal{D}_0}(h) + \mathrm{Err}_{\mathcal{D}_1}(h)\big) \\ \geq {} & \frac{1}{2}\min\{\alpha, 1-\alpha\}\big[\big(W_1(\mathcal{D}_0(Y), \mathcal{D}_1(Y)) - W_1(h_\sharp\mathcal{D}_0, h_\sharp\mathcal{D}_1)\big)_+\big]^2. \quad \text{(Theorem 6.3.1)} \end{aligned}$$

□

Theorem 6.3.2. If Assumption 6.2.1 holds, then for $\forall h \in \mathcal{H}$, let $\widehat{Y} = h(X)$, the following inequality holds:

$$\Delta_{\mathrm{Err}}(h) \leq 8M^2 d_{\mathrm{TV}}(\mathcal{D}_0(Y), \mathcal{D}_1(Y)) + 3M\min\{\mathbb{E}_{\mathcal{D}_0}[|\mathbb{E}_{\mathcal{D}_0^y}[\widehat{Y}] - \mathbb{E}_{\mathcal{D}_1^y}[\widehat{Y}]|], \mathbb{E}_{\mathcal{D}_1}[|\mathbb{E}_{\mathcal{D}_0^y}[\widehat{Y}] - \mathbb{E}_{\mathcal{D}_1^y}[\widehat{Y}]|]\}.$$

Proof. First, we show that for $a \in \{0, 1\}$:

$$\mathrm{Err}_{\mathcal{D}_a}(h) = \mathbb{E}_{\mathcal{D}_a}[(h(X) - Y)^2] = \mathbb{E}_{\mathcal{D}_a}[h^2(X) - 2Yh(X) + Y^2] = \mathbb{E}_{\mathcal{D}_a}[h^2(X) - 2Yh(X)] + \mathbb{E}_{\mathcal{D}_a}[Y^2].$$

Next, we bound the error gap:

$$\begin{aligned} & |\mathrm{Err}_{\mathcal{D}_0}(h) - \mathrm{Err}_{\mathcal{D}_1}(h)| \\ = {} & |\mathbb{E}_{\mathcal{D}_0}[h^2(X) - 2Yh(X)] + \mathbb{E}_{\mathcal{D}_0}[Y^2] - \mathbb{E}_{\mathcal{D}_1}[h^2(X) - 2Yh(X)] - \mathbb{E}_{\mathcal{D}_1}[Y^2]| \\ \leq {} & |\mathbb{E}_{\mathcal{D}_0}[h^2(X) - 2Yh(X)] - \mathbb{E}_{\mathcal{D}_1}[h^2(X) - 2Yh(X)]| + |\mathbb{E}_{\mathcal{D}_0}[Y^2] - \mathbb{E}_{\mathcal{D}_1}[Y^2]|. \quad \text{(Triangle inequality)} \end{aligned}$$

For the second term, we can easily prove that

$$|\mathbb{E}_{\mathcal{D}_0}[Y^2] - \mathbb{E}_{\mathcal{D}_1}[Y^2]| = |\langle Y^2, d\mathcal{D}_0 - d\mathcal{D}_1\rangle| \leq \|Y\|_\infty^2 \|d\mathcal{D}_0 - d\mathcal{D}_1\|_1 \leq 2M^2 d_{\mathrm{TV}}(\mathcal{D}_0(Y), \mathcal{D}_1(Y)),$$

where the second equation follows Hölder's inequality and the last equation follow the definition of total variation distance. Now it suffices to bound the remaining term:

$$\begin{aligned}
&|\mathbb{E}_{\mathcal{D}_0}[h^2(X) - 2Yh(X)] - \mathbb{E}_{\mathcal{D}_1}[h^2(X) - 2Yh(X)]| \\
= &\left| \int h(\mathbf{x})(h(\mathbf{x}) - 2y)\,\mathrm{d}\mu_0(\mathbf{x}, y) - \int h(\mathbf{x})(h(\mathbf{x}) - 2y)\,\mathrm{d}\mu_1(\mathbf{x}, y) \right| \\
\leq &\left| \iint h(\mathbf{x})(h(\mathbf{x}) - 2y)\,\mathrm{d}\mu_0(\mathbf{x}|y) d\mu_0(y) - \iint h(\mathbf{x})(h(\mathbf{x}) - 2y)\,\mathrm{d}\mu_0(\mathbf{x}|y) d\mu_1(y) \right| \qquad \text{(Triangle inequality)} \\
&+ \left| \iint h(\mathbf{x})(h(\mathbf{x}) - 2y)\,\mathrm{d}\mu_1(\mathbf{x}|y) d\mu_1(y) - \iint h(\mathbf{x})(h(\mathbf{x}) - 2y)\,\mathrm{d}\mu_0(\mathbf{x}|y) d\mu_1(y) \right|.
\end{aligned}$$

We upper bound the first term:

$$\begin{aligned}
&\left| \iint h(\mathbf{x})(h(\mathbf{x}) - 2y)\,\mathrm{d}\mu_0(\mathbf{x}|y)\,\mathrm{d}\mu_0(y) - \iint h(\mathbf{x})(h(\mathbf{x}) - 2y)\,\mathrm{d}\mu_0(\mathbf{x}|y)\,\mathrm{d}\mu_1(y) \right| \\
\leq &\iint \big| h(\mathbf{x})(h(\mathbf{x}) - 2y)(\mathrm{d}\mu_0(y) - \mathrm{d}\mu_1(y)) \big|\,\mathrm{d}\mu_0(\mathbf{x}|y) \\
\leq &\int \big|\,\mathrm{d}\mu_0(y) - \mathrm{d}\mu_1(y) \big| \int \big| \sup_{\mathbf{x}} h(\mathbf{x}) \big| \big| h(\mathbf{x}) - 2y \big|\,\mathrm{d}\mu_0(\mathbf{x}|y) \\
\leq &M \int \mathbb{E}_{\mathcal{D}_0}[|h(X) - 2Y||Y = y]\,\big|\,\mathrm{d}\mu_0(y) - \mathrm{d}\mu_1(y) \big| \qquad \text{(Assumption 6.2.1)} \\
\leq &3M^2 \int \big|\,\mathrm{d}\mu_0(y) - \mathrm{d}\mu_1(y) \big| \qquad \text{(Assumption 6.2.1)} \\
\leq &6M^2 d_{\mathrm{TV}}(\mathcal{D}_0(Y), \mathcal{D}_1(Y)).
\end{aligned}$$

Note that the last equation follows the definition of total variation distance. For the second term, we have:

$$\begin{aligned}
&\left| \iint h(\mathbf{x})(h(\mathbf{x}) - 2y)\,\mathrm{d}\mu_1(\mathbf{x}|y)\,\mathrm{d}\mu_1(y) - \iint h(\mathbf{x})(h(\mathbf{x}) - 2y)\,\mathrm{d}\mu_0(\mathbf{x}|y)\,\mathrm{d}\mu_1(y) \right| \\
\leq &\left| \iint h^2(\mathbf{x})(\mathrm{d}\mu_1(\mathbf{x}|y) - \mathrm{d}\mu_0(\mathbf{x}|y))\,\mathrm{d}\mu_1(y) \right| + \left| \iint 2y\, h(\mathbf{x})(\mathrm{d}\mu_1(\mathbf{x}|y) - \mathrm{d}\mu_0(\mathbf{x}|y))\,\mathrm{d}\mu_1(y) \right| \qquad \text{(Triangle inequality)} \\
\leq &3M\ \mathbb{E}_{\mathcal{D}_1}[|\mathbb{E}_{\mathcal{D}_0^y}[\widehat{Y}] - \mathbb{E}_{\mathcal{D}_1^y}[\widehat{Y}]|]. \qquad \text{(Assumption 6.2.1)}
\end{aligned}$$

To prove the last equation, we first see that:

$$\begin{aligned}
&\left|\iint h^2(\mathbf{x})(\mathrm{d}\mu_1(\mathbf{x}|y) - \mathrm{d}\mu_0(\mathbf{x}|y))\,\mathrm{d}\mu_1(y)\right| \\
\leq &\left|\iint \big(\sup_{\mathbf{x}} h(\mathbf{x})\big) h(\mathbf{x})(\mathrm{d}\mu_1(\mathbf{x}|y) - \mathrm{d}\mu_0(\mathbf{x}|y))\,\mathrm{d}\mu_1(y)\right| \\
\leq &\, M \int \big|\mathbb{E}_{\mathcal{D}_0}[h(X)|Y=y] - \mathbb{E}_{\mathcal{D}_1}[h(X)|Y=y]\big|\,\mathrm{d}\mu_1(y) \quad \text{(Assumption 6.2.1)} \\
= &\, M\ \mathbb{E}_{\mathcal{D}_1}[|\mathbb{E}_{\mathcal{D}_0^y}[\widehat{Y}] - \mathbb{E}_{\mathcal{D}_1^y}[\widehat{Y}]|].
\end{aligned}$$

Similarly, we also have:

$$\begin{aligned}
&\left|\iint 2y\, h(\mathbf{x})(\mathrm{d}\mu_1(\mathbf{x}|y) - \mathrm{d}\mu_0(\mathbf{x}|y))\,\mathrm{d}\mu_1(y)\right| \\
\leq &\, 2\left|\iint (\sup y) h(\mathbf{x})(\mathrm{d}\mu_1(\mathbf{x}|y) - \mathrm{d}\mu_0(\mathbf{x}|y))\,\mathrm{d}\mu_1(y)\right| \\
\leq &\, 2M \int \big|\mathbb{E}_{\mathcal{D}_0}[h(X)|Y=y] - \mathbb{E}_{\mathcal{D}_1}[h(X)|Y=y]\big|\, d\mu_1(y) \quad \text{(Assumption 6.2.1)} \\
= &\, 2M\ \mathbb{E}_{\mathcal{D}_1}[|\mathbb{E}_{\mathcal{D}_0^y}[\widehat{Y}] - \mathbb{E}_{\mathcal{D}_1^y}[\widehat{Y}]|].
\end{aligned}$$

By symmetry, we can also see that:

$$|\mathbb{E}_{\mathcal{D}_0}[h^2(X) - 2Yh(X)] - \mathbb{E}_{\mathcal{D}_1}[h^2(X) - 2Yh(X)]| \leq 6M^2 d_{\mathrm{TV}}(\mathcal{D}_0(Y), \mathcal{D}_1(Y)) + 3M\,\mathbb{E}_{\mathcal{D}_1}[|\mathbb{E}_{\mathcal{D}_0^y}[\widehat{Y}] - \mathbb{E}_{\mathcal{D}_1^y}[\widehat{Y}]|].$$

Combine the above two equations yielding:

$$\begin{aligned}
&|\mathbb{E}_{\mathcal{D}_0}[h^2(X) - 2Yh(X)] - \mathbb{E}_{\mathcal{D}_1}[h^2(X) - 2Yh(X)]| \\
\leq &\, 6M^2 d_{\mathrm{TV}}(\mathcal{D}_0(Y), \mathcal{D}_1(Y)) + 3M \min\{\mathbb{E}_{\mathcal{D}_0}[|\mathbb{E}_{\mathcal{D}_0^y}[\widehat{Y}] - \mathbb{E}_{\mathcal{D}_1^y}[\widehat{Y}]|], \mathbb{E}_{\mathcal{D}_1}[|\mathbb{E}_{\mathcal{D}_0^y}[\widehat{Y}] - \mathbb{E}_{\mathcal{D}_1^y}[\widehat{Y}]|]\}.
\end{aligned}$$

Incorporating the terms back to the upper bound of the error gap then completes the proof. □

Theorem 6.3.3. Consider the minimax game in (6.1). The equilibrium (g^*, f^*) of the game is attained when 1). $Z = g^*(X)$ is independent of A conditioned on Y; 2). $f^*(Z, Y) = \mathcal{D}(A = 1 \mid Y, Z)$.

Proof. To prove Theorem 6.3.3, we first give Proposition 6.7.1.

Proposition 6.7.1. For any feature map $g : \mathcal{X} \to \mathcal{Z}$, assume that $\mathcal{F}$ contains all the randomized binary classifiers and $\mathcal{F} \ni f : \mathcal{Z} \times \mathcal{Y} \to \mathcal{A}$, then $\min_{f\in\mathcal{F}} \mathrm{CE}_{\mathcal{D}}(A \parallel f(g(X), Y)) = H(A \mid Z, Y)$.

Proof. By the definition of cross-entropy loss, we have:

$$\begin{aligned}
\mathrm{CE}_{\mathcal{D}}(A \parallel f) &= -\mathbb{E}_{\mathcal{D}}\left[\mathbb{I}(A=0)\log(1-f(g(X),Y)) + \mathbb{I}(A=1)\log(f(g(X),Y))\right] \\
&= -\mathbb{E}_{g_\sharp\mathcal{D}}\left[\mathbb{I}(A=0)\log(1-f(Z,Y)) + \mathbb{I}(A=1)\log(f(Z,Y))\right] \\
&= -\mathbb{E}_{Z,Y}\mathbb{E}_{A\mid Z,Y}\left[\mathbb{I}(A=0)\log(1-f(Z,Y)) + \mathbb{I}(A=1)\log(f(Z,Y))\right] \\
&= -\mathbb{E}_{Z,Y}\left[\mathcal{D}(A=0\mid Z,Y)\log(1-f(Z,Y)) + \mathcal{D}(A=1\mid Z,Y)\log(f(Z,Y))\right] \\
&= \mathbb{E}_{Z,Y}\left[D_{\mathrm{KL}}(\mathcal{D}(A\mid Z,Y)\parallel f(Z,Y))\right] + H(A\mid Z,Y) \\
&\geq H(A\mid Z,Y),
\end{aligned}$$

where $D_{\mathrm{KL}}(\cdot\|\cdot)$ denotes the KL divergence between two distributions. From the above inequality, it is also clear that the minimum value of the cross-entropy loss is achieved when $f(Z, Y)$ equals the conditional probability $\mathcal{D}(A = 1 \mid Z, Y)$, i.e., $f^*(Z, Y) = \mathcal{D}(A = 1 \mid Z = g(X), Y)$. □

Proposition 6.7.1 states that the minimum cross-entropy loss that the discriminator can achieve is $H(A \mid Z, Y)$ when f is the conditional distribution $\mathcal{D}(A = 1 \mid Z = g(X), Y)$. By the basic property of conditional entropy, we have:

$$\min_{f\in\mathcal{F}} \mathrm{CE}_{\mathcal{D}}(A \parallel f(g(X), Y)) = H(A \mid Z, Y) = H(A \mid Y) - I(A; Z \mid Y).$$

Note that $H(A \mid Y)$ is a constant given the distribution $\mathcal{D}$, so the maximization of g is equivalent to the minimization of $\min_{Z=g(X)}\ I(A; Z \mid Y)$, and it follows that the optimal strategy for the transformation g is the one that induces conditionally invariant features, e.g., $I(A; Z \mid Y) = 0$. On the other hand, if g^* plays optimally, then the optimal response of the discriminator f is given by

$$f^*(Z, Y) = \mathcal{D}(A = 1 \mid Z = g^*(X), Y) = \mathcal{D}(A = 1 \mid Y).$$

□

Theorem 6.3.4. Let the optimal feature transformation $g^* := \arg\min_g W_1(\mathcal{D}_0(g(X), Y), \mathcal{D}_1(g(X), Y))$, then $\mathcal{D}_0^Y(Z = g^*(X)) = \mathcal{D}_1^Y(Z = g^*(X))$ almost surely.

Proof. By the definition of Wasstertein distance, we have:

$$\begin{aligned}
W_1(\mathcal{D}_0(Z,Y),\mathcal{D}_1(Z,Y)) &= \inf_{\gamma\in\Gamma(\mathcal{D}_0,\mathcal{D}_1)} \int d((\mathbf{z}_0,y_0),(\mathbf{z}_1,y_1))\,\mathrm{d}\gamma((\mathbf{z}_0,y_0),(\mathbf{z}_1,y_1)) \\
&= \inf_{\gamma\in\Gamma(\mathcal{D}_0,\mathcal{D}_1)} \iint d((\mathbf{z}_0,y_0),(\mathbf{z}_1,y_1))\,\mathrm{d}\gamma(\mathbf{z}_0,\mathbf{z}_1 \mid y_0,y_1)\,\mathrm{d}\gamma(y_0,y_1) \\
&= \inf_{\gamma\in\Gamma(\mathcal{D}_0,\mathcal{D}_1)} \iint \|\mathbf{z}_0-\mathbf{z}_1\|_1 + |y_0-y_1|\,\mathrm{d}\gamma(\mathbf{z}_0,\mathbf{z}_1 \mid y_0,y_1)\,\mathrm{d}\gamma(y_0,y_1) \\
&\geq \inf_{\gamma\in\Gamma(\mathcal{D}_0,\mathcal{D}_1)} \iint |y_0-y_1|\,\mathrm{d}\gamma(y_0,y_1)\,\mathrm{d}\gamma(\mathbf{z}_0,\mathbf{z}_1 \mid y_0,y_1) \\
&= \inf_{\gamma\in\Gamma(\mathcal{D}_0(Y),\mathcal{D}_1(Y))} \int |y_0-y_1|\,\mathrm{d}\gamma(y_0,y_1) \\
&= W_1(\mathcal{D}_0(Y),\mathcal{D}_1(Y)).
\end{aligned}$$

To finish the proof, next we prove the lower bound is achieved when $\mathcal{D}_0^Y(Z=g^*(X)) = \mathcal{D}_1^Y(Z=g^*(X))$: it is easy to see $W_1(\mathcal{D}_0^Y(Z),\mathcal{D}_0^Y(Z)) = \int \|\mathbf{z}_0-\mathbf{z}_1\|_1\,\mathrm{d}\gamma(\mathbf{z}_0,\mathbf{z}_1 \mid y_0,y_1) = 0$ when the conditional distributions are equal. In this case, when the Wasserstein distance is minimized, then Z is conditionally independent of A given Y almost surely. □

6.7.2 Experimental Details

Adult The Adult dataset contains 48,842 examples for income prediction. The task is to predict whether the annual income of an individual is greater or less than 50K/year based on the attributes of the individual, such as education level, age, occupation, etc. In our experiment, we use gender (binary) as the sensitive attribute. The target variable (income) is an ordinal binary variable: 0 if < 50K/year otherwise 1. After data pre-processing, the dataset contains 30,162/15,060 training/test instances where the input dimension of each instance is 113. We show the data distributions for different demographic subgroups in Table 6.1.

To preprocess the dataset, we first filter out the data records that contain the missing values. We then remove the sensitive attribute from the input features and normalize the input features with its means and standard deviations. Note that we use one-hot encoding for the categorical attributes.

For our proposed methods, we use a three-layer neural network with ReLU as the activation function of the hidden layers and the sigmoid function as the output function for the prediction task (we take the first two layers as the feature mapping). The number of neurons in the hidden layers is 60. We train the neural networks with the Adadelta algorithm with the learning rate 0.1 and a batch size of 512. The models are trained in 50 epochs. For the adversary networks in CENet and

WassersteinNet, we use a two-layer neural network with ReLU as the activation function. The number of neurons in the hidden layers of the adversary networks is 60. The adversary network in CENet also uses sigmoid function as the output function. The weight clipping norm in the adversary network of WassersteinNet is 0.005. We use the gradient reversal layer [247] to implement the gradient descent ascent (GDA) algorithm for optimization of the minimax problem since it makes the training process more stable [248]. For the rest of the datasets we used in our experiments, we also use a gradient reversal layer to implement our algorithms.

We use the Fairlearn toolkit [249] to implement BGL: we use the exponentiated-gradient algorithm with the default setting as the mitigator and vary the upper bound $\epsilon \in \{0.1, 0.2, 0.3, 0.5\}$ of the bounded group loss constraint. For each value of ϵ, we average the results of ten different random seeds.

COMPAS The COMPAS dataset contains 6,172 instances to predict whether a criminal defendant will recidivate within two years or not. It contains attributes such as age, race, etc. In our experiment, we use race (white or non-white) as the sensitive attribute and recidivism as the target variable. We split the dataset into train and test sets with the ratio 7/3. We show the data distributions for different demographic subgroups in Table 6.2.

For all methods, we use a two-layer neural network with ReLU as the activation function of the hidden layers and the sigmoid function as the output function for the prediction task (we take the first layer as the feature mapping). The number of neurons in the hidden layers is 60. We train the neural networks with the Adadelta algorithm with the learning rate 1.0 and a batch size of 512. The models are trained in 50 epochs. For the adversary networks in CENet and WassersteinNet, we use a two-layer neural network with ReLU as the activation function. The number of neurons in the hidden layers of the adversary networks is 10. The adversary network in CENet also uses sigmoid function as the output function. The weight clipping norm in the adversary network of WassersteinNet is 0.05.

We use the Fairlearn toolkit to implement BGL: we use the exponentiated-gradient algorithm with the default setting as the mitigator and vary the upper bound $\epsilon \in \{0.1, 0.2, 0.3, 0.5\}$ of the bounded group loss constraint. For each value of ϵ, we average the results of ten different random seeds.

As for CoD, we follow the source implementation.[3] We use the same hyper-parameter settings

[3] https://github.com/jkomiyama/fairregresion

as [228]: We use the kernelized optimization with the random Fourier features and the RBF kernel (we vary hyper-parameter of the RBF kernel $\gamma \in \{0.1, 1.0, 10, 100\}$) and report the best results with minimal MSE loss for each time we change the fairness budget ϵ. We also vary $\epsilon \in \{0.01, 0.1, 0.5, 1.0\}$ and average the results of ten different random seeds.

Table 6.1: Data distribution of Y and A in Adult dataset.

	$Y = 0$	$Y = 1$
$A = 0$	20988	9539
$A = 1$	13026	1669

Table 6.2: Data distribution of Y and A in COMPAS dataset.

	$Y = 0$	$Y = 1$
$A = 0$	1849	1148
$A = 1$	1514	1661

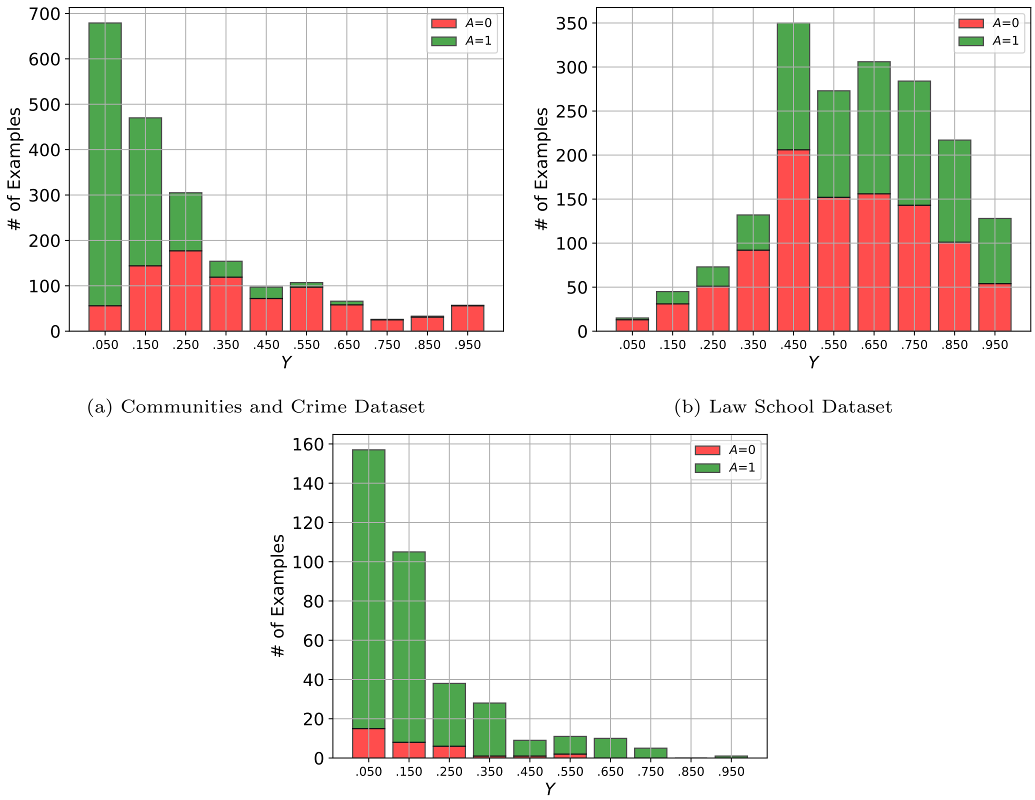
Communities and Crime The Communities and Crime dataset contains 1,994 examples of socio-economic, law enforcement, and crime data about communities in the United States. The task is to predict the number of violent crimes per 100K population. All attributes in the dataset have been curated and normalized to $[0, 1]$. In our experiment, we use race (binary) as the sensitive attribute: 1 if the population percentage of the white is greater or equal to 80% otherwise 0. After data pre-processing, the dataset contains 1,595/399 training/test instances where the input dimension of each instance is 96. We visualize the data distributions for different demographic subgroups in Figure 6.5a.

To preprocess the dataset, we first remove the non-predictive attributes and sensitive attributes from the input features. Note that all features in the dataset have already been normalized in $[0, 1]$ so that we do not perform additional normalization to the features. We then replace the missing values with the mean values of the corresponding attributes.

For all methods, we use a two-layer neural network with ReLU as the activation function of the hidden layers and the sigmoid function as the output function for the prediction task (we take the first layer as the feature mapping). The number of neurons in the hidden layers is 50. We train the neural networks with the Adadelta algorithm with the learning rate 0.1 and a batch size of 256. The models are trained in 100 epochs. For the adversary networks in CENet and WassersteinNet, we use a two-layer neural network with ReLU as the activation function. The number of neurons in the hidden layers of the adversary networks is 100. The adversary network in CENet also uses sigmoid function as the output function. The weight clipping norm in the adversary network of WassersteinNet is 0.002.

We use the Fairlearn toolkit to implement BGL: we use the exponentiated-gradient algorithm with the default setting as the mitigator and vary the upper bound $\epsilon \in \{0.01, 0.02, 0.03, 0.05\}$ of the bounded group loss constraint. For each value of ϵ, we average the results of ten different random seeds. Note that our experiment setup is different from [227], so our results cannot be directly compared to theirs.

As for CoD, we follow the same hyper-parameter settings as [228]: We use the kernelized optimization with the random Fourier features and the RBF kernel (we vary hyper-parameter of the RBF kernel $\gamma \in \{0.1, 1.0, 10, 100\}$) and report the best results with minimal MSE loss for each time we change the fairness budget ϵ. The hyper-parameter settings follow from [228]. We also vary $\epsilon \in \{0.01, 0.1, 0.5, 1.0\}$ and average the results of ten different random seeds. Note that our experiment setup is different from [228], so our results cannot be directly compared to theirs.



(a) Communities and Crime Dataset

(b) Law School Dataset

(c) Medical Insurance Cost Dataset

Figure 6.5: Data distributions for different demographic subgroups in three datasets.

Law School The Law School dataset contains 1,823 records for law students who took the bar passage study for Law School Admission[4]. The features in the dataset include variables such as

[4] We use the edited public version of the dataset which can be download here: `https://github.com/algowatchpenn/GerryFair/blob/master/dataset/lawschool.csv`

undergraduate GPA, LSAT score, full-time status, family income, gender, etc. In our experiment, we use gender as the sensitive attribute and undergraduate GPA as the target variable. We split the dataset into train and test sets with the ratio 8/2. We show the data distributions for different demographic subgroups in Figure 6.5b.

For all methods, we use a two-layer neural network with ReLU as the activation function of the hidden layers and the sigmoid function as the output function for the prediction task (we take the first layer as the feature mapping). The number of neurons in the hidden layers is 10. We train the neural networks with the Adadelta algorithm with the learning rate 0.1 and a batch size of 256. The models are trained in 100 epochs. For the adversary networks in CENet and WassersteinNet, we use a two-layer neural network with ReLU as the activation function. The number of neurons in the hidden layers of the adversary networks is 10. The adversary network in CENet also uses sigmoid function as the output function. The weight clipping norm in the adversary network of WassersteinNet is 0.2.

We use the Fairlearn toolkit to implement BGL: we use the exponentiated-gradient algorithm with the default setting as the mitigator and vary the upper bound $\epsilon \in \{0.01, 0.02, 0.03, 0.05\}$ of the bounded group loss constraint. For each value of ϵ, we average the results of ten different random seeds. Note that our experiment setup is different from [227], so our results cannot be directly compared to theirs.

As for CoD, we follow the same hyper-parameter settings as [228]: We use the kernelized optimization with the random Fourier features and the RBF kernel (we vary hyper-parameter of the RBF kernel $\gamma \in \{0.1, 1.0, 10, 100\}$) and report the best results with minimal MSE loss for each time we change the fairness budget ϵ. The hyper-parameter settings follow from [228]. We also vary $\epsilon \in \{0.01, 0.1, 0.5, 1.0\}$ and average the results of ten different random seeds. Note that our experiment setup is different from [228], so our results cannot be directly compared to theirs.

Medical Insurance Cost The medical insurance cost dataset [222] is a simulated dataset which was created using real-world demographic statistics from the U.S. Census Bureau.[5] The dataset reflect approximately reflect real-world conditions and has been used in the research of regression [250–252]. It contains 1,338 medical expense examples for patients in the United States, with features such as gender, age, BMI, etc., indicating characteristics of the patient and total annual medical expenses charged to the patients. In our experiment, we use gender as the sensitive attribute and the charged

[5] We download the public version of data here: `https://www.kaggle.com/mirichoi0218/insurance`

medical expenses as the target variable. In order to reflect the real-world scenarios where the accuracy disparity is significant due to the small and imbalanced dataset, we sub-sample the dataset: we randomly subsample 5% of examples with gender as male and 50% of examples with gender as female. After sub-sampling, we get 364 examples in total (33 male examples and 331 female examples). We split the dataset into train and test sets with the ratio 7/3. We visualize the data distributions for different demographic subgroups in Figure 6.5c.

For all methods, we use a two-layer neural network with ReLU as the activation function of the hidden layers and the sigmoid function as the output function for the prediction task (we take the first layer as the feature mapping). The number of neurons in the hidden layers is 7. We train the neural networks with the SGD algorithm with the learning rate 0.1 and a batch size of 64. The models are trained in 750 epochs. For the adversary networks in CENet and WassersteinNet, we use a two-layer neural network with ReLU as the activation function. The number of neurons in the hidden layers of the adversary networks is 7. The adversary network in CENet also uses sigmoid function as the output function. The weight clipping norm in the adversary network of WassersteinNet is 0.2.

We use the Fairlearn toolkit to implement BGL: we use the exponentiated-gradient algorithm with the default setting as the mitigator and vary the upper bound $\epsilon \in \{0.01, 0.1, 0.5, 1.0\}$ of the bounded group loss constraint. For each value of ϵ, we average the results of ten different random seeds.

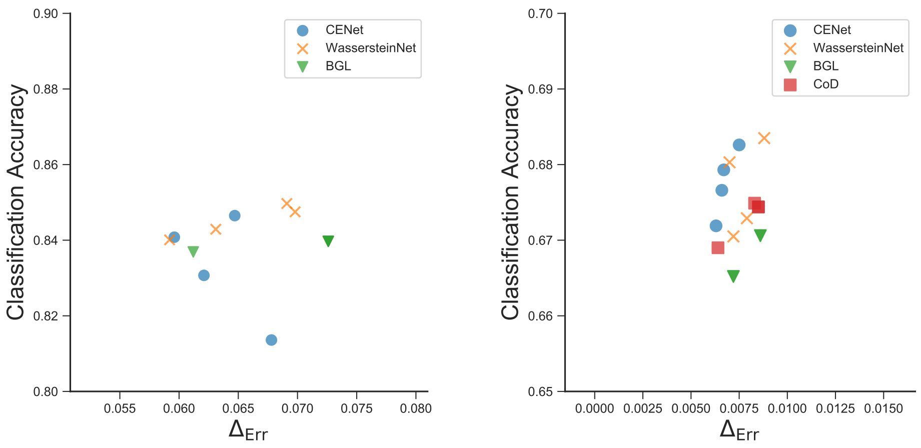
As for CoD, we follow the same hyper-parameter settings as [228]: We use the kernelized optimization with the random Fourier features and the RBF kernel (we vary hyper-parameter of the RBF kernel $\gamma \in \{0.1, 1.0, 10, 100\}$) and report the best results with minimal MSE loss for each time we change the fairness budget ϵ. The hyper-parameter settings follow from [228]. We also vary $\epsilon \in \{0.01, 0.1, 0.5, 1.0\}$ and average the results of ten different random seeds.

6.7.3 Additional Experimental Results and Analysis

In this section, we provide additional experimental results and analysis.

6.7.4 Classification Accuracy vs. Error Gaps in Adult and COMPAS Datasets

We also report the corresponding classification accuracy for Adult and COMPAS datasets here. In Figure 6.6, we can see that our proposed methods achieve the best trade-offs in terms of classification



(a) Adult

(b) COMPAS

Figure 6.6: Classification accuracy and error gaps of different methods in Adult and COMPAS datasets.

accuracies and error gap values.

6.7.5 Impact of Fairness Trade-off in the Baseline Methods

We present additional experimental results and analyses to gain more insights into how the fairness trade-off parameters (e.g., ϵ) affect the performance of the model predictive performance and accuracy disparity in baseline methods.

Table 6.3: R^2 regression scores and error gaps when ϵ changes in BGL.

Adult	ϵ	0.1	0.2	0.3	0.5
	R^2	0.3508	0.3696	0.3696	0.3696
	Δ_{Err}	0.0612	0.0726	0.0726	0.0726
COMPAS	ϵ	0.1	0.2	0.3	0.5
	R^2	0.1478	0.1478	0.1507	0.1507
	Δ_{Err}	0.0072	0.0072	0.0086	0.0086
Crime	ϵ	0.01	0.02	0.03	0.05
	R^2	0.3922	0.3922	0.5380	0.5380
	Δ_{Err}	0.0189	0.0189	0.0238	0.0238
Law	ϵ	0.01	0.02	0.03	0.05
	R^2	0.1407	0.1407	0.1407	0.1412
	Δ_{Err}	0.0094	0.0094	0.0094	0.0101
Insurance	ϵ	0.0001	0.01	0.05	0.1
	R^2	0.6804	0.6855	0.6855	0.6855
	Δ_{Err}	0.0145	0.0144	0.0144	0.0144

Table 6.3 shows R^2 regression scores and error gaps when ϵ changes in BGL. We see that with the decrease of the trade-off parameter ϵ, both the values of R^2 and error gaps decrease. This is because

when the upper bound of ϵ in BGL is small, the accuracy disparity is also mitigated. When ϵ is above/below a certain threshold, R^2 scores and error gap values then increase/decrease.

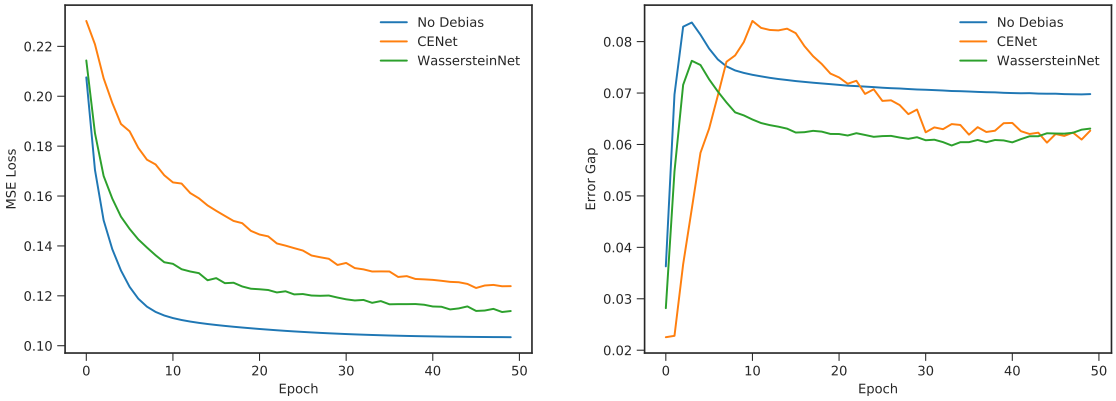
Table 6.4: R^2 regression scores and error gaps when ϵ changes in CoD.

COMPAS	ϵ	0.01	0.1	0.5	1.0
	R^2	0.1033	0.1144	0.1146	0.1146
	Δ_{Err}	0.0064	0.0083	0.0085	0.0085
Crime	ϵ	0.01	0.1	0.5	1.0
	R^2	0.1262	0.3284	0.3603	0.3603
	Δ_{Err}	0.0312	0.0307	0.0343	0.0343
Law	ϵ	0.01	0.1	0.5	1.0
	R^2	0.1262	0.3284	0.3606	0.3603
	Δ_{Err}	0.0312	0.0307	0.0343	0.0343
Insurance	ϵ	0.01	0.1	0.5	1.0
	R^2	0.2711	0.2691	0.2689	0.2689
	Δ_{Err}	0.0203	0.0210	0.0211	0.0211

Table 6.4 shows R^2 regression scores and error gaps when ϵ changes in CoD. We see that with the decrease of the trade-off parameter ϵ, both the values of R^2 and error gaps decrease in general.

6.7.6 Visualization of Training Processes

We visualize the training processes of our proposed methods CENet and WassersteinNet in the Adult dataset and COMPAS dataset in Figure 6.7 and Figure 6.8, respectively. We also compare their training dynamics with the model performance when we solely minimize the MSE loss (i.e., $\lambda = 0$) and we term it as No Debias.

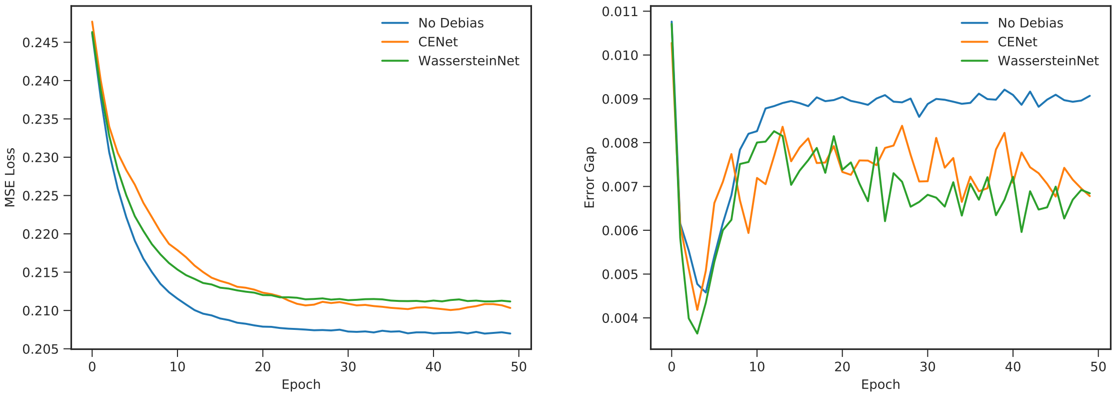


(a) MSE Loss

(b) Error Gap

Figure 6.7: Training visualization of CENet, WassersteinNet ($\lambda = 50$) and No Debias ($\lambda = 0$) in the Adult dataset.

In Figure 6.7 and Figure 6.8, we can see that as the training progresses go on, the MSE losses in both datasets are decreasing and finally converge. However, the training dynamics of error gaps are much more complex even in the No Debias case. Before convergence, the training dynamics of error



(a) MSE Loss

(b) Error Gap

Figure 6.8: Training visualization of CENet, WassersteinNet ($\lambda = 5$) and No Debias ($\lambda = 0$) in the COMPAS dataset.

gaps differs among different datasets. Our methods enforce the models to converge to the points where error gap are smaller while preserving the models' predictive performance. It is also worth to note that minimax optimization makes the training processes somehow unstable, especially when training CENet.

Chapter 7

Towards Return Parity in Markov Decision Processes

7.1 Introduction

The increasing use of automated decision-making systems trained with real-world data has raised serious concerns with potential unfairness caused by biased data, learning algorithms, and models. Decisions made by these systems have lasting and diverse effects on different social groups. For example, in predictive policing [253], each time, the decisions about the locations of future crimes are made by the predictive models. As a result, the discovered crime rates in different communities might change dynamically and interactively as feedback to the decisions being made by the models. Thus, the error rates of the predictive models for different communities might change over time, and error gaps among communities could possibly exacerbate in the long run. Similar feedback loops could also exist in the applications such as recommender systems (e.g., the user satisfactions from different demographic groups diverge over time), temporal resource allocation systems (e.g., the uneven resource allocation become more skew towards one group over the others), etc. This interplay between algorithmic decisions and the heterogeneous reactions caused by the decisions further complicates the analysis of (un)fairness problems in the dynamic environment.

Most prior works mainly focus on studying static fairness notions (e.g., demographic parity [51] and equalized odds [52]) in the settings of classification [52, 213, 254, 255] and regression [30, 63, 227, 228].

In a seminal work, [256] show that somewhat contrary to the common belief, enforcing static fairness constraints could do harm to the minority group even in a one-step feedback model. Motivated by this observation, a line of work aims to extend static fairness notions in the settings of online learning [257, 258] and multi-armed bandit [259–261]. However, these works do not take into account the interactions between the models and the environment: the decisions made by the models could potentially influence the state of our environment as well, as demonstrated by the predictive policing example. Other works study the interplay between (static) fairness notions and the population dynamics under simplified temporal dynamic models [262–268]. However, those simplified temporal dynamic models explicitly make task-specific assumptions on temporal dynamics and might not be able to precisely characterize the complex dynamics of a more general changing environment.

In this chapter, we study a fairness problem in Markov decision processes (MDPs) to understand the dynamics of performance disparity in the changing environments, taking into account the feedback loop caused by policies to the environments. Specifically, we propose *return parity*, a novel fairness notion that requires MDPs from different demographic groups that share the same state and action spaces to achieve approximately the same expected time-discounted rewards. To the best of our knowledge, our work is the first of this kind, in the sense that we study the long-term impact of policy functions in general MDPs. First, we formally show exact return parity cannot always be satisfied for any two MDPs and provide sufficient conditions that ensure return parity in terms of transitions, initial distributions, and the reward functions. Next, we derive a decomposition theorem for return disparity that decomposes it into the distance between group-wise reward functions, the discrepancy of group policies, and the discrepancy between state visitation distributions induced by the group policies. Motivated by the decomposition theorem, we propose algorithms to mitigate return disparity via learning a shared group policy with state visitation distributional alignment using integral probability metrics. We conduct experiments on two real-world benchmark datasets to corroborate the effectiveness of our proposed algorithms in reducing return disparity. Experimental results demonstrate that our proposed algorithms help to mitigate return disparity while maintaining the performance of policies.

7.2 Preliminaries

Notation and Problem Setup Throughout the paper, we mainly focus on discrete MDPs, where both the state and action spaces are finite.[1] A Markov decision process is a tuple $(\mathcal{S}, \mathcal{A}, \mu, T, r, \gamma)$, where $\mathcal{S}$ and $\mathcal{A}$ are the state space and the action space, respectively; μ is initial state distribution; $T \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{A}|\times|\mathcal{S}|}$ is the state transition function where $T(s' \mid s, a) = \Pr[S_{t+1} = s' \mid S_t = s, A_t = a]$; $r \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{A}|}$ is the reward function where $r(s, a)$ represents the reward obtained when taking action a in state s. Throughout the paper, we assume that the reward function is uniformly bounded, i.e., $\exists\ R > 0,\ \|r\|_\infty \le R$. We use $\gamma \in (0, 1)$ to denote the discount factor. Given a policy $\pi \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{A}|}$ (i.e., $\pi(a \mid s) = \Pr[A_t = a \mid S_t = s]$), the induced state transition under π is $P^\pi \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{S}|}$ where $P^\pi(s' \mid s) = \sum_{a\in\mathcal{A}} \pi(a \mid s)T(s' \mid s, a)$. The induced distribution over states under the policy π at time step t is

$$\mu^{(\pi,t)} = \begin{cases} \mu & \text{if } t = 0 \\ P^\pi \mu^{(\pi,t-1)} & \text{otherwise.} \end{cases}$$

The state visitation distribution (i.e., time-discounted distribution over states) is $\mu^\pi = (1 - \gamma)\sum_{t=0}^\infty \gamma^t \mu^{(\pi,t)} = (1 - \gamma)\sum_{t=0}^\infty (\gamma P^\pi)^t \mu$ and the occupancy measure (i.e., time-discounted distribution over state-action pairs) is $\rho^\pi(s, a) = \mu^\pi(s)\pi(a \mid s)$. We then define the value function w.r.t. the reward function r under the policy π as $v^\pi(s) = \mathbb{E}_\pi[\sum_{t=0}^\infty \gamma^t r_t \mid S_0 = s]$ and the Q-function as $q^\pi(s, a) = \mathbb{E}_\pi[\sum_{t=0}^\infty \gamma^t r_t \mid S_0 = s, A_0 = a]$, where r_t is the immediate reward at time step t. With all the notation defined above, the goal of reinforcement learning is to find a policy to maximize the value (expected return) under the initial state distribution:

$$\eta^\pi := \mathbb{E}_{s\in\mu}[v^\pi(s)] = \frac{1}{1-\gamma}\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r(s, a)\rho^\pi(s, a) = \frac{1}{1-\gamma}\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r(s, a)\mu^\pi(s)\pi(a \mid s).$$

Let $|\mathcal{S}| = m$ and $|\mathcal{A}| = n$. In practice, each state $s \in \mathcal{S}$ might represent features of an individual and the action enforced on the individual could lead to the change of features of the individual. We also assume there are two Markov decision processes that represent two different demographic groups (e.g., male/female, white/non-white) and the two MDPs share the same state space, action space, and discount factor but might differ in initial distributions, transitions, and reward functions. We use the subscript $g \in \{0, 1\}$ to denote the two groups. For example, μ_0 and μ_1 are the initial distributions of the two groups, respectively. Note that in our paper, we mainly discuss the setting there are two different demographic groups and follow the standard definition of return in the time-discounted

[1] The main results in Section 7.3 could be extended to continuous state space and action space.

MDPs, but the underlying theory and algorithms could easily be extended to the cases with finite $K > 2$ groups and undiscounted MDPs with finite time-horizon.

Next, we define ϵ-*return parity* as a fairness criterion to ask that different demographic groups share approximately the same long-term rewards under a policy:

Definition 7.2.1 (ϵ-Return Parity)**.** For $g \in \{0, 1\}$, two MDPs satisfy ϵ-*return parity* if $\Delta_{\text{Ret}} := |\mathbb{E}_{s\in\mu_0}[v^{\pi_0}(s)] - \mathbb{E}_{s\in\mu_1}[v^{\pi_1}(s)]| \leq \epsilon$.

Return parity could have different implications depending on the scenarios we consider: In recommender systems, if reward function corresponds to be users' satisfaction, return parity seeks similar users' satisfaction across different demographic groups in the long run; In predictive policing, if we define the reward function as the ratio between truly discovered incidents of crime (i.e., those directly observed by dispatched police as a result of the predictive policing algorithm) and the overall predicted incidents of crime in a time period, return parity requires a similar ratio for different communities over time. The complex temporal dynamic of the above scenarios could be modeled by MDPs. The goal in our setting is then to find two policies that maximize the weighted combination of expected returns of two MDPs respectively while satisfying ϵ-return parity:

$$\max_{\pi_0,\pi_1} \quad \lambda\, \eta^{\pi_0} + (1 - \lambda)\, \eta^{\pi_1}, \qquad \text{s.t.} \quad \Delta_{\text{Ret}} \leq \epsilon,$$

where $\lambda \in [0, 1]$ represents the proportion of group 0 over the entire population.

Integral Probability Metrics The integral probability metrics (IPMs) are a class of distance measures on probability distributions over the same probability space [269]. Formally, given two probability distributions $\mathcal{P}$ and $\mathcal{Q}$, the IPMs are defined as $d_{\mathcal{F}}(\mathcal{P}, \mathcal{Q}) = \sup_{f\in\mathcal{F}} |\int f\, \mathrm{d}\mathcal{P} - \int f\, \mathrm{d}\mathcal{Q}|$, where $\mathcal{F}$ is a class of real-valued bounded measurable functions on the space where the distributions are defined on. Different choices of $\mathcal{F}$ recover different distance metrics: If we choose $\mathcal{F} = \{f : \|f\|_L \leq 1\}$ where $\|\cdot\|_L$ denotes the Lipschitz semi-norm, then $d_{\mathcal{F}}(\mathcal{P}, \mathcal{Q})$ becomes *Wasserstein-1 distance* $W_1(\mathcal{P}, \mathcal{Q})$; If we choose $\mathcal{F} = \{f : \|f\|_{\mathcal{H}} \leq 1\}$ where $\|\cdot\|_{\mathcal{H}}$ denotes the norm in a reproducing kernel Hilbert space (RKHS), then $d_{\mathcal{F}}(\mathcal{P}, \mathcal{Q})$ becomes *maximum mean discrepancy* $\text{MMD}(\mathcal{P}, \mathcal{Q})$.

7.3 Analysis of Return Parity

In this section, we first show that return parity cannot always be satisfied between two MDPs that share the same state and action spaces and provide sufficient conditions under which return parity is possible. Then, we provide more insights into return disparity by proving an upper bound of the return gap between two MDPs. We defer all the detailed proofs to Appendix 7.9.1 due to the space limit.

7.3.1 Is Return Parity Always Possible?

Before we provide a rigorous analysis of return disparity in MDPs, it is vital to ask whether the exact return parity is always achievable. The following proposition gives a negative answer to this question:

Proposition 7.3.1. For any constant $c > 0$, there exist two MDPs that share the same state and action spaces, such that $\forall\ \pi_0, \pi_1 \in \Pi$, the return disparity $\Delta_{\text{Ret}} \geq c$.

For example, consider two MDPs share two states s_1 and s_2. Let $r(s_1, a) = c(1-\gamma) > 0$ and $r(s_2, a) = 0,\ \forall a \in \mathcal{A}$, $T(s_2 \mid s_1, a) = T(s_1 \mid s_2, a) = 0$ and $T(s_1 \mid s_1, a) = T(s_2 \mid s_2, a) = 1,\ \forall a \in \mathcal{A}$. Given $\mu_0 = [1, 0]^T$ and $\mu_1 = [0, 1]^T$, then the return gap $\Delta_{\text{Ret}} = c > 0$. In this case, it is impossible to find policies to satisfy ϵ-return parity for any $\epsilon < c$.

In addition, it is also natural to ask whether it is feasible to maximize the expected returns of the two MDPs simultaneously while achieving ϵ-return parity in general. Formally, with the help of the linear programming (LP) approach for MDPs [270, 271] and the duality of LP, it is equivalent to solve the following dual LP:

$$
\begin{aligned}
\max \sum_s \sum_a & \hat{\rho}_0(s,a) r_0(s,a) + \hat{\rho}_1(s,a) r_1(s,a) - \epsilon(b_0 + b_1) \\
\text{s.t.} \sum_a \hat{\rho}_0(s_i, a) - \gamma & \sum_s \sum_a T_0(s_i \mid s, a) \hat{\rho}_0(s,a) \\
& \qquad = (\lambda + b_0 - b_1)(\mu_0)_i \qquad \forall\ i \in [m] \\
\sum_a \hat{\rho}_1(s_i, a) - \gamma & \sum_s \sum_a T_1(s_i \mid s, a) \hat{\rho}_1(s,a) \\
& \qquad = (1 - \lambda + b_1 - b_0)(\mu_1)_i \ \ \forall\ i \in [m],
\end{aligned}
$$

where $\hat{\rho}_0(s,a), \hat{\rho}_1(s,a), b_0, b_1 \geq 0,\ \forall\ s, a$ are dual variables. Note that $\hat{\rho}_0(s,a)$ and $\hat{\rho}_1(s,a)$ have the interpretation of discounted state-action counts of the policy when $b_0 = b_1$, and b_0, b_1 are the corresponding dual variables of the ϵ-return parity constraint, representing the "per unit cost" of the overall return to achieve ϵ-return parity. The dual constraints are state transitions under the learned policies. We can now characterize a sufficient condition for the optimal policies π_0 and π_1 to satisfy ϵ-return parity with the dual formulation above.

Proposition 7.3.2. For $\forall\ \hat{\rho}_0(s,a), \hat{\rho}_1(s,a), b_0, b_1 \geq 0$, if there exists $i \in [m]$, such that

$$\sum_a \hat{\rho}_0(s_i, a) - \gamma \sum_s \sum_a T_0(s_i \mid s, a)\hat{\rho}_0(s,a) > (b_0 - b_1)(\mu_0)_i$$

$$\sum_a \hat{\rho}_1(s_i, a) - \gamma \sum_s \sum_a T_1(s_i \mid s, a)\hat{\rho}_1(s,a) > (b_1 - b_0)(\mu_1)_i$$

$$\sum_s \sum_a \hat{\rho}_0(s,a) r_0(s,a) + \hat{\rho}_1(s,a) r_1(s,a) \leq (b_0 + b_1)\epsilon$$

then the optimal policies π_0^* and π_1^* that maximize the expected returns of two MDPs satisfy ϵ-return parity.

In light of the dual LP formulation, the first two inequalities in Proposition 7.3.2 indicate the probability masses of the initial distributions in at least one state are greater than zero, and the last inequality requires the maximum value of the objective function in the dual LP formulation is no greater than zero.

7.3.2 A Decomposition Theorem for Return Disparity

The linear programming methods to solve the return disparity problem of MDPs become impractical in continuous or high-dimensional discrete state and action spaces. However, in many real-world scenarios where return parity is desired, the number of the states or actions is often large (e.g., recommend items to different demographic groups of users). In this section, we shall provide an upper bound to (1) quantitatively characterize return disparity in terms of the distance between the reward functions, the discrepancy of group policies, and the discrepancy between state visitation distributions and, (2) motivate our algorithm design to mitigate return disparity in continuous or high-dimensional discrete state and action spaces (Sec. 7.4).

Theorem 7.3.1. For $g \in \{0,1\}$, given policies $\pi_0, \pi_1 \in \Pi$ and assume there exists a witness function class $\mathcal{F} = \{f : \mathcal{S} \times \mathcal{A} \to \mathbb{R}\}$, such that the reward functions $r_g(s) = \mathbb{E}_{a \sim \pi_g(\cdot|s)}[r_g(a,s) \mid s] \in \mathcal{F}$ for $\forall\ s \in \mathcal{S}$, $a \in \mathcal{A}$, and $g \in \{0,1\}$, then the following holds:

$$\Delta_{\text{Ret}} \leq \frac{1}{1-\gamma}\left(\|r_0 - r_1\|_\infty + R \cdot \min\left\{\mathbb{E}_{s\sim\mu^{\pi_0}}\left[d_{\pi_0,\pi_1}(s)\right], \mathbb{E}_{s\sim\mu^{\pi_1}}\left[d_{\pi_0,\pi_1}(s)\right]\right\} + d_{\mathcal{F}}(\mu^{\pi_0}(s), \mu^{\pi_1}(s))\right),$$

where $d_{\pi_0,\pi_1}(s) := \|\pi_0(\cdot \mid s) - \pi_1(\cdot \mid s)\|_1$.

Remark We see that return disparity is upper bounded by three terms: the distance between group-wise reward functions, the discrepancy of group policies, and the discrepancy between state visitation distributions of the two MDPs. Given any two MDPs, the distance between group-wise reward functions is constant. It suggests that if the reward functions from two groups are drastically different, it may not be possible to ensure return parity by only looking at the policies and the state-visitation distributions. If we further assume the two MDPs share the same reward functions (i.e., $r_0 = r_1$) and the same policy (i.e., $\pi_0 = \pi_1$), then the upper bound in Theorem 7.3.1 is simplified as

$$\Delta_{\text{Ret}} \leq d_{\mathcal{F}}\big(\mu^{\pi_0}(s), \mu^{\pi_1}(s)\big).$$

In this case, it implies a sufficient condition to minimize return disparity is to find policies $\pi \in \Pi$ that minimize the distance between induced state visitation distributions in the two MDPs. In the subsequent sections, we assume the reward functions of different groups are (approximately) the same and propose algorithmic interventions to mitigate return disparity. For completeness, we also provide another decomposition theorem (Theorem 7.9.1) in Appendix 7.9.3 and motivate the design of another family of algorithms based on Theorem 7.9.1 in Appendix 7.9.4. We present the Theorem 7.3.1 in the main text since the algorithms (see Sec. 7.4) motivated by it are more efficient and stable in the application scenarios (e.g., recommender system) we consider.

7.4 Mitigation of Return Disparity

Inspired by the theoretical results in Theorem 7.3.1, we introduce a state visitation distribution alignment procedure, which can be naturally incorporated into existing RL methods, to encourage policies to maximize expected returns while keeping state visitation distribution similar to each other. We use deep Q-networks [272] as our baseline backbone algorithm, which has demonstrated its superior performance in recommender application [273], in which we will conduct experiments next. In what follows, we first briefly introduce how to learn the deep Q-networks and then discuss state visitation distributional alignment via IPMs. We give the main pipeline of our algorithms in Algorithm 1.

7.4.1 Preliminaries: Learning Deep Q-networks

The main idea behind learning deep Q-networks (Q-learning) is to approximate the value functions in high dimensional state and action spaces. Specifically, we aim at learning deep Q-network

$Q(s, a; \theta) : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ to approximate the reward when taking an action in a given state. Given the deep Q-network, we can construct the policy that maximizes the rewards: $\pi(s) = \arg\max_a Q(s, a)$. When the action space is discrete such as recommendation applications, the Q-network is often implemented as $Q(s, a; \theta) : \mathcal{S} \rightarrow \mathbb{R}^{|\mathcal{A}|}$ for efficiency, where the value in output dimension i represents estimated reward when taking action a_i given the state s.

During model training, the parameter θ of the Q-network is trained through a trial-and-error process. Take interactive recommendation process as an example: at each time step t, the recommender agent obtains a user's state s_t, and takes an action a_t (e.g., recommend an item) via the ε-greedy policy (i.e., with probability $1 - \varepsilon$ taking an action with the max Q-value, and with probability ε choosing a random action). The agent then receives the reward r_t (e.g., rating score on the recommended item) and the updated state s_{t+1} from the user's feedback and stores the experience (s_t, a_t, r_t, s_{t+1}) in replay buffer $\boldsymbol{D}$. After updating the replay buffer with batches of experiences from different users, the agent then optimizes the following loss function to improve the Q-network:

$$\mathcal{L}_Q(\theta) = \mathbb{E}_{(s_t, a_t, r_t, s_{t+1}) \sim \boldsymbol{D}}[(y_t - Q(s, a; \theta))^2], \tag{7.1}$$

with

$$y_t = r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta). \tag{7.2}$$

To avoid the overestimation problem [274] in original DQN, we adopt the double DQN architecture [275]: a target network Q' parameterized by θ' is utilized along with the online network Q. The online network is updated at each model update step, and the target network is a duplicate of the online network and updated with delay (soft update):

$$\theta' = \tau\theta + (1 - \tau)\theta', \tag{7.3}$$

where τ controls the update frequency. With the double DQN architecture, y_t in Eq. 7.2 is changed to

$$y_t = r_t + \gamma \max_{a_{t+1}} Q'(s_{t+1}, \arg\max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta); \theta'). \tag{7.4}$$

Algorithm 1 Algorithm to mitigate return disparity under the framework of DQN.

```
Initialize Q-functions $Q_\theta$ and $Q'_{\theta'}$, feature extractors $f_{\psi_0}$ and $f_{\psi_1}$, environment buffers $\boldsymbol{D}_0$ and $\boldsymbol{D}_1$
for each iteration do
    for each environment step do
        for $g \in \{0, 1\}$ do
            Sample an action $a_g$ using $\varepsilon$-greedy policy; Add the experience $(s_g, a_g, s'_g, r_g)$ to $\boldsymbol{D}_g$;
        end for
    end for
    for each model update step do
        Sample a mini-batch of experiences from $\boldsymbol{D}_0$ and $\boldsymbol{D}_1$; Update online and target DQNs and feature extractors $f_{\psi_g},\ g \in \{0, 1\}$ using Eq. 7.1 and Eq. 7.3;
    end for
    for each state visitation distributional alignment step do
        Update feature extractors $f_{\psi_0}$ and $f_{\psi_1}$ using Eq. 7.7 or Eq. 7.8;
    end for
end for
```

7.4.2 State Visitation distributional Alignment

Inspired by Theorem 7.3.1, we propose to align the state visitation distribution via learning group-invariant representation. Specifically, we introduce two feature extractors $f_{\psi_g},\ g \in \{0, 1\}$ before inputting s_t to Q-function. We alternatively optimize f_{ψ_g} between minimizing the loss of Q-function (Eq. 7.1) and minimizing the loss of state visitation distributional alignment. At each iteration, the feature extractors and the Q-function, parameterized by $\theta' = \{\psi_g \circ \theta\}$, are first trained jointly to minimize $\mathcal{L}_q(\theta')$. After updating the models, the feature extractors are then updated to minimize the loss of state visitation distributional alignment via minimizing the estimated integral probability metrics between different groups. The whole algorithm is shown in Algorithm 1. Next, we introduce the detailed steps of state visitation distributional alignment via Wasserstein-1 distance, which is a kind of IPM choosing $\mathcal{F} = \{f : \|f\|_L \leq 1\}$ and is more favorable than total variation distance from the optimization perspective [218].

Wasserstein-1 Distance Given samples s_0 from the state visitation distribution of group 0 and samples s_1 from the state visitation distribution of group 1, two feature extractors f_0 and f_1, with parameters ψ_0 and ψ_1, map the samples to feature representations: $h_0 = f_0(s_0)$ and $h_1 = f_1(s_1)$. Upon receiving the feature representations from both groups, we use a critic function f_c, parameterized by ω, to estimate Wasserstein-1 distance [218] by maximizing the following objective function:

$$\mathcal{L}_{\text{Wass}}(\psi_0, \psi_1, \omega) = \frac{1}{N_0} \sum_{i=1}^{N_0} f_c(h_0^i) - \frac{1}{N_1} \sum_{j=1}^{N_1} f_c(h_1^j). \tag{7.5}$$

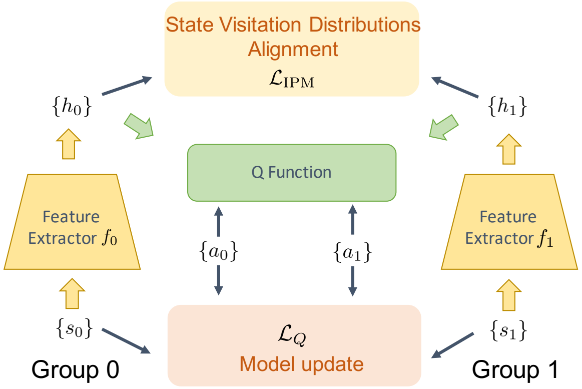


Figure 7.1: Illustration of our algorithms. At every iteration, we first update the models (e.g., Q-function and feature extractors). The feature extractors are then updated to minimize the loss of state visitation distributional alignment via minimizing the estimated integral probability metrics between different groups.

To enforce the Lipschitz constraint on the critic function f_c, we choose to minimize the gradient penalty loss [276] meanwhile:

$$\mathcal{L}_{\text{GP}}(\omega) = \mathbb{E}_{\hat{h}\sim\hat{\boldsymbol{D}}}[(\|\nabla f_c(\hat{h})\|_2 - 1)^2], \tag{7.6}$$

where $\hat{\boldsymbol{D}}$ represents the distribution of a uniformly distributed linear interpolations between the group visitation distributions. Finally, the overall minimax objective becomes

$$\min_{\psi_0,\psi_1} \max_{\omega} \mathcal{L}_{\text{Wass}}(\psi_0, \psi_1, \omega) - \beta \mathcal{L}_{\text{GP}}(\omega), \tag{7.7}$$

where β is the balancing coefficient and by convention it is set to be 10 [276]. To ensure the training stability and do less harm to the group that obtains lower return, we use block coordinate descent algorithm [277] to update the feature extractors: at each iteration of state visitation distributional alignment, we first identify the group with higher return choose to update its feature extractor while fixing the parameters of the feature extractor in the other group. Figure 7.1 illustrates our algorithm framework.

MMD Variant Similarly to Wasserstein distance, we can also use maximum mean discrepancy to align state visitation distributions. Let k be the kernel of the corresponding RKHS $\mathcal{H}$ on the feature space, then the squared MMD between the induced feature distributions of two groups $\boldsymbol{D}_{h_0}$ and $\boldsymbol{D}_{h_1}$ is

$$\text{MMD}^2(\boldsymbol{D}_{h_0}, \boldsymbol{D}_{h_1}) := \mathbb{E}_{h_0,h_0'}[k(h_0, h_0')] + \mathbb{E}_{h_1,h_1'}[k(h_1, h_1')] - 2\mathbb{E}_{h_0,h_1}[k(h_0, h_1)].$$

In practice, given samples from $\{h_0^1, \dots, h_0^{N_0}\} \sim \boldsymbol{D}_{h_0}$ and $\{h_1^1, \dots, h_1^{N_1}\} \sim \boldsymbol{D}_{h_1}$, then unbiased estimation of the squared MMD is

$$\mathcal{L}_{\text{MMD}}(\psi_0, \psi_1) = \frac{1}{N_0(N_0-1)} \sum_{i \neq i'} k(h_0^i, h_0^{i'}) + \frac{1}{N_1(N_1-1)} \sum_{j \neq j'} k(h_1^j, h_1^{j'}) - \frac{2}{N_0 N_1} \sum_i \sum_j k(h_0^i, h_1^j) \tag{7.8}$$

To align state visitation distributions, we optimize the feature extractor using Eq. 7.8. In the implementation of the MMD variant, we use a linear combination of RBF kernels with bandwidths $\{0.001, 0.005, 0.01, 0.05, 0.1, 1, 5, 10\}$ since it remains an open problem on choosing the optimal kernels.

Extension for Multi-group Fairness We could extend our algorithm for multi-group fairness by learning one feature extractor for each group. The model update step remains the same. To reduce model complexity, we might choose to align the state visitation distributions between the two groups with the largest return disparity. We leave this extension for future study.

7.5 Experiments

In the following, we conduct empirical evaluation on two real-world datasets, showing that our proposed algorithms help to mitigate return disparity while maintaining the performance of policies.[2]

7.5.1 Experimental Setup

Datasets The two real-world datasets we use are benchmark recommender system datasets with user demographic information (e.g., gender and age):

- MovieLens-1M[3] is a benchmark dataset consisting of 1 million ratings from more than 6000 users on more than 4000 movies on the MovieLens website. The movie ratings range from 1 to 5 and the users are provided with demographic information such as gender and age.

- Book-Crossing[4] is a book rating dataset collected from the Book-Crossing community. It consists of more than 1 million ratings from more than 278k users on about 271k books. The

[2] Our code is publicly available at: `https://github.com/JFChi/Return-Parity-MDP`
[3] `https://grouplens.org/datasets/movielens/`
[4] `http://www2.informatik.uni-freiburg.de/~cziegler/BX/`

book ratings range from 0 to 10, and the users are provided with demographic information such as age.

Our goal of using these datasets is to learn a recommender system that maximizes the expected long-term user satisfaction while keeping the user satisfaction in different demographic groups similar.

Environment Simulator We focus on evaluating our proposed RL algorithms on the two benchmark datasets by setting up an environment simulator [278, 279] to mimic online environments. Specifically, we normalize the user ratings in each dataset into the range $[-1, 1]$. Given a user's historical interaction with the recommender system at time step t (state s_t), the recommender system recommends an item (action a_t), and the user gives a rating to the recommended item (reward r_t). Following [278], we give the details of the state representation scheme used in this chapter in Figure 7.2.

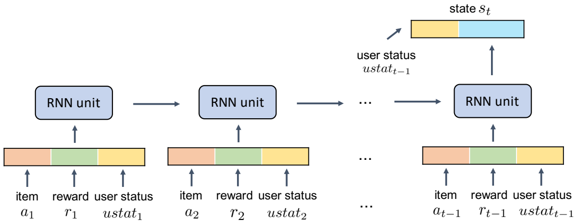


Figure 7.2: State representation in our RL environments.

As shown in Figure 7.2, the user state s_t is constructed by concatenating the user status at $t-1$ and the $t-1$-th output of a recurrent neural network (RNN). The user status at $t-1$ contains statistics information such as the number of positive/negative rewards before time step t. The input of the RNN at each time step is composed of three signals: the recommended item, the reward gained by recommending the corresponding item, and the user status. Note that each item and reward are mapped to an embedding vector and a one-hot vector before inputting to the RNN. We perform matrix factorization [280] to train an item embedding for each item to recommend.

For each dataset, we randomly split the users into two parts: 80% of the users are used for training, and the other 20% are used for testing. Due to the way we perform the train/test split in our datasets, our experiments are cold-start scenarios: the users in the test set have never been seen during training, and there is no interaction between the recommender system and the users at first. To deal with the problem, our recommender system recommends a popular item among the training users to a test user at time step t_0 and recommends non-repeated items to the user interactively

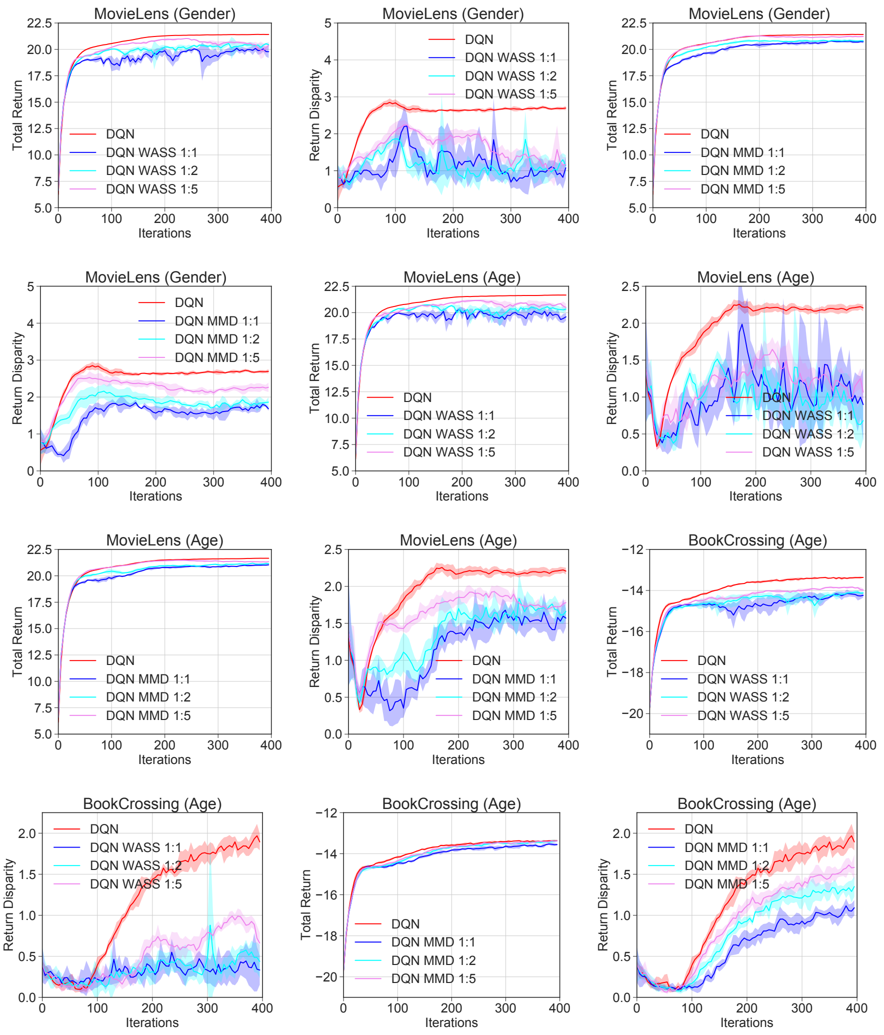


Figure 7.3: Learning curves of DQN, DQN-WASS and DQN-MMD in three different settings. The legend DQN WASS (DQN MMD) X:Y indicates the interval of model update versus the interval of state visitation distributional alignment with Wasserstein-1 distance (MMD distance) is X:Y (i.e., smaller numbers means more frequent updates). With the increase of the frequency of state visitation distributional alignment, return disparities are decreasing at the cost of performances of policies in all environments.

according to users' feedback. The episode length is set to be 32 for each user in the two datasets in our experiments.

Methods and Implementation Details We implement the DQN algorithm and our proposed algorithms that perform state visitation distributional alignment via Wasserstein-1 distance (DQN-WASS) and maximum mean discrepancy (DQN-MMD). To the best of our knowledge, our work is the first work that studies the return disparity problem in MDPs. We also adapt the state-of-the-art reduction approach, constrained policy optimization (CPO) [281] to our problem setting and find its training processes cannot converge, so we do not include the results here. We suspect the failure of

CPO is because their setting is different from ours: they assume the constraint of the policy is a deterministic function determined by states and actions, while in our setting, the reward gap in each environment step is dynamically changing, making it hard to estimate.

We take gender and age in MovieLens dataset and age in Book-Crossing dataset as the binary demographic groups (e.g., male/female, young/old). In each environment, we vary the model update frequency and the state visitation distributional alignment update frequency in DQN-Wass and DQN-MMD and report the corresponding overall return and the return disparity between groups. We average the results over five different random seeds and visualize the performance curves in each setting. The detailed data pre-processing pipelines and hyper-parameter configurations in our experiments are presented in Appendix 7.9.2.

7.5.2 Results and Analysis

The performance curves of DQN, DQN-WASS and DQN-MMD are shown in Figure 7.3. We can see that with the increase of the frequency of state visitation distributional alignment, return disparities are decreasing at the cost of performances of policies in all environments, which demonstrates the trade-offs between return maximization and return parity. Our methods can flexibly tune the trade-off between return maximization and return parity by controlling the relative update frequency ratio between the model update step and state visitation distributional alignment update step. Compared to DQN-WASS, DQN-MMD leads to more stable training processes with slightly higher overall returns and return disparities on average.

Next, we provide more insights into how our algorithms work. Since the MMD distance can be estimated analytically using Eq. 7.8, we visualize the learning curves of estimated MMD distances between the (induced) state visitation distributions in different MMD update settings in MovieLens (Gender) in Figure 7.4. We can see that the more frequent the MMD update is, the smaller the MMD distance. We also perform principal component analysis (PCA) on sampled state representations of different groups for DQN and DQN-MMD in MovieLens (Gender). The visualization results are presented in Figure 7.5. We can see from Figure 7.5 that DQN-MMD helps to align the state visitation distributions of different groups, which is consistent with our theoretical findings in Theorem 7.3.1.

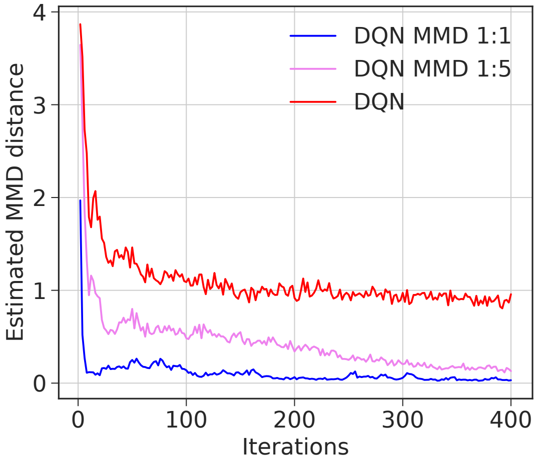
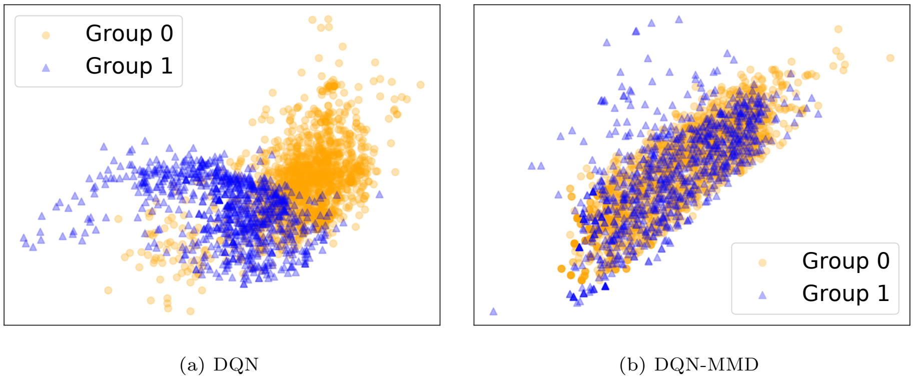


Figure 7.4: Training visualization of estimated MMD distance in MovieLens (Gender). With the increased frequency of MMD update, the MMD distance becomes smaller.



(a) DQN

(b) DQN-MMD

Figure 7.5: PCA visualization of sampled state visitation representations of different groups for DQN and DQN-MMD in MovieLens (Gender). Our method helps to align the state visitation distributions of different groups.

7.6 Related Work

Fairness in MDPs [282] propose an individual fairness notion which requires an algorithm to select an action if the long-term (discounted) reward of choosing that action is higher than the others in MDPs; [283] extend static fairness notions to Markov decision processes and propose model-based and model-free algorithms to maximize expected return while satisfying the fairness constraints. However, their proposed algorithms are based on linear programming or evolutionary algorithm, which cannot be scaled up to solve complex tasks compared to deep reinforcement learning approaches; [284] consider the fairness of multi-objective MDPs and propose to learn a policy with objective function satisfy the generalized Gini social welfare function [285]; [286] consider the fairness of item exposure problem in recommendation systems and extend CPO algorithm [281] to satisfy the fairness constraint; [287] propose an offline RL algorithm called quasi-Sheldonian reinforcement learning algorithm to determine whether given sets of policy distributions satisfy a set of return constraints with guarantees; [288] analyze how static fairness constraints affect the dynamics of group qualification rates. In contrast to their work, we propose the notion of return parity to quantify the dynamics of performance disparity in general changing environments: we theoretically analyze the notion of return parity by providing sufficient conditions where return parity can be satisfied and propose a decomposition theorem for return disparity. Based on our decomposition theorem, we propose algorithms to mitigate return disparity and empirically show the effectiveness of our algorithms.

Fairness under Other Temporal Models A series of works focus on study fairness in the setting of online learning [257, 258, 289], multi-armed bandit [259–261], and one-step feedback model [256,262,290]. In particular, [264] show that empirical risk minimization amplifies representation disparity over time with a low group retention rate for the underrepresented group. They further propose distributionally robust optimization to minimize the worst-case risk overall group distributions. [291] propose a time-dependent individual fairness notion that requires similar individuals should receive similar outcomes during the same time epoch. [263] model predictive policing problem using Pólya urn model and show that all police officers will be allocated to one location if more officers are constantly assigned to the locations with higher predicted crime rates. [292] propose effort-based fairness, which measures unfairness as the disparity in the effort made by individuals from each group to get a target outcome. [266] quantify the condition of the exacerbation of relative group ratio under the fairness constraints such as demographic parity and equalized odds.

7.7 Limitations and Future Work

One limitation of our approach is that experimental results show trade-offs between return parity and return maximization, which is possible if state (feature) visitation distributions induced by the optimal group policies for the MDPs are largely different. As a future direction, we plan to propose more stable and efficient algorithms to achieve Pareto optimality for return maximization and mitigation of return disparity beyond binary demographic groups.

It is also worth noting that in the special case when the length of time horizon is 1, then our notion of return parity corresponds to accuracy parity in classification settings (i.e., return maximization is reduced to accuracy maximization). However, it remains unclear whether our proposed fairness notion is compatible with other group fairness notions, such as demographic parity and equalized odds. We also leave this analysis as future work as it is a fundamental question that warrants an independent study.

7.8 Conclusions

In this work, we investigate the problem of return parity in MDPs both theoretically and empirically. In particular, we prove a decomposition theorem for return disparity which decomposes the return disparity of any two MDPs into the distance between group-wise reward functions, the discrepancy of group policies, and the discrepancy between state visitation distributions. We then provide algorithmic interventions to mitigate return disparity via state visitation distributional alignment with IPMs. To corroborate our theoretical results, we conduct experiments on two real-world benchmark datasets. Experimental results suggest that our proposed algorithms help to mitigate return disparity while maintaining the performance of policies. We believe our work takes an important step towards better understanding the dynamics of performance disparity in changing environments.

7.9 Appendix of Chapter 7

7.9.1 Missing Proofs

Proof of Proposition 7.3.1

Proposition 7.3.1. *For any constant $c > 0$, there exist two MDPs that share the same state and action spaces, such that $\forall\ \pi_0, \pi_1 \in \Pi$, the return disparity $\Delta_{\mathrm{Ret}} \geq c$.*

Proof. Consider two MDPs share two states s_1 and s_2. Let $r(s_1, a) = c(1-\gamma) > 0$ and $r(s_2, a) = 0,\ \forall a \in \mathcal{A}$, $T(s_2 \mid s_1, a) = T(s_1 \mid s_2, a) = 0$ and $T(s_1 \mid s_1, a) = T(s_2 \mid s_2, a) = 1,\ \forall a \in \mathcal{A}$. Given $\mu_0 = [1, 0]^T$ and $\mu_1 = [0, 1]^T$, then the expected return for group 0 is c, while the expected return for group 1 is 0. In this case, the return gap $\Delta_{\mathrm{Ret}} = c \geq 0$. □

Proof of Proposition 7.3.2

Proposition 7.3.2. For $\forall\ \hat{\rho}_0(s,a), \hat{\rho}_1(s,a), b_0, b_1 \geq 0$, if there exists $i \in [m]$, such that

$$\sum_a \hat{\rho}_0(s_i,a) - \gamma \sum_s \sum_a T_0(s_i \mid s,a)\hat{\rho}_0(s,a) > (b_0 - b_1)(\mu_0)_i$$

$$\sum_a \hat{\rho}_1(s_i,a) - \gamma \sum_s \sum_a T_1(s_i \mid s,a)\hat{\rho}_1(s,a) > (b_1 - b_0)(\mu_1)_i$$

$$\sum_s \sum_a \hat{\rho}_0(s,a) r_0(s,a) + \hat{\rho}_1(s,a) r_1(s,a) \leq (b_0 + b_1)\epsilon$$

then the optimal policies π_0^* and π_1^* that maximize the expected returns of two MDPs satisfy ϵ-return parity.

Proof. Let μ_0 and μ_1 be initial distributions of group 0 and 1, respectively, and denote $V^{\pi_0} = [v^{\pi_0}(s_1), \ldots, v^{\pi_0}(s_m)]^T$ and $V^{\pi_1} = [v^{\pi_1}(s_1), \ldots, v^{\pi_1}(s_m)]^T$. By definition, in order to satisfy return parity, the following equation should be satisfied:

$$\mu_0^T V^{\pi_0} = \mu_0^T V^{\pi_1}.$$

Consider solving the optimal value function via linear programming while satisfying the constraint of return parity, then the whole optimization problem becomes

$$\begin{aligned}
\min_{V^{\pi_0}, V^{\pi_1}} \quad & \lambda \mu_0^T V^{\pi_0} + (1-\lambda)\mu_1^T V^{\pi_1} \\
\text{s.t.} \quad & v^{\pi_0}(s) \geq r_0(s,a) + \gamma \sum_{s'} T_0(s' \mid s,a) v^{\pi_0}(s'), \quad \forall a \in \mathcal{A}, s \in \mathcal{S} \\
& v^{\pi_1}(s) \geq r_1(s,a) + \gamma \sum_{s'} T_1(s' \mid s,a) v^{\pi_1}(s'), \quad \forall a \in \mathcal{A}, s \in \mathcal{S} \\
& |\mu_0^T V^{\pi_0} - \mu_1^T V^{\pi_1}| \leq \epsilon
\end{aligned}$$

Next, we convert the constraints of the LP in the standard form:

$$\begin{aligned}
(\gamma T_0(s \mid s,a) - 1) v^{\pi_0}(s) + \gamma \sum_{s' \neq s} T_0(s' \mid s,a) v^{\pi_0}(s') \leq -r_0(s,a) \quad & \forall a \in \mathcal{A}, s \in \mathcal{S} \\
(\gamma T_1(s \mid s,a) - 1) v^{\pi_1}(s) + \gamma \sum_{s' \neq s} T_1(s' \mid s,a) v^{\pi_1}(s') \leq -r_1(s,a) \quad & \forall a \in \mathcal{A}, s \in \mathcal{S} \\
\mu_0^T V^{\pi_0} - \mu_1^T V^{\pi_1} \leq \epsilon & \\
\mu_1^T V^{\pi_1} - \mu_0^T V^{\pi_0} \leq \epsilon. &
\end{aligned}$$

The variant of Farkas' Lemma states that either the system $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ has a solution with $\mathbf{x} \geq 0$, or the system $\mathbf{A}^{\mathsf{T}}\mathbf{y} \geq 0$ has a solution with $\mathbf{b}^{\mathsf{T}}\mathbf{y} < 0$ and $\mathbf{y} \geq 0$. In other words, if we want the above system has a non-negative solution, we can easily show that $\nexists\ \hat{\rho}_0(s,a), \hat{\rho}_1(s,a), b_0, b_1 \geq 0$, $\forall\ i \in [m]$ such that

$$\begin{aligned}
\sum_s \sum_a T_0(s_i \mid s,a)\hat{\rho}_0(s,a) - \sum_a \hat{\rho}_0(s_i,a) + (b_0 - b_1)(\mu_0)_i \geq 0 \\
\sum_s \sum_a T_1(s_i \mid s,a)\hat{\rho}_1(s,a) - \sum_a \hat{\rho}_1(s_i,a) + (b_1 - b_0)(\mu_1)_i \geq 0 \\
\sum_s \sum_a \hat{\rho}_0(s,a)r_0(s,a) + \hat{\rho}_1(s,a)r_1(s,a) - (b_0 + b_1)\epsilon > 0
\end{aligned}$$

By the law of contraposition, it is equivalent to its contrapositive: $\forall\ \hat{\rho}_0(s,a), \hat{\rho}_1(s,a), b_0, b_1 \geq 0$, $\exists\ i \in [m]$ such that

$$\begin{aligned}
\sum_s \sum_a T_0(s_i \mid s,a)\hat{\rho}_0(s,a) - \sum_a \hat{\rho}_0(s_i,a) + (b_0 - b_1)(\mu_0)_i < 0 \\
\sum_s \sum_a T_1(s_i \mid s,a)\hat{\rho}_1(s,a) - \sum_a \hat{\rho}_1(s_i,a) + (b_1 - b_0)(\mu_1)_i < 0 \\
\sum_s \sum_a \hat{\rho}_0(s,a)r_0(s,a) + \hat{\rho}_1(s,a)r_1(s,a) - (b_0 + b_1)\epsilon \leq 0
\end{aligned}$$

Reorganizing the above equations then completes the proof. □

Proof of Theorem 7.3.1

Theorem 7.3.1. For $g \in \{0,1\}$, given policies $\pi_0, \pi_1 \in \Pi$ and assume there exists a witness function class $\mathcal{F} = \{f : \mathcal{S} \times \mathcal{A} \to \mathbb{R}\}$, such that the reward functions $r_g(s) = \mathbb{E}_{a \sim \pi_g(\cdot|s)}[r_g(a,s) \mid s] \in \mathcal{F}$ for $\forall\ s \in \mathcal{S}$, $a \in \mathcal{A}$, and $g \in \{0,1\}$, then the following holds:

$$\Delta_{\text{Ret}} \leq \frac{1}{1-\gamma}\Big(\|r_0 - r_1\|_\infty + R \cdot \min\left\{\mathbb{E}_{s\sim\mu^{\pi_0}}\left[d_{\pi_0,\pi_1}(s)\right], \mathbb{E}_{s\sim\mu^{\pi_1}}\left[d_{\pi_0,\pi_1}(s)\right]\right\} + d_{\mathcal{F}}(\mu^{\pi_0}(s), \mu^{\pi_1}(s))\Big),$$

where $d_{\pi_0,\pi_1}(s) := \|\pi_0(\cdot \mid s) - \pi_1(\cdot \mid s)\|_1$.

Proof. By definition, the return disparity is

$$\begin{aligned}
\Delta_{\text{Ret}} &= \left|\mathbb{E}_{s\in\mu_0}[v^\pi(s)] - \mathbb{E}_{s\in\mu_1}[v^\pi(s)]\right| \\
&= \frac{1}{1-\gamma}\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_0(s,a)\rho^{\pi_0}(s,a) - \sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\rho^{\pi_1}(s,a)\right| \\
&= \frac{1}{1-\gamma}\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_0(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_1}(s,a)\right| \\
&= \frac{1}{1-\gamma}\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_0(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_0}(s,a) + r_1(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_1}(s,a)\right| \\
&\le \frac{1}{1-\gamma}\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_0(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_0}(s,a)\right| + \frac{1}{1-\gamma}\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_1}(s,a)\right|,
\end{aligned}$$

where the first term is upper bounded by

$$\begin{aligned}
\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_0(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_0}(s,a)\right| &= \left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} \big(r_0(s,a) - r_1(s,a)\big)\rho^{\pi_0}(s,a)\right| \\
&\le \max_{s,a}|r_0(s,a) - r_1(s,a)| \sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} \rho^{\pi_0}(s,a) = \|r_0 - r_1\|_\infty,
\end{aligned}$$

and the second term is upper bounded by

$$\begin{aligned}
&\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_1}(s,a)\right| \\
&= \left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\mu^{\pi_0}(s)\pi_0(a \mid s) - r_1(s,a)\mu^{\pi_1}(s)\pi_1(a \mid s)\right| \\
&= \left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\mu^{\pi_0}(s)\pi_0(a \mid s) - r_1(s,a)\mu^{\pi_0}(s)\pi_1(a \mid s) + r_1(s,a)\mu^{\pi_0}(s)\pi_1(a \mid s) - r_1(s,a)\mu^{\pi_1}(s)\pi_1(a \mid s)\right| \\
&\le \left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\mu^{\pi_0}(s)\pi_0(a \mid s) - r_1(s,a)\mu^{\pi_0}(s)\pi_1(a \mid s)\right| \\
&\quad + \left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\mu^{\pi_0}(s)\pi_1(a \mid s) - r_1(s,a)\mu^{\pi_1}(s)\pi_1(a \mid s)\right|.
\end{aligned} \tag{7.9}$$

The first term in the upper bound of 7.9 is

$$\begin{aligned}
\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\mu^{\pi_0}(s)\pi_0(a \mid s) - r_1(s,a)\mu^{\pi_0}(s)\pi_1(a \mid s)\right| &\le R \sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} \mu^{\pi_0}(s)\Big|\pi_0(a \mid s) - \pi_1(a \mid s)\Big| \\
&\le R\ \mathbb{E}_{s\sim\mu^{\pi_0}}\left[\sum_{a\in\mathcal{A}}\Big|\pi_0(a \mid s) - \pi_1(a \mid s)\Big|\right] \\
&= R\ \mathbb{E}_{s\sim\mu^{\pi_0}}\left[\sum_{a\in\mathcal{A}}\|\pi_0(\cdot \mid s) - \pi_1(\cdot \mid s)\|_1\right],
\end{aligned}$$

and the second term in the upper bound of 7.9 is

$$\begin{aligned}
&\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\mu^{\pi_0}(s)\pi_1(a \mid s) - r_1(s,a)\mu^{\pi_1}(s)\pi_1(a \mid s)\right| \\
&= \left|\sum_{s\in\mathcal{S}}\mu^{\pi_0}(s)\sum_{a\in\mathcal{A}} r_1(s,a)\pi_1(a \mid s) - \sum_{s\in\mathcal{S}}\mu^{\pi_1}(s)\sum_{a\in\mathcal{A}} r_1(s,a)\pi_1(a \mid s)\right| \\
&= \left|\sum_{s\in\mathcal{S}}\mu^{\pi_0}(s)\ r_1(s) - \sum_{s\in\mathcal{S}}\mu^{\pi_1}(s)\ r_1(s)\right| \\
&\leq \sup_{r_1(s)\in\mathcal{F}}\left|\sum_{s\in\mathcal{S}}\mu^{\pi_0}(s)\ r_1(s) - \sum_{s\in\mathcal{S}}\mu^{\pi_1}(s)\ r_1(s)\right| \\
&= d_{\mathcal{F}}(\mu^{\pi_0}(s), \mu^{\pi_1}(s)).
\end{aligned}$$

By symmetry of 7.9, we also have

$$\begin{aligned}
&\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_1}(s,a)\right| \\
&\leq R\ \mathbb{E}_{s\sim\mu^{\pi_1}}\left[\sum_{a\in\mathcal{A}}\|\pi_0(\cdot \mid s) - \pi_1(\cdot \mid s)\|_1\right] + d_{\mathcal{F}}(\mu^{\pi_0}(s), \mu^{\pi_1}(s))
\end{aligned}$$

Combining the above results then completes the proof. □

7.9.2 Experimental Details

MovieLen-1M The original rating matrix of the dataset is a sparse matrix. In order to learn a good item embedding for each user, we first filter out the items which get less than 64 ratings (the dataset has ensured that each user has at least 20 ratings). We take gender (e.g., male/female) and age (e.g., ≥ 45 / < 45 years old) as the binary demographic groups. Similar to [279], we also perform the matrix completion [280] on the original rating matrix to avoid the sparsity of rating signals in the original data. To mimic the real-world scenarios where the number of users could be skewed across different demographic groups, we downsample one group's data by a factor of 10. Similar to [278, 293], we fix the item embedding when updating our models and pre-train the RNNs.[5] The dimension of the item embedding vector, reward vector and user status (i.e., the inputs of RNNs) are 50, 20, and 9, respectively. Thus, the state dimension is 88. We give the details of training hyperparameters in Table 7.1.

Book-Crossing Similar to MovieLen-1M, we first filter out the items which get less than 32 ratings and users who have less than 16 ratings. We take age (e.g., ≥ 35 / < 35 years old) as the binary

[5] `https://github.com/chenhaokun/TPGR`

demographic groups. The rest of data pre-processing pipelines are similar to those in the MovieLen dataset. The dimension of the item embedding vector, reward vector and user status (i.e., the inputs of RNNs) are 60, 20 and 9, respectively. Thus, the state dimension is 98. We give the details of training hyperparameters in Table 7.1.

	MovieLens (Gender)	MovieLens (Age)	Book-Crossing (Age)
Q-network	MLP with hidden size [128]		
Soft-update frequency τ	0.99		
Iteration	400		
Learning Rate	1e-3		
Q-network Weight Decay	1e-6		
ε-greedy Policy	Linear decay from max $\varepsilon = 1.0$ to min $\varepsilon = 0.1$ with decay steps 160		
Sample Batch Size	1000		
Update Batch Size	10000		
Update per Iteration	10		
Buffer Size	200000		
Feature Extractor	MLP with hidden size the same as the state dimension		

Table 7.1: Hyperparameter settings in our experiments.

7.9.3 Another Decomposition Theorem for Return Disparity

Theorem 7.9.1. For $g \in \{0, 1\}$, given policies $\pi_0, \pi_1 \in \Pi$ and assume there exists a witness function class $\mathcal{F} = \{f : \mathcal{S} \times \mathcal{A} \to \mathbb{R}\}$, such that the reward functions $r_g(\cdot, \cdot) \in \mathcal{F}$ for $\forall\ s \in \mathcal{S}, a \in \mathcal{A}$ and $g \in \{0, 1\}$, then the following holds:

$$\Delta_{\text{Ret}} \leq \frac{1}{1-\gamma}\bigg(\|r_0 - r_1\|_\infty + d_{\mathcal{F}}\big(\rho^{\pi_0}(s,a), \rho^{\pi_1}(s,a)\big)\bigg),$$

Remark We see that return disparity is upper bounded by two terms: the distance between group-wise reward functions and the discrepancy between occupancy measures of the two MDPs. Given any two MDPs, the distance between group-wise reward functions is constant. If we further assume the two MDPs share the same reward function (i.e., $r_0(s,a) = r_1(s,a) = r(s,a),\ \forall s \in \mathcal{S}, a \in \mathcal{A}$), then the upper bound is simplified as

$$\Delta_{\text{Ret}} \leq d_{\mathcal{F}}\big(\rho^{\pi_0}(s,a), \rho^{\pi_1}(s,a)\big).$$

In this case, Theorem 7.9.1 implies a sufficient condition to minimize return disparity is to find policies $\pi_0, \pi_1 \in \Pi$ that minimize the distance between induced occupancy measures in the two

MDPs. In what follows, we first give the detailed proof of Theorem 7.9.1 and give the algorithm design inspired by Theorem 7.9.1 in the subsequent section.

Proof. By definition, the return disparity is

$$\begin{aligned}\Delta_{\text{Ret}} =& \left|\mathbb{E}_{s\in\mu_0}[v^\pi(s)] - \mathbb{E}_{s\in\mu_1}[v^\pi(s)]\right| \\ =&\frac{1}{1-\gamma}\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_0(s,a)\rho^{\pi_0}(s,a) - \sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\rho^{\pi_1}(s,a)\right| \\ =&\frac{1}{1-\gamma}\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_0(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_1}(s,a)\right| \\ =&\frac{1}{1-\gamma}\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_0(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_0}(s,a) + r_1(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_1}(s,a)\right| \\ \leq&\frac{1}{1-\gamma}\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_0(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_0}(s,a)\right| + \frac{1}{1-\gamma}\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_1}(s,a)\right|,\end{aligned}$$

where the first term is upper bounded by

$$\begin{aligned}\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_0(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_0}(s,a)\right| =& \left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} \big(r_0(s,a) - r_1(s,a)\big)\rho^{\pi_0}(s,a)\right| \\ \leq& \max_{s,a}\left|r_0(s,a) - r_1(s,a)\right| \sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} \rho^{\pi_0}(s,a) = \|r_0 - r_1\|_\infty,\end{aligned}$$

and the second term is upper bounded by

$$\begin{aligned}\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_1}(s,a)\right| =& \left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\big(\rho^{\pi_0}(s,a) - \rho^{\pi_1}(s,a)\big)\right| \\ \leq& \sup_{r_1\in\mathcal{F}}\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_1(s,a)\big(\rho^{\pi_0}(s,a) - \rho^{\pi_1}(s,a)\big)\right| \\ =& d_{\mathcal{F}}\big(\rho^{\pi_0}(s,a), \rho^{\pi_1}(s,a)\big).\end{aligned}$$

By symmetry, we also have

$$\left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r_0(s,a)\rho^{\pi_0}(s,a) - r_1(s,a)\rho^{\pi_1}(s,a)\right| \leq d_{\mathcal{F}}\big(\rho^{\pi_0}(s,a), \rho^{\pi_1}(s,a)\big).$$

Combining the above results then completes the proof. □

7.9.4 Algorithms to Mitigate Return Disparity via Occupancy Measures Alignment

Algorithm 2 Algorithm to mitigate return disparity via occupancy measures alignment under the framework of [294].

```
Initialize policies $\pi_{\phi_0}$, $\pi_{\phi_1}$, Q-functions $q_{\theta_1}$ and $q_{\theta_2}$, feature extractors $f_{\psi_0}$ and $f_{\psi_1}$, environment buffers $\boldsymbol{D}_0$ and $\boldsymbol{D}_1$
for each iteration do
    for each environment step do
        for $g \in \{0, 1\}$ do
            Sample an action $a_g$ using policy $\pi_{\phi_g}$; add the sample $(s_g, a_g, s'_g, r_g)$ to $\boldsymbol{D}_g$;
        end for
    end for
    for each model update step do
        for $g \in \{0, 1\}$ do
            Update policy $\pi_{\phi_g}$, Q-function $q_{\theta_g}$, and feature extractor $f_{\psi_g}$ following [294];
        end for
    end for
    for each occupancy measures alignment c do
        Update feature extractors $f_{\psi_0}$ and $f_{\psi_1}$ via occupancy measures alignment via IPM (Wasserstein-1 distance/MMD);
    end for
end for
```

Inspired by Theorem 7.9.1, we introduce an occupancy measures alignment procedure when learning group policies for the two MDPs. We use the Wolpertinger policy [294] based on Deep Deterministic Policy Gradient (DDPG) [295] as as our baseline backbone algorithm, which is an actor-critic approach designed for learning policy in large discrete action spaces. We give an outline of our algorithm in Algorithm 2 and readers can refer to [295] for more clarification of the Wolpertinger policy. The details of the occupancy measures alignment step are similar to those of state visitation distributional alignment step. The experimental results of the implementation of Algorithm 2 are shown in Figure 7.6.

The overall results in Figure 7.6 are similar to those in Figure 7.3. However, compared to the Algorithm 1, the training processes of Algorithm 2 are more unstable. Besides, it also incurs additional learning costs since we have to train one policy for each MDP instead of one policy for both MDPs.

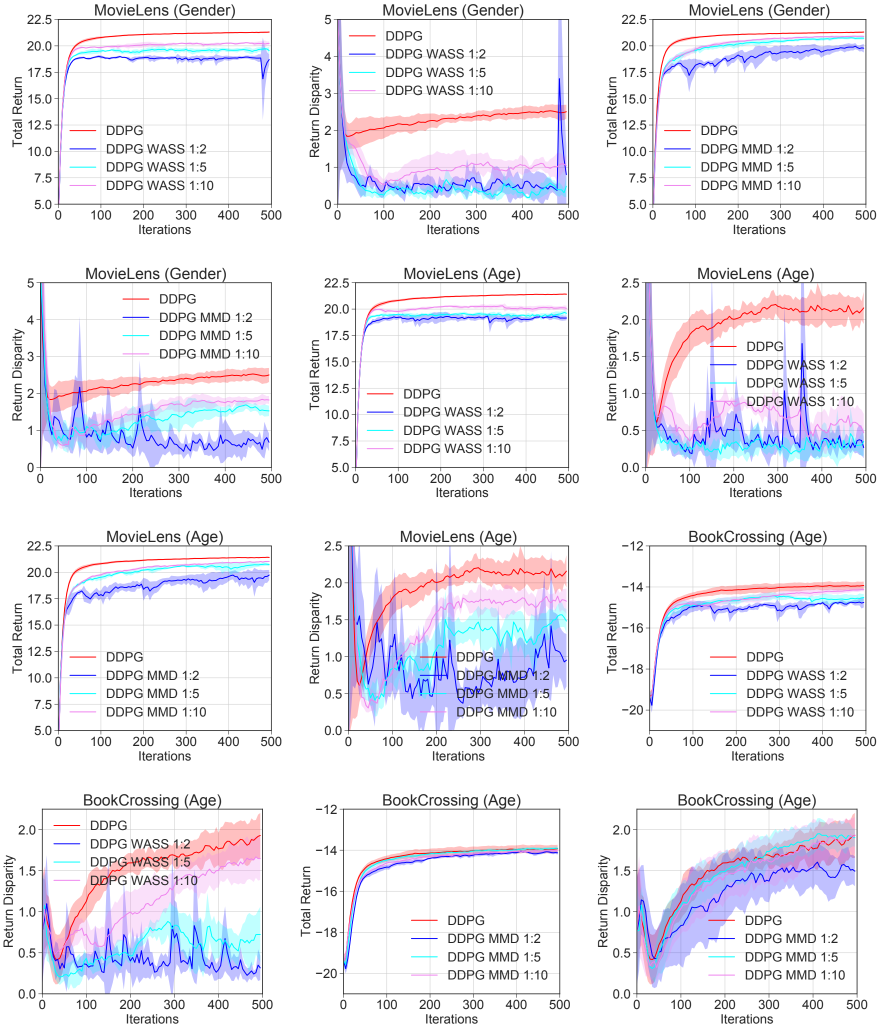


Figure 7.6: Learning curves of DDPG, DDPG-WASS and DDPG-MMD in three different settings.

Chapter 8

Trade-offs and Guarantees of Adversarial Representation Learning for Information Obfuscation

8.1 Introduction

With the growing demand for machine learning systems provided as services, a massive amount of data containing sensitive information, such as race, income level, age, etc., are generated and collected from local users. This poses a substantial challenge and it has become an imperative object of study in machine learning [81], computer vision [104, 296], healthcare [297, 298], speech recognition [299], and many other domains. In this chapter, we consider a practical scenario where the prediction vendor requests crowdsourced data for a target task, e.g, scientific modeling. The data owner agrees on the data usage for the target task while she does not want her other sensitive information (e.g., age, race) to be leaked. The goal in this context is then to obfuscate sensitive attributes of the sanitized data released by data owner from potential attribute inference attacks from a malicious adversary. For example, in an online advertising scenario, while the user (data owner) may agree to share her historical purchasing events, she also wants to protect her age information so that no malicious adversary can infer her age range from the shared data. Note that simply removing age attribute from the shared data is insufficient for this purpose, due to the redundant encoding in data, i.e., other attributes may have a high correlation with age.

Under this scenario, a line of work [99–107] aims to address the problem in the framework of (constrained) minimax problem. However, the theory behind these methods is little known. Such a gap between theory and practice calls for an important and appealing challenge:

> *Can we prevent the information leakage of the sensitive attribute while still maximizing the task accuracy? Furthermore, what is the fundamental trade-off between attribute obfuscation and accuracy maximization in the minimax problem?*

Under the setting of attribute obfuscation, the notion of information confidentiality should be attribute-specific: the goal is to protect specific attributes from being inferred by malicious adversaries as much as possible. Note that this is in sharp contrast with differential privacy (we systematically compare the related notions in Sec. 8.5 Related Work), where mechanisms are usually designed to resist worst-case membership query among all the data owners instead of preventing information leakage of the sensitive attribute [300]. From this perspective, our relaxed definition of attribute obfuscation against adversaries also allows for a more flexible design of algorithms with better accuracy.

Our Contributions In this chapter, we first formally define the notion of attribute inference attack in our setting and justify why our definitions are particularly suited under our setting. Through the lens of representation learning, we formulate the problem of accuracy maximization with information obfuscation constraint as a minimax optimization problem. To provide a formal guarantee on attribute obfuscation, we prove an information-theoretic lower bound on the inference error of the protected attribute under attacks from arbitrary adversaries. To investigate the relationship between attribute obfuscation and accuracy maximization, we also prove a theorem that formally characterizes the inherent trade-off between these two concepts. We conduct experiments to corroborate our formal guarantees and validate the inherent trade-offs in different attribute obfuscation algorithms. From our empirical results, we conclude that the adversarial representation learning approach achieves the best trade-off in terms of attribute obfuscation and accuracy maximization, among various state-of-the-art attribute obfuscation algorithms.

8.2 Preliminaries

Problem Setup We focus on the setting where the goal of the adversary is to perform *attribute inference*. This setting is ubiquitous in sever-client paradigm where machine learning is provided as a service (MLaaS, [301]). Formally, there are two parties in the system, namely the prediction vendor and the data owner. We consider the practical scenarios where users agree to contribute their data for specific purposes (e.g., training a machine learning model) but do not want others to infer their sensitive attributes in the data, such as health information, race, gender, etc. The prediction vendor will not collect raw user data but processed user data and the target attribute for the target task. In our setting, we assume the adversary cannot get other auxiliary information than the processed user data. In this case, the adversary can be anyone who can get access to the processed user data to some extent and wants to infer other private information. For example, malicious machine learning service providers are motivated to infer more information from users to do user profiling and targeted advertisements. The goal of the data owner is to provide as much information as possible to the prediction vendor to maximize the vendor's own accuracy, but under the constraint that the data owner should also protect the private information of the data source, i.e., *attribute obfuscation*. For ease of discussion, in our following analysis, we assume the the prediction vendor performs binary classification on the processed data. Extensions to multi-class classification is straightforward.

Notation We use $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{A}$ to denote the input, output and adversary's output space, respectively. Accordingly, we use X, Y, A to denote the random variables which take values in $\mathcal{X}, \mathcal{Y}$ and $\mathcal{A}$. We note that in our framework the input space $\mathcal{X}$ may or may not contain the sensitive attribute A. For two random variables X and Y, $I(X;Y)$ denotes the mutual information between X and Y. We use $H(X)$ to mean the Shannon entropy of random variable X. Similarly, we use $H(X \mid Y)$ to denote the conditional entropy of X given Y. We assume there is a joint distribution $\mathcal{D}$ over $\mathcal{X}\times\mathcal{Y}\times\mathcal{A}$ from which the data are sampled. To make our notation consistent, we use $\mathcal{D}_\mathcal{X}$, $\mathcal{D}_\mathcal{Y}$ and $\mathcal{D}_\mathcal{A}$ to denote the marginal distribution of $\mathcal{D}$ over $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{A}$. Given a feature map function $f : \mathcal{X} \to \mathcal{Z}$ that maps instances from the input space $\mathcal{X}$ to feature space $\mathcal{Z}$, we define $\mathcal{D}^f := \mathcal{D} \circ f^{-1}$ to be the induced (pushforward) distribution of $\mathcal{D}$ under f, i.e., for any event $E' \subseteq \mathcal{Z}$, $\Pr_{\mathcal{D}^f}(E') := \Pr_\mathcal{D}(\{x \in \mathcal{X} \mid f(x) \in E'\})$.

To simplify the exposition, we mainly discuss the setting where $\mathcal{X} \subseteq \mathbb{R}^d, \mathcal{Y} = \mathcal{A} = \{0, 1\}$, but the underlying theory and methodology could easily be extended to the categorical case as well. In what follows, we first formally define both the *accuracy* of the prediction vendor for the individualized service and the *attribute inference advantage* of an adversary. It is worth pointing out that our definition of inference advantage is *attribute-specific*. In particular, we seek to keep the data useful while being robust to an adversary on protecting specific attribute information from attack.

A *hypothesis* is a function $h : \mathcal{X} \to \mathcal{Y}$. The *error* of a hypothesis h under the distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$ is defined as: $\mathrm{Err}(h) := \mathbb{E}_{\mathcal{D}}\big[|Y - h(X)|\big]$. Similarly, we use $\widehat{\mathrm{Err}}(h)$ to denote the empirical error of h on a sample from $\mathcal{D}$. For binary classification problem, when $h(\mathbf{x}) \in \{0, 1\}$, the above loss also reduces to the error rate of classification. Let $\mathcal{H}$ be the space of hypotheses. In the context of binary classification, we define the accuracy of a hypothesis $h \in \mathcal{H}$ as:

Definition 8.2.1 (Accuracy)**.** The accuracy of $h \in \mathcal{H}$ is $\textsc{Acc}(h) := 1 - \mathbb{E}_{\mathcal{D}}\big[|Y - h(X)|\big]$.

For binary classification, we always have $0 \leq \textsc{Acc}(h) \leq 1$, $\forall h \in \mathcal{H}$. Similarly, an *adversarial hypothesis* is a function of $h_A : \mathcal{X} \to \mathcal{A}$. Next we define a measure of how much advantage of attribute inference gained from a particular attack space in our framework:

Definition 8.2.2 (Attribute Inference Advantage)**.** The inference advantage w.r.t. attribute A under attacks from $\mathcal{H}_A$ is defined as $\textsc{Adv}(\mathcal{H}_A) := \max_{h_A \in \mathcal{H}_A} \big|\Pr_{\mathcal{D}}(h_A(X) = 1 \mid A = 1) - \Pr_{\mathcal{D}}(h_A(X) = 1 \mid A = 0)\big|$.

Again, it is straightforward to verify that $0 \leq \textsc{Adv}(\mathcal{H}_A) \leq 1$. Based on our definition, $\textsc{Adv}(\mathcal{H}_A)$ then measures maximal inference advantage that the adversary in $\mathcal{H}_A$ can gain. We can also refine the above definition to a particular hypothesis $h_A : \mathcal{X} \to \{0, 1\}$ to measure its ability to steal information about A: $\textsc{Adv}(h_A) = \big|\Pr_{\mathcal{D}}(h_A(X) = 1 \mid A = 1) - \Pr_{\mathcal{D}}(h_A(X) = 1 \mid A = 0)\big|$.

Proposition 8.2.1. Let $h_A : \mathcal{X} \to \{0, 1\}$ be a hypothesis, then $\textsc{Adv}(h_A) = 0$ iff $I(h_A(X); A) = 0$ and $\textsc{Adv}(h_A) = 1$ iff $h_A(X) = A$ almost surely or $h_A(X) = 1 - A$ almost surely.

Proposition 8.2.1 justifies Definition 8.2.2 on how well an adversary h_A can infer about A from X: when $\textsc{Adv}(h_A) = 0$, it means that $h_A(X)$ contains no information about the sensitive attribute A. On the other hand, if $\textsc{Adv}(h_A) = 1$, then $h_A(X)$ fully predicts A (or equivalently, $1 - A$) from input X. In the latter case $h_A(X)$ also contains perfect information of A in the sense that $I(h_A(X); A) = H(A)$, i.e., the Shannon entropy of A. It is worth pointing out that Definition 8.2.2 is insensitive to the marginal distribution of A, and hence is more robust than other definitions such as the error rate of predicting A. In that case, if A is extremely imbalanced, even a naive predictor can attain small prediction error by simply outputting constant. We call a hypothesis space $\mathcal{H}_A$

symmetric if $\forall h_A \in \mathcal{H}_A$, $1 - h_A \in \mathcal{H}_A$ as well. When $\mathcal{H}_A$ is symmetric, we can also relate $\textsc{Adv}(\mathcal{H}_A)$ to a binary classification problem:

Proposition 8.2.2. If $\mathcal{H}_A$ is symmetric, then $\textsc{Adv}(\mathcal{H}_A) + \min_{h_A \in \mathcal{H}_A} \Pr(h_A(X) = 0 \mid A = 1) + \Pr(h_A(X) = 1 \mid A = 0) = 1$.

Consider the confusion matrix between the actual sensitive attribute A and its predicted variable $h_A(X)$. The false positive rate (eqv. Type-I error) is defined as FPR = FP / (FP + TN) and the false negative rate (eqv. Type-II error) is similarly defined as FNR = FN / (FN + TP). Using the terminology of confusion matrix, it is then clear that $\Pr(h_A(X) = 0 \mid A = 1) = \text{FNR}$ and $\Pr(h_A(X) = 1 \mid A = 0) = \text{FPR}$. In other words, Proposition 8.2.2 says that if $\mathcal{H}_A$ is symmetric, then the larger the attribute inference advantage of $\mathcal{H}_A$, the smaller the minimum sum of Type-I and Type-II error under attacks from $\mathcal{H}_A$.

8.3 Main Results

Given a set of samples $\mathbf{S} = \{(\mathbf{x}_i, y_i, a_i)\}_{i=1}^n$ drawn i.i.d. from the joint distribution $\mathcal{D}$, how can the data owner keep the data useful while keeping the sensitive attribute A obfuscated under potential attacks from malicious adversaries? Through the lens of representation learning, we seek to find a (non-linear) feature representation $f : \mathcal{X} \to \mathcal{Z}$ from input space $\mathcal{X}$ to feature space $\mathcal{Z}$ such that f still preserves relevant information w.r.t. the target task of inferring Y while hiding sensitive attribute A. Specifically, we can solve the following unconstrained regularized problem with $\lambda > 0$:

$$\min_{h \in \mathcal{H}, f} \max_{h_A \in \mathcal{H}_A} \quad \widehat{\text{Err}}(h \circ f) - \lambda\big(\Pr_{\mathbf{S}}(h_A(f(X)) = 0 \mid A = 1) + \Pr_{\mathbf{S}}(h_A(f(X)) = 1 \mid A = 0)\big) \tag{8.1}$$

It is worth pointing out that the optimization formulation in (8.1) admits an interesting game-theoretic interpretation, where two agents f and h_A play a game whose score is defined by the objective function in (8.1). Intuitively, h_A seeks to minimize the sum of Type-I and Type-II error while f plays against h_A by learning transformation to removing information about the sensitive attribute A. Algorithmically, for the data owner to achieve the goal of hiding information about the sensitive attribute A from malicious adversary, it suffices to learn a representation that is independent of A:

Proposition 8.3.1. Let $f : \mathcal{X} \to \mathcal{Z}$ be a deterministic function and $\mathcal{H}_A \subseteq 2^{\mathcal{Z}}$ be a hypothesis class over $\mathcal{Z}$. For any joint distribution $\mathcal{D}$ over X, A, Y, if $I(f(X); A) = 0$, then $\textsc{Adv}(\mathcal{H}_A \circ f) = 0$.

Note that in this sequential game, f is the first-mover and h_A is the second. Hence without explicit constraint f possesses a first-mover advantage so that f can dominate the game by simply mapping all the input X to a constant or uniformly random noise[1]. To avoid these degenerate cases, the first term in the objective function of (8.1) acts as an incentive to encourage f to preserve task-related information. But will this incentive compromise the information of A? As an extreme case if the target variable Y and the sensitive attribute A are perfectly correlated, then it should be clear that there is a trade-off in achieving accuracy and preventing information leakage of the attribute. In Sec. 8.3.2 we shall provide an information-theoretic bound to precisely characterize such trade-off.

8.3.1 Formal Guarantees against Attribute Inference

In the unconstrained minimax formulation (8.1), the hyperparameter λ measures the trade-off between accuracy and information obfuscation. On one hand, if $\lambda \to 0$, we barely care about the information obfuscation of A and devote all the focus to maximize our accuracy. On the other extreme, if $\lambda \to \infty$, we are only interested in obfuscating the sensitive information. In what follows we analyze the true error that an optimal adversary has to incur in the limit when both the task classifier and the adversary have unlimited capacity, i.e., they can be any randomized functions from $\mathcal{Z}$ to $\{0, 1\}$. To study the true error, we hence use the population loss rather than the empirical loss in our objective function. Furthermore, since the binary classification error in (8.1) is NP-hard to optimize even for hypothesis class of linear predictors, in practice we consider the cross-entropy loss function as a convex surrogate loss. With a slight abuse of notation, the cross-entropy loss $\mathrm{CE}_Y(h)$ of a probabilistic hypothesis $h : \mathcal{X} \to [0, 1]$ w.r.t. Y on a distribution $\mathcal{D}$ is defined as follows:

$$\mathrm{CE}_Y(h) := -\mathbb{E}_{\mathcal{D}}[\mathbb{I}(Y = 0) \log(1 - h(X)) + \mathbb{I}(Y = 1) \log(h(X))].$$

We also use $\mathrm{CE}_A(h_A)$ to mean the cross-entropy loss of the adversary h_A w.r.t. A. Using the same notation, the optimization formulation with cross-entropy loss becomes:

$$\min_{h \in \mathcal{H}, f} \max_{h_A \in \mathcal{H}_A} \quad \mathrm{CE}_Y(h \circ f) - \lambda \cdot \mathrm{CE}_A(h_A \circ f) \tag{8.2}$$

Given a feature map $f : \mathcal{X} \to \mathcal{Z}$, assume that $\mathcal{H}$ contains all the possible probabilistic classifiers from the feature space $\mathcal{Z}$ to $[0, 1]$. For example, a probabilistic classifier can be constructed by first defining a function $h : \mathcal{Z} \to [0, 1]$ followed by a random coin flipping to determine the output label,

[1] The extension of Proposition 8.3.1 to randomized function is staightforward as long as the randomness is independent of the sensitive attribute A.

where the probability of the coin being 1 is given by $h(Z)$. Under such assumptions, the following lemma shows that the optimal target classifier under f is given by the conditional distribution $h^*(Z) := \Pr(Y = 1 \mid Z)$.

Lemma 8.3.1. For any feature map $f : \mathcal{X} \to \mathcal{Z}$, assume that $\mathcal{H}$ contains all the probabilistic classifiers, then $\min_{h\in\mathcal{H}} \mathrm{CE}_Y(h \circ f) = H(Y \mid Z)$ and $h^*(Z) := \arg\min_{h\in\mathcal{H}} \mathrm{CE}_Y(h\circ f) = \Pr(Y = 1 \mid Z = f(X))$.

By a symmetric argument, we can also see that the worst-case (optimal) adversary under f is the conditional distribution $h_A^*(Z) := \Pr(A = 1 \mid Z)$ and $\min_{h_A\in\mathcal{H}_A} \mathrm{CE}_A(h_A \circ f) = H(A \mid Z)$. Hence we can further simplify the optimization formulation (8.2) to the following form where the only optimization variable is the feature map f:

$$\min_f \quad H(Y \mid Z = f(X)) - \lambda H(A \mid Z = f(X)) \tag{8.3}$$

Since $Z = f(X)$ is a deterministic feature map, it follows from the basic properties of Shannon entropy that

$$H(Y \mid X) \leq H(Y \mid Z = f(X)) \leq H(Y), \qquad H(A \mid X) \leq H(A \mid Z = f(X)) \leq H(A)$$

which means that $H(Y \mid X) - \lambda H(A)$ is a lower bound of the optimum of the objective function in (8.3). However, such lower bound is not necessarily achievable. To see this, consider the simple case where $Y = A$ almost surely. In this case there exists no deterministic feature map $Z = f(X)$ that is both a sufficient statistics of X w.r.t. Y while simultaneously filters out all the information w.r.t. A except in the degenerate case where $A(Y)$ is constant. Next, to show that solving the optimization problem in (8.3) helps to remove sensitive information, the following theorem gives a bound of attribute inference in terms of the error that has to be incurred by the optimal adversary:

Theorem 8.3.1. Let f^* be the optimal feature map of (8.3) and define $H^* := H(A \mid Z = f^*(X))$. Then for any adversary $\widehat{A}$ such that $I(\widehat{A}; A \mid Z) = 0$, $\Pr_{\mathcal{D}^{f^*}}(\widehat{A} \neq A) \geq H^*/2\lg(6/H^*)$.

Remark Theorem 8.3.1 shows that whenever the conditional entropy $H^* = H(A \mid Z = f^*(X))$ is large, then the inference error of the protected attribute incurred by any (randomized) adversary has to be at least $\Omega(H^*/\log(1/H^*))$. The assumption $I(\widehat{A}; A \mid Z) = 0$ says that, given the processed feature Z, the adversary $\widehat{A}$ could not use any external information that depends on the true sensitive attribute A. As we have already shown above, the conditional entropy essentially corresponds to the second term in our objective function, whose optimal value could further be flexibly adjusted by tuning the trade-off parameter λ. As a final note, Theorem 8.3.1 also shows that representation learning helps to remove the information about A since we always have $H(A \mid Z = f(X)) \geq H(A \mid X)$ for any deterministic feature map f so that the lower bound of inference error by any adversary is larger after learning the representation $Z = f(X)$.

8.3.2 Inherent trade-off between Accuracy and Attribute Obfuscation

In this section we shall provide an information-theoretic bound to quantitatively characterize the inherent trade-off between these accuracy maximization and attribute obfuscation, due to the discrepancy between the conditional distributions of the target variable given the sensitive attribute. Our result is algorithm-independent, hence it applies to a general setting where there is a need to preserve both terms. To the best of our knowledge, this is the first information-theoretic result to precisely quantify such trade-off. Due to space limit, we defer all the proofs to the appendix.

Before we proceed, we first define several information-theoretic concepts that will be used in our analysis. For two distributions $\mathcal{D}$ and $\mathcal{D}'$, the Jensen-Shannon (JS) divergence $D_{\mathrm{JS}}(\mathcal{D}, \mathcal{D}')$ is: $D_{\mathrm{JS}}(\mathcal{D}, \mathcal{D}') := \frac{1}{2} D_{\mathrm{KL}}(\mathcal{D} \parallel \mathcal{D}_M) + \frac{1}{2} D_{\mathrm{KL}}(\mathcal{D}' \parallel \mathcal{D}_M)$, where $D_{\mathrm{KL}}(\cdot \parallel \cdot)$ is the Kullback–Leibler (KL) divergence and $\mathcal{D}_M := (\mathcal{D} + \mathcal{D}')/2$. The JS divergence can be viewed as a symmetrized and smoothed version of the KL divergence, However, unlike the KL divergence, the JS divergence is bounded: $0 \leq D_{\mathrm{JS}}(\mathcal{D}, \mathcal{D}') \leq 1$. Additionally, from the JS divergence, we can define a distance metric between two distributions as well, known as the JS distance [302]: $d_{\mathrm{JS}}(\mathcal{D}, \mathcal{D}') := \sqrt{D_{\mathrm{JS}}(\mathcal{D}, \mathcal{D}')}$. With respect to the JS distance, for any feature space $\mathcal{Z}$ and any deterministic mapping $f : \mathcal{X} \to \mathcal{Z}$, we can prove the following lemma via the celebrated data processing inequality:

Lemma 8.3.2. Let $\mathcal{D}_0$ and $\mathcal{D}_1$ be two distributions over $\mathcal{X}$ and let $\mathcal{D}_0^f$ and $\mathcal{D}_1^f$ be the induced distributions of $\mathcal{D}_0$ and $\mathcal{D}_1$ over $\mathcal{Z}$ by function f, then $d_{\mathrm{JS}}(\mathcal{D}_0^f, \mathcal{D}_1^f) \leq d_{\mathrm{JS}}(\mathcal{D}_0, \mathcal{D}_1)$.

Without loss of generality, any method aiming to predict the target variable Y defines a Markov chain as $X \xrightarrow{f} Z \xrightarrow{h} \hat{Y}$, where $\hat{Y}$ is the predicted target variable given by hypothesis h and Z is the intermediate representation defined by the feature mapping f. Hence for any distribution $\mathcal{D}_0(\mathcal{D}_1)$ of X, this Markov chain also induces a distribution $\mathcal{D}_0^{h\circ f}(\mathcal{D}_1^{h\circ f})$ of $\hat{Y}$ and a distribution $\mathcal{D}_0^f(\mathcal{D}_1^f)$ of Z. Now let $\mathcal{D}_0^Y(\mathcal{D}_1^Y)$ be the underlying true conditional distribution of Y given $A = 0(A = 1)$. Realize that the JS distance is a metric, the following chain of triangular inequalities holds:

$$d_{\mathrm{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y) \leq d_{\mathrm{JS}}(\mathcal{D}_0^Y, \mathcal{D}_0^{h\circ f}) + d_{\mathrm{JS}}(\mathcal{D}_0^{h\circ f}, \mathcal{D}_1^{h\circ f}) + d_{\mathrm{JS}}(\mathcal{D}_1^{h\circ f}, \mathcal{D}_1^Y).$$

Combining the above inequality with Lemma 8.3.2 to show $d_{\mathrm{JS}}(\mathcal{D}_0^{h\circ f}, \mathcal{D}_1^{h\circ f}) \leq d_{\mathrm{JS}}(\mathcal{D}_0^f, \mathcal{D}_1^f)$, we immediately have:

$$d_{\mathrm{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y) \leq \; d_{\mathrm{JS}}(\mathcal{D}_0^Y, \mathcal{D}_0^{h\circ f}) + d_{\mathrm{JS}}(\mathcal{D}_0^f, \mathcal{D}_1^f) + d_{\mathrm{JS}}(\mathcal{D}_1^{h\circ f}, \mathcal{D}_1^Y).$$

Intuitively, $d_{\mathrm{JS}}(\mathcal{D}_0^Y, \mathcal{D}_0^{h\circ f})$ and $d_{\mathrm{JS}}(\mathcal{D}_1^Y, \mathcal{D}_1^{h\circ f})$ measure the distance between the predicted and the true target distribution on $A = 0/1$ cases, respectively. Formally, let $\mathrm{Err}_a(h \circ f)$ be the prediction error of function $h \circ f$ conditioned on $A = a$. With the help of Lemma 8.7.2, the following result establishes a relationship between $d_{\mathrm{JS}}(\mathcal{D}_a^Y, \mathcal{D}_a^{h\circ f})$ and the accuracy of $h \circ f$:

Lemma 8.3.3. Let $\hat{Y} = h(f(X)) \in \{0, 1\}$ be the predictor, then for $a \in \{0, 1\}$, $d_{\mathrm{JS}}(\mathcal{D}_a^Y, \mathcal{D}_a^{h\circ f}) \leq \sqrt{\mathrm{Err}_a(h \circ f)}$.

Combine Lemma 8.3.2 and Lemma 8.3.3, we get the following key lemma that is the backbone for proving the main results in this section:

Lemma 8.3.4 (Key lemma)**.** Let $\mathcal{D}_0$, $\mathcal{D}_1$ be two distributions over $\mathcal{X} \times \mathcal{Y}$ conditioned on $A = 0$ and $A = 1$ respectively. Assume the Markov chain $X \xrightarrow{f} Z \xrightarrow{h} \hat{Y}$ holds, then $\forall h \in \mathcal{H}$:

$$d_{\mathrm{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y) \leq \sqrt{\mathrm{Err}_0(h \circ f)} + d_{\mathrm{JS}}(\mathcal{D}_0^f, \mathcal{D}_1^f) + \sqrt{\mathrm{Err}_1(h \circ f)}.$$

We emphasize that for $a \in \{0, 1\}$, the term $\mathrm{Err}_a(h \circ f)$ measures the conditional error of the predicted variable $\hat{Y}$ by the composite function $h \circ f$ over $\mathcal{D}_a$. Similarly, we can define the *conditional accuracy* for $a \in \{0, 1\}$: $\textsc{Acc}_a(h \circ f) := 1 - \mathrm{Err}_a(h \circ f)$. The following main theorem then characterizes a fundamental trade-off between accuracy and attribute obfuscation:

Theorem 8.3.2. Let $\mathcal{H}_A \subseteq 2^{\mathcal{Z}}$ be the hypothesis space of all the classifiers from $\mathcal{Z}$ to $\{0, 1\}$. Assume the conditions in Lemma 8.3.4 hold, then $\forall h \in \mathcal{H}$, $\textsc{Acc}_0(h \circ f) + \textsc{Acc}_1(h \circ f) \leq 2 - \frac{1}{3} D_{\mathrm{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y) + \textsc{Adv}(\mathcal{H}_A \circ f)$.

The upper bound given in Theorem 8.3.2 shows that when the marginal distribution of the target variable Y differ between two cases $A = 0$ or $A = 1$, then it is impossible to perfectly maximize accuracy and prevent the sensitive attribute being inferred. Furthermore, the trade-off due to the difference in marginal distributions is precisely given by the JS divergence $D_{\mathrm{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y)$. Next, if we would like to decrease the advantage of adversaries, $\textsc{Adv}(\mathcal{H}_A \circ f)$, through learning proper feature transformation f, then the upper bound on the sum of conditional accuracy also becomes smaller, for any predictor h. Note that in Theorem 8.3.2 the upper bound holds for *any* adversarial hypothesis h_A in the richest hypothesis class $\mathcal{H}_A$ that contains all the possible binary classifiers. Put it another way, if we would like to maximally obfuscate information w.r.t. sensitive attribute A, then we have to incur a large joint error:

Theorem 8.3.3. Assume the conditions in Theorem 8.3.2 hold. If $\textsc{Adv}(\mathcal{H}_A \circ f) \leq D_{\mathrm{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y)$, then $\forall h \in \mathcal{H}$, $\mathrm{Err}_0(h \circ f) + \mathrm{Err}_1(h \circ f) \geq \frac{1}{2}\big(d_{\mathrm{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y) - \sqrt{\textsc{Adv}(\mathcal{H}_A \circ f)}\big)^2$.

Remark The above lower bound characterizes a fundamental trade-off between information obfuscation of the sensitive attribute and joint error of target task. In particular, up to a certain level $D_{\mathrm{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y)$, the larger the inference advantage that the adversary can gain, the smaller the joint error. In light of Proposition 8.3.1, this means that although the data-owner, or the first-mover f, could try to maximally filter out the sensitive information via constructing f such that $f(X)$ is independent of A, such construction will also inevitably compromise the joint accuracy of the prediction vendor. It is also worth pointing out that our results in both Theorem 8.3.2 and Theorem 8.3.3 are attribute-independent in the sense that neither of the bounds depends on the marginal distribution of A. Instead, all the terms in our results only depend on the conditional distributions given $A = 0$ and $A = 1$. This is often more desirable than bounds involving mutual information, e.g., $I(A, Y)$, since $I(A, Y)$ is close to 0 if the marginal distribution of A is highly imbalanced.

8.4 Experiments

In this section, we conduct experiments to investigate the following questions:

Q1 Are our formal guarantees valid for different attribute obfuscation methods and the inherent trade-offs between attribute information obfuscation and accuracy maximization exist in all methods?

Q2 Which attribute obfuscation algorithms achieve the best trade-offs in terms of attribute obfuscation and accuracy maximization?

8.4.1 Datasets and Setup

In our experiments, we use: (1) Adult dataset [219]: The Adult dataset is a benchmark dataset for classification. The task is to predict whether an individual's income is greater or less than 50K/year based on census data. In our experiment we set the target task as income prediction and the malicious task done by the adversary as inferring gender, age and education, respectively. (2) UTKFace dataset [303]: The UTKFace dataset is a large-scale face benchmark dataset containing more than 20,000 images with annotations of age, gender, and ethnicity. In our experiment, we set our target task as gender classification and we use the age and ethnicity as the protected attributes. We refer readers to Sec. 8.7.3 in the appendix for detailed descriptions about the data pre-processing pipeline and the data distribution for each dataset.

We conduct experiments with the following methods to verify our theoretical results and provide a thorough practical comparison among these methods. 1). Privacy Partial Least Squares (PPLS) [99], 2). Privacy Linear Discriminant Analysis (PLDA) [100], 3). Minimax filter with alternative update (ALT-UP) [101], 4) Maximum Entropy Adversarial Representation Learning (MAX-ENT) [304] 5). Gradient Reversal Layer (GRL) [247] 6). Principal Component Analysis (PCA) 7). Normal Training (NORM-TRAIN), 8) Local Differential Privacy (LDP) with Laplacian mechanism, 9). Differentially Private SGD (DPSGD) [78]. Among the first seven methods, the first five are state-of-the-art minimax methods for protecting against attribute inference attacks while the latter two are normal representation learning baselines for comprehensive comparison. Although DP is not tailored to attribute obfuscation, we can still add two DP baselines to examine the accuracy and attribute obfuscation trade-off for comparison[2]. To ensure the comparison is fair among different methods, we conduct a controlled experiment by using the same network structure as the baseline hypothesis among all the methods for each dataset. For each experiment on the Adult dataset and UTKFace dataset, we repeat the experiments for ten times to report both the average performance and their standard deviations. Sec. 8.7.3 in the appendix provides detailed descriptions about the methods and the hyperparameter settings.

Note that in practice due to the non-convex nature of optimizing deep neural nets, we cannot guarantee to find the global optimal conditional entropy H^*. Hence in order to compute the formal guarantee given by our lower bound in Theorem 8.3.1, we use the cross-entropy loss of the optimal adversary found by our algorithm on inferring the sensitive attribute A. Furthermore, since our

[2] Some other methods [105, 305] in the literature are close variants of the above, so we do not include them here due to the space limit.

analysis only applies to representation learning based approaches, we do not have similar guarantee for DP-related methods in our context. We visualize the performances of the aforementioned algorithms on attribute obfuscation and accuracy maximization in Figure 8.1 and Figure 8.2, respectively.

8.4.2 Results and Analysis

Validation of Our Theory (Q1) From Figure 8.1, we can see that the formal guarantees are valid for all representation learning approaches. With the results in Figure 8.2, we also see that no methods are perfect in both achieving both attribute obfuscation and accuracy maximization: the methods with small accuracy loss comes with relative low inference errors and vice versa.

Comparison with Different Methods (Q2) Among all methods, LDP, PLDA, ALT-UP, MAX-ENT and GRL are effective in attribute obfuscation by forcing the optimal adversary to incur a large inference error in Figure 8.1. On the other hand, PCA and NORM-TRAIN are the least effective ones. This is expected as neither NORM-TRAIN nor PCA filters information in data about the sensitive attribute A.

From Figure 8.2, we can also see a sharp contrast between DP-based methods and other methods in terms of the joint conditional error on the target task: both LDP and DPSGD could incur significant accuracy loss compared with other methods. Combining this one with our previous observation from Figure 8.1, we can see that DP-based methods either make data private by adding large amount of noise to filter out both target-related information and sensitive-related information available in the data, or add insufficient amount of noise so that both target-related and sensitive-related information is well preserved. As a comparison, representation learning based approaches leads to a better trade-off.

Among the representation learning methods, PLDA, ALT-UP, MAX-ENT and GRL perform the best in attribute obfuscation. Compared to PLDA and GRL, ALT-UP and MAX-ENT incur significant drops in accuracy when λ is large. It is also worth to note that different adversarial representation learning methods have different sensitivity on λ: a large λ for MAX-ENT might lead to an unstable model training process and result in a large accuracy loss. In contrast, GRL is often more stable, which is consistent to the results shown in [248].

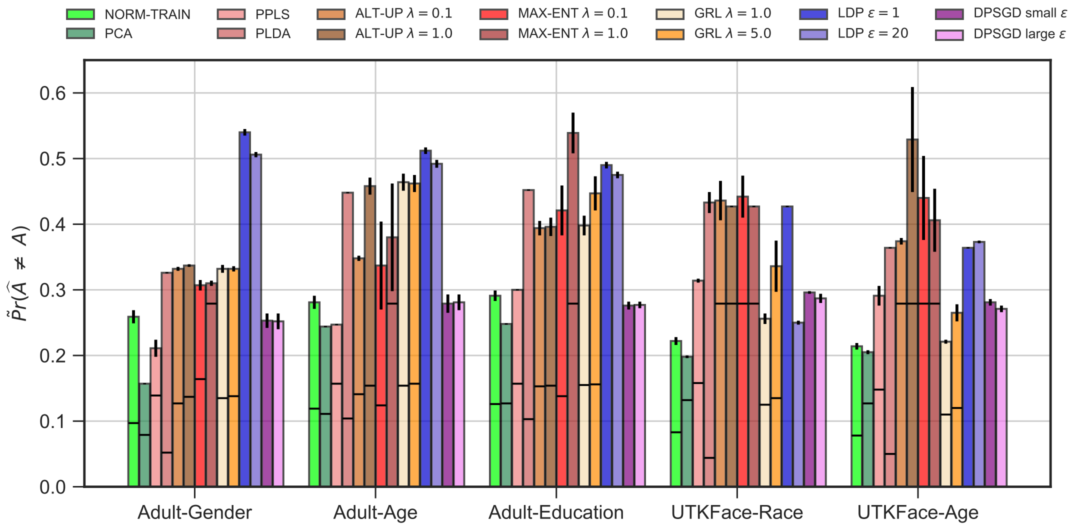


Figure 8.1: Performance on attribute obfuscation of different methods (the larger the better). The horizontal lines across the bars indicate the corresponding formal guarantees given by our lower bound in Theorem 8.3.1.

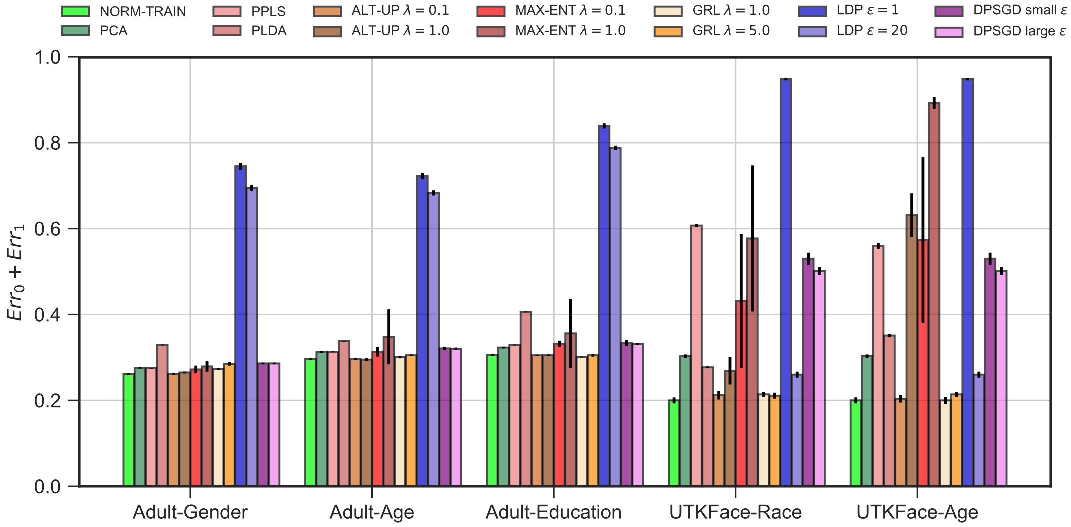


Figure 8.2: The joint conditional error ($\mathrm{Err}_0 + \mathrm{Err}_1$, the smaller the better) of different methods.

8.5 Related Work

Attribute Obfuscation Various minimax formulations and algorithms have been proposed to defend against inference attack in different scenarios [99–102, 104–107, 306]. [106] proposed the optimization problem where the terms in the objective function are defined in terms of mutual information. Under their formulation, they analyze a trade-off between between utility loss and attribute obfuscation: under the constraint of the attribute obfuscation $I(A; Z) \leq k$, what the maximum utility loss $I(Y; X \mid Z)$ is. Compared with these works, we study the inherent trade-off between the accuracy and attribute obfuscation and provide formal guarantees to quantify worst-case inference error given the transformation.

Differential Privacy Differential privacy (DP) has been proposed and extensively investigated to protect the individual privacy of collected data [77] and DP mechanisms were used in the training of deep neural network recently [78,79]. DP ensures the output distribution of a randomized mechanism to be statistically indistinguishable between any two neighboring datasets, and provides formal guarantees for privacy problems such as defending against the membership query attacks [12,307]. From this perspective, DP is closely related to the well-known membership inference attack [12] instead. As a comparison, our goal of attribute obfuscation is to learn a representation such that the sensitive attributes cannot be accurately inferred. Although the two goals differ, [70] show the there are deep connections between membership inference and attribute inference. An interesting direction to explore is to draw more formal connections to these two notions. Last but not least, It is also worth to mention that the notion of individual fairness may be viewed as a generalization of DP [51].

Algorithmic Fairness Recent work has shown that unfair models could lead to the leakage of users' sensitive information [211]. In particular, adversarial learning methods have been used as a tool in both fields to achieve the corresponding goals [101,243]. However, the motivations and goals significantly differ between these two fields. Specifically, the widely adopted notion of group fairness, namely equalized odds [52], requires equalized false positive and false negative rates across different demographic subgroups. As a comparison, in applications where information leakage is a concern, we mainly want to ensure that adversaries cannot steal sensitive information from the data. Hence our goal is to give a worst case guarantee on the inference error that any adversary has at least to incur. To the best of our knowledge, our results in Theorem 8.3.1 is the first one to analyze the performance of attribute obfuscation in such scenarios. Furthermore, no prior theoretical results exist on discussing the trade-off between attribute obfuscation and accuracy under the setting of representation learning. Our proof techniques developed in this chapter could also be used to derive information-theoretic lower bounds in related problems as well [213,308]. On a final note, the relationships of the above notions are visualized in Figure 8.3.

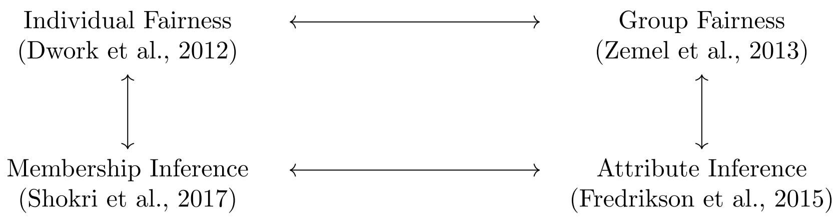


Figure 8.3: Relationships between different notions of fairness and inference attack.

8.6 Conclusion

We develop a theoretical framework for analyzing attribute obfuscation through adversarial representation learning. Specifically, the framework suggests using adversarial learning techniques to obfuscate the sensitive attribute and we also analyze the formal guarantees of such techniques in the limit of worst-case adversaries. We also prove an information-theoretic lower bound to quantify the inherent trade-off between accuracy and obfuscation of attribute information. Following our formulation, we conduct experiments to corroborate our theoretical results and to empirically compare different state-of-the-art attribute obfuscation algorithms. Experimental results show that the adversarial representation learning approaches are effective against attribute inference attacks and often achieve the best trade-off in terms of attribute obfuscation and accuracy maximization. We believe our work takes an important step towards better understanding the trade-off between accuracy maximization and attribute obfuscation, and it also helps inspire the future design of attribute obfuscation algorithms with adversarial learning techniques.

8.7 Appendix of Chapter 8

In this appendix we provide the missing proofs of theorems and claims in our main paper. We also describe detailed experimental settings here.

8.7.1 Technical Tools

In this section we list the lemmas and theorems used during our proof.

Lemma 8.7.1 (Theorem 2.2, [309])**.** Let $H_2^{-1}(s)$ be the inverse binary entropy function for $s \in [0, 1]$, then $H_2^{-1}(s) \geq s/2\lg(6/s)$.

Lemma 8.7.2 ([310])**.** Let $\mathcal{D}$ and $\mathcal{D}'$ be two distributions, then $D_{\mathrm{JS}}(\mathcal{D}, \mathcal{D}') \leq \frac{1}{2}\|\mathcal{D} - \mathcal{D}'\|_1$.

Theorem 8.7.1 (Data processing inequality)**.** Let $X \perp Y \mid Z$, then $I(X; Z) \geq I(X; Y)$.

8.7.2 Missing Proofs

Proposition 8.2.1. Let $h_A : \mathcal{X} \to \{0, 1\}$ be a hypothesis, then $\textsc{Adv}(h_A) = 0$ iff $I(h_A(X); A) = 0$ and $\textsc{Adv}(h_A) = 1$ iff $h_A(X) = A$ almost surely or $h_A(X) = 1 - A$ almost surely.

Proof. We first prove the first part of the proposition. By definition, $\textsc{Adv}(h_A) = 0$ iff $\Pr_{\mathcal{D}}(h_A(X) = 1 \mid A = 1) = \Pr_{\mathcal{D}}(h_A(X) = 1 \mid A = 0)$, which is also equivalent to $h_A(X) \perp A$. It then follows that $h_A(X) \perp A \Leftrightarrow I(h_A(X); A) = 0$.

For the second part of the proposition, again, by definition of $\textsc{Adv}(h_A)$, it is clear to see that we either have $\Pr_{\mathcal{D}}(h_A(X) = 1 \mid A = 1) = 1$ and $\Pr_{\mathcal{D}}(h_A(X) = 1 \mid A = 0) = 0$, or $\Pr_{\mathcal{D}}(h_A(X) = 1 \mid A = 1) = 0$ and $\Pr_{\mathcal{D}}(h_A(X) = 1 \mid A = 0) = 1$. Hence we discuss by these two cases. For ease of notation, we omit the subscript $\mathcal{D}$ from $\Pr_{\mathcal{D}}$ when it is obvious from the context which probability distribution we are referring to.

1. If $\Pr(h(X) = 1 \mid A = 1) = 1$ and $\Pr(h(X) = 1 \mid A = 0) = 0$, then we know that:

$$\begin{aligned}\Pr(h_A(X) \neq A) &= \Pr(A = 0) \Pr(h_A(X) \neq A \mid A = 0) + \Pr(A = 1) \Pr(h_A(X) \neq A \mid A = 1)\\ &= \Pr(A = 0) \Pr(h_A(X) = 1 \mid A = 0) + \Pr(A = 1) \Pr(h_A(X) = 0 \mid A = 1)\\ &= \Pr(A = 0) \cdot 0 + \Pr(A = 1) \cdot 0\\ &= 0.\end{aligned}$$

2. If $\Pr(h_A(X) = 1 \mid A = 1) = 0$ and $\Pr(h_A(X) = 1 \mid A = 0) = 1$, similarly, we have:

$$\begin{aligned}\Pr(h_A(X) \neq 1 - A) &= \Pr(A = 0) \Pr(h_A(X) \neq 1 - A \mid A = 0) + \Pr(A = 1) \Pr(h_A(X) \neq 1 - A \mid A = 1)\\ &= \Pr(A = 0) \Pr(h_A(X) = 0 \mid A = 0) + \Pr(A = 1) \Pr(h_A(X) = 1 \mid A = 1)\\ &= \Pr(A = 0) \cdot 0 + \Pr(A = 1) \cdot 0\\ &= 0.\end{aligned}$$

Combining the above two parts completes the proof. □

Proposition 8.2.2. If $\mathcal{H}_A$ is symmetric, then $\textsc{Adv}(\mathcal{H}_A) + \min_{h_A \in \mathcal{H}_A} \Pr(h_A(X) = 0 \mid A = 1) + \Pr(h_A(X) = 1 \mid A = 0) = 1$.

Proof. By definition, we have:

$$\begin{aligned} 1 - \textsc{Adv}(\mathcal{H}_A) &:= 1 - \max_{h_A \in \mathcal{H}_A} \ \textsc{Adv}(h_A) \\ &= \min_{h_A \in \mathcal{H}_A} \ 1 - \big| \Pr(h_A(X) = 1 \mid A = 1) - \Pr(h_A(X) = 1 \mid A = 0) \big| \\ &= \min_{h_A \in \mathcal{H}_A} \ 1 - \big(\Pr(h_A(X) = 1 \mid A = 1) - \Pr(h_A(X) = 1 \mid A = 0) \big) \\ &= \min_{h \in \mathcal{H}} \ \Pr(h_A(X) = 0 \mid A = 1) + \Pr(h_A(X) = 1 \mid A = 0), \end{aligned}$$

where the third equality holds due to the fact that $\max_{h_A \in \mathcal{H}_A} \big| \Pr(h_A(X) = 1 \mid A = 1) - \Pr(h_A(X) = 1 \mid A = 0) \big| = \max_{h_A \in \mathcal{H}_A} \big(\Pr(h_A(X) = 1 \mid A = 1) - \Pr(h_A(X) = 1 \mid A = 0) \big)$. To see this, for any specific h_A such that the term inside the absolute value is negative, we can find $1 - h_A \in \mathcal{H}_A$ such that it becomes positive, due to the assumption that $\mathcal{H}_A$ is symmetric. □

Proposition 8.3.1. Let $f : \mathcal{X} \to \mathcal{Z}$ be a deterministic function and $\mathcal{H}_A \subseteq 2^{\mathcal{Z}}$ be a hypothesis class over $\mathcal{Z}$. For any joint distribution $\mathcal{D}$ over X, A, Y, if $I(f(X); A) = 0$, then $\textsc{Adv}(\mathcal{H}_A \circ f) = 0$.

Proof. First, by the celebrated data-processing inequality, $\forall h_A \in \mathcal{H}_A$:

$$0 \leq I(h_A(f(X)); A) \leq I(f(X); A) = 0.$$

By Proposition 8.2.1, this means that $\forall h_A \in \mathcal{H}_A$, $\textsc{Adv}(h_A) = 0$, which further implies that $\textsc{Adv}(\mathcal{H}_A \circ f) = 0$ by definition. □

Lemma 8.3.1. For any feature map $f : \mathcal{X} \to \mathcal{Z}$, assume that $\mathcal{H}$ contains all the probabilistic classifiers, then $\min_{h \in \mathcal{H}} \mathrm{CE}_Y(h \circ f) = H(Y \mid Z)$ and $h^*(Z) := \arg\min_{h \in \mathcal{H}} \mathrm{CE}_Y(h \circ f) = \Pr(Y = 1 \mid Z = f(X))$.

Proof. Let $\mathcal{D}^f$ be the induced (pushforward) distribution of $\mathcal{D}$ under the map $f : \mathcal{X} \to \mathcal{Z}$. By the definition of cross-entropy loss, we have:

$$\begin{aligned}
\mathrm{CE}_Y(h \circ f) &= -\mathbb{E}_{\mathcal{D}}\left[\mathbb{I}(Y=0)\log(1-h(f(X))) + \mathbb{I}(Y=1)\log(h(f(X)))\right] \\
&= -\mathbb{E}_{\mathcal{D}^f}\left[\mathbb{I}(Y=0)\log(1-h(Z)) + \mathbb{I}(Y=1)\log(h(Z))\right] \\
&= -\mathbb{E}_Z\mathbb{E}_{Y|Z}\left[\mathbb{I}(Y=0)\log(1-h(Z)) + \mathbb{I}(Y=1)\log(h(Z))\right] \\
&= -\mathbb{E}_Z\left[\Pr(Y=0 \mid Z)\log(1-h(Z)) + \Pr(Y=1 \mid Z)\log(h(Z))\right] \\
&= \mathbb{E}_Z\left[D_{\mathrm{KL}}(\Pr(Y \mid Z) \mid\mid h(Z))\right] + H(Y \mid Z) \\
&\geq H(Y \mid Z).
\end{aligned}$$

It is also clear from the above proof that the minimum value of the cross-entropy loss is achieved when $h(Z)$ equals the conditional probability $\Pr(Y = 1 \mid Z)$, i.e., $h^*(Z) = \Pr(Y = 1 \mid Z = f(X))$. □

Theorem 8.3.1. Let f^* be the optimal feature map of (8.3) and define $H^* := H(A \mid Z = f^*(X))$. Then for any adversary $\widehat{A}$ such that $I(\widehat{A}; A \mid Z) = 0$, $\Pr_{\mathcal{D}^{f^*}}(\widehat{A} \neq A) \geq H^*/2\lg(6/H^*)$.

Proof. To prove this theorem, let E be the binary random variable that takes value 1 iff $A \neq \widehat{A}$, i.e., $E = \mathbb{I}(A \neq \widehat{A})$. Now consider the joint entropy of $A, \widehat{A}$ and E. On one hand, we have:

$$H(A, \widehat{A}, E) = H(A, \widehat{A}) + H(E \mid A, \widehat{A}) = H(A, \widehat{A}) + 0 = H(A \mid \widehat{A}) + H(\widehat{A}).$$

Note that the second equation holds because E is a deterministic function of A and $\widehat{A}$, that is, once A and $\widehat{A}$ are known, E is also known, hence $H(E \mid A, \widehat{A}) = 0$. On the other hand, we can also decompose $H(A, \widehat{A}, E)$ as follows:

$$H(A, \widehat{A}, E) = H(\widehat{A}) + H(A \mid \widehat{A}, E) + H(E \mid \widehat{A}).$$

Combining the above two equalities yields

$$H(A \mid \widehat{A}, E) + H(E \mid \widehat{A}) = H(A \mid \widehat{A}).$$

Furthermore, since conditioning cannot increase entropy, we have $H(E \mid \widehat{A}) \leq H(E)$, which further implies

$$H(A \mid \widehat{A}) \leq H(E) + H(A \mid \widehat{A}, E).$$

Now consider $H(A \mid \widehat{A}, E)$. Since $A \in \{0, 1\}$, by definition of the conditional entropy, we have:

$$H(A \mid \widehat{A}, E) = \Pr(E = 1)H(A \mid \widehat{A}, E = 1) + \Pr(E = 0)H(A \mid \widehat{A}, E = 0) = 0 + 0 = 0.$$

To lower bound $H(A \mid \widehat{A})$, realize that

$$I(A; \widehat{A}) + H(A \mid \widehat{A}) = H(A) = I(A; Z) + H(A \mid Z).$$

Since $\widehat{A}$ is a randomized function of Z such that $A \perp \widehat{A} \mid Z$, due to the celebrated data-processing inequality, we have $I(A; \widehat{A}) \leq I(A; Z)$, which implies

$$H(A \mid \widehat{A}) \geq H(A \mid Z).$$

Combine everything above, we have the following chain of inequalities hold:

$$H(A \mid Z) \leq H(A \mid \widehat{A}) \leq H(E) + H(A \mid \widehat{A}, E) = H(E),$$

which implies

$$\Pr_{\mathcal{D}^{f^*}}(A \neq \widehat{A}) = \Pr_{\mathcal{D}^{f^*}}(E = 1) \geq H_2^{-1}(H(A \mid Z)),$$

where $H_2^{-1}(\cdot)$ is the inverse function of the binary entropy $H(t) := -t \log t - (1 - t) \log(1 - t)$ when $t \in [0, 1]$. To conclude the proof, we apply Lemma 8.7.1 to further lower bound the inverse binary entropy function by

$$H_2^{-1}(H(A \mid Z)) \geq H(A \mid Z)/2 \lg(6/H(A \mid Z)),$$

completing the proof. □

Lemma 8.3.2. Let $\mathcal{D}_0$ and $\mathcal{D}_1$ be two distributions over $\mathcal{X}$ and let $\mathcal{D}_0^f$ and $\mathcal{D}_1^f$ be the induced distributions of $\mathcal{D}_0$ and $\mathcal{D}_1$ over $\mathcal{Z}$ by function f, then $d_{\text{JS}}(\mathcal{D}_0^f, \mathcal{D}_1^f) \leq d_{\text{JS}}(\mathcal{D}_0, \mathcal{D}_1)$.

Proof. Let B be a uniform random variable taking value in $\{0, 1\}$ and let the random variable Z_B with distribution $\mathcal{D}_B^f$ (resp. X_B with distribution $\mathcal{D}_B$) be the mixture of $\mathcal{D}_0^f$ and $\mathcal{D}_1^f$ (resp. $\mathcal{D}_0$ and

$\mathcal{D}_1$) according to B. It is easy to see that $\mathcal{D}_B = (\mathcal{D}_0 + \mathcal{D}_1)/2$, and we have:

$$\begin{aligned}
I(B; X_B) &= H(X_B) - H(X_B \mid B) \\
&= -\sum \mathcal{D}_B \log \mathcal{D}_B + \frac{1}{2}\left(\sum \mathcal{D}_0 \log \mathcal{D}_0 + \sum \mathcal{D}_1 \log \mathcal{D}_1\right) \\
&= -\frac{1}{2}\sum \mathcal{D}_0 \log \mathcal{D}_B - \frac{1}{2}\sum \mathcal{D}_1 \log \mathcal{D}_B + \frac{1}{2}\left(\sum \mathcal{D}_0 \log \mathcal{D}_0 + \sum \mathcal{D}_1 \log \mathcal{D}_1\right) \\
&= \frac{1}{2}\sum \mathcal{D}_0 \log \frac{\mathcal{D}_0}{\mathcal{D}_B} + \frac{1}{2}\sum \mathcal{D}_1 \log \frac{\mathcal{D}_1}{\mathcal{D}_B} \\
&= \frac{1}{2} D_{\mathrm{KL}}(\mathcal{D}_0 \parallel \mathcal{D}_B) + \frac{1}{2} D_{\mathrm{KL}}(\mathcal{D}_1 \parallel \mathcal{D}_B) \\
&= D_{\mathrm{JS}}(\mathcal{D}_0, \mathcal{D}_1).
\end{aligned}$$

Similarly, we have:

$$D_{\mathrm{JS}}(\mathcal{D}_0^f, \mathcal{D}_1^f) = I(B; Z_B).$$

Since $\mathcal{D}_0^f$ (resp. $\mathcal{D}_1^f$) is induced by f from $\mathcal{D}_0$ (resp. $\mathcal{D}_1$), by linearity, $\mathcal{D}_B^f$ is also induced by f from $\mathcal{D}_B$. Hence $Z_B = f(X_B)$ and the following Markov chain holds:

$$B \to X_B \to Z_B.$$

Apply the data processing inequality, we have

$$D_{\mathrm{JS}}(\mathcal{D}_0, \mathcal{D}_1) = I(B; X_B) \geq I(B; Z_B) = D_{\mathrm{JS}}(\mathcal{D}_0^f, \mathcal{D}_1^f).$$

Taking square root on both sides of the above inequality completes the proof. □

Lemma 8.3.3. Let $\hat{Y} = h(f(X)) \in \{0, 1\}$ be the predictor, then for $a \in \{0, 1\}$, $d_{\mathrm{JS}}(\mathcal{D}_a^Y, \mathcal{D}_a^{h\circ f}) \leq \sqrt{\mathrm{Err}_a(h \circ f)}$.

Proof. For $a \in \{0, 1\}$, by definition of the JS distance:

$$\begin{aligned}
d^2_{\mathrm{JS}}(\mathcal{D}^Y_a, \mathcal{D}^{h\circ f}_a) &= D_{\mathrm{JS}}(\mathcal{D}^Y_a, \mathcal{D}^{h\circ f}_a) \\
&\leq \|\mathcal{D}^Y_a - \mathcal{D}^{h\circ f}_a\|_1/2 && \text{(Lemma 8.7.2)} \\
&= (|\Pr(Y = 0 \mid A = a) - \Pr(h(f(X)) = 0 \mid A = a)| \\
&\qquad + |\Pr(Y = 1 \mid A = a) - \Pr(h(f(X)) = 1 \mid A = a)|)/2 \\
&= |\Pr(Y = 1 \mid A = a) - \Pr(h(f(X)) = 1 \mid A = a)| \\
&= |\mathbb{E}[Y \mid A = a] - \mathbb{E}[h(f(X)) \mid A = a]| \\
&\leq \mathbb{E}[|Y - h(f(X))| \mid A = a] \\
&= \mathrm{Err}_a(h \circ f),
\end{aligned}$$

where the expectation is taken over the joint distribution of X, Y. Taking square root at both sides then completes the proof. □

Theorem 8.3.2. Let $\mathcal{H}_A \subseteq 2^{\mathcal{Z}}$ be the hypothesis space of all the classifiers from $\mathcal{Z}$ to $\{0, 1\}$. Assume the conditions in Lemma 8.3.4 hold, then $\forall h \in \mathcal{H}$, $\textsc{Acc}_0(h \circ f) + \textsc{Acc}_1(h \circ f) \leq 2 - \frac{1}{3} D_{\mathrm{JS}}(\mathcal{D}^Y_0, \mathcal{D}^Y_1) + \textsc{Adv}(\mathcal{H}_A \circ f)$.

Proof. Before we delve into the details, we first give a high-level sketch of the main idea. The proof could be basically partitioned into two parts. In the first part, we will show that when $\mathcal{H}_A$ contains all the measurable prediction functions, $\textsc{Adv}(\mathcal{H}_A \circ f)$ could be used to upper bound $D_{\mathrm{JS}}(\mathcal{D}^f_0, \mathcal{D}^f_1)$. The second part combines Lemma 8.3.3 and Lemma 8.3.2 to complete the proof.

In this part we first show that $D_{\mathrm{JS}}(\mathcal{D}^f_0, \mathcal{D}^f_1) \leq \textsc{Adv}(\mathcal{H} \circ f)$:

$$\begin{aligned}
D_{\mathrm{JS}}(\mathcal{D}^f_0, \mathcal{D}^f_1) &\leq \frac{1}{2}\|\mathcal{D}^f_0 - \mathcal{D}^f_1\|_1 \\
&= d_{\mathrm{TV}}(\mathcal{D}^f_0, \mathcal{D}^f_1) \\
&= \sup_{A \in \mathscr{B}} |\mathcal{D}^f_0(A) - \mathcal{D}^f_1(A)|,
\end{aligned}$$

where $d_{\mathrm{TV}}(\cdot, \cdot)$ denotes the total variation distance and $\mathscr{B}$ is the sigma algebra that contains all the measurable subsets of $\mathcal{Z}$. On the other hand, when $\mathcal{H}_A$ contains all the measurable functions in $2^{\mathcal{Z}}$, we have:

$$\begin{aligned}\textsc{Adv}(\mathcal{H}_A \circ f) &= \max_{h_A \in \mathcal{H}_A} |\Pr(h_A(Z) = 1 \mid A = 0) - \Pr(h_A(Z) = 1 \mid A = 1)| \\ &= \max_{h_A \in \mathcal{H}_A} |\mathcal{D}_0(h_A^{-1}(1)) - \mathcal{D}_1(h_A^{-1}(1))| \\ &= \sup_{A \in \mathscr{B}} |\mathcal{D}_0^f(A) - \mathcal{D}_1^f(A)|,\end{aligned}$$

where the last equality follows from the fact that $\mathcal{H}_A$ is complete and contains all the measurable functions. Combine the above two parts we immediately have $D_{\text{JS}}(\mathcal{D}_0^f, \mathcal{D}_1^f) \leq \textsc{Adv}(\mathcal{H}_A \circ f)$.

Now using the key lemma, we have:

$$\begin{aligned}d_{\text{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y) &\leq d_{\text{JS}}(\mathcal{D}_0^Y, \mathcal{D}_0^{h\circ f}) + d_{\text{JS}}(\mathcal{D}_0^f, \mathcal{D}_1^f) + d_{\text{JS}}(\mathcal{D}_1^{h\circ f}, \mathcal{D}_1^Y) \\ &\leq \sqrt{\text{Err}_0(h \circ f)} + \sqrt{\textsc{Adv}(\mathcal{H}_A \circ f)} + \sqrt{\text{Err}_1(h \circ f)} \\ &= \sqrt{1 - \textsc{Acc}_0(h \circ f)} + \sqrt{\textsc{Adv}(\mathcal{H}_A \circ f)} + \sqrt{1 - \textsc{Acc}_1(h \circ f)} \\ &\leq \sqrt{3(1 - \textsc{Acc}_0(h \circ f) + 1 - \textsc{Acc}_1(h \circ f) + \textsc{Adv}(\mathcal{H}_A \circ f))} \\ &= \sqrt{3\big(2 - (\textsc{Acc}_0(h \circ f) + \textsc{Acc}_1(h \circ f) - \textsc{Adv}(\mathcal{H}_A \circ f))\big)}.\end{aligned}$$

Taking square at both sides and then rearrange the terms then completes the proof. □

Theorem 8.3.3. Assume the conditions in Theorem 8.3.2 hold. If $\textsc{Adv}(\mathcal{H}_A \circ f) \leq D_{\text{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y)$, then $\forall h \in \mathcal{H}$, $\text{Err}_0(h \circ f) + \text{Err}_1(h \circ f) \geq \frac{1}{2}\big(d_{\text{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y) - \sqrt{\textsc{Adv}(\mathcal{H}_A \circ f)}\big)^2$.

Proof. Similarly, using the key lemma, we have:

$$\begin{aligned}d_{\text{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y) &\leq d_{\text{JS}}(\mathcal{D}_0^Y, \mathcal{D}_0^{h\circ f}) + d_{\text{JS}}(\mathcal{D}_0, \mathcal{D}_1) + d_{\text{JS}}(\mathcal{D}_1^{h\circ f}, \mathcal{D}_1^Y) \\ &\leq \sqrt{\text{Err}_0(h \circ f)} + \sqrt{\textsc{Adv}(\mathcal{H}_A \circ f)} + \sqrt{\text{Err}_1(h \circ f)}\end{aligned}$$

Under the assumption that $\textsc{Adv}(\mathcal{H}_A \circ f) \leq D_{\text{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y)$, we have $d_{\text{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y) \geq \sqrt{\textsc{Adv}(\mathcal{H}_A \circ f)}$, hence by AM-GM inequality:

$$\sqrt{2\big(\text{Err}_0(h \circ f) + \text{Err}_1(h \circ f)\big)} \geq \sqrt{\text{Err}_0(h \circ f) + \text{Err}_1(h \circ f)} \geq d_{\text{JS}}(\mathcal{D}_0^Y, \mathcal{D}_1^Y) - \sqrt{\textsc{Adv}(\mathcal{H}_A \circ f)}.$$

Taking square at both sides then completes the proof. □

8.7.3 Detailed Experiments

In this section, we provide more details of the experiments. First we provide the details of different existing methods we evaluate. Then we elaborate more dataset description, model architecture and training parameters in different experiments.

Details on Methods

We provide a detailed description of each method here:

1). Privacy Partial Least Squares (PPLS): It learns $n \times X_d$ matrix for the feature transformation. The matrix is learned by maximizing the covariance of the learned representation and target attribute while minimizing the covariance of the learned representation and sensitive attribute.

2). Privacy Linear Discriminant Analysis (PLDA): It learns $n \times X_d$ matrix for the feature transformation. The matrix is learned by maximizing the Fisher's linear discriminability of the learned representation and target attribute while minimizing the Fisher's linear discriminability of the learned representation and sensitive attribute.

3). Minimax filter with alternative update (ALT-UP): The representation is learned via optimizing Equation 8.2 in an alternative way, first we update the parameters of the feature transformation module and the target attribute classifier, and then accordingly update the sensitive attribute classifier.

4). Maximum Entropy Adversarial Representation Learning (MAX-ENT): The objective equation is the slightly different from ALT-UP. The latter term contains additional entropy term to maximize unpredictability of the sensitive attribute.

5). Gradient Reversal Layer (GRL): The objective equation is the same as ALT-UP, and we train the feature transformation module by adding a gradient reversal layer between the feature transformation module and the sensitive attribute classifier.

6). Principal Component Analysis (PCA): It generates a $n \times X_d$ matrix for the feature transformation where the rows of the matrix are the n largest eigenvectors of the input dataset X.

7). Normal Training (NORM-TRAIN): It is equivalent to normal training by setting $\lambda = 0$ in Equation 8.2.

8). Local Differential Privacy (LDP): Standard Laplace mechanism of local differential privacy, where the noise is added to the raw representation for erasing the information of the sensitive attribute.

9). Differentially private SGD (DPSGD): It is one of the state-of-the-art differential privacy methods on deep learning. It adds Gaussian noise to the gradients when training the model.

Details on UCI Adult Dataset Evaluation

UCI Adult dataset is a benchmark machine learning dataset for income prediction. Each data record contains 14 categorical or numerical attributes, such as occupation, education and gender, to predict whether individual annual income exceeds $50K/year. The dataset is divided into training set (24130 examples), validation (6032 examples), and test set (15060 examples). We choose gender, age, and education as the sensitive attributes, respectively.

Table 8.1: Data distribution of income (Y) and gender (A) in UCI Adult dataset.

	$Y = 0$	$Y = 1$
$A = 0$	20988	9539
$A = 1$	13026	1669

Table 8.2: Data distribution of income (Y) and age (A) in UCI Adult dataset.

	$Y = 0$	$Y = 1$
$A = 0$	18042	2473
$A = 1$	15972	8735

Table 8.3: Data distribution of income (Y) and education (A) in UCI Adult dataset.

	$Y = 0$	$Y = 1$
$A = 0$	20447	4248
$A = 1$	13567	6960

We process each sensitive attribute as binary label for each experiment: for age label, 0 if the person is no greater than 35 years old and 1 otherwise; for education label, 0 if the person has not entered college or receive higher education than college, and 1 otherwise. In the mean time, we also remove corresponding sensitive attribute from the input, so the dimension of input data for each experiment is different. The input dimensions for income-gender experiment, income-age experiment, and income-education experiment are 113, 104 and 99, respectively. Table 8.1, Table 8.2 and Table 8.3 summarize the data distribution of UCI Adult dataset for protecting different sensitive attributes.

We use the two-layer ReLU-based neural net for f and one-layer neural net for h. The output dimensions of f are 64. We train all methods using SGD with the initial learning late 0.001 and momentum 0.9 for 40 epochs. In the DP-SGD experiment, we set the noise multiplier as 0.45 and 4.0 for small noise and large noise, respectively, and set the clipping norm as 1.0. (ϵ, δ) for DPSGD small noise and DPSGD large noise are $(33.7, 10^{-5})$ and $(0.572, 10^{-5})$, respectively. Among all methods, we report the one achieving the best performance on the target task in the validation set. We run the experiments for ten random seeds and compute the average.

Details on UTKFace Dataset Evaluation

UTKFace dataset is a large scale face dataset with annotations of age (range from 0 to 116 years old), gender (male and female), and ethnicity (White, Black, Asian, Indian, and Others). It contains 23,705 64×64 aligned and cropped RGB face images and we split the dataset into training set (15171 examples), validation set (3793 examples) and test set (4741 examples), respectively. We further process age label and ethnicity label as binary labels: 0 if the person is not greater than 35 years old for age label (is white for ethnicity label), and 1 if the the person is greater than 35 years old for age label (is non-white for ethnicity label). Table 8.4 and Table 8.5 summarize the data distribution of UTKFace dataset for protecting different sensitive attributes.

Table 8.4: Data distribution of gender (Y) and race (A) in UTKFace dataset.

	$Y=0$	$Y=1$
$A=0$	5477	4601
$A=1$	6914	6713

Table 8.5: Data distribution of gender (Y) and age (A) in UTKFace dataset.

	$Y=0$	$Y=1$
$A=0$	6889	8218
$A=1$	5502	3096

Since NORM-TRAIN, ALT-UP, GRL and DP can directly enjoy the benefits of using the state-of-the-art neural network architecture as feature extraction module, so we use the feature extraction module of Wide Residual Network [311] for the (non-linear) feature transformation module, while PPLS, PLDA, and PCA learn 12288×2048 matrix filter for f. We train all methods using SGD with the initial learning late 0.01 and momentum 0.9 for 30 epochs. The learning rate is decayed by a factor of 0.1 for every 10 epochs. In the DP-SGD experiment, we set the noise multiplier as 0.45 and 1.0 for small noise and large noise, respectively, and set the clipping norm as 1.0. (ϵ, δ) for DPSGD small noise and DPSGD large noise are $(25.7, 10^{-5})$ and $(2.7, 10^{-5})$, respectively. Among all methods, we report the one achieving the best performance on the target task in the validation set. We run the experiments for ten times and compute the average.

8.7.4 Additional Experimental Results

In this section, we present additional experimental results to gain more insights into how the trade-off parameter λ affects the performances of different adversarial presentation learning methods. We varies the values of λ and report the accuracies of both tasks using the Adult dataset when the sensitive attribute is gender. Note that all hyperparameter settings follow the previous experiments. The results are shown in Table 8.6. We can see that the overall trend is that when λ increases, the accuracies for both tasks decrease. Compared to ALT-UP and GRL, the training of MAX-ENT is unstable when λ is large.

Table 8.6: Performances of different adversarial representation learning methods when λ changes.

		λ	0	0.1	1	5
Gender	ALT-UP	TAR. ACC.	0.8501±0.0010	0.8496±0.0013	0.8483±0.0010	0.8456±0.0014
		SEN. ACC.	0.7408±0.0096	0.6682±0.0026	0.6627±0.0021	0.6737±0.0005
		λ	0	0.1	1	5
	GRL	TAR. ACC.	0.8501±0.0010	0.8465±0.0017	0.8449±0.0010	0.8387±0.0019
		SEN. ACC.	0.7408±0.0096	0.6677±0.0060	0.6677±0.0039	0.6764±0.0054
		λ	0	0.1	1	5
	MAX-ENT	TAR. ACC.	0.8501±0.0010	0.8450±0.0038	0.8411±0.0055	0.7891±0.0449
		SEN. ACC.	0.7408±0.0096	0.6928±0.0084	0.6897±0.0038	0.5695±0.1679
		λ	0	0.1	1	5
Age	ALT-UP	TAR. ACC.	0.8467±0.0011	0.8468±0.0009	0.8472±0.0011	0.8451±0.0008
		SEN. ACC.	0.7190±0.010	0.6516±0.0038	0.5422±0.0133	0.5573±0.0438
		λ	0	0.1	1	5
	GRL	TAR. ACC.	0.8467±0.0011	0.8444±0.0009	0.8445±0.0012	0.8422±0.0013
		SEN. ACC.	0.7190±0.010	0.6486±0.0067	0.5361±0.0134	0.5381±0.0133
		λ	0	0.1	1	5
	MAX-ENT	TAR. ACC.	0.8467±0.0011	0.8379±0.0056	0.8194±0.0345	0.7795±0.0406
		SEN. ACC.	0.7190±0.0100	0.6633±0.0669	0.6201±0.0820	0.5400±0.0316
		λ	0	0.1	1	5
Education	ALT-UP	TAR. ACC.	0.8494±0.0008	0.8498±0.0004	0.8497±0.0012	0.8494±0.0015
		SEN. ACC.	0.7088±0.0080	0.6062±0.0108	0.6044±0.0145	0.5462±0.0358
		λ	0	0.1	1	5
	GRL	TAR. ACC.	0.8494±0.0008	0.8525±0.0010	0.8518±0.0007	0.8500±0.0013
		SEN. ACC.	0.7088±0.0080	0.6082±0.0119	0.6015±0.0154	0.5528±0.0260
		λ	0	0.1	1	5
	MAX-ENT	TAR. ACC.	0.8494±0.0008	0.8365±0.0033	0.8253±0.0376	0.8087±0.0468
		SEN. ACC.	0.7088±0.0080	0.5790±0.0383	0.5484±0.0001	0.5386±0.0305

Chapter 9

Conclusion

This dissertation presents two parts of works that aim to help build ethical ML systems. In the first part of the dissertation, we develop NLP tools to extract fine-grained and structured information extraction from privacy policies, a type of policy documents describing the practices of using, sharing, and protecting users' data. The developed NLP tools could help users better understand how ML/AI service providers deal with their data and codify that information for ML/AI practitioners to build ethical AI systems. The second part of the dissertation provides the theoretical understanding and development of algorithmic interventions for ethical AI. In particular, we study the fairness problems for various machine learning tasks, such as classification, regression, and sequential decision-making. In addition, we also study adversarial representation learning, a technique that has been widely used for algorithmic fairness, and its implications for information obfuscation. The outcomes in the second part could help (1) answer the questions such as whether ethical AI requirements provided in the policy documents are achievable; (2) provide methods to achieve the requirements (3) understand the costs to achieve such requirements.

The presented works in this dissertation by no means cover all practices toward building ethical AI systems. For example, there might be other ways to present ethical AI requirements (e.g., model cards [312]) to users and ML/AI practitioners. Besides, there are a variety of notions of ethical AI, such as fairness and privacy, that the research in the dissertation could not cover. We view the outcome of this dissertation as a supportive step for building ethical ML systems. We hope the research presented in the thesis will facilitate the practices of building ethical machine learning systems and help increase the understanding and trust among stakeholders towards the machine learning systems.

Bibliography

[1] Shomir Wilson, Florian Schaub, Aswarth Abhilash Dara, Frederick Liu, Sushain Cherivirala, Pedro Giovanni Leon, Mads Schaarup Andersen, Sebastian Zimmeck, Kanthashree Mysore Sathyendra, N. Cameron Russell, Thomas B. Norton, Eduard Hovy, Joel Reidenberg, and Norman Sadeh. The creation and analysis of a website privacy policy corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1330–1340, August 2016.

[2] Martin Arjovsky and Léon Bottou. Towards principled methods for training generative adversarial networks. arxiv e-prints, art. *arXiv preprint arXiv:1701.04862*, 2017.

[3] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[5] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

[6] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[7] Wayne Xiong, Lingfeng Wu, Fil Alleva, Jasha Droppo, Xuedong Huang, and Andreas Stolcke. The microsoft 2017 conversational speech recognition system. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5934–5938. IEEE, 2018.

[8] Sorin Grigorescu, Bogdan Trasnea, Tiberiu Cocias, and Gigel Macesanu. A survey of deep learning techniques for autonomous driving. *Journal of Field Robotics*, 37(3):362–386, 2020.

[9] Javier Sánchez-Monedero, Lina Dencik, and Lilian Edwards. What does it mean to'solve'the problem of discrimination in hiring? social, technical and legal perspectives from the uk on automated hiring systems. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pages 458–468, 2020.

[10] Jon Walker. Artificial intelligence applications for lending and loan management. *Emerj. com*, 2019.

[11] Irene Y Chen, Emma Pierson, Sherri Rose, Shalmali Joshi, Kadija Ferryman, and Marzyeh Ghassemi. Ethical machine learning in healthcare. *Annual Review of Biomedical Data Science*, 4, 2020.

[12] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18. IEEE, 2017.

[13] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pages 1322–1333, 2015.

[14] Matthew Fredrikson, Eric Lantz, Somesh Jha, Simon Lin, David Page, and Thomas Ristenpart. Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. In *23rd {USENIX} Security Symposium ({USENIX} Security 14)*, pages 17–32, 2014.

[15] Solon Barocas and Andrew D Selbst. Big data's disparate impact. *Calif. L. Rev.*, 104:671, 2016.

[16] Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186, 2017.

[17] Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650, 2021.

[18] Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *Advances in neural information processing systems*, 29:4349–4357, 2016.

[19] Brent Mittelstadt. Principles alone cannot guarantee ethical ai. *Nature Machine Intelligence*, 1(11):501–507, 2019.

[20] Steven A Wright. Ai in the law: Towards assessing ethical risks. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 2160–2169. IEEE, 2020.

[21] Rachel Cummings, Gabriel Kaptchuk, and Elissa M Redmiles. " i need a better description": An investigation into user expectations for differential privacy. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 3037–3052, 2021.

[22] Kenneth Holstein, Jennifer Wortman Vaughan, Hal Daumé III, Miro Dudik, and Hanna Wallach. Improving fairness in machine learning systems: What do industry practitioners need? In *Proceedings of the 2019 CHI conference on human factors in computing systems*, pages 1–16, 2019.

[23] Michael Madaio, Lisa Egede, Hariharan Subramonyam, Jennifer Wortman Vaughan, and Hanna Wallach. Assessing the fairness of ai systems: Ai practitioners' processes, challenges, and needs for support. *arXiv preprint arXiv:2112.05675*, 2021.

[24] Eugene Bagdasaryan, Omid Poursaeed, and Vitaly Shmatikov. Differential privacy has disparate impact on model accuracy. *Advances in Neural Information Processing Systems*, 32:15479–15488, 2019.

[25] Rachel Cummings, Varun Gupta, Dhamma Kimpara, and Jamie Morgenstern. On the compatibility of privacy and fairness. In *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*, pages 309–315, 2019.

[26] Bashir Rastegarpanah, Mark Crovella, and Krishna P Gummadi. Fair inputs and fair outputs: The incompatibility of fairness in privacy and accuracy. In *Adjunct Publication of the 28th ACM Conference on User Modeling, Adaptation and Personalization*, pages 260–267, 2020.

[27] Wasi Ahmad, Jianfeng Chi, Yuan Tian, and Kai-Wei Chang. PolicyQA: A reading comprehension dataset for privacy policies. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 743–749, Online, November 2020. Association for Computational Linguistics.

[28] Wasi Ahmad, Jianfeng Chi, Tu Le, Thomas Norton, Yuan Tian, and Kai-Wei Chang. Intent classification and slot filling for privacy policies. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4402–4417, Online, August 2021. Association for Computational Linguistics.

[29] Jianfeng Chi, William Shand, Yaodong Yu, Kai-Wei Chang, Han Zhao, and Yuan Tian. Conditional supervised contrastive learning for fair text classification. *arXiv preprint arXiv:2205.11485*, 2022.

[30] Jianfeng Chi, Yuan Tian, Geoffrey J Gordon, and Han Zhao. Understanding and mitigating accuracy disparity in regression. In *International conference on machine learning*, 2021.

[31] Jianfeng Chi, Jian Shen, Xinyi Dai, Weinan Zhang, Yuan Tian, and Han Zhao. Towards return parity in markov decision processes. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 1161–1178. PMLR, 28–30 Mar 2022.

[32] Han Zhao, Jianfeng Chi, Yuan Tian, and Geoffrey J Gordon. Trade-offs and guarantees of adversarial representation learning for information obfuscation. *Advances in Neural Information Processing Systems*, 33, 2020.

[33] Aleecia M McDonald and Lorrie Faith Cranor. The cost of reading privacy policies. *Isjlp*, 4:543, 2008.

[34] Joel R Reidenberg, Jaspreet Bhatia, Travis D Breaux, and Thomas B Norton. Ambiguity in privacy policies and the impact of regulation. *The Journal of Legal Studies*, 45(S2):S163–S190, 2016.

[35] Shomir Wilson, Florian Schaub, Aswarth Abhilash Dara, Frederick Liu, Sushain Cherivirala, Pedro Giovanni Leon, Mads Schaarup Andersen, Sebastian Zimmeck, Kanthashree Mysore Sathyendra, N. Cameron Russell, Thomas B. Norton, Eduard Hovy, Joel Reidenberg, and Norman Sadeh. The creation and analysis of a website privacy policy corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1330–1340, Berlin, Germany, August 2016. Association for Computational Linguistics.

[36] Kanthashree Mysore Sathyendra, Shomir Wilson, Florian Schaub, Sebastian Zimmeck, and Norman Sadeh. Identifying the provision of choices in privacy policy text. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2774–2779, 2017.

[37] Hamza Harkous, Kassem Fawaz, Rémi Lebret, Florian Schaub, Kang G Shin, and Karl Aberer. Polisis: Automated analysis and presentation of privacy policies using deep learning. In *27th {USENIX} security symposium ({USENIX} security 18)*, pages 531–548, 2018.

[38] Sebastian Zimmeck, Peter Story, Daniel Smullen, Abhilasha Ravichander, Ziqi Wang, Joel R Reidenberg, N Cameron Russell, and Norman Sadeh. Maps: Scaling privacy compliance analysis to a million apps. *Proc. Priv. Enhancing Tech.*, 2019:66, 2019.

[39] Fei Liu, Rohan Ramanath, Norman Sadeh, and Noah A. Smith. A step towards usable privacy policy: Automatic alignment of privacy statements. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 884–894, Dublin, Ireland, August 2014. Dublin City University and Association for Computational Linguistics.

[40] Rohan Ramanath, Fei Liu, Norman Sadeh, and Noah A Smith. Unsupervised alignment of privacy policies using hidden markov models. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 605–610, 2014.

[41] Mitra Bokaie Hosseini, Pragyan K C, Irwin Reyes, and Serge Egelman. Identifying and classifying third-party entities in natural language privacy policies. In *Proceedings of the Second Workshop on Privacy in NLP*, pages 18–27, Online, November 2020. Association for Computational Linguistics.

[42] Duc Bui, Kang G Shin, Jong-Min Choi, and Junbum Shin. Automated extraction and presentation of data practices in privacy policies. *Proc. Priv. Enhancing Technol.*, 2021(2):88–110, 2021.

[43] Yan Shvartzshanider, Ananth Balashankar, Thomas Wies, and Lakshminarayanan Subramanian. RECIPE: Applying open domain question answering to privacy policies. In *Proceedings of the Workshop on Machine Reading for Question Answering*, pages 71–77, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[44] Abhilasha Ravichander, Alan W Black, Shomir Wilson, Thomas Norton, and Norman Sadeh. Question answering for privacy policies: Combining computational and legal perspectives. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4949–4959, Hong Kong, China, November 2019. Association for Computational Linguistics.

[45] Bingfeng Luo, Yansong Feng, Jianbo Xu, Xiang Zhang, and Dongyan Zhao. Learning to predict charges for criminal cases with legal basis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2727–2736, 2017.

[46] Paheli Bhattacharya, Kaustubh Hiware, Subham Rajgaria, Nilay Pochhi, Kripabandhu Ghosh, and Saptarshi Ghosh. A comparative study of summarization algorithms applied to legal case judgments. In *European Conference on Information Retrieval*, pages 413–428. Springer, 2019.

[47] Ilias Chalkidis, Ion Androutsopoulos, and Nikolaos Aletras. Neural legal judgment prediction in english. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4317–4323, 2019.

[48] Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. How does nlp benefit legal system: A summary of legal artificial intelligence. In *Proceedings of ACL*, 2020.

[49] Arpan Mandal, Kripabandhu Ghosh, Saptarshi Ghosh, and Sekhar Mandal. A sequence labeling model for catchphrase identification from legal case documents. *Artificial Intelligence and Law*, pages 1–34, 2021.

[50] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021.

[51] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226, 2012.

[52] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29:3315–3323, 2016.

[53] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. On fairness and calibration. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 5684–5693, 2017.

[54] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. *Advances in Neural Information Processing Systems*, 30, 2017.

[55] Faisal Kamiran and Toon Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, 2012.

[56] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 259–268, 2015.

[57] Flavio P Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. Optimized pre-processing for discrimination prevention. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 3995–4004, 2017.

[58] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 335–340, 2018.

[59] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *International conference on machine learning*, pages 325–333. PMLR, 2013.

[60] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 35–50. Springer, 2012.

[61] L Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K Vishnoi. Classification with fairness constraints: A meta-algorithm with provable guarantees. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 319–328, 2019.

[62] Faisal Kamiran, Asim Karim, and Xiangliang Zhang. Decision theory for discrimination-aware classification. In *2012 IEEE 12th International Conference on Data Mining*, pages 924–929. IEEE, 2012.

[63] Evgenii Chzhen, Christophe Denis, Mohamed Hebiri, Luca Oneto, and Massimiliano Pontil. Fair regression with wasserstein barycenters. *arXiv preprint arXiv:2006.07286*, 2020.

[64] Chandler May, Alex Wang, Shikha Bordia, Samuel R Bowman, and Rachel Rudinger. On measuring social biases in sentence encoders. In *2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2019*, pages 622–628. Association for Computational Linguistics (ACL), 2019.

[65] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Gender bias in coreference resolution: Evaluation and debiasing methods. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 15–20, 2018.

[66] Shikha Bordia and Samuel R Bowman. Identifying and reducing gender bias in word-level language models. In *NAACL-HLT (Student Research Workshop)*, 2019.

[67] Joel Escudé Font and Marta R Costa-jussà. Equalizing gender bias in neural machine translation with word embeddings techniques. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 147–154, 2019.

[68] Kate Crawford. The trouble with bias. In *Conference on Neural Information Processing Systems, invited speaker*, 2017.

[69] Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. Language (technology) is power: A critical survey of "bias" in nlp. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, 2020.

[70] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*, pages 268–282. IEEE, 2018.

[71] Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, Yann Ollivier, and Hervé Jégou. White-box vs black-box: Bayes optimal strategies for membership inference. In *International Conference on Machine Learning*, pages 5558–5567. PMLR, 2019.

[72] Zecheng He, Tianwei Zhang, and Ruby B Lee. Model inversion attacks against collaborative inference. In *Proceedings of the 35th Annual Computer Security Applications Conference*, pages 148–162, 2019.

[73] Florian Tramèr, Fan Zhang, Ari Juels, Michael K Reiter, and Thomas Ristenpart. Stealing machine learning models via prediction apis. In *25th {USENIX} Security Symposium ({USENIX} Security 16)*, pages 601–618, 2016.

[74] Binghui Wang and Neil Zhenqiang Gong. Stealing hyperparameters in machine learning. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 36–52. IEEE, 2018.

[75] Karan Ganju, Qi Wang, Wei Yang, Carl A Gunter, and Nikita Borisov. Property inference attacks on fully connected neural networks using permutation invariant representations. In *Proceedings of the 2018 ACM SIGSAC conference on computer and communications security*, pages 619–633, 2018.

[76] Ahmed Salem, Apratim Bhattacharya, Michael Backes, Mario Fritz, and Yang Zhang. Updates-leak: Data set inference and reconstruction attacks in online learning. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*, pages 1291–1308, 2020.

[77] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

[78] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016.

[79] Nicolas Papernot, Martín Abadi, Ulfar Erlingsson, Ian Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private training data. *arXiv preprint arXiv:1610.05755*, 2016.

[80] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 169–178, 2009.

[81] Ran Gilad-Bachrach, Nathan Dowlin, Kim Laine, Kristin Lauter, Michael Naehrig, and John Wernsing. Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy. In *International conference on machine learning*, pages 201–210. PMLR, 2016.

[82] Hervé Chabanne, Amaury De Wargny, Jonathan Milgram, Constance Morel, and Emmanuel Prouff. Privacy-preserving classification on deep neural network. *Cryptology ePrint Archive*, 2017.

[83] Florian Bourse, Michele Minelli, Matthias Minihold, and Pascal Paillier. Fast homomorphic evaluation of deep discretized neural networks. In *Annual International Cryptology Conference*, pages 483–512. Springer, 2018.

[84] Amartya Sanyal, Matt Kusner, Adria Gascon, and Varun Kanade. TAPAS: Tricks to accelerate (encrypted) prediction as a service. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4490–4499. PMLR, 10–15 Jul 2018.

[85] M Sadegh Riazi, Mohammad Samragh, Hao Chen, Kim Laine, Kristin Lauter, and Farinaz Koushanfar. {XONN}: Xnor-based oblivious deep neural network inference. In *28th {USENIX} Security Symposium ({USENIX} Security 19)*, pages 1501–1518, 2019.

[86] Eleftheria Makri, Dragos Rotaru, Nigel P Smart, and Frederik Vercauteren. Epic: efficient private image classification (or: Learning from the masters). In *Cryptographers' Track at the RSA Conference*, pages 473–492. Springer, 2019.

[87] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1175–1191, 2017.

[88] Nitin Agrawal, Ali Shahin Shamsabadi, Matt J Kusner, and Adrià Gascón. Quotient: two-party secure neural network training and prediction. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 1231–1247, 2019.

[89] Sameer Wagh, Divya Gupta, and Nishanth Chandran. Securenn: 3-party secure computation for neural network training. *Proc. Priv. Enhancing Technol.*, 2019(3):26–49, 2019.

[90] Payman Mohassel and Peter Rindal. Aby3: A mixed protocol framework for machine learning. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pages 35–52, 2018.

[91] Harsh Chaudhari, Rahul Rachuri, and Ajith Suresh. Trident: Efficient 4pc framework for privacy preserving machine learning. *arXiv preprint arXiv:1912.02631*, 2019.

[92] Bita Darvish Rouhani, M Sadegh Riazi, and Farinaz Koushanfar. Deepsecure: Scalable provably-secure deep learning. In *Proceedings of the 55th Annual Design Automation Conference*, pages 1–6, 2018.

[93] M Sadegh Riazi, Christian Weinert, Oleksandr Tkachenko, Ebrahim M Songhori, Thomas Schneider, and Farinaz Koushanfar. Chameleon: A hybrid secure computation framework for machine learning applications. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, pages 707–721, 2018.

[94] Nishant Kumar, Mayank Rathee, Nishanth Chandran, Divya Gupta, Aseem Rastogi, and Rahul Sharma. Cryptflow: Secure tensorflow inference. In *2020 IEEE Symposium on Security and Privacy (SP)*, pages 336–353. IEEE, 2020.

[95] Raphael Bost, Raluca Ada Popa, Stephen Tu, and Shafi Goldwasser. Machine learning classification over encrypted data. *Cryptology ePrint Archive*, 2014.

[96] Jian Liu, Mika Juuti, Yao Lu, and Nadarajah Asokan. Oblivious neural network predictions via minionn transformations. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 619–631, 2017.

[97] Chiraag Juvekar, Vinod Vaikuntanathan, and Anantha Chandrakasan. {GAZELLE}: A low latency framework for secure neural network inference. In *27th {USENIX} Security Symposium ({USENIX} Security 18)*, pages 1651–1669, 2018.

[98] Pratyush Mishra, Ryan Lehmkuhl, Akshayaram Srinivasan, Wenting Zheng, and Raluca Ada Popa. Delphi: A cryptographic inference service for neural networks. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*, pages 2505–2522, 2020.

[99] Miro Enev, Jaeyeon Jung, Liefeng Bo, Xiaofeng Ren, and Tadayoshi Kohno. Sensorsift: balancing sensor data privacy and utility in automated face understanding. In *Proceedings of the 28th Annual Computer Security Applications Conference*, pages 149–158. ACM, 2012.

[100] Jacob Whitehill and Javier Movellan. Discriminately decreasing discriminability with learned image filters. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2488–2495. IEEE, 2012.

[101] Jihun Hamm. Minimax filter: Learning to preserve privacy from inference attacks. *The Journal of Machine Learning Research*, 18(1):4704–4734, 2017.

[102] Yitong Li, Timothy Baldwin, and Trevor Cohn. Towards robust and privacy-preserving text representations. *arXiv preprint arXiv:1805.06093*, 2018.

[103] Ye Wang, Yuksel Ozan Basciftci, and Prakash Ishwar. Privacy-utility tradeoffs under constrained data release mechanisms. *arXiv preprint arXiv:1710.09295*, 2017.

[104] Zhenyu Wu, Zhangyang Wang, Zhaowen Wang, and Hailin Jin. Towards privacy-preserving visual recognition via adversarial training: A pilot study. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 606–624, 2018.

[105] Seyed Ali Osia, Ali Taheri, Ali Shahin Shamsabadi, Kleomenis Katevas, Hamed Haddadi, and Hamid R Rabiee. Deep private-feature extraction. *IEEE Transactions on Knowledge and Data Engineering*, 32(1):54–66, 2018.

[106] Martin Bertran, Natalia Martinez, Afroditi Papadaki, Qiang Qiu, Miguel Rodrigues, Galen Reeves, and Guillermo Sapiro. Adversarially learned representations for information ob-

fuscation and inference. In *International Conference on Machine Learning*, pages 614–623, 2019.

[107] Seyed Ali Osia, Ali Shahin Shamsabadi, Sina Sajadmanesh, Ali Taheri, Kleomenis Katevas, Hamid R Rabiee, Nicholas D Lane, and Hamed Haddadi. A hybrid deep learning architecture for privacy-preserving mobile analytics. *IEEE Internet of Things Journal*, 2020.

[108] Abhilasha Ravichander, Alan W Black, Shomir Wilson, Thomas Norton, and Norman Sadeh. Question answering for privacy policies: Combining computational and legal perspectives. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4947–4958, November 2019.

[109] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*, 2016.

[110] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, June 2019.

[111] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics.

[112] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.

[113] Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Well-read students learn better: The impact of student initialization on knowledge distillation. *arXiv preprint arXiv:1908.08962*, 2019.

[114] Norman Sadeh, Alessandro Acquisti, Travis D Breaux, Lorrie Faith Cranor, Aleecia M McDonald, Joel R Reidenberg, Noah A Smith, Fei Liu, N Cameron Russell, Florian Schaub, et al. The usable privacy policy project. *Technical report, Technical Report, CMU-ISR-13-119*, 2013.

[115] Shomir Wilson, Florian Schaub, Rohan Ramanath, Norman Sadeh, Fei Liu, Noah A Smith, and Frederick Liu. Crowdsourcing annotations for websites' privacy policies: Can it really work? In *Proceedings of the 25th International Conference on World Wide Web*, pages 133–143. International World Wide Web Conferences Steering Committee, 2016.

[116] Kanthashree Mysore Sathyendra, Florian Schaub, Shomir Wilson, and Norman Sadeh. Automatic extraction of opt-out choices from privacy policies. In *2016 AAAI Fall Symposium Series*, 2016.

[117] Jaspreet Bhatia and Travis D Breaux. Towards an information type lexicon for privacy policies. In *2015 IEEE eighth international workshop on requirements engineering and law (RELAW)*, pages 19–24. IEEE, 2015.

[118] Jaspreet Bhatia, Morgan C Evans, Sudarshan Wadkar, and Travis D Breaux. Automated extraction of regulated information types using hyponymy relations. In *2016 IEEE 24th*

International Requirements Engineering Conference Workshops (REW), pages 19–25. IEEE, 2016.

[119] Jasmin Kaur, Rozita A Dara, Charlie Obimbo, Fei Song, and Karen Menard. A comprehensive keyword analysis of online privacy policies. *Information Security Journal: A Global Perspective*, 27(5-6):260–275, 2018.

[120] Fei Liu, Rohan Ramanath, Norman Sadeh, and Noah A. Smith. A step towards usable privacy policy: Automatic alignment of privacy statements. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 884–894, Dublin, Ireland, August 2014.

[121] Yan Shvartzshanider, Ananth Balashankar, Thomas Wies, and Lakshminarayanan Subramanian. RECIPE: Applying open domain question answering to privacy policies. In *Proceedings of the Workshop on Machine Reading for Question Answering*, pages 71–77, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[122] Federal Trade Commission et al. Protecting consumer privacy in an era of rapid change. *FTC report*, 2012.

[123] Joshua Gluck, Florian Schaub, Amy Friedman, Hana Habib, Norman Sadeh, Lorrie Faith Cranor, and Yuvraj Agarwal. How short is too short? implications of length and framing on the effectiveness of privacy notices. In *Twelfth Symposium on Usable Privacy and Security ({SOUPS} 2016)*, pages 321–340, 2016.

[124] Florencia Marotta-Wurgler. Does "notice and choice" disclosure regulation work? an empirical study of privacy policies,". In *Michigan Law: Law and Economics Workshop*, 2015.

[125] Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. Brat: a web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107, 2012.

[126] Krippendorff Klaus. Content analysis: An introduction to its methodology, 1980.

[127] Ron Artstein and Massimo Poesio. Survey article: Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596, 2008.

[128] Dennis Reidsma and Jean Carletta. Reliability measurement without limits. *Computational Linguistics*, 34(3):319–326, 2008.

[129] Milan Straka, Jan Hajic, and Jana Straková. Udpipe: trainable pipeline for processing conll-u files performing tokenization, morphological analysis, pos tagging and parsing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4290–4297, 2016.

[130] Qian Chen, Zhu Zhuo, and Wen Wang. Bert for joint intent classification and slot filling. *arXiv preprint arXiv:1902.10909*, 2019.

[131] Subendhu Rongali, Luca Soldaini, Emilio Monti, and Wael Hamza. Don't parse, generate! a sequence to sequence architecture for task-oriented semantic parsing. In *Proceedings of The Web Conference 2020*, pages 2962–2968, 2020.

[132] Haoran Li, Abhinav Arora, Shuohui Chen, Anchit Gupta, Sonal Gupta, and Yashar Mehdad. MTOP: A comprehensive multilingual task-oriented semantic parsing benchmark. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational*

Linguistics: Main Volume, pages 2950–2962, Online, April 2021. Association for Computational Linguistics.

[133] David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. Entity, relation, and event extraction with contextualized span representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5784–5789, Hong Kong, China, November 2019. Association for Computational Linguistics.

[134] Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi. A general framework for information extraction using dynamic span graphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3036–3046, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[135] Bing Liu and Ian Lane. Attention-based recurrent neural network models for joint intent detection and slot filling. In *Interspeech 2016, 17th Annual Conference of the International Speech Communication Association*, pages 685–689, 09 2016.

[136] Xiaodong Zhang and Houfeng Wang. A joint model of intent determination and slot filling for spoken language understanding. In *IJCAI*, volume 16, pages 2993–2999, 2016.

[137] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc., 2017.

[138] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[139] Jie Zhou and Wei Xu. End-to-end learning of semantic role labeling using recurrent neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1127–1137, Beijing, China, July 2015. Association for Computational Linguistics.

[140] Ryan Cotterell and Kevin Duh. Low-resource named entity recognition with cross-lingual, character-level neural conditional random fields. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 91–96, Taipei, Taiwan, November 2017. Asian Federation of Natural Language Processing.

[141] John D Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289, 2001.

[142] Xuezhe Ma and Eduard Hovy. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074, Berlin, Germany, August 2016. Association for Computational Linguistics.

[143] Qile Zhu, Haidar Khan, Saleh Soltan, Stephen Rawls, and Wael Hamza. Don't parse, insert: Multilingual semantic parsing with insertion based decoding. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 496–506, Online, November 2020. Association for Computational Linguistics.

[144] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Advances in Neural Information Processing Systems*, 2020.

[145] Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. Unified language model pre-training for natural language understanding and generation. In *Advances in Neural Information Processing Systems*, pages 13063–13075, 2019.

[146] Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Jianfeng Gao, Songhao Piao, Ming Zhou, et al. Unilmv2: Pseudo-masked language models for unified language model pre-training. In *International Conference on Machine Learning*, pages 642–652. PMLR, 2020.

[147] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mass: Masked sequence to sequence pre-training for language generation. In *International Conference on Machine Learning*, pages 5926–5936, 2019.

[148] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July 2020. Association for Computational Linguistics.

[149] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics.

[150] Elisa Costante, Jerry den Hartog, and Milan Petković. What websites know about you. In *Data Privacy Management and Autonomous Spontaneous Security*, pages 146–159. Springer, 2012.

[151] Mitra Bokaei Hosseini, Sudarshan Wadkar, Travis D Breaux, and Jianwei Niu. Lexical similarity of information type hypernyms, meronyms and synonyms in privacy policies. In *2016 AAAI Fall Symposium Series*, 2016.

[152] Vinayshekhar Bannihatti Kumar, Roger Iyengar, Namita Nisal, Yuanyuan Feng, Hana Habib, Peter Story, Sushain Cherivirala, Margaret Hagan, Lorrie Cranor, Shomir Wilson, et al. Finding a choice in a haystack: Automatic extraction of opt-out statements from privacy policy text. In *Proceedings of The Web Conference 2020*, pages 1943–1954, 2020.

[153] Md Rizwan Parvez, Jianfeng Chi, Wasi Uddin Ahmad, Yuan Tian, and Kai-Wei Chang. Retrieval enhanced data augmentation for question answering on privacy policies. *arXiv preprint arXiv:2204.08952*, 2022.

[154] Charles T Hemphill, John J Godfrey, and George R Doddington. The atis spoken language systems pilot corpus. In *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27, 1990*, 1990.

[155] Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. Snips

voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*, 2018.

[156] Abhirut Gupta, Anupama Ray, Gargi Dasgupta, Gautam Singh, Pooja Aggarwal, and Prateeti Mohapatra. Semantic parsing for technical support questions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3251–3259, Santa Fe, New Mexico, USA, August 2018. Association for Computational Linguistics.

[157] Shyam Upadhyay, Manaal Faruqui, Gokhan Tür, Hakkani-Tür Dilek, and Larry Heck. (almost) zero-shot cross-lingual spoken language understanding. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6034–6038. IEEE, 2018.

[158] Sebastian Schuster, Sonal Gupta, Rushin Shah, and Mike Lewis. Cross-lingual transfer learning for multilingual task oriented dialog. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3795–3805, 2019.

[159] Weijia Xu, Batool Haider, and Saab Mansour. End-to-end slot alignment and recognition for cross-lingual NLU. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5052–5063, Online, November 2020. Association for Computational Linguistics.

[160] Ben Hutchinson, Vinodkumar Prabhakaran, Emily Denton, Kellie Webster, Yu Zhong, and Stephen Denuyl. Social biases in nlp models as barriers for persons with disabilities. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5491–5501, 2020.

[161] Kellie Webster, Xuezhi Wang, Ian Tenney, Alex Beutel, Emily Pitler, Ellie Pavlick, Jilin Chen, Ed Chi, and Slav Petrov. Measuring and reducing gendered correlations in pre-trained models. *arXiv preprint arXiv:2010.06032*, 2020.

[162] Daniel de Vassimon Manela, David Errington, Thomas Fisher, Boris van Breugel, and Pasquale Minervini. Stereotype and skew: Quantifying gender bias in pre-trained and fine-tuned language models. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2232–2242, 2021.

[163] Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, et al. Ethical and social risks of harm from language models. *arXiv preprint arXiv:2112.04359*, 2021.

[164] Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 67–73, 2018.

[165] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006.

[166] Lajanugen Logeswaran and Honglak Lee. An efficient framework for learning sentence representations. *arXiv preprint arXiv:1803.02893*, 2018.

[167] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

[168] Olivier Henaff. Data-efficient image recognition with contrastive predictive coding. In *International Conference on Machine Learning*, pages 4182–4192. PMLR, 2020.

[169] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[170] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33, 2020.

[171] Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2021.

[172] Aili Shen, Xudong Han, Trevor Cohn, Timothy Baldwin, and Lea Frermann. Contrastive learning for fair representations. *arXiv preprint arXiv:2109.10645*, 2021.

[173] Yao-Hung Hubert Tsai, Martin Q Ma, Han Zhao, Kun Zhang, Louis-Philippe Morency, and Ruslan Salakhutdinov. Conditional contrastive learning: Removing undesirable information in self-supervised representations. *arXiv preprint arXiv:2106.02866*, 2021.

[174] Yao-Hung Hubert Tsai, Tianqin Li, Martin Q. Ma, Han Zhao, Kun Zhang, Louis-Philippe Morency, and Ruslan Salakhutdinov. Conditional contrastive learning with kernel. In *International Conference on Learning Representations*, 2022.

[175] Han Zhao and Geoff Gordon. Inherent tradeoffs in learning fair representations. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[176] Han Zhao, Amanda Coston, Tameem Adel, and Geoffrey J Gordon. Conditional learning of fair representations. In *International Conference on Learning Representations*, 2019.

[177] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[178] Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.

[179] Ioana Baldini, Dennis Wei, Karthikeyan Natesan Ramamurthy, Mikhail Yurochkin, and Moninder Singh. Your fairness may vary: group fairness of pretrained language models in toxic text classification. *arXiv preprint arXiv:2108.01250*, 2021.

[180] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR, 2020.

[181] Farzan Farnia and David Tse. A minimax approach to supervised learning. *Advances in Neural Information Processing Systems*, 29, 2016.

[182] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[183] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In Hal Daumé III and Aarti Singh,

editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 13–18 Jul 2020.

[184] Beliz Gunel, Jingfei Du, Alexis Conneau, and Veselin Stoyanov. Supervised contrastive learning for pre-trained language model fine-tuning. In *International Conference on Learning Representations*, 2021.

[185] Jiequan Cui, Zhisheng Zhong, Shu Liu, Bei Yu, and Jiaya Jia. Parametric contrastive learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 715–724, 2021.

[186] Maria De-Arteaga, Alexey Romanov, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnaram Kenthapadi, and Adam Tauman Kalai. Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 120–128, 2019.

[187] Yada Pruksachatkun, Satyapriya Krishna, Jwala Dhamala, Rahul Gupta, and Kai-Wei Chang. Does robustness improve fairness? approaching fairness with word substitution robustness methods for text classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3320–3331, Online, August 2021. Association for Computational Linguistics.

[188] Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Nuanced metrics for measuring unintended bias with real data for text classification. In *Companion Proceedings of The 2019 World Wide Web Conference*, WWW '19, page 491–500, New York, NY, USA, 2019. Association for Computing Machinery.

[189] Luyu Gao, Yunyi Zhang, Jiawei Han, and Jamie Callan. Scaling deep contrastive learning batch size under memory limited setup. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pages 316–321, Online, August 2021. Association for Computational Linguistics.

[190] Yanai Elazar and Yoav Goldberg. Adversarial removal of demographic attributes from text data. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 11–21, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[191] Jon M. Kleinberg, Sendhil Mullainathan, and M. Raghavan. Inherent trade-offs in the fair determination of risk scores. *ArXiv*, abs/1609.05807, 2017.

[192] Ari Ball-Burack, Michelle Seng Ah Lee, Jennifer Cobbe, and Jatinder Singh. Differential tweetment: Mitigating racial dialect bias in harmful tweet detection. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 116–128, New York, NY, USA, 2021. Association for Computing Machinery.

[193] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. Learning adversarially fair and transferable representations. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 3384–3393. PMLR, 10–15 Jul 2018.

[194] Xudong Han, Timothy Baldwin, and Trevor Cohn. Diverse adversaries for mitigating bias in training. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2760–2765, Online, April 2021. Association for Computational Linguistics.

[195] Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null it out: Guarding protected attributes by iterative nullspace projection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7237–7256, Online, July 2020. Association for Computational Linguistics.

[196] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Gender bias in coreference resolution: Evaluation and debiasing methods. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 15–20, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[197] Shikha Bordia and Samuel R. Bowman. Identifying and reducing gender bias in word-level language models. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 7–15, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[198] Ioana Baldini, Dennis Wei, Karthikeyan Natesan Ramamurthy, Moninder Singh, and Mikhail Yurochkin. Your fairness may vary: Pretrained language model fairness in toxic text classification. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2245–2262, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[199] Pengyu Cheng, Weituo Hao, Siyang Yuan, Shijing Si, and Lawrence Carin. Fairfil: Contrastive neural debiasing method for pretrained text encoders. In *International Conference on Learning Representations*, 2021.

[200] Pengyu Cheng, Martin Renqiang Min, Dinghan Shen, Christopher Malon, Yizhe Zhang, Yitong Li, and Lawrence Carin. Improving disentangled text representation learning with information-theoretic guidance. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7530–7541, Online, July 2020. Association for Computational Linguistics.

[201] Pierre Colombo, Pablo Piantanida, and Chloé Clavel. A novel estimator of mutual information for learning to disentangle textual representations. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6539–6550, Online, August 2021. Association for Computational Linguistics.

[202] Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2564–2572. PMLR, 10–15 Jul 2018.

[203] Forest Yang, Mouhamadou Cisse, and Sanmi Koyejo. Fairness with overlapping groups; a probabilistic perspective. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 4067–4078. Curran Associates, Inc., 2020.

[204] XuanLong Nguyen, Martin J. Wainwright, and Michael I. Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861, 2010.

[205] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton A. Earnshaw, Imran S. Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and

Percy Liang. WILDS: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning (ICML)*, 2021.

[206] Jason Wei and Kai Zou. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China, November 2019. Association for Computational Linguistics.

[207] Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. Understanding back-translation at scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 489–500, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[208] Sosuke Kobayashi. Contextual augmentation: Data augmentation by words with paradigmatic relations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 452–457, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[209] Pauline T Kim. Data-driven discrimination at work. *Wm. & Mary L. Rev.*, 58:857, 2016.

[210] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91, 2018.

[211] Mohammad Yaghini, Bogdan Kulynych, and Carmela Troncoso. Disparate vulnerability: On the unfairness of privacy attacks against machine learning. *arXiv preprint arXiv:1906.00389*, 2019.

[212] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. In *Artificial Intelligence and Statistics*, pages 962–970, 2017.

[213] Han Zhao and Geoffrey J Gordon. Inherent tradeoffs in learning fair representations. In *Advances in neural information processing systems*, 2019.

[214] Ray Jiang, Aldo Pacchiano, Tom Stepleton, Heinrich Jiang, and Silvia Chiappa. Wasserstein fair classification. *arXiv preprint arXiv:1907.12059*, 2019.

[215] Irene Chen, Fredrik D Johansson, and David Sontag. Why is my classifier discriminatory? In *Advances in Neural Information Processing Systems*, pages 3539–3550, 2018.

[216] Arvind Narayanan. Translation tutorial: 21 fairness definitions and their politics. In *Proc. Conf. Fairness Accountability Transp., New York, USA*, 2018.

[217] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

[218] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pages 214–223, 2017.

[219] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

[220] William Dieterich, Christina Mendoza, and Tim Brennan. Compas risk scales: Demonstrating accuracy equity and predictive parity. *Northpointe Inc*, 2016.

[221] Linda F Wightman and Henry Ramsey. *LSAC national longitudinal bar passage study.* Law School Admission Council, 1998.

[222] Brett Lantz. *Machine learning with R.* Packt publishing ltd, 2013.

[223] Qichao Que and Mikhail Belkin. Back to the future: Radial basis function networks revisited. In Arthur Gretton and Christian C. Robert, editors, *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 1375–1383, Cadiz, Spain, 09–11 May 2016. PMLR.

[224] Vidya Muthukumar, Adhyyan Narang, Vignesh Subramanian, Mikhail Belkin, Daniel Hsu, and Anant Sahai. Classification vs regression in overparameterized regimes: Does the loss function matter? *arXiv preprint arXiv:2005.08054*, 2020.

[225] Like Hui and Mikhail Belkin. Evaluation of neural architectures trained with square loss vs cross-entropy in classification tasks. *arXiv preprint arXiv:2006.07322*, 2020.

[226] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32:8026–8037, 2019.

[227] Alekh Agarwal, Miroslav Dudik, and Zhiwei Steven Wu. Fair regression: Quantitative definitions and reduction-based algorithms. In *International Conference on Machine Learning*, pages 120–129, 2019.

[228] Junpei Komiyama, Akiko Takeda, Junya Honda, and Hajime Shimao. Nonconvex optimization for regression with fairness constraints. In *International conference on machine learning*, pages 2737–2746, 2018.

[229] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1171–1180. International World Wide Web Conferences Steering Committee, 2017.

[230] Tatsunori B Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. Fairness without demographics in repeated loss minimization. *arXiv preprint arXiv:1806.08010*, 2018.

[231] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. Fairness through causal awareness: Learning causal latent-variable models for biased data. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 349–358, 2019.

[232] Fereshte Khani and Percy Liang. Noise induces loss discrepancy across groups for linear regression. *arXiv preprint arXiv:1911.09876*, 2019.

[233] Fereshte Khani and Percy Liang. Removing spurious features can hurt accuracy and affect groups disproportionately. *arXiv preprint arXiv:2012.04104*, 2020.

[234] Toon Calders, Asim Karim, Faisal Kamiran, Wasif Ali, and Xiangliang Zhang. Controlling attribute effect in linear regression. In *2013 IEEE 13th international conference on data mining*, pages 71–80. IEEE, 2013.

[235] Kory D Johnson, Dean P Foster, and Robert A Stine. Impartial predictive modeling: Ensuring fairness in arbitrary models. *arXiv preprint arXiv:1608.00528*, 2016.

[236] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research*, page 0049124118782533, 2018.

[237] Evgenii Chzhen, Christophe Denis, Mohamed Hebiri, Luca Oneto, and Massimiliano Pontil. Fair regression via plug-in estimator and recalibration with statistical guarantees. *Advances in Neural Information Processing Systems*, 33, 2020.

[238] Jérémie Bigot. Statistical data analysis in the wasserstein space. *ESAIM: Proceedings and Surveys*, 68:1–19, 2020.

[239] Jérémie Mary, Clément Calauzènes, and Noureddine El Karoui. Fairness-aware learning for continuous attributes and treatments. In *International Conference on Machine Learning*, pages 4382–4391, 2019.

[240] Anna Zink and Sherri Rose. Fair regression for health care spending. *Biometrics*, 76(3):973–982, 2020.

[241] Harikrishna Narasimhan, Andrew Cotter, Maya R Gupta, and Serena Wang. Pairwise fairness for ranking and regression. In *AAAI*, pages 5248–5255, 2020.

[242] Evgenii Chzhen, Christophe Denis, Mohamed Hebiri, Luca Oneto, and Massimiliano Pontil. Fair regression with wasserstein barycenters. *arXiv preprint arXiv:2006.07286*, 2020.

[243] Harrison Edwards and Amos Storkey. Censoring representations with an adversary. *arXiv preprint arXiv:1511.05897*, 2015.

[244] Alex Beutel, Jilin Chen, Zhe Zhao, and Ed H Chi. Data decisions and theoretical implications when adversarially learning fair representations. *arXiv preprint arXiv:1707.00075*, 2017.

[245] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. Learning adversarially fair and transferable representations. *arXiv preprint arXiv:1802.06309*, 2018.

[246] Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard Zemel. The variational fair autoencoder. *arXiv preprint arXiv:1511.00830*, 2015.

[247] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[248] Constantinos Daskalakis and Ioannis Panageas. The limit points of (optimistic) gradient descent in min-max optimization. In *Advances in Neural Information Processing Systems*, pages 9236–9246, 2018.

[249] Sarah Bird, Miro Dudík, Richard Edgar, Brandon Horn, Roman Lutz, Vanessa Milan, Mehrnoosh Sameki, Hanna Wallach, and Kathleen Walker. Fairlearn: A toolkit for assessing and improving fairness in AI. Technical Report MSR-TR-2020-32, Microsoft, May 2020.

[250] Belisario Panay, Nelson Baloian, José A Pino, Sergio Peñafiel, Horacio Sanson, and Nicolas Bersano. Predicting health care costs using evidence regression. In *Multidisciplinary Digital Publishing Institute Proceedings*, volume 31, page 74, 2019.

[251] Markus Hittmeir, Andreas Ekelhart, and Rudolf Mayer. Utility and privacy assessments of synthetic data for regression tasks. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 5763–5772. IEEE, 2019.

[252] Yangchen Pan, Ehsan Imani, Amir-massoud Farahmand, and Martha White. An implicit function learning approach for parametric modal regression. *Advances in Neural Information Processing Systems*, 33, 2020.

[253] Kristian Lum and William Isaac. To predict and serve? *Significance*, 13(5):14–19, 2016.

[254] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. In *Artificial Intelligence and Statistics*, pages 962–970, 2017.

[255] Ray Jiang, Aldo Pacchiano, Tom Stepleton, Heinrich Jiang, and Silvia Chiappa. Wasserstein fair classification. In *Uncertainty in Artificial Intelligence*, pages 862–872. PMLR, 2020.

[256] Lydia T Liu, Sarah Dean, Esther Rolf, Max Simchowitz, and Moritz Hardt. Delayed impact of fair machine learning. In *International Conference on Machine Learning*, pages 3150–3158, 2018.

[257] Avrim Blum, Suriya Gunasekar, Thodoris Lykouris, and Nati Srebro. On preserving non-discrimination when combining expert advice. In *Advances in Neural Information Processing Systems*, pages 8376–8387, 2018.

[258] Yahav Bechavod, Katrina Ligett, Aaron Roth, Bo Waggoner, and Steven Z Wu. Equal opportunity in online classification with partial feedback. In *Advances in Neural Information Processing Systems*, pages 8974–8984, 2019.

[259] Matthew Joseph, Michael Kearns, Jamie H Morgenstern, and Aaron Roth. Fairness in learning: Classic and contextual bandits. In *Advances in Neural Information Processing Systems*, pages 325–333, 2016.

[260] Vishakha Patil, Ganesh Ghalme, Vineet Nair, and Y Narahari. Achieving fairness in the stochastic multi-armed bandit problem. In *AAAI*, pages 5379–5386, 2020.

[261] Yifang Chen, Alex Cuellar, Haipeng Luo, Jignesh Modi, Heramb Nemlekar, and Stefanos Nikolaidis. Fair contextual multi-armed bandits: Theory and experiments. In *Conference on Uncertainty in Artificial Intelligence*, pages 181–190. PMLR, 2020.

[262] Lily Hu and Yiling Chen. A short-term intervention for long-term fairness in the labor market. In *Proceedings of the 2018 World Wide Web Conference*, pages 1389–1398, 2018.

[263] Danielle Ensign, Sorelle A Friedler, Scott Neville, Carlos Scheidegger, and Suresh Venkatasubramanian. Runaway feedback loops in predictive policing. In *Conference on Fairness, Accountability and Transparency*, pages 160–171, 2018.

[264] Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. Fairness without demographics in repeated loss minimization. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1929–1938, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.

[265] Hussein Mouzannar, Mesrob I Ohannessian, and Nathan Srebro. From fair decision making to social equality. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 359–368, 2019.

[266] Xueru Zhang, Mohammadmahdi Khaliligarekani, Cem Tekin, et al. Group retention when using machine learning in sequential decision making: the interplay between user dynamics

and fairness. In *Advances in Neural Information Processing Systems*, pages 15269–15278, 2019.

[267] Hadi Elzayn, Shahin Jabbari, Christopher Jung, Michael Kearns, Seth Neel, Aaron Roth, and Zachary Schutzman. Fair algorithms for learning in allocation problems. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 170–179, 2019.

[268] Lydia T Liu, Ashia Wilson, Nika Haghtalab, Adam Tauman Kalai, Christian Borgs, and Jennifer Chayes. The disparate equilibria of algorithmic decision making when individuals invest rationally. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 381–391, 2020.

[269] Alfred Müller. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 29(2):429–443, 1997.

[270] Daniela Pucci De Farias. *The linear programming approach to approximate dynamic programming: Theory and application.* PhD thesis, stanford university, 2002.

[271] Daniela Pucci De Farias and Benjamin Van Roy. The linear programming approach to approximate dynamic programming. *Operations research*, 51(6):850–865, 2003.

[272] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[273] Guanjie Zheng, Fuzheng Zhang, Zihan Zheng, Yang Xiang, Nicholas Jing Yuan, Xing Xie, and Zhenhui Li. Drn: A deep reinforcement learning framework for news recommendation. In *Proceedings of the 2018 World Wide Web Conference*, pages 167–176, 2018.

[274] Sebastian Thrun and Anton Schwartz. Issues in using function approximation for reinforcement learning. In *Proceedings of the Fourth Connectionist Models Summer School*, pages 255–263. Hillsdale, NJ, 1993.

[275] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

[276] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 5769–5779, Red Hook, NY, USA, 2017. Curran Associates Inc.

[277] Stephen J Wright. Coordinate descent algorithms. *Mathematical Programming*, 151(1):3–34, 2015.

[278] Haokun Chen, Xinyi Dai, Han Cai, Weinan Zhang, Xuejian Wang, Ruiming Tang, Yuzhou Zhang, and Yong Yu. Large-scale interactive recommendation with tree-structured policy gradient. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3312–3320, 2019.

[279] Sijin Zhou, Xinyi Dai, Haokun Chen, Weinan Zhang, Kan Ren, Ruiming Tang, Xiuqiang He, and Yong Yu. Interactive recommender system via knowledge graph-enhanced reinforcement learning. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 179–188, 2020.

[280] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[281] Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International Conference on Machine Learning*, pages 22–31. PMLR, 2017.

[282] Shahin Jabbari, Matthew Joseph, Michael Kearns, Jamie Morgenstern, and Aaron Roth. Fairness in reinforcement learning. In *International Conference on Machine Learning*, pages 1617–1626, 2017.

[283] Min Wen, Osbert Bastani, and Ufuk Topcu. Fairness with dynamics. *arXiv preprint arXiv:1901.08568*, 2019.

[284] Umer Siddique, Paul Weng, and Matthieu Zimmer. Learning fair policies in multi-objective (deep) reinforcement learning with average and discounted rewards. In *International Conference on Machine Learning*, pages 8905–8915. PMLR, 2020.

[285] John A Weymark. Generalized gini inequality indices. *Mathematical Social Sciences*, 1(4):409–430, 1981.

[286] Yingqiang Ge, Shuchang Liu, Ruoyuan Gao, Yikun Xian, Yunqi Li, Xiangyu Zhao, Changhua Pei, Fei Sun, Junfeng Ge, Wenwu Ou, et al. Towards long-term fairness in recommendation. *arXiv preprint arXiv:2101.03584*, 2021.

[287] Philip S Thomas, Bruno Castro da Silva, Andrew G Barto, Stephen Giguere, Yuriy Brun, and Emma Brunskill. Preventing undesirable behavior of intelligent machines. *Science*, 366(6468):999–1004, 2019.

[288] Xueru Zhang, Ruibo Tu, Yang Liu, Hedvig Kjellstrom, Kun Zhang, Cheng Zhang, et al. How do fair decisions fare in long-term qualification? *Advances in Neural Information Processing Systems*, 33, 2020.

[289] Swati Gupta and Vijay Kamble. Individual fairness in hindsight. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pages 805–806, 2019.

[290] Sampath Kannan, Aaron Roth, and Juba Ziani. Downstream effects of affirmative action. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 240–248, 2019.

[291] Hoda Heidari and Andreas Krause. Preventing disparate treatment in sequential decision making. In *IJCAI*, pages 2248–2254, 2018.

[292] Hoda Heidari, Vedant Nanda, and Krishna Gummadi. On the long-term impact of algorithmic decision policies: Effort unfairness and feature segregation through social learning. In *International Conference on Machine Learning*, pages 2692–2701, 2019.

[293] Feng Liu, Huifeng Guo, Xutao Li, Ruiming Tang, Yunming Ye, and Xiuqiang He. End-to-end deep reinforcement learning based recommendation with supervised embedding. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 384–392, 2020.

[294] Gabriel Dulac-Arnold, Richard Evans, Hado van Hasselt, Peter Sunehag, Timothy Lillicrap, Jonathan Hunt, Timothy Mann, Theophane Weber, Thomas Degris, and Ben Coppin. Deep reinforcement learning in large discrete action spaces. *arXiv preprint arXiv:1512.07679*, 2015.

[295] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[296] Edward Chou, Josh Beal, Daniel Levy, Serena Yeung, Albert Haque, and Li Fei-Fei. Faster cryptonets: Leveraging sparsity for real-world encrypted inference. *arXiv preprint arXiv:1811.09953*, 2018.

[297] Brett K Beaulieu-Jones, William Yuan, Samuel G Finlayson, and Zhiwei Steven Wu. Privacy-preserving distributed deep learning for clinical data. *arXiv preprint arXiv:1812.01484*, 2018.

[298] Brett K Beaulieu-Jones, Zhiwei Steven Wu, Chris Williams, Ran Lee, Sanjeev P Bhavnani, James Brian Byrd, and Casey S Greene. Privacy-preserving generative deep neural networks support clinical data sharing. *BioRxiv*, page 159756, 2018.

[299] Brij Mohan Lal Srivastava, Aurélien Bellet, Marc Tommasi, and Emmanuel Vincent. Privacy-preserving adversarial representation learning in asr: Reality or illusion? *arXiv preprint arXiv:1911.04913*, 2019.

[300] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

[301] Mauro Ribeiro, Katarina Grolinger, and Miriam AM Capretz. Mlaas: Machine learning as a service. In *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, pages 896–902. IEEE, 2015.

[302] Dominik Maria Endres and Johannes E Schindelin. A new metric for probability distributions. *IEEE Transactions on Information theory*, 2003.

[303] Zhifei Zhang, Yang Song, and Hairong Qi. Age progression/regression by conditional adversarial autoencoder. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5810–5818, 2017.

[304] Proteek Chandan Roy and Vishnu Naresh Boddeti. Mitigating information leakage in image representations: A maximum entropy approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2586–2594, 2019.

[305] Ji Wang, Jianguo Zhang, Weidong Bao, Xiaomin Zhu, Bokai Cao, and Philip S Yu. Not just privacy: Improving performance of private deep learning in mobile cloud. In *KDD*, 2018.

[306] Clément Feutry, Pablo Piantanida, Yoshua Bengio, and Pierre Duhamel. Learning anonymized representations with adversarial neural networks. *arXiv preprint arXiv:1802.09386*, 2018.

[307] Nils Homer, Szabolcs Szelinger, Margot Redman, David Duggan, Waibhav Tembe, Jill Muehling, John V Pearson, Dietrich A Stephan, Stanley F Nelson, and David W Craig. Resolving individuals contributing trace amounts of dna to highly complex mixtures using high-density snp genotyping microarrays. *PLoS genetics*, 4(8), 2008.

[308] Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. On learning invariant representations for domain adaptation. In *International Conference on Machine Learning*, pages 7523–7532, 2019.

[309] Chris Calabro. *The exponential complexity of satisfiability problems*. PhD thesis, UC San Diego, 2009.

[310] Jianhua Lin. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991.

[311] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[312] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 220–229, 2019.