Building Robust Text Classification Models under Provenance Shift: Methods of Adjustment and a Framework for Evaluation

Xiruo Ding

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2024

Reading Committee:
Trevor A. Cohen, Chair
Meliha Yetisgen
Serguei Pakhomov

Program Authorized to Offer Degree:
Biomedical Informatics and Medical Education

University of Washington

Abstract

Building Robust Text Classification Models under Provenance Shift:
Methods of Adjustment and a Framework for Evaluation

Xiruo Ding

Chair of the Supervisory Committee:
Trevor A. Cohen
Department of Biomedical Informatics and Medical Education

Machine learning and deep learning have consistently delivered groundbreaking contributions across a wide range of disciplines. Biomedical research also benefits from such methods at every scale, from the molecular level (such as in structural biology) to the population level. Many learning algorithms require adequate amount of data to fully train a model, and also assume no difference between the training data and test data. This may be achievable for problems in the general domain. For example, large datasets exist for computer vision (CIFAR-10, CIFAR-100, etc.) and natural language processing (Amazon Reviews, Yelp Reviews, Wikipedia, etc.). However, in biomedical research, it is challenging to collect data on the order of millions when high quality patient-related data are needed. One feasible solution is to combine data from several sites. This approach can also increase the variety of data, thus helping to build robust models. However, the models trained on such settings may recognize spurious correlations between data provenance and the target of interest. Naturally, this can also happen when subpopulations exist, each of which has different characteristics. This effect can be detrimental when model is deployed in a new setting where provenance composition shifts.

This thesis builds on such scenarios where confounding by provenance and provenance shift are the main concerns. Formal definitions and a simulation framework are introduced

first. Building upon these, the aim is to find useful ways to build models that are robust to such provenance shift while maintaining reasonable performance. This goal is attained through different means, from statistical adjustment through distribution adjustment to architecture adjustment. Two key contributions are: (1) a framework for experimentally simulating different degrees of provenance shift and evaluating model robustness and performance; (2) several effective adjustment methods to build more robust models.

The framework and adjustment methods were tested on three datasets, two from the biomedical domain and one from the general domain, to validate their generalizability. Results indicate that the methods, focusing on different aspects of the modeling procedure, can help improve model robustness, and that model performance can also be improved when provenance shift is extreme.

This work contributes to our understanding of how provenance shift impacts model performance, and provides methods to develop more robust models that can withstand the challenges posed by such shifts, ultimately leading to algorithms that are more reliable and trustworthy, and less biased.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I want to begin with my deep gratitude to my advisor, Trevor Cohen, for guiding me through such a long journey of my doctoral life. He has always been insightful in identifying the problem and providing directions, during times when I came to a seemingly dead end. I deeply appreciate his patience for going through my endless Slack messages (any time of the day) and reading and editing drafts that are many steps away from a polished work. I have learned a lot under his mentorship and will forever be grateful for this wonderful relationship.

I would love to thank my committee members, Serguei Pakhomov and Meliha Yetişgen. During countless meetings, they are always helpful with their insights, guidance, and support, that together solidify my ideas and move them forward.

I want to thank wonderful collaborators, Justin Tauscher and Dror Ben-Zeev, from the Behavioral Research in Technology and Engineering (BRiTE) Center in the University of Washington for insightful clinical guidance across multiple projects, providing valuable data, and helping to shape this current work. This work is also made possible by other great collaborators, and I would like to thank you, Zhecheng Sheng, Brian Hur, Hannah A Burkhardt, Kevin Lybarger, Justin Mower, and Devika Subramanian.

I want to thank my friends, Yue Guo, Oliver Li, Weipeng Zhou, Nick Reid, Mu Yang, from the Department. Your accompany makes the tough graduate school life a fun experience for me. I also want to thank friends outside school, whose constant support is key for making life colorful.

Lastly, I am incredibly thankful for my wife, Hongyuan Zhang. When I need someone to talk to, share my worries with, or simply unwind, you were always there – providing a

listening ear, wise counsel, and a calming presence that soothed my mind and heart. I offer my deepest, abiding gratitude to my parents and extended family for your unwavering love, suport, and guidance througout the years since I was just a kid. They have shaped me into the person I am today, and I'm forever grateful.

This work is kindly supported by several funding agencies during different periods, including the UW Medicine Garvey Institute for Brain Health Solutions; National Institute of Mental Health grant (R56MH109554); U.S. National Library of Medicine Grant (R01LM014056). I also give thanks for University of Washington eScience Institute for kindly awarding us Azure Credit Awards, and the UW Research Computing Club, both of which provided key computing resources to make this work possible.

DEDICATION

To my family.

Chapter 1
INTRODUCTION

1.1 Problem

Machine learning and deep learning models have been successfully applied for predictive modeling in many domains [1]–[3]. This includes their application to biomedical problems, including those at the molecular level, such as identifying potential new antibiotics [4] and predicting protein structure [5], [6]; at the patient level, such as electronic health record phenotyping [7], [8] and cancer diagnosis [9]; and at the institutional and policy levels, such as predicting demand at emergency departments [10]. Such methods are also well-established in clinical natural language processing (NLP) [11], and the successes of deep neural network models have generated interest and enthusiasm for their applications in biomedical research [12].

A key requirement for successfully training and then deploying such systems is to use adequate amounts of diverse training data. However, this need for quantity and diversity presents a considerable challenge for clinical data collection from an individual institute [13]. One feasible solution is to integrate data from multiple sources. In addition to supporting the development of robust models, this can also increase the diversity of the training population, improving generalizability.

However, combining data from different sites may introduce bias and highly parameterized models may learn the source of a data element, and use this information to inform its predictions (which may well be correct within that particular dataset but are based on an incidental feature and therefore will not generalize to other data with a different distribution). Recent work in NLP has drawn attention to the potential of confounding variables – variables

that influence both the predictors (incoming text) and outcomes (an assigned category) of a text categorization system – to harm model performance at the point of deployment [14], [15]. Essentially, the concern is that when the distribution of category assignment in the training data varies in accordance with the value of a confounding variable (e.g. gender of the author), the model will erroneously learn to make category assignments using words that are associated with this variable (e.g. gender differences in pronoun use) [16], rather than words that meaningfully inform the categorization task at hand. This bias, where spurious associations between the source (provenance) and target (for prediction - e.g. assigning a diagnosis) distributions are learned by models trained on such datasets, is called *confounding by provenance* [17]–[20].

Such bias is detrimental when the composition of datasets with differing provenance at the deployment time shifts away from the composition in the training set. When source-specific data distributions differ at deployment, this may harm model performance, which is important especially for high-stakes applications in healthcare (such as mental health diagnosis classification and cancer prognosis prediction). The idea of confounding by provenance is related to some existing research areas, specifically confounding effects in causal inference, class imbalance, and dataset shift. However, this work fills a gap in literature from these areas, within which provenance is not explicitly considered as a confounding variable. However, when taking provenance into account, some methods studied in these areas are applicable with modifications. We evaluate their usefulness under the developed framework for confounding by provenance.

This work intends to address the issue of confounding by provenance with natural language data from high-stakes clinical settings. The range of available models encompasses a broad spectrum, from fundamental statistical approaches to modified neural networks. In the area of NLP, rapid advancements in research and engineering have led to many new models within relatively short timeframes. Model sizes also vary greatly, ranging from hundreds of parameters to billions, or even tens of billions. It is notable that such large models are usually out of reach for small health institutions with limited computing resources. This

work aims to explore the usability, generalizability, robustness, and performance of models varying in size, with the ultimate goal of identifying simple and efficient methods for those models.

We first establish a simulation and evaluation framework for this specific problem such that different degrees of provenance shift can be manually manipulated in experimental settings. Following this, a wide range of adjustment methods, from statistical adjustment, to distributional adjustment and architecture adjustment in neural networks, are developed and evaluated in the proposed framework. The ultimate goal is to enhance the reliability and robustness of machine learning and deep learning methods for clinical NLP.

1.2 Aims of This Work

This work is driven by two key questions: how can we simulate different degrees of provenance shift, and what strategies can be employed to construct robust and effective models that remain viable in the presence of such changes? My specific aims are designed to answer these two questions. A high-level overview of the research plan is shown in Figure 1.1.

Aim 1: Define an evaluation framework for confounding by provenance

As a first step, a framework is developed to (1) manipulate data distributions to introduce different degrees of provenance shifts; (2) quantify the degree of shift introduced; and (3) evaluate model performance under different degrees of shift. A range of clinical datasets are used to evaluate mitigation strategies under this framework, which provides the fundamental evaluation component for current work. This framework extends the evaluation approach proposed by Landeiro & Culotta, who examined the utility of adapting Pearl's backdoor adjustment method to text categorization with logistic regression in the context of confounding shift [15]. In our work, Backdoor Adjustment is adapted for the provenance shift issue and further extended with different text representation methods. This method serves as a statistical adjustment and is evaluated within the framework.

Aim 2: Mitigation through distribution adjustments

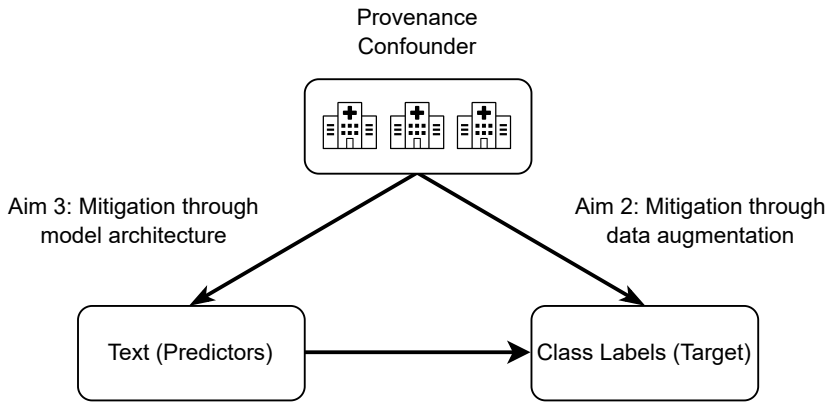


Figure 1.1: Overview of the aims. **Aim 1** (not shown in the figure) provides an evaluation framework for robustness to confounding by provenance. **Aim 2**, mitigation through distribution adjustments, targets eliminating hospital-wide differences in class label distribution, thereby eliminating the causal link between site and target label prevalence that affects distributions directly. **Aim 3**, mitigation through model modification, focuses on eliminating the ability of models to perceive hospital-specific language differences, thereby eliminating the link between site and text representation.

Data augmentation has been widely used in computer vision and is gaining attention in NLP. For the purpose of this work, the key idea of applying data augmentation is to purposely manipulate class distributions in training data to mitigate the effect of provenance shift. I adapt and evaluate data augmentation methods originating in computer vision for NLP, specifically rule-, interpolation-, and model-based techniques. Their utility as a means to mitigate confounding by provenance is tested in both logistic regression and deep learning models, using the evaluation framework developed in Aim 1.

Aim 3: Mitigation through model modifications

Deep learning approaches have been widely adopted in clinical natural language processing to improve performance. Notably, adversarial training and domain adaptation techniques have gained significant attention in recent years as promising strategies for facilitating learn-

ing of domain-invariant representations. These methods were conceived with the aim to enable models to generalize well across different datasets, medical contexts, and even languages. I take two methodological approaches. First, I propose a novel approach that builds upon the concept of task vectors. Specifically, I adapt the idea of performing arithmetic operations on task vectors derived from the differences between fine-tuned models and pretrained models. Second, I adapt robust learning strategies from the field of domain adaptation, with the aim to build robust models under provenance shift. I employ the evaluation methodology from Aim 1 to test whether these methods align with the broader research objectives and standards.

1.3 Innovation

Developing an evaluation framework is the first step proposed, as this will provide a grounding point for method development. The first adjustment method implemented is through Backdoor Adjustment, based on the work of Landeiro and Culotta [15]. I extend this method in two ways: (1) sentence embeddings (from Sentence-BERT) are introduced; and (2) some constraints have been relaxed. As one example of the latter extension, previous work [15] enforces that the prevalence of each value of the confounding variable is held constant across training and testing set. This constraint limits the range of distributional shifts that can be synthetically imposed for evaluation purposes, and it is unrealistic to force data proportions from all sources to be the same across the training and testing set.

Data augmentation is widely used in computer vision (CV) [21], [22]. However, little research has been done on utilizing this technique to adjust distributions to compensate for provenance shift. With a variety of proposed methods in CV, I adapt transferable ones for NLP. Furthermore, there is no established consensus concerning the optimal class distribution in the presence of an imbalance in distributions. For text classification, I propose the DistMatch framework, which promotes a "balanced" distribution across provenances and combine this framework with data augmentation and simple resampling techniques.

Besides the DistMatch framework, I propose to modify deep learning architecture, with

one goal being the learning of domain-invariant representations. Domain-invariant representations, in the scenario of confounding by provenance, refer to (text) encodings that show no difference across sources. Several proposed methods, such as Domain Adversarial Neural Network (DANN) [23] and domain alignment through Maximum Mean Discrepancy (MMD) [24], have been studied in the areas of domain adaptation and out-of-distribution generalization, that assume domain-level difference between training and testing class distributions. This could be considered as an extreme case of confounding by provenance, where the latter still assumes mixed proportions from different sources. Though these methods have proven utility for domain adaptation, their utility as means to compensate for confounding by provenance has yet to be evaluated. I also propose Dominance-Aligned Polarized Provenance Effect Reduction, which is based on the idea of task vectors and serves as a post-hoc editing of model weights, in order to remove undesired correlations with provenance.

1.4 Overview

The dissertation is organized according to the following outline:

Chapter 2 introduces definitions of confounding by provenance and provenance shift, which are key concepts in this work. Background information is provided.

Chapter 3 develops the simulation and evaluation framework that is used across experiments in this work. This framework simulates varying degrees of provenance shift in the experimental settings.

Chapter 4 introduces three datasets which are main training materials for this work. Their characteristics, collection methods, and classification goals are described in detail.

Chapter 5 presents a modification of Backdoor Adjustment for the problem of confounding by provenance. This method provides a statistical adjustment that builds upon logistic regression models.

Chapter 6 explores different ways of performing data augmentation for NLP tasks when data imbalance is present. A framework for adjusting training distributions is introduced.

Chapter 7 focuses on the manipulation of hidden spaces for provenance shift adjustment.

First, a novel method through post-hoc manipulation of hidden spaces in neural networks is introduced. This method is model-agnostic and can work on models of all sizes. It also works well with low rank adaptation techniques for partially training large models. Next, robust learning approaches, that adjust hidden spaces through the training process, are discussed and tested.

Chapter 8 summarizes results from all methods used in this work. Models are compared across three dimensions: robustness, worst-case performance, and best-case performance. The chapter provides a holistic view of pros and cons for adjustment methods.

Chapter 9 summarizes this work's key findings and primary contributions, and outlines future research directions.

Chapter 2

BACKGROUND AND DEFINITIONS

Two fundamental concepts, confounding by provenance and provenance shift, form the foundation for this work. They fall within broader research topics of distribution shift and confounding. This chapter serves as an introduction to various types of distribution shift, which have important connections to the current study. Establishing these relationships provides a contextual framework to situate my work within the broader field, where I formally define the key terms used in the current work.

2.1 Common Types of Distribution Shift

Distribution shift is extensively discussed in the book *Dataset Shift in Machine Learning* by Quinonero-Candela, Sugiyama, Schwaighofer, *et al.* [25]. The authors categorized different forms of shift into several groups, as discussed in the first chapter of the book [26]: Covariate Shift; Prior Probability Shift (or Target Shift); Domain Shift (changes in measurement); Source Component Shift (changes in strength of contributing components). Broadly speaking, it can also include Sample Selection Bias (distributions differ as a result of sampling process); Imbalanced Data. Besides these, other related shifts have been discussed recently, especially for methods in domain adaptation: Conditional Shift, when $P(Y)$ remains the same and $P(X|Y)$ changes [27], [28]; Concept Drift, when $P_{train}(Y|X) \neq P_{test}(Y|X)$ [29], [30].

I will proceed to discuss several kinds of shifts that are highly related to the current work.

2.1.1 Covariate Shift

Covariate shift occurs when the following requirements are met:

$$P_{train}(X) \neq P_{test}(X) \tag{2.1}$$

$$P_{train}(Y|X) = P_{test}(Y|X) \tag{2.2}$$

where P_{train} and P_{test} represents distributions over the training and test set, separately. The first statement represents the covariate shift between training and testing scenarios. The second statement, originating from the causal model $P(Y|X)P(X)$ (illustrated in Figure 2.1), indicates that the target (Y) distribution solely depends on covariates, and the relationship between X and Y does not change [31]–[33]. For example, the relationship between smoking behavior (X) and developing lung cancer (Y) is established to be causal [34]. At some point, if a smoking ban is enforced, smoking behavior (X) will change. However, this causal relationship will remain the same, thus $P(Y|X)$ is unchanged.

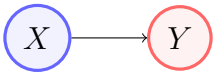


Figure 2.1: Covariate shift.

2.1.2 Prior Probability Shift (or Target Shift)

Prior probability shift, or Target shift, happens when:

$$P_{train}(Y) \neq P_{test}(Y) \tag{2.3}$$

$$P_{train}(X|Y) = P_{test}(X|Y) \tag{2.4}$$

The second statement is a commonly used assumption for prior probability shift [35], based on the causal model demonstrated in Figure 2.2. For example, we take X to be symptoms like fever or cough and Y to be contracting influenza. The prevalence of influenza

will change over time, e.g., increasing during flu season (Statement (1)). However, the dependence of symptoms of fever or cough on whether a person contracts influenza or not remains the same, given all other conditions are unchanged (Statement (2)). Nonetheless a model trained during flu season may overestimate its probability at other times.

Figure 2.2: Prior probability shift, also known as Target Shift.

2.1.3 Concept Drift

Concept drift has been discussed in literature with different definitions, based on sources of the shift. Gama, Žliobaitė, Bifet, *et al.* [36] define two categories:

- *Real concept drift*: $P_{train}(Y|X) \neq P_{test}(Y|X)$, regardless of $P(X)$.
- *Virtual concept drift*: $P_{train}(X) \neq P_{test}(X)$, while $P(Y|X)$ is unchanged.

It is acknowledged that *Virtual Concept Drift* shares the same form as covariate shift. Lu, Liu, Dong, *et al.* [37] further categorized the problem into three forms, based on if either one or both of $P(X)$ and $P(Y|X)$ shifts are present.

2.1.4 Source Component Shift

In the general sense, source component shift deals with the scenario where data are collected from a variety of sources, each of which has its own characteristics, and these components vary between training and test times. For example, voting expectations are dependent on type of work, and different places across the country have very different distributions of jobs [26]. This effect is illustrated in Figure 2.3.

There are three types of source component shift:

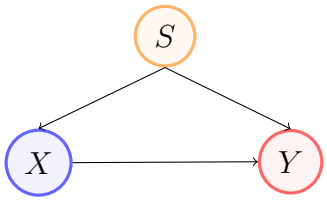


Figure 2.3: Source Component Shift.

- *Mixture Component Shift*: samples come from several sub-populations, and targets of interest may vary. It is usually assumed that $P(X, Y|S)$ remains the same across scenarios, but $P(S)$ shifts.

- *Factor Component Shift*: samples are subject to uncontrolled factors, which vary in different scenarios. It is assumed that the factors (components) are the same, however, the strength of these factors changes.

- *Mixing Component Shift*: targets are aggregated over a potentially varying population. This is the same as Mixture Component Shift, with the only difference being the target of interest being aggregated.

2.1.5 Class Imbalance

The problem of class imbalance concerns cases where the class (i.e. target) distribution is skewed. For example, in the classification of stage-I acute kidney injury using the MIMIC-III dataset, a group of authors found positive:control ratio to be around 1:10 when using 3-day data and 1:15 when using 4-day data for prediction in next 7 days [38]. This is the case for many rare disease classification problems, as well as for other applications outside of the biomedical domain (e.g. fraud detection with 6% positive rate [39]). Such imbalanced data distributions bring up downstream issues of small sample size, class separability, and within-class concepts [40]. Class separability can be problematic when discriminative patterns across classes do not exist, and it can be further exacerbated by small sample sizes. It is usually the

case for the classification problem that a single class is a combination of several subclasses, or subconcepts. When data scarcity influences sample sizes for minority subclasses, it is referred as within-class imbalance. This phenomena increases the learning concept complexity and is sometimes hard to detect [41].

A range of class-balancing techniques have been applied in biomedical research:

1. Traditional techniques:

   (a) Random under-sampling strategies

   (b) Cluster Centroids: this under-samples from the majority class by identifying K cluster centroids. Representative subsets from the majority class are then mixed with the target class for classification [42].

   (c) Instance hardness thresholding: it is an undersampling technique. A classifier is first trained within the majority class and samples that are hard to classify (than some threshold) are removed [43].

   (d) One-sided selection method: this combines Tomek links for under-sampling and 1-NN rule to remove borderline and noisy examples [44].

   (e) Edited Nearest Neighbor Rule (ENN): this method removes an example whose label is different from the class of at least two of its three nearest neighbors [45].

   (f) Synthetic Minority Oversampling Technique (SMOTE): SMOTE over-samples from the minority class by generating synthetic examples from the feature space of that class. Specifically authors proposed using line segments between two selected samples to generate a new one [46].

2. The Case-Control Matching Strategy: for example, the Charlson comorbidity index [47] is a useful indicator for matching in clinical settings [40].

3. Individualized Predictive Modeling: personalized models are customized for each individual by sampling similar examples to that individual from both positive training set

and control training set [48].

4. Cost-sensitive models: the key idea is to impose different degrees of cost onto different classes, usually a higher penalty for misclassification on rare classes [49].

5. One-class learning: class-level models are trained on examples of one class at a time (no negative examples present). This framework has been studied through application of two classifiers: a neural network [50] and an SVM [51].

On top of these foundational ideas, some combinations have also been proposed for better performance, such as SMOTE and Tomek links, SMOTE and ENN [52], and SMOTE and under-sampling [46].

For the above approaches that are operated on the data sample level (such as random under-sampling, instance hardness thresholding, and case-control matching), one pending question concerns optimal class distribution in terms of performance under specific testing scenarios and model's robustness to different degrees of shift. This determines to what degree to perform resampling. Weiss and Provost showed empirically that a balanced class distribution generally performs well, although may not be optimal, which in turn is dependent on datasets [53]. This will be further discussed in Chapter 6.

2.2 Confounding by Provenance

2.2.1 Background: Confounding Effects

Confounding effects are frequent and have been widely studied. In statistics, a confounder is a variable that affects both the independent variable and the dependent variable. For example, when investigating the effect of chronic kidney disease on mortality, age is a recognized risk factor which affect both risks of developing chronic kidney disease and mortality [54], [55]. Other factors, such as severity (severity of illness, such that more severe patients are likely to receive more intensive treatment), insurance status, etc., have also been recognized as confounders in clinical settings [56]. Outside of clinical research, typical examples include:

movie reviews can be confounded by genre, text-based geolocation identification by whether tweets were GPS-tagged [57], and volunteered geographic information (such as in Twitter, Flickr, and Forusquare check-ins) by urban vs rural users [58].

Randomized controlled trials (RCTs) with careful design are the gold standard for controlling known confounding factors [59], but this option is sometimes unnecessary, inappropriate, impossible, or inadequate [60]. According to Kyriacou and Lewis [56], commonly-used approaches include: (1) study design procedures: randomization, restriction, and matching; (2) statistical procedures: stratified analyses, regression, and propensity scoring. For propensity scores, there are four general ways in which these can be used [61]: propensity score matching, stratification on the propensity score, covariate adjustment using the propensity score, and inverse probability of treatment weighting using the propensity score. In the book *Causality: models, reasoning, and inference*, Pearl presents a universal way of adjusting for confounders, the Back-Door Adjustment method (Theorem 3.3.2) [62]. This also provides the baseline method for our current work.

2.2.2 Confounding by Provenance

Confounding by provenance was observed in previous work when data from multiple sources were combined together for model development [17], [18]. The effect is illustrated by the Directed Acyclic Graph (DAG) in Figure 2.4, which was adapted from [15] and first presented in [19]. Take text classification as an example, where text features vary by provenance (source), for example different dialects may be spoken across study sites, or different headers may be used in the clinical notes. In addition, the prevalence of the outcome of interest (e.g., cognitive distortions, or substance abuse) may vary across the samples drawn from each site to serve as training data. Both these text features (Arrow 3 in Figure 2.4) and the distribution of positive examples from each source will affect the prediction of the outcome of interest. This occurs because the distributions of language and labels from distinct data sources will affect both the text concerned (the predictors) (Arrow 1 in Figure 2.4) and the frequency (and predicted probability, if the model can recognize provenance) of the primary

target (Arrow 2 in Figure 2.4).

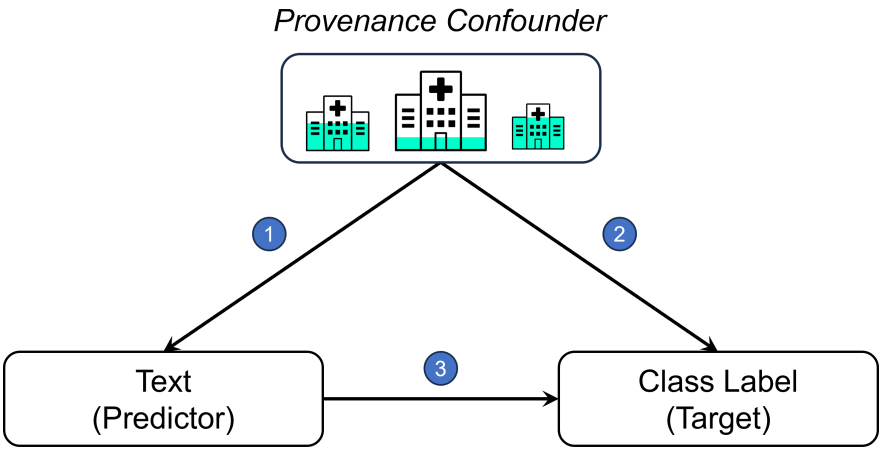


Figure 2.4: An illustration of confounding by provenance in a text classification task, where text provides the predictors. The provenance confounder includes several hospitals, with different positive rates for the primary target, illustrated by opaque shading.

2.3 Provenance Shift

From the discussion above, we learn that confounding by provenance is different from the aforementioned forms of dataset shift, because of the introduced provenance confounding variable Z. In our case it affects the target distribution by definition, and could potentially affect covariates as well. By considering existing methods for dataset shift, I will integrate confounding by provenance into this broader framework.

We define provenance shift as a type of a distribution shift (difference in target label distributions) between the training and test sets. Formally, this can be described as follows:

$$P_{train}(Y|Z) \neq P_{test}(Y|Z) \tag{2.5}$$

where P_{train} and P_{test} stand for the distribution for the training and test set, separately. Y is the variable of main interest. Z is the variable indicating provenance, e.g., different

subpopulations, hospitals or health systems. The deleterious confounding effect of a provenance shift is illustrated in Figure 2.5, where training and testing scenarios are separated to highlight this difference.

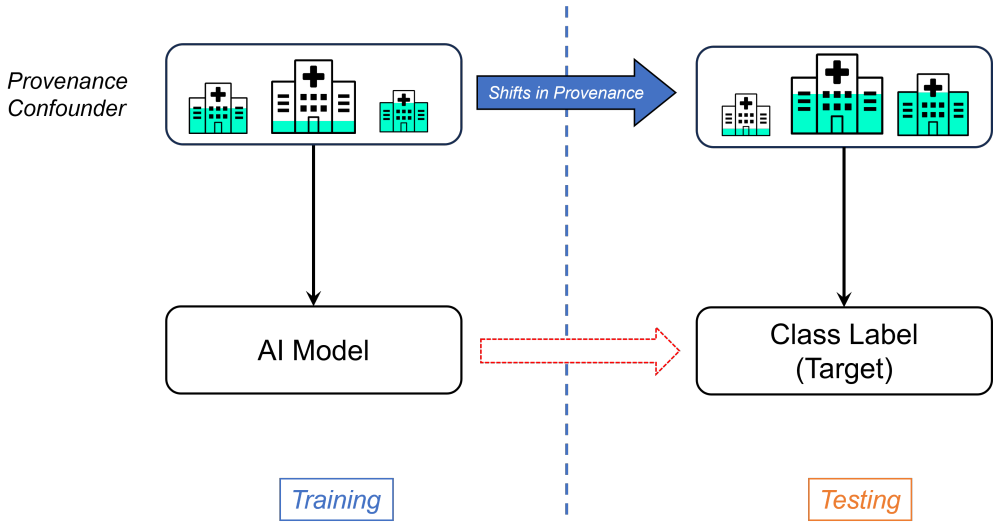


Figure 2.5: Provenance shift with confounding effect. During training (left side), a specific composition of datasets is used for model building. In the testing/deployment period (right side), that composition may change, leading to new distributions of both predictors and the primary target.

I further denote this difference between two conditional distributions as the degrees of the provenance shift. I focus on the setting where the main outcome is binary ($Y = \{0, 1\}$) and only two subpopulations exist to be merged ($Z = \{z_1, z_2\}$). In order to measure the provenance shift, we introduce two variables for measuring the positive rate (of the main outcome) ratios between subpopulations, defined as:

$$\alpha_{train} := \frac{P_{train}(Y = 1|Z = z_2)}{P_{train}(Y = 1|Z = z_1)} \tag{2.6}$$

$$\alpha_{test} := \frac{P_{test}(Y = 1|Z = z_2)}{P_{test}(Y = 1|Z = z_1)} \tag{2.7}$$

The ratio between α_{test} and α_{train} provides a quantitative measurement of the degree of

provenance shift. We take log scales (base of 10) for α's. For example, for a specific training set where $\alpha_{train} = 1$, the α_{test} value different from 1 on either side (towards 0 or infinity) indicates shift from the provenance-specific class distributions in the training set in different directions. $\alpha_{test} = 0.1$ and 10 will be considered as the same amount of shift from $\alpha_{train} = 1$, and $\alpha_{test} = 0.01$ indicates a larger shift than 0.1 and 10 (as illustrated in Figure 2.6).

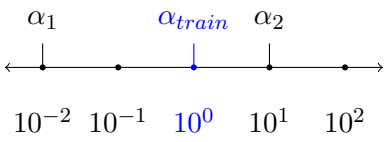


Figure 2.6: Provenance shift indicated by the distance between α_{test} and α_{train}, in log scale. For a given training set with $\alpha_{train} = 1$, different α_{test} values (α_1, α_2) indicate different degrees of provenance shift.

The provenance shift shares a similar framework as source component shift, but the former assumes a shift in $P(Y|Z)$. Additionally, the framework for provenance shift is not strictly causal and focuses more on making predictions. Covariates (texts) are implicitly assumed to remain unchanged within each provenance class between the training and test time.

We can also consider confounding by provenance as a conditional version of data imbalance problem, where $P(X, Y|Z)$, rather than $P(Y)$, is skewed. Some of aforementioned techniques may be of value for this problem also, especially over- and/or under-sampling, synthetic techniques, one-class learning, and cost-sensitive models.

Chapter 3

SIMULATION AND EVALUATION FRAMEWORK

3.1 Introduction

In previous chapters, confounding by provenance and its related shift problem, provenance shift, was defined and illustrated by examples. To measure this shift, two helper variables were introduced, α_{train} and α_{test}. In this chapter, we use these measurements as foundations for building a simulation framework.

Under experimental settings, a simulation framework is needed to introduce different degrees of provenance shift for a holistic evaluation of the developed models. Simulation is especially important when the shift is more extreme, which may or may not happen in a natural setting. For example, population composition may change over time with slight variations in disease prevalence and demographics. It may also change drastically during unexpected events like COVID-19 and roll-outs of vaccinations, as reported by Gray, Navaratnam, Day, *et al.* [63] where they observed changes in the role of ethnicity, sex and comorbidity on in-hospital mortality in England from March to September 2020.

When investigating confounding shift, Landeiro and Culotta [14] used a set of bias terms, $P_{train}(y = 1|z = 1) = b_{train}$, $P_{test}(y = 1|z = 1) = b_{test}$, and their differences to indicate the level of shift. Based on this idea, I formally introduce the simulation framework that provides the capability to simulate different degrees of shift. Moreover, my framework allows for customization, such as removing constraints on the overall positive rates or sub-population compositions. The current focus is on building a simulation framework for scenarios with binary outcomes and two provenance sources. The multi-class case is also briefly discussed. Lastly, we introduce the evaluation framework that was applied across our related studies for investigating models under provenance shift.

3.2 Simulation Framework for Binary Cases

Provenance shift was defined in Chapter 2 as a type of distribution shift between the training and test sets when label distributions among sub-populations differ. Two auxiliary variables were introduced to measure the prevalence differences among subgroups for the training and test set, respectively:

$$\alpha_{train} := \frac{P_{train}(Y = 1|Z = z_2)}{P_{train}(Y = 1|Z = z_1)}$$
$$\alpha_{test} := \frac{P_{test}(Y = 1|Z = z_2)}{P_{test}(Y = 1|Z = z_1)}$$

The difference between α_{train} and α_{test} provides a quantitative measurement of the degree of provenance shift, as discussed in Chapter 2. We use those two variables as building blocks for simulating a wide range of α_{train} and α_{test} combinations.

Table 3.1 shows an example of the 80%:20% train/test split where $\alpha_{train} = \frac{300/600}{20/200} = 5.0$ and $\alpha_{test} = \frac{8/80}{60/120} = 0.2$. This difference in α_{train} and α_{test} indicates a shift in provenance-specific positive rates at test time. The difference of composition, $P_{train}(Z = \textit{Hospital A}) = 0.25$ vs $P_{test}(Z = \textit{Hospital A}) = 0.6$, indicates a shift in sub-population sizes drawn from each source.

		Train	Test
Hospital A	Positive	20	60
	Negative	180	60
Hospital B	Positive	300	8
	Negative	300	72
	Total	800	200

Table 3.1: An example of binary labels (Y) from two hospitals (Z). This follows an 80%:20% split for training/test datasets. In this example, $\alpha_{train} = 5$ and $\alpha_{test} = 0.2$, indicating a severe shift from the training scenario at test time.

3.2.1 With Constrained Positive Rates and Composition

To support the evaluation of robustness to confounding shift, we designed a perturbation framework for distributions so that different degrees of shift could be simulated by sampling. For problems of binary classification with two sub-populations, the following probabilities govern the distribution of $P(Y, Z)$:

$$P_{train}(Y = 1|Z = z_1) \tag{3.1}$$

$$P_{test}(Y = 1|Z = z_1) \tag{3.2}$$

$$P_{train}(Y = 1|Z = z_2) \tag{3.3}$$

$$P_{test}(Y = 1|Z = z_2) \tag{3.4}$$

$$P_{train}(Y = 1) = P_{test}(Y = 1) = Const_y \tag{3.5}$$

$$P_{train}(Z = z_1) = P_{test}(Z = z_1) = Const_z \tag{3.6}$$

$$P_{train}(Z = z_2) = P_{test}(Z = z_2) = 1 - Const_z \tag{3.7}$$

These probabilities determine the source-specific class distributions at training and test time. Of these constraints, (3.5) and (3.6) are held constant across experiments while the

others are varied.

Now, given (3.1), (3.3), (3.6), α_{train} and α_{test}, we can obtain the remaining probabilities for (3.2), (3.4), (3.5). Intuitively, this means all that is needed to set up an evaluation of robustness to confounding shift is to control the training distributions and set a positive ratio between sources for the testing set. This simulates a setting where, in general, we only know our training set, and different degrees of shift are simulated. Following the example in Table 3.1 which represents only one train/test setting, this framework can automatically generate many similar splits with desired configurations. This framework has been used in our paper [19].

3.2.2 With Unconstrained Positive Rates and Composition

The simulation framework in the last section provides tools for systematically manipulating distributions of α_{train} and α_{test}, thus simulating a wide range of degrees of provenance shift. In this framework, constraints were introduced to control factors that are unrelated to confounding by provenance: (1) $P_{train}(Y) = P_{test}(Y)$ (the overall prevalence of the positive class is the same at training and test time); (2) $P_{train}(Z) = P_{test}(Z)$ (the relative contribution of examples from each site to the training and the test set is held constant). However, as simulating an extreme distribution shift requires subsampling of positive examples from a particular site, these constraints may limit the extent of the shift that can be simulated, especially when only a small number of positive examples emanate from one of the sites. Therefore, we proposed an updated framework with those two constraints from (3.5) and (3.6) removed.

3.2.3 Simulation Configurations

In the simulation of different degrees of distribution shifts by provenance, 4 parameters are required: $P_{train}(Y = 1|Z = z_1), P_{train}(Y = 1|Z = z_2), P_{train}(Z = z_2)$, and α_{test}. The first three are sampled from 0 to 1 evenly in linear space, with a step size of 0.05 or 0.1, depending on scenarios.

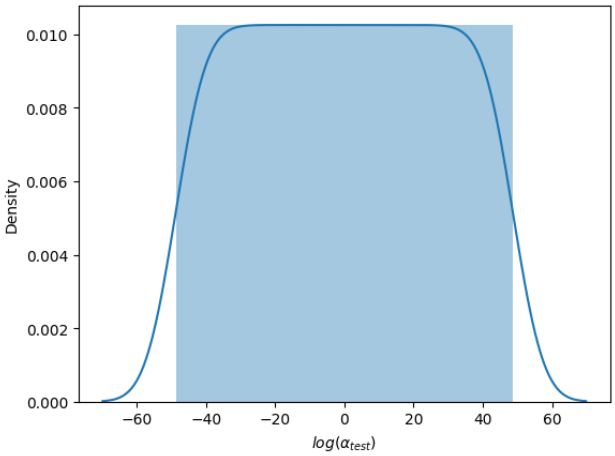


Figure 3.1: α_{test} distribution.

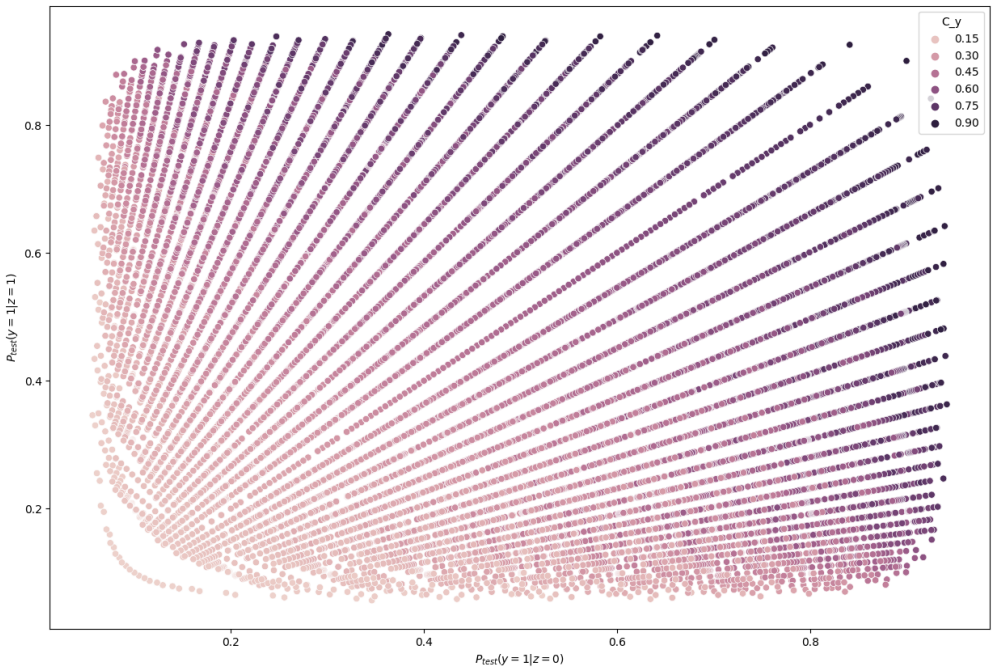


Figure 3.2: Theoretical sampling from the joint distribution of $P_{test}(Y = 1|Z = z_1)$ and $P_{test}(Y = 1|Z = z_2)$. Z categories are coded as $\{0, 1\}$. C_y represents $P_{test}(Y)$.

Since α represents the ratio for positive rates from two sources and $\alpha = 1$ means the same prevalence rates across sources, we sampled α_{test} in a reciprocal manner while centering around 1.0. One such example is $\{..., 1/8, 1/4, 1/2, 1, 2, 4, 8, ...\}$. This can ensure a uniform distribution of α_{test} in the log space, as shown in Figure 3.1.

To break down α_{test} into detailed samplings of its two subpopulations, Figure 3.2 shows the theoretical sampling for the joint distribution of $P_{test}(Y = 1|Z = z_1)$ and $P_{test}(Y = 1|Z = z_2)$. It is worth noting that not all sampling configurations from Figure 3.2 can lead to a valid training/testing split, given how many positive/negative samples from both sources are required for that setting.

3.3 Multi-label Cases

In the previous sections, the simulation framework was developed specifically for binary outcomes of two provenance sources. For generation of simulation scenarios, we provide the full distribution before sampling: $P_{train}(Y|Z), P_{test}(Y|Z)$. An example of the training set composition with different label rates is shown in Table 3.2. The example includes four data sources (hospitals) for three cancer statuses (none, benign, malignant). The table represents one configuration for $P_{train}(Y|Z)$. Test data can be generated with $P_{test}(Y|Z)$ in the same format.

When it comes to measuring the provenance shift in the multi-level setting, the auxiliary variable α becomes infeasible, unless a pair-wise comparison is applied or this is represented as a vector. To quantify provenance shift under such circumstances, statistical distance between distributions is more suitable [64]. For example, Kullback–Leibler (KL) divergence can be adjusted to measure the shift by constructing:

$$D_{KL}(P_{train}(Y|Z)||P_{test}(Y|Z)) \tag{3.8}$$

KL-divergence is an asymmetric measure. Alternatively, Jensen–Shannon divergence

		Y		
Z	$P(Z)$	none	benign	malignant
Hospital A	0.1	0.1	0.1	0.8
Hospital B	0.2	0.2	0	0.8
Hospital C	0.2	0.7	0.3	0
Hospital D	0.5	0.3	0.3	0.4

Table 3.2: A hypothetical example of cancer prevalence (Y) at multiple hospitals (Z). Provenances and target labels are multi-level. The label frequencies within each hospital (data source) sum to 1.

could be used to ensure symmetry [65]:

$$JSD(P_{train}(Y|Z)||P_{test}(Y|Z)) = \frac{1}{2}D_{KL}(P_{train}(Y|Z)||P_{test}(Y|Z)) + \tag{3.9}$$

$$\frac{1}{2}D_{KL}(P_{test}(Y|Z)||P_{train}(Y|Z)) \tag{3.10}$$

Given the expanded generation procedure and new measure of distribution distance, a simulation framework for multi-level classification can then be built.

3.4 Evaluating Robustness to Provenance Shift

Models were tested under the proposed simulation frameworks with or without constraints on positive rates (in Equation (3.5)) and composition (in Equation (3.6)).

For each test set, we use the Area Under the Precision Recall Curve (AUPRC) for its better discriminant ability in rare-case scenarios as compared with the Area under the Receiver Operating Characteristic curve [66]. In the simulation framework, which tests different degrees of provenance shift, one metric of evaluation is *robustness*. This is quantified as the absolute value of the coefficient from the fitted regression line with AUPRC as the dependent variable and α_{test}, in the log scale, as the independent variable, inspired by Taori, Dave,

Shankar, *et al.* [67]. Specifically, across all different testing scenarios, robustness is assessed as $|\beta|$ from:

$$AUPRC = intercept + \beta \times log_{10}(\alpha_{test}) \tag{3.11}$$

Intuitively, this measures the slope of the trend on a model's AUPRC values under different degrees of provenance shift. The lower the absolute value of the fitted coefficient, the more robust a model is to confounding shift, with a value of zero indicating equivalent performance irrespective of this shift.

However, the magnitude of a model's performance is also of importance. A model exhibiting a perfectly flat line in the simulation framework, but with very low AUPRC, is not a useful model. With each specific testing scenario, performance is directly evaluated as an AUPRC value. When the framework is applied, we identify two points corresponding to specific α_{test} values at which AUPRC's are of interest:

- $AUPRC_{best}$: the $AUPRC$ value where $\alpha_{test} = \alpha_{train}$ (no confounding shift)
- $AUPRC_{worst}$: the $AUPRC$ value where $\alpha_{test} = 1/\alpha_{train}$ (severe confounding shift)

Figure 3.3 shows an example of the results. It includes models trained on a dataset with $\alpha_{train} = 0.2$ (as indicated in the title) and evaluated on a range of different α_{test} values (the x-axis in log scale). At each evaluation point (corresponding with one α_{test} value), AUPRC is reported as the y-axis. The green vertical line marked with "best" indicates where the α_{test} value equals to α_{train} (0.2 in this case) and thus $AUPRC_{best}$ is reported with this α_{test} value. The gray vertical line marked with "worst" indicates $\alpha_{test} = 1/\alpha_{train} = 5$ where $AUPRC_{worst}$ is reported. The robustness of the model is shown as the slope which can be visually checked and is also reported as a coefficient calculated from (3.11) in the legend (in this case it is -0.149). The x-axis is reversed in this case to position the worst-case scenario on the left side and best-case scenario on the right hand side, but the coefficient is calculated without this reversion.

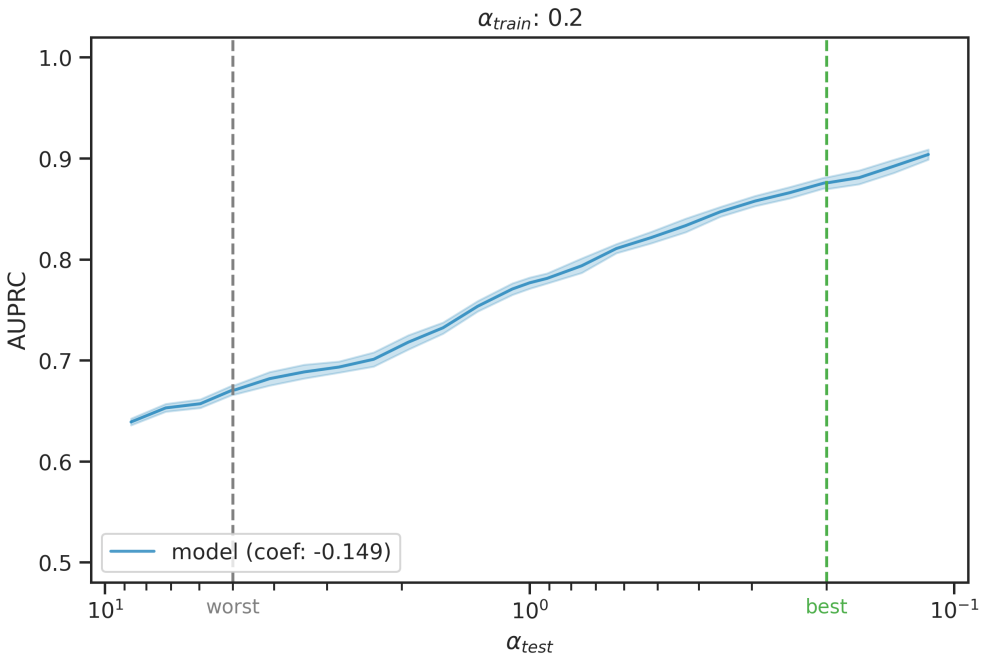


Figure 3.3: An example of the result figure. Models are trained on datasets with $\alpha_{train} = 0.2$. The x-axis is α_{test} in log scale. The y-axis is the AUPRC.

Chapter 4

DATASETS

4.1 Introduction

Three datasets were used in this work as primary materials for evaluating the effect of confounding by provenance and adjustment methods under the proposed simulation framework. They include two biomedical datasets and one from the general domain. The biomedical datasets are a locally-developed Cognitive Distortion Detection (CD) dataset [68], and the publicly-available Social History Annotation Corpus (SHAC)[69], [70]. The general domain dataset, Hate Speech Detection (HateSpeech) [71], [72], was used to assess the generalizability of methods beyond the biomedical domain.

4.2 Cognitive Distortion Detection Datasets

The CD set was derived from two previous studies.

TM set The first was a randomized controlled trial of a community-based text-message intervention (referred as the TM set), which involved 51 participants and 49 of them were included in the original analysis [68]. 37 participants were randomly assigned to the text intervention arm (vs. 12 in the control group) and they provided data for this current work. The data collection period was between December 2017 and October 2019. As part of the study, clients participating in standard care engaged with trained clinicians in text-message conversations up to three times a day for 12-weeks. In total, 14,312 messages between clinicians and their clients were collected, with 7,354 coming from the clients. These data were annotated for the presence of cognitive distortions - maladaptive patterns of thinking that can be addressed by cognitive behavioral therapy. Ten cognitive distortions were selected

for annotation [73]: Dichotomous Thinking (DT), Labeling (L), Overgeneralization (O), Mental filter (MF), Disqualifying the positive (DP), Should statements (SM), Personalization (P), Jumping to conclusions (JC), Emotional reasoning (ER), and Catastrophizing (C). In addition, we added the label Any Distortion (AD), generated in accordance with the other assigned distortions. Two mental health specialists annotated all messages from clients by assigning these categories, which are not mutually exclusive [74]. This provided ground truth labels. This dataset was previously used for training NLP systems for detecting different cognitive distortions [74]–[76].

AVH set The second dataset, referred to as the AVH set, was derived from a study that assessed the experience of Auditory Verbal Hallucinations (AVH), with the goal of identifying factors associated with increased use of health care services [77]. The study involved 384 participants from 41 US states, 295 (77%) of whom received inpatient, outpatient, or combination treatments for AVH, and 89 of whom (23%) did not. A total of 4,809 audio diaries (recordings of participants describing their experiences in up to three minutes) were collected during a 30-day period. Only recordings of thirty seconds or more in duration were eligible for the study (a total of 3,033) and 511 of these were randomly selected for the analysis of cognitive distortions. The 511 audio diaries were then professionally transcribed. Given the length of the diaries and variation in flow of speech, we further split transcribed diaries into sentences, with a median (1st and 3rd quartiles) of 10 (6 - 19) words for each split (sample) and a total of 5,424 samples. These samples were then annotated for their cognitive distortion status according to the same guideline used in TM set above.

In this work, I focus on the aggregated "Any Distortion" label for simplicity and use it as the primary prediction target. Table 4.1 provides an overview of this dataset.

4.3 Social History Annotation Corpus (SHAC)

The Social History Annotation Corpus (SHAC) is a dataset of notes from electronic health records annotated for social determinants of health (SDoH) that provided the basis for the

	Total Number	Identified Cognitive Distorted Cases	Positive Rate
TM	7,354	2,109	28.70%
AVH	5,424	1,170	21.60%

Table 4.1: Cognitive Distortion Detection dataset summary.

n2c2/UW SDoH Challenge [69], [70]. SHAC was designed to support extracting Social Determinants of Health (SDoH) from clinical notes. The notes were collected from two sources: clinical notes of chronic pain patients from the University of Washington Medical Center (the **UW Dataset**), and discharge notes of intensive care unit patients from MIMIC-III (the **MIMIC Dataset**). As described in previous work [69], only the social history sections from these notes were selected (using pattern matching) and retained. These served as our text samples for classification. The SHAC corpus was constructed through an active learning framework. SHAC consists of 4,405 (2,528 from UW and 1,877 from MIMIC) annotated social history sections (original distribution: 70% train, 10% development, and 20% test). All development and test data were randomly sampled. Training samples were 29% randomly selected and 71% actively selected. For the 71%, active learning was used to increase the prevalence of critical risk factors in the annotated training data including positive substance use, unemployment, disability, and homelessness. The annotation was completed by four medical students.

In this work, we collected all notes from the train, development, and test sets into one pool and resampled from it. For the main outcome, we selected "drug abuse", among other substance abuse (such as alcohol and tobacco), employment, and living status categories. We made this selection because of the outcome variables concerned, drug abuse had the greatest difference in positive class prevalence across the two sources, and as such presented the best opportunity to explore confounding by provenance. The original annotation was developed to support a span extraction task, and included more granularity than is required for text

categorization, including the extraction of modifiers indicating the status of the documented drug abuse. For our purpose we only used the Status arguments of "current" and "past" to construct a positive label for drug abuse. All others were considered as negative.

Documented cases of drug abuse are distributed very differently in the two sources, with a positive class prevalence of 41.1% for the UW dataset and 19.8% for the MIMIC dataset (Table 4.2). These differences likely reflect differences in sampling strategies across the two sites, rather than differences in the prevalence of drug abuse at each location [69]. This, in the meanwhile, presents the best opportunity to explore confounding by provenance. Dataset characteristics are shown in Table 4.2.

	Total Number	Identified Drug Abuse Cases	Positive Rate
UW	2,528	1,040	41.1%
MIMIC	1,877	371	19.8%

Table 4.2: SHAC dataset descriptive statistics.

4.4 Hate Speech Detection Datasets

Two publicly available datasets were used for the hate speech detection task.

DynGen set The first was dynamically generated by Vidgen, Thrush, Waseem, *et al.* [71] (referred as DynGEN set), through four rounds of data creation, each generating around 10,000 entries. The first round collected synthetic texts, created by the annotation team to closely mimic real-world content. This was then followed by perturbations in the texts to create new examples for the next three rounds. In the end, a total of around 40,000 entries, around 45% of which contained hate speech, were collected. This work implemented binary coding of HATE and NOHATE, with the former being further categorized into secondary

labels of Derogation, Animosity, Threatening Language, Support for Hateful Entities, Dehumanization, and "Not Given". The original study reported a majority of hate speeches are in type Derogation (9,907) and "Not given" (7,197), followed by Animosity (3,438). In our work, only the primary binary label was used for the classification. This dataset is publicly available[1].

WSF set The second dataset (WSF set) comes from a real-world white supremacist forum, Stormfront[2], which is characterized by pseudo-rational discussions of race [78]. Posts published between 2002 and 2017 were collected from a subset of 22 sub-forums covering diverse topics and nationalities from the website in a previous study, for annotating hate speech [72]. The posts were segmented into sentences for manual annotation, which guarantees the minimum unit containing hate speech and avoids noise from other nearby sentences that are potentially "clean". Authors used four categories for annotation: (1) HATE; (2) NOHATE; (3) RELATION, where the sentence itself does not convey any information and must be put in its context to be correctly identified, such as a reply to a hate speech comment; and (4) SKIP, where HATE/NOHATE cannot be decided. Table 4.3 shows numbers for each annotated label in the corpus. For the sake of simplicity, we only retained examples with the HATE and NOHATE labels for our work, preserving the majority of the texts (10,703 records out of 10,944 in total). The dataset is publicly available[3].

Table 4.4 provides a summary of the Hate Speech Detection datasets.

---

[1] https://github.com/bvidgen/Dynamically-Generated-Hate-Speech-Dataset

[2] www.stormfront.org

[3] https://github.com/Vicomtech/hate-speech-dataset

Labels	Number of entries
HATE	1,196
NOHATE	9,507
SKIP	73
RELATION	168
Total	10,944

Table 4.3: Labels used for manual annotation of the WSF set in the original work. Only HATE and NOHATE are used in our work.

	Total Number	Identified Hate Speech	Positive Rate
DynGEN	41,144	18,969	46.1%
WSF	10,703	1,196	12.2%

Table 4.4: Hate Speech dataset summary.

Chapter 5

STATISTICAL ADJUSTMENT - BACKDOOR ADJUSTMENT

The previous chapter introduced a unified simulation and evaluation framework for systematically testing methods for adjusting provenance effects, which is the focus for the rest of the thesis. In this chapter, I start with the statistical adjustment. Backdoor Adjustment is the key in this category, which tries to cut the relationship of the provenance variable with the text (predictor) and target label at the same time.

A version of this chapter was previously published by the American Association for Medical Informatics (AMIA) as an open-access article. ©AMIA.

Ding, X., Sheng, Z., Yetişgen, M., Pakhomov, S., & Cohen, T. (2023). Backdoor adjustment of confounding by provenance for robust text classification of multi-institutional clinical notes. In *AMIA Annual Symposium Proceedings* (Vol. 2023, p. 923). American Medical Informatics Association.

5.1 Introduction

Machine learning methods are well-established in clinical natural language processing (NLP) [11], and the recent successes of deep neural network models have generated interest and enthusiasm in their applications in the clinical domain [12]. Adequate amounts of diverse data are key to successful training of such models. However, this need for size and diversity presents a considerable challenge for clinical data collection from an individual institute [13]. Integrating data from multiple institutions is a natural way of increasing dataset size, and also promoting collaboration. Data coming from different sources will benefit model training, but at the same time, may also introduce bias. Recent work in NLP has drawn attention to the potential of confounding variables – variables that influence both the pre-

dictors (incoming text) and outcomes (an assigned category) of a text categorization system – to harm model performance at the point of deployment [14], [15]. Essentially, the concern is that when the distribution of category assignment in the training data varies in accordance with the value of a confounding variable (e.g. gender of the author), the model will erroneously learn to make category assignments using words that are associated with this variable (e.g. gender differences in pronoun use)[16], rather than words that meaningfully inform the categorization task at hand. In our recent work, we have identified *confounding by provenance* as a variant of this confounding effect in which models learn to associate features that indicate the source (provenance) of a component of a multi-institutional dataset with this component's label distribution [17], [18]. Confounding by provenance threatens the integrity of machine learning models trained on multi-institutional data sets, and may lead to inaccurate predictions once they are deployed in settings where source-specific distributions of the category of interest differ from training data, with potential to limit adoption of AI models and threaten patient safety.

In this work, we use a backdoor adjustment for text classification method developed by Landeiro and Culotta [14], [15], which follows a similar form to backdoor adjustment for causal inference, as introduced by Pearl [62]. This method was shown to reduce confounding bias in a set of clinical notes drawn from different services within an institution in our recent work on detection of documented goals-of-care discussions [17]. A limitation of this prior work, including our own, concerns how texts were represented: as binary unigram vectors. The availability of large language models presents representational alternatives that involve encoding text as continuous vectors (embeddings). Methods such as Bidirectional Encoder Representations from Transformers (BERT)[79] and Sentence-BERT[80] generate text embeddings that naturally address the perennial NLP concerns of synonymy (because similar words will produce similar embeddings) and polysemy (because representations of the same word will differ in accordance with local context). One goal of the current research is to evaluate their amenability to backdoor adjustment as a means to address the problem of confounding by provenance, in comparison with binary unigram vectors. In addition, we

wished to go beyond our prior evaluation efforts to develop a principled framework for the evaluation of robustness to *confounding shift* [15] - a shift in positive class probability given a confounding variable between training and test data - in the context of confounding by provenance.

The main contributions of our work are:

- The formal definition of the problem of confounding by provenance (Section 5.2).
- The development of an evaluation framework for robustness to provenance-related confounding shift (Section 5.3.2).
- The deployment of this framework in an evaluation of the effectiveness of Landeiro and Culotta's backdoor adjustment method when applied to Sentence-BERT embeddings (Section 5.3.4).

5.2 Problem Definition and Dataset

5.2.1 Confounding by Provenance

In our recent work, we observed confounding by provenance when data from multiple sources were combined together for model development [17], [18]. In both of the settings concerned, some level of data integration from different sources was applied. In one case, a transfer learning framework [18] was applied. In the other, the application involved deliberately merging data from different types of clinical notes [17]. Though confounding by provenance was recognized in this work, the papers concerned do not include a formal definition of this phenomenon, nor do they provide framework for the evaluation of robustness to it in the context of confounding shift.

Here, we formulate the combination of confounding by provenance and distribution shift as follows. In causal inference, a Directed Acyclic Graph (DAG) provides a convenient way to represent confounding variables [62]. The example in Figure 5.1(a) shows a case where the goal is to investigate the effect of chronic kidney disease on mortality. Age is a recognized

risk factor for chronic kidney disease [54], [55], and at the same time it affects mortality rates. Thus, age is considered as a *confounder* in causal inference literature - a variable that influences both the predictor (chronic kidney disease) and the outcome (mortality) of a study.

In the setting of text classification, a similar "non-causal" DAG for predictive models was adopted [14], [15]. Instantiated for confounding by provenance in the context of the dataset used for the current experiments (this is derived from the Social History Annotation Corpus - SHAC [69] and described in Chapter 4.3), text features from clinical notes ($\mathbf{X}$) serve as inputs into classification models (Figure 5.1(b)). Text features will vary by source (Z), and both these features and the distribution of positive examples from each source will affect the prediction of drug abuse (Y). The distributions of language and labels from two distinct data sources (Z) will affect both the text in clinical notes (the predictors) and the frequency (and predicted probability [66]) of documentation of drug abuse (the outcome).

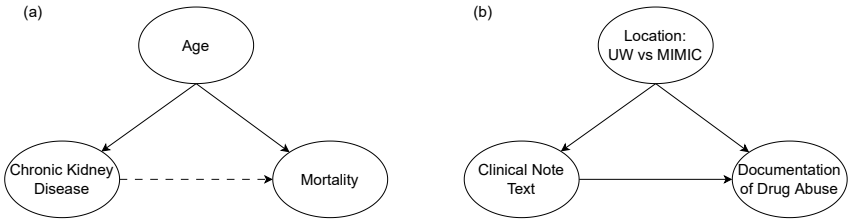


Figure 5.1: (a) Causal DAG depicting age acts as a confounder in effect of CKD on mortality (b) Non-causal DAG in text classification setting for SHAC dataset, with confounding by provenance of two sources: UW and MIMIC.

In the context of distribution shift, the problem can be formalized as follows:

$$P_{UW}(X,Y) \neq P_{MIMIC}(X,Y) \tag{5.1}$$

$$P_{UW,train}(X,Y) \neq P_{UW,test}(X,Y) \tag{5.2}$$

$$P_{MIMIC,train}(X,Y) \neq P_{MIMIC,test}(X,Y) \tag{5.3}$$

where UW and MIMIC are the two sources of data for the multi-institutional set. The equations indicate that the two sources have different positive class prevalence ($P(Y|Z)$), and that *within* each source this prevalence differs at training and test time. These differences and their implications are discussed in greater detail in Section 5.3.2, which describes our evaluation framework in which different degrees of distribution shift are imposed by experimental perturbation.

5.3 Methods

5.3.1 Backdoor Adjustment in Text Classification

Backdoor adjustment is a widely used technique and has been studied across different domains where causal inference is needed [81]–[84] (for an illustrative recent example we refer the interested reader to this prospective observational study evaluating COVID-19 vaccine side effects when non-causal paths need to be adjusted [85]).

According to Pearl (Causality, Equation (3.19) in pg. 80), under the simple setting shown in (Figure 5.1(a)), the confounding effect can be controlled for by the following summation over Z [62]:

$$P(y|\hat{x}) = \sum_{z} P(y|x,z)P(z) \tag{5.4}$$

Landeiro et al. proposed a similar adjustment for text classification [15]. At the point of prediction when the true confounding variable is unavailable, the adjusted estimated (in change of adjusted) causal effect can be calculated as:

$$P(y|x) = \sum_{z=c}^{C} P(y|x,z_c)P(z_c) \tag{5.5}$$

Essentially, the prediction probability is the sum of the probabilities across all possible Z values (for all known categories,$1, 2, ..., C$). This estimate is used in the absence of knowledge of the true Z, in order to compute $P(y|x, z_c)$. Under the logistic regression framework, for

any potential $Z = c$, the logit for a single case i is given by:

$$logit(P(\hat{y}|\boldsymbol{X_i}, \boldsymbol{z_{ic}})) = \beta_0 + \boldsymbol{\beta_1 X_i} + \boldsymbol{\beta_2 z_{ic}} + \epsilon_i \tag{5.6}$$

In our experiments on the SHAC dataset, $logit(P(\hat{y}|\boldsymbol{X_i}, \boldsymbol{z_{ic}}))$ is the estimated logit that example i belongs to c (UW or MIMIC), $\boldsymbol{X_i}$ represents the text encoding features (binary unigram vectors or Sentence-BERT embeddings). $P(z_c)$ is empirically inferred from the training set, using frequencies for each category:

$$P(z_c) = \frac{\sum_{i \in D_{train}} \mathbb{1}(y_i = c)}{|D_{train}|} \tag{5.7}$$

At this point, we can combine Equations 5.6 and Equation 5.7 with Equation 5.4 to estimate the probability of documented drug abuse for any example i in the testing set.

In our experimental setting for the SHAC dataset, we only control one confounding variable: the source, and it is binary ($z \in \{UW, MIMIC\}$). It is encoded using a modified one-hot encoding. For example for each instance, we concatenate ($v \times \mathbb{1}(UW)$, $v \times \mathbb{1}(MIMIC)$) as additional features to the vector representing the text. Here, v is a hyperparameter controlling how much to constrain emphasis on the text-derived features, $\boldsymbol{X}$ (relative to Z). $v = 1$ corresponds to one-hot encoding. Other settings of $v = 10$ and $v = 100$ were also tested in our experiments.

5.3.2 Perturbation of Distributions

These probabilities determine the source-specific class distributions at training and test time. Of these constraints, (3.5) and (3.6) are held constant across experiments while the others are varied. Furthermore, we introduced two auxiliary helper variables α_{train} and α_{test}, which measure the train and test set ratios between positive class prevalence in data from each site:

$$\alpha_{train} = \frac{p_{train}(y = 1|z = 1)}{p_{train}(y = 1|z = 0)} \tag{5.8}$$

$$\alpha_{test} = \frac{p_{test}(y = 1|z = 1)}{p_{test}(y = 1|z = 0)} \tag{5.9}$$

Now, given (3.1), (3.3), (3.6), (5.9), we can obtain the remaining probabilities for (3.2), (3.4), (3.5). Intuitively, this means all that is needed to set up an evaluation of robustness to confounding shift is to control the training distributions and set a positive ratio between sources for the testing set. This simulates a setting where, in general, we only know our training set, and different degrees of shift are simulated.

We proceed to describe our experiments in detail. The Z variable was assigned as 0 for UW, and 1 for MIMIC. We fixed the training set size at 2,000 and testing set size at 500, so that for all settings we introduced confounding shift by undersampling and no instance was drawn more than once. We first sampled our training set to have positive rates similar to those shown in Table 4.2:

$$p_{train}(y = 1|z = UW) = 0.5$$
$$p_{train}(y = 1|z = MIMIC) = 0.2$$

This corresponds to $\alpha_{train} = 0.2/0.5 = 0.4$. Next, $p_{train}(z = 1) = p_{test}(z = 1) = Const_z$ was drawn from range 0.1 to 0.9 (inclusive) at a step size of 0.05, α_{test} from range 0 to 10 (inclusive) at a step size of 0.05. Excluding extreme combination settings where training and testing set sizes could not be successfully drawn without replacement left with 1,287 valid settings. Each setting represents a different amount of provenance-related confounding shift. Shift toward increased representation of positive examples from MIMIC in the test set will result in higher α values. Shift toward increased representation of positive examples from UW in this set will result in lower α values.

5.3.3 Discrete Text Representation Using Binary Unigram Vectors

As a baseline, binary unigram vectors (i.e. one-hot vectors with a coordinate corresponding to each word in the vocabulary) were used following the work of Landeiro et al. [14], [15] This is a constrained case of an n-gram representation (n=1), where only one word is considered when constructing the dictionary. Furthermore, only a binary indicator (0 for absent, 1 for present) of each word is recorded into the final vector for a given document.

5.3.4 Continuous Text Representation Using Sentence-BERT

Proposed in 2019 by Reimers et al., Sentence-BERT is a BERT-based framework [79] with modifications in training using siamese and triplet network structures to reduce the distance between vector representations of sentence with similar meaning [80]. In addition to improving performance on sentence similarity benchmarks, the siamese network architecture, pre-training starting from BERT and RoBERTa, and a smart batching strategy all help reduce complexity for training the model and comparing sentence similarities. Several pretrained Sentence-BERT models have been released since then, with varying neural network depths and training corpus. We used the `all-MiniLM-L6-v2` version, which with only 6 transformer layers provides a good balance between performance and computational efficiency. This pretrained model is publicly available from the HuggingFace repository [1]. Each document was provided as input to the model and the output, a 384-dimensional dense vector then served as a continuous document representation.

5.3.5 Evaluation Setup

With perturbed distributions, we applied backdoor adjustment for text classification using the logistic regression, using (5.6). A "vanilla" version of logistic regression without the provenance confounder variable Z was also fit using the same settings as a baseline:

$$logit(y_{ic}) = \beta_0 + \boldsymbol{\beta_1 x_i} + \epsilon_i \tag{5.10}$$

Each experiment was repeated five times, with a different random seed for subsampling instances to generate train/test splits with confounding shift. On the testing set, we report the mean and standard deviation of Area Under the Precision-Recall Curve (AUPRC) values for each setting. AUPRC was chosen for its better discriminant ability in rare-case scenarios [86].

---

[1] `https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2`

5.4 Results

Figure 5.2 shows results for experiments using binary unigrams as text representations. Figure 5.3 shows results for experiments using Sentence-BERT embeddings as text representations. In both cases results for each $P(z = MIMIC)$ setting are shown separately, simulating different mixture ratios of the two data sources. For example, when $P(z = MIMIC) = 0.2$, according to (3.6), for both training and testing set 20% of data come from UW and the remaining 80% from MIMIC. For the sake of conciseness, and given the subtle differences between results when varying $P(z = MIMIC)$, only a few settings over the span are shown.

The figures can be interpreted as follows. The y axis shows model performance on the testing set, measured using the AUPRC. The x axis shows the α value of the test set, which indicates the ratio between the positive class prevalence with $z = 1$ (MIMIC) and that with $z = 0$ (UW). The α_{test} across settings could not be aligned (as shown from different x-axis ranges) because for some experimental combinations of probabilities, there is no valid way of drawing samples satisfying that constraint. As this ratio moves further from that of the training set (α_{train}), indicated by the dashed vertical line, the degree of confounding shift increases. Therefore, a curve that does not drop precipitously while starting from the dashed vertical line and moving to the right on the x-axis indicates robustness to confounding shift in this direction. This is the case with plots of backdoor-adjusted (BA) models (blue lines) only, indicating that this approach is effective at mitigating for confounding shift involving an increase in the proportion of positive examples from MIMIC at test time with both discrete (unigram) and distributed (Sentence-BERT) text representations. However, increasing the proportion of positive examples from the UW site leads to worse performance with backdoor adjustment than with the baseline ("vanilla") models.

For models using binary unigram representations, BA models lead to much more stable AUPRC results in range of 0.90-0.94 across $P(z = 1)$ (MIMIC) of 0.3, 0.5, and 0.6. In comparison, the baseline models show a wider range of 0.87-0.95 in performance. The slopes of the lines in Figure 5.2 for the two models also indicate the robustness of BA models to

confounding shift. For each setting of $P(z = 1)$, when α_{test} is small (to the left side of the red dashed line for α_{train} marker indicating an increase in the number of positive examples drawn from the UW site), the "vanilla" model performs better than BA with an increase of AUPRC of 0.01-0.02. When moving further to the right side region, the "vanilla" model's performance drops and quickly falls beneath that of the BA model (a difference of 0.02-0.04).

Binary unigram representations lead to AUPRC on the testing set in the range of 0.88-0.95 for both of models: backdoor adjustment logistic regression (BA) and logistic regression without provenance confounders ("vanilla"). However, for each setting, the BA model appears much more stable than the "vanilla" model, even though this stability does not guarantee better performance. For example when $P(z = 1)$ is increasing (in the current experiment to level 0.6), BA performance gets closer to that of the "vanilla" model (Figure 5.2). Similarly, for each setting of $P(z = 1)$, when α_{test} is small (to the left side of the red dashed line for α_{train} marker indicating an increase in the number of positive examples drawn from the UW site), the "vanilla" model also performs better than BA. Except for those benefits from specific scenarios or regions in one setting for the "vanilla" model, the BA model shows little variance to provenance shift (indicated by varying α_{test} in the x axis).

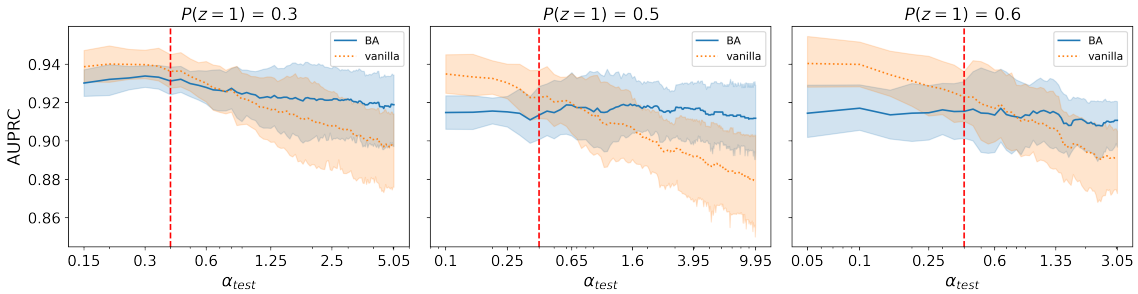


Figure 5.2: AUPRC performance with binary unigram representations ($log10$ scale for x). $v = 10$. Vertical red dashed line in each plot represents $\alpha_{train} = 0.4$, where α_{test} matches the training set and distribution difference is minimal. Shaded areas represents 95% CI for 5 random runs. BA: backdoor adjustment text classification logistic regression. vanilla: logistic regression without provenance confounders.

Overall, Sentence-BERT embeddings (Figure 5.3) lead to AUPRCs on the testing set in the range 0.82-0.94. Similarly to models using binary unigrams, models using Sentence-BERT embeddings also show different performance according to α_{test} values. In general, when α_{test} is small (to the left side of the red dashed line), the "vanilla" models outperform BA models by 0.01-0.02 AUPRC units. To the right of the graph, where α_{test} is large and the proportion of positive examples drawn from MIMIC is high, "vanilla" model performance drops. At the rightmost point, BA models performs better by 0.04-0.06 units of AUPRC, where the proportion of positive examples from MIMIC in the testing set is high. Overall, BA models show gentler slopes than baseline models, suggesting robustness provided by backdoor adjustment.

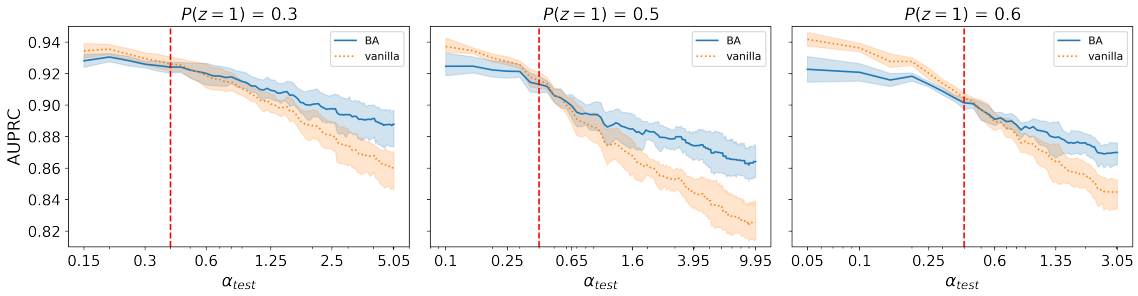


Figure 5.3: AUPRC performance with Sentence-BERT representations ($log10$ scale for x). $v = 10$. Vertical red dashed line in each plot represents $\alpha_{train} = 0.4$, where α_{test} matches the training set and distribution difference is minimal. Shaded areas represents 95% CI for 5 random runs. BA: backdoor adjustment text classification logistic regression. vanilla: logistic regression without provenance confounders.

When comparing binary unigram vectors and Sentence-BERT embeddings, in setting of $P(z = MIMIC) = 0.5$, Sentence-BERT BA model performance is in the range 0.87-0.93, while binary unigram representations result in an AUPRC of 0.90-0.94. The best performance at optimal α is obtained using Sentence-BERT representations, which is consistent with the general trend in NLP of better performance with deep learning derived distributed represen-

tations. However, while there are observable effects relative to the unadjusted baseline, the Sentence-BERT representations appear to be less conducive to backdoor adjustment. When moving to the right-side region of the vertical red dashed lines (the more α_{test} increases from 0.4), Sentence-BERT models show more reduction in performance, indicating sensitivity to confounding shift. With unigram representations, the slopes of the blue lines (standing for BA models) in Figure 5.2 are in general much gentler than those in Figure 5.3, suggesting more robustness to BA models when using binary unigram vectors. When α_{test} is small, especially less than 0.4 (to the left-side region of the red dashed lines), both embedding methods do not help the BA model's performance. Here, the "vanilla" model outperforms it. All of the above observations hold for other settings of $P(z = MIMIC) = 0.5, 0.6$. We also performed simulations using different v values: 1, 100 (not included in the paper), and those results remain similar, except that both $v = 1$ and $v = 100$ lead to a higher variance across repeated experiments. This suggests that $v = 10$ provides a good balance between stability and adjustment effect.

5.5 Discussion

In this work, we presented the problem of confounding by provenance, devised a framework for evaluation of robustness to it, and applied this framework to evaluate a method to address it (backdoor adjustment) using the SHAC dataset. To represent text for regression modeling, we used two approaches: binary unigram vectors and Sentence-BERT embeddings. Our findings indicate that the effects of backdoor adjustment vary with this representational choice, as well as with the *direction* of confounding shift (with this direction determined by the value of α_{test}). With a direction of provenance shift promoting higher proportions of MIMIC-derived positive examples in the test set, models with Sentence-BERT embeddings usually perform better in terms of AUPRC (for $P(z = MIMIC) = 0.5, 0.6$). However, in the opposite direction of provenance shift promoting higher proportions of UW-derived positive examples in the test set, models with binary unigram vectors slowly drop in performance and at very high α_{test} they outperform models using Sentence-BERT embeddings. Provenance-

related confounding shift can happen in both directions (promoting the number of positive test set examples drawn from one source over those from another). Our results suggest that the utility of adjusting for these shifts using the backdoor adjustment methods may vary depending on whether the shifts are toward (to the right of the red dashed lines in figures) or away from (to the left of the red dashed lines in figures) the source with lower prevalence in the training set (MIMIC).

In terms of the relationship between choice of representation and robustness, results from Figure 5.2 and Figure 5.3 show that models with binary unigram vectors usually have gentler slopes, suggesting stronger robustness to provenance shift in both directions. This is true even for logistic regression models without any adjustment for confounders (when comparing orange lines in Figure 5.2 and Figure 5.3).

In general, results show that backdoor adjustment for text classification is an appropriate method to mitigate for provenance-related confounding shift, and can provide the models with robustness to this shift. The effect of the adjustment, however, varies with different modeling choices, including the selection of hyperparameter v for one-hot encoding, text embedding choices, penalization for logistic regression, degree of penalization. v and degree of L2 penalization for logistic regression have modest effects on adjustment, so not all results are shown in this chapter.

While these results make a foundational contribution to the study of remediation of confounding by provenance, both our evaluation framework for robustness to it and the methods developed to address this problem have broader implications. They can also be applied in the context of other potential sources of bias by partitioning data sets with this variable, rather than by provenance. For example, one might partition a data set of patient-generated language by patient ethnicity, and use backdoor adjustment to ensure that dialect differences are not being used as spurious cues to predict some unrelated outcome. In this way, both our framework for evaluation and the methods under consideration have the potential to be applied to address other types of bias also.

5.6 Limitations

In this work, we utilized fixed training source-specific positive class rates (0.5 for UW, 0.2 for MIMIC) that are close to those of the full SHAC dataset. Since the positive rates are imbalanced for *training* the models, in the coming chapters, we will evaluate the BA method under different settings during training and check its validity when imposing different degrees of provenance shift. This will allow us to assess the relationship between source-specific class distributions at training time and the utility of backdoor adjustment. It is also noted that, from the figures, the densities of points are not uniform in log scale across different α values, due to the fact that we performed uniform sampling of α in the linear space. To avoid potential bias on interpretation, as future work we will update this sampling strategy for a more balanced distribution.

In terms of text feature extractions, we only tested binary unigram vectors as previously used by others in work with backdoor adjustment for text classification [14], [15], [17]. Normalized counts of n-grams could be another option for retaining more information. While our results did not show clear benefits for using Sentence-BERT embeddings across all ranges of confounding shift, this may be due to the model capacity limitation of regularized logistic regression. Other statistical and machine learning models, SVM [87], XGBoost [88] could be explored under backdoor adjustment. Moreover, according to Sentence-BERT model benchmarks[2], the pretrained model "all-MiniLM-L6-v2" we used in the paper is not the one with the best performance. Other pretrained models, such as "all-MiniLM-L12-v2" and "all-mpnet-base-v2", are good candidates for next steps. Finally, to better utilize embeddings generated from large language models, deep neural networks could be used for text classification, in addition to their use for representational purposes. The application of backdoor adjustment while fine-tuning a deep learning model for text categorization remains an interesting direction for future work, though we anticipate many methodological details remain to be resolved.

---

[2] `https://www.sbert.net/docs/pretrained_models.html`

5.7 Conclusion

In this work, we evaluated the utility of backdoor adjustment as a mean to address confounding by provenance for text categorization. Our result indicate that given the imbalanced source-specific class distributions in our training set, models with the backdoor adjustment generate more stable results than those without it, with both unigram and deep learning derived text representations. Models using binary unigram vectors as input features with adjustment show strong robustness to provenance shift, though this only leads to advantages in performance over a baseline model when the shift is in the direction of the minority source in the training set, raising interesting questions about the scope of applicability of this method. To determine its validity, in the coming chapters, I will discuss situations under different training set distributions.

Chapter 6

DISTRIBUTION MATCHING AND AUGMENTATION FOR ADJUSTMENT

The statistical adjustment discussed in Chapter 5 focuses on using Backdoor Adjustment to cut the relationship from provenance variable to both text and target label. In this chapter, I will introduce another method which approaches the problem of provenance shift in a more direct way by manipulating the distributions. This category of methods aim at the relationship between the provenance variable and the target label, even though some methods will also deal with the relationship between the provenance variable and texts (predictors) as a by-product.

Ding, X., Lybarger, K., Tauscher, J., & Cohen, T. (2022, July). Improving classification of infrequent cognitive distortions: domain-specific model vs. data augmentation. In *Proceedings of the 2022 conference of the North American chapter of the association for computational linguistics: human language technologies: Student Research Workshop* (pp. 68-75).

6.1 Introduction

Data augmentation first became a popular topic in computer vision, where deep neural networks have performed remarkably well. Complex architectures, such as AlexNet [89], ResNet [90], DenseNet [91], generally require sufficient training data for model convergence, even with

the help of dropout regularization and batch normalization. This situation also occurs in natural language processing (NLP) with deep learning methods and can become more problematic when limited to small datasets by data collection or data annotation constraints. In imaging, data augmentation, involving transformations such as cropping and shearing, is a common strategy to expand the amount of data available for training. Analogously, several methods have been proposed to perform data augmentation in NLP, including Easy Data Augmentation [92], Back Translation [93], GPT-2 Augmentation [94], and `mixup` [95]. Kumar, Choudhary, and Cho [96] applied some of these methods to pretrained transformer models and showed an average improvement in accuracy of 1-6%.

However, the low-resource scenario was simulated by simply constraining the training data from large corpora. It remains unclear how these methods might perform when used in realistic applications, where certain classes may be of very low frequency. One exemplary case concerns NLP analysis of online therapy sessions, where large amounts of patient-generated texts must be classified, but only well-trained specialists with relevant mental health domain knowledge can perform annotation manually to ensure clinical accuracy. In this study, we used a dataset from text message conversations between clients and therapists, previously used for detecting distorted thoughts [74]. Besides the limitation in size, we found that some types of distorted thinking are very rare, resulting in worse classification performance. To address these issues, we investigate the extent to which data augmentation methods can improve performance of the best-performing BERT model from these experiments. We compare the utility of this augmentation approach to the use of a domain-specific pretrained language model, MentalBERT. In doing so, we evaluate the utility of data and model augmentation techniques to improve the identification of rare classes in the context of real-world data.

Given the methods for making augmentation, the following question conerns which targeted distribution data should be augmented. This targeted distribution matters in the problem of confounding shift by provenance, since we specifically examine cases where the training and test distributions differ, usually by a large amount (extreme shifts).

One popular framework, proposed in the SMOTE (Synthetic Minority Over-sampling

TEchnique) paper, uses synthetic minority class over-sampling combined with majority class under-sampling [46]. The over-sampling rate was tested at 50%, 100%, 200%, 300%, 400%, 500%. The majority class was under-sampled at rates ranging from 10% to 2000%. The authors only reported the best combination from under-sampling and over-sampling rates on final ROC curves and showed SMOTE performed better than under-sampling. Zhu, Liu, Li, *et al.* [97] used data augmentation through Generative Adversarial Networks (GAN) for emotion classification. They deliberately aimed for equal sizes from all categories after augmentation in the toy example or, in real world datasets, made sizes of two minority classes (with the least samples) and one reference class ("neutral") equal. Another framework generates a fixed amount of augmentations per original sample. For example, Escobar Díaz Guerrero, Carvalho, Bocklitz, *et al.* [98] proposed a modified copy-paste data augmentation technique [99], coupled with weight-balancing methods, and showed its effectiveness in histopathology image classification tasks when class imbalance exists. Our previous work based on Easy Data Augmentation [75], which generates four versions on each original sentence, also falls into this framework. During test time, it is a common practice to randomly sample from all test data or artificially create a set with balanced distribution for each class category.

Liu, Xu, Xu, *et al.* [100] empirically tested the idea of utilizing data augmentation techniques for the problem of distribution shift and building more robust models. The results show Empirical Risk Minimization (ERM) [101] combined with data augmentation can achieve state-of-the-art performance in out-of-distribution accuracy. However, they still used fixed number of augmentations per sample, including random resizing and cropping, and random horizontal flipping for the imaging datasets.

In this Chapter, I propose a new framework with strictly pre-defined targets for performing distribution matching. This framework can utilize any augmentation techniques and also resampling (oversampling and undersampling).

6.2 Methods for Augmentation

Approaches for making augmentations.

6.2.1 Easy Data Augmentation

Wei and Zou [92] proposed Easy Data Augmentation (EDA) which comprises four main operations on the original text:

- Synonym Replacement (SR): Randomly choose a non-stopword and replace it with its synonyms.
- Random Insertion (RI): Find a synonym of a random non-stop word of the original sentence. Insert it randomly into the original sentence.
- Random Swap (RS): Randomly swap two words.
- Random Deletion (RD): Randomly remove a word with probability p.

We adopted authors' recommended setting for the parameter α, 0.1, that controls the percentage of words in a sentence changed by each augmentation method. For Random Deletion, we kept $p = \alpha = 0.1$.

6.2.2 Back Translation

Sennrich, Haddow, and Birch [93] proposed Back Translation for data augmentation, where sentences are first translated into another language and then back to the original language. This technique has been explored for the task of neural machine translation by Sugiyama and Yoshinaga [102].

To generate new texts, we applied Back Translation with two intermediate languages: German and Spanish. During the augmentation, each original message was translated into German or Spanish and then back to English to get a corresponding message. Class labels of

the original text were inherited, because this auto-encoder-like operation should not change the underlying idea of the message. We did not repeat these experiments because we found little to no variation in generated texts upon repetition. We used OPUS-MT, an open-source machine translation project that provides over 1,000 pre-trained translation models[1] [103], [104].

6.2.3 Augmentation of Hidden Spaces: mixup

Zhang, Cissé, Dauphin, *et al.* [95] proposed `mixup` for data augmentation. The authors claim that this method extends the training distribution by incorporating the prior knowledge that linear interpolations of feature vectors (x) should lead to linear interpolations of the associated targets (y), providing data are modeled on vicinity relation across examples of different classes. `mixup` operates as follows:

$$\tilde{x} = \lambda x_i + (1 - \lambda) x_j,$$
$$\tilde{y} = \lambda y_i + (1 - \lambda) y_j$$

where $\lambda \sim Beta(\alpha, \alpha)$ for $\alpha \in (0, +\infty)$. This paper did not examine the hyperparameter α across different NLP applications, with results reported only for Google speech commands, a dataset of 65,000 one-second utterances[2]. However, the authors did report improved results when using $\alpha = 0.3$ for this task, and in general proposed a small $\alpha \in [0.1, 0.4]$, based on results on ImageNet-2012. They also acknowledge that model error is less sensitive to large α when increasing model capacity. Sun, Xia, Yin, *et al.* [105] applied `mixup` to the transformer architecture and showed improvements on eight GLUE benchmarks. Across all of their experiments, α was fixed at 0.5.

From the previous two studies [95], [105], it is not clear what hyperparameter setting of α should be used with other data sets. Given the probability density function controlled by α (demonstrated in Figure 6.1), other settings when α is large may make more sense for

[1] `https://huggingface.co/Helsinki-NLP`

[2] `https://ai.googleblog.com/2017/08/launching-speech-commands-dataset.html`

scenarios in which we want to make two examples contribute more evenly. This leads to augmented examples lying in the margin between two categories, which may be appropriate for categories that are difficult to distinguish.

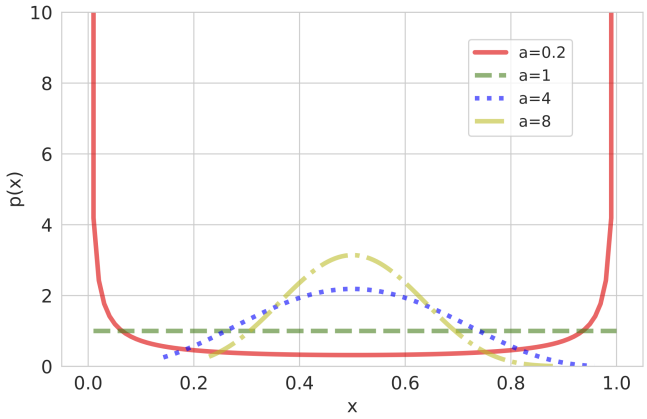


Figure 6.1: Probability Density Function of beta distribution with equal shape parameters, $Beta(\alpha, \beta)$, where $\alpha = \beta$.

In our case, the cognitive distortion dataset is relatively small compared with those evaluated previously, and some classes (O, SM) are quite rare. We wished to assess whether the `mixup` method could help with data augmentation in this context. We did an extended search in the hyperparameter space of α: 0.02, 0.2, 0.5, 1, 2, 4, and 8.

6.2.4 Fine-tune GPT-2 for Generative Augmentation

Anaby-Tavor, Carmeli, Goldbraich, *et al.* [94] propose using GPT-2 for data augmentation, by fine-tuning the model to generate text corresponding to a class of interest. Following their proposed approach, and using a publicly available GPT-2 model[3], we implemented two variations of GPT-2 for data augmentation.

[3] `https://huggingface.co/gpt2`

Context-agnostic GPT-2 We first reconstructed our text messages as follows:

$$y_i[SEP]x_i[EOS]$$

for each of the messages i, where y_i indicates the label of a message, and x_i the message content. GPT-2 was then fine-tuned on this new structure of data for 20 epochs. New messages were generated by feeding in the prompt of "$y[SEP]$".

Contextual GPT-2 Texts in our dataset are derived from conversations. To utilizing this contextual information, we reorganized inputs as follows:

$$y_i[SEP]x_{i-1}[SEP]x_i[EOS]$$

where x_{i-1} is the previous message. The GPT-2 model was then fine-tuned on this structure. Given the prompt of "$y_i[SEP]x_{i-1}[SEP]$", new messages were generated according to the class label y_i and the preceding message for a representative example as context.

6.2.5 Rephrasing using LLM

Research on augmentation in natural language data has adopted the current trend of large language models, especially their generative capability, for generating new texts. This augmentation through LLMs can happen on different levels, namely word level, syntax level, and sentence or discourse level [106]. Some studies directly generated new texts through rephrasing the original ones [107], [108]. Hu, He, Wang, *et al.* [109] used LLM to generate analysis as augmentation when enhancing a personality detection model on social media posts .

In this work, I utilized rephrasing as augmentation. Specifically by asking the LLM to rephrase any given sentence, I collected 10 different rephrases per sample. The LLM used in my experiments is the openly available LLama-3.1 with 405 billion parameters[4] through

[4] `https://ai.meta.com/blog/meta-llama-3-1/`

4-bit quantization via Exllama V2[5]. We adapted the instruction from AugGPT [108], by providing:

`Instruction: You are a helpful assistant that rephrase text and make sentence smooth. I will give you a sample, please rephrase it, then give me 10 rephrased answers. Sample test:`

This is then followed by one sample text to be rephrased. Usually the outcome is an ordered list of 10 rephrases, if no error occurred.

6.2.6 Resample

I used resampling as a baseline due to its simplicity. Depending on the specific goals, when more examples are needed, new ones are generated with sampling with replacement; when fewer examples were needed, I performed under-sampling as described in the SMOTE paper [46].

6.3 Methods for Adjustment through Augmentation - DistMatch

In the setting of confounding by provenance, spurious correlations in the training set can arise from the imbalance of subpopulations. Under my simulation framework, the α_{train} is an indicator for such imbalance. In the training set, $\alpha_{train} > 1$, for example, indicates the prevalence rate from subpopulation $Z = z_2$ is higher than that from $Z = z_1$. In order to build robust models, I propose the DistMatch framework, with the goal of matching the distributions across subpopulations in the training set. Specifically, it targets toward building $\alpha_{train} = 1$, where both provenances provide subsets with the same positive rates. I choose this balanced ratio of 1 as the target to adjust the original training distribution, under which scenario there is no provenance-related bias to learn. This method aims to remove the relationship between the provenance variable and the target label, thus promoting increased robustness under provenance shift. The DistMatch framework can work with any data

[5] `https://github.com/turboderp/exllamav2`

augmentation techniques and resampling, providing a rigorous guideline for making data augmentation and/or resampling.

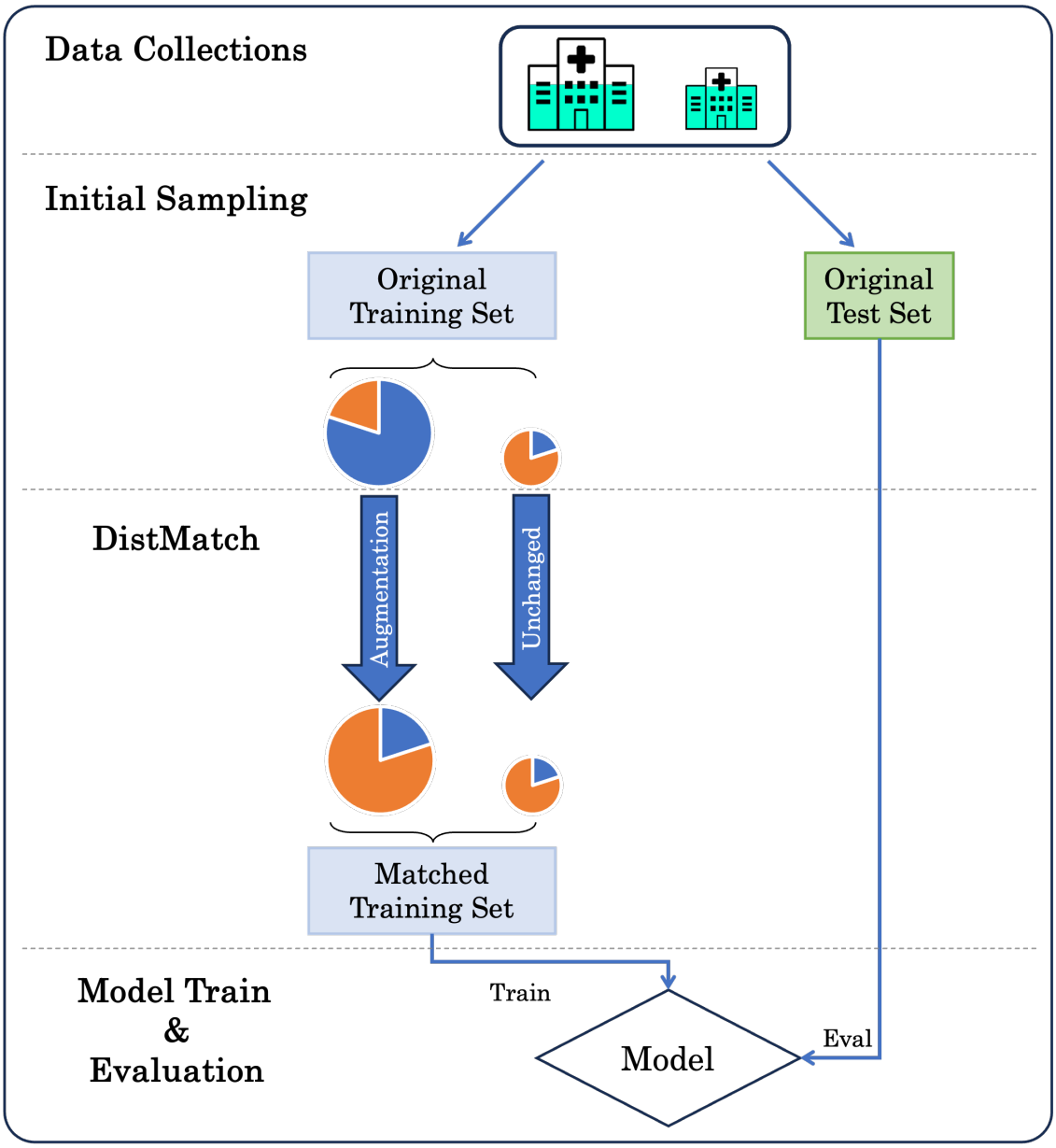


Figure 6.2: DistMatch framework diagram.

The full architecture of the DistMatch framework and information flow are shown in

Figure 6.2. DistMatch only focuses on α_{train} and tries to make a balanced training set (target: $\alpha_{train} = 1$) by performing over-sampling of the positive class together with under-sampling of the negative class on the subset with lower positive rates, while keeping the other subset unchanged. The targeted sizes for subsets and mixture proportions were kept unchanged in order to remove the effect of subpopulation composition in the training set.

Specifically, for an $\alpha_{train} > 1$, this means $P_{train}(Y = 1|Z = z_2) > P_{train}(Y = 1|Z = z_1)$. The DistMatch framework then keeps the training subset of $Z = z_2$ unchanged, as the target for distribution matching. For the training subset of $Z = z_1$, which has a lower positive rates, the aim is to bring this up to match $P_{train}(Y = 1|Z = z_2)$. Given the binary setting, there are two components consisting of positive cases and negative ones, denoted as two sets $\mathcal{D}^+_{z_1}$ and $\mathcal{D}^-_{z_1}$, respectively. Then over-sampling through data augmentation is performed on the $\mathcal{D}^+_{z_1}$ set to make its cardinality equal to $|\mathcal{D}_{z_1}| * P_{train}(Y = 1|Z = z_2)$, and under-sampling on the $\mathcal{D}^-_{z_1}$ to make its cardinality equal to $|\mathcal{D}_{z_1}| * (1 - P_{train}(Y = 1|Z = z_2))$. It can be observed that the size of $\mathcal{D}_{z_1}$ (set from $Z = z_1$) after DistMatch remains the same. The exact procedure is applied when $\alpha_{train} < 1$, except that the over-sampling (through augmentation) and under-sampling procedures now are applied in the subpopulation of $Z = z_2$ at this time.

6.4 Datasets and Experiment Settings

6.4.1 Comparing Augmentation Methods

In the first step of building an augmentation framework and evaluating the effectiveness of different data augmentation methods, I used the community-based text-message intervention dataset for cognitive distortion detection (the TM set, which was fully described in Chapter 4.2). For this specific task, five common distortions were selected [110]: Mental Filter (MF), Jumping to Conclusions (JC), Catastrophizing (C), Should Statements (SM), Overgeneralization (O). In addition, we added the label Any Distortion (AD), generated in accordance with the other assigned distortions. It is worth noting that any message could be identified as having multiple distortions, or no distortions at all, making this a multi-label

multi-class problem. Table 6.1 shows the label frequency.

	AD	**C**	**MF**	**JC**	**O**	**SM**
Frequency	24.4%	14.8%	8.6%	8.1%	3.6%	2.6%

Table 6.1: Label frequency for five common distortions and AD in the TM set.

Due to the algorithms used in each augmentation method, there are often limits to how many augmentations can be made per sample, which can be expressed in the form of a ratio. For Back Translation, it is 1:1. For Easy Data Augmentation, given there are 4 main operations (Synonym Replacement, Random Insertion, Random Swap, and Random Deletion), I only generated one augmentation for each operation per sample, thus the ratio is 4:1. For two fine-tuned GPT-2 generative models, I kept this ratio to 0.5:1. These ratios were tested in a range search and the best within each method was selected.

For this experiment, I used BERT as the text classifier [79]. The main framework for evaluation is 5-fold cross validation, and out-of-sample predictions were collected for the whole dataset. For the BERT model, following the original paper [74], I used the best hyperparameter settings for each of the iterations, as shown in Table 6.2. Also, losses were weighted inversely proportional to label frequencies.

Iteration	**#1**	**#2**	**#3**	**#4**	**#5**
number of epochs	14	14	10	14	8
dropout	0.2	0.3	0.1	0.2	0.2

Table 6.2: BERT hyperparameter settings.

Easy Data Augmentation was labeled as “BERT (EDA)”; the two back translation models as “BERT (BT:German)” and “BERT (BT:Spanish)”, depending on which intermediate

language was used; context-agnostic GPT-2 generation as "BERT (GPT-2: no context)"; contextual GPT-2 generation as "BERT (GPT-2: contextual)"; `mixup` as "BERT (`mixup`: $\alpha = X$)", with varying hyperparameter α settings.

To investigate the utility of domain-specific models for transfer learning, I identified a domain-specific pretrained language model. Ji, Zhang, Ansari, *et al.* [111] describe MentalBERT and MentalRoBERTa, two language models developed specifically for mental health NLP. Starting with pretrained base models, and following standard BERT and RoBERTa pretraining protocols, MentalBERT and MentalRoBERTa were further pretrained on subreddits in the mental health domain, including "r/depression", "r/SuicideWatch", "r/Anxiety", "r/offmychest", "r/bipolar", "r/mentalillness", and "r/mentalhealth". These subreddits made up a pretraining corpus of over 13 million sentences. Upon evaluation, this additional pretraining improved performance in classifying mental health conditions, including depression, stress, and anorexia. However, the evaluation sets used texts from online or SMS-like platforms, which were not fully annotated by specialists. In our work, we used MentalBERT, available from HuggingFace [6]. The same hyperparameters as the BERT model were used for comparison purposes. The baseline MentalBERT model is referred as "MentalBERT (no aug)". We also applied the best-performing data augmentation methods to MentalBERT, including back translation (Spanish) and explored some α settings for `mixup`.

6.4.2 Rephrasing

For data augmentation through rephrasing using open-source Large Language Models, the Cognitive Distortion set was used for evaluation. The model used was Llama-3.1 405B[7] with through 4-bit quantization. I tested rephrase generations on one set of 800 examples (equal sizes from the TM and AVH set), with 270 labeled cognitive distortion samples.

---

[6] `https://huggingface.co/mental/mental-bert-base-uncased`

[7] `https://ai.meta.com/blog/meta-llama-3-1/`

6.4.3 Models for Evaluating the DistMatch Framework

To evaluate the effectiveness of the DistMatch framework, I first tested it in the established settings of a logistic regression model. Under this model, I used `mixup` and resampling as techniques for generating examples, and only applied LLM rephrasing to the Cognitive Distortion dataset.

Next, I fine-tuned RoBERTa models under the DistMatch framework. I used `mixup` and resampling techniques as described in previous sections, and added Easy Data Augmentation (EDA) as another approach. To further check the effect of the DistMatch framework, I implemented for each example baselines under standard augmentation or upsampling scheme by generating samples 4 times in the original training set. This scheme is in accordance with EDA where 1 new example is generated for each of 4 permutation operations (SR, RI, RS, RD) on one input sample. This approach 4 other baseline models (besides no DistMatch): "mixup-4 times", "resample-4 times", and "EDA-4 times".

For all the settings, I fine-tuned RoBERTa model for 20 epochs on the Cognitive Distortion and Hate Speech Detection datasets and 6 epochs on the SHAC dataset.

6.5 Results on Low-Resource Classification with Data Augmentation

Performance for all models is shown in Table 6.3.

BERT: For the baseline BERT model, BERT (no aug), results show an AUPRC of 0.518 for the most frequent classes (AD,C). When frequency decreases (classes MF,JC), the AUPRC also drops to 0.372, and it drops further to 0.214 for the rarest classes of O,SM. This trend applies to all models. When data augmentation is applied to the base BERT model, there are improved results with different models. For the most frequent class of AD,C, back translation using Spanish achieves the highest AUPRC of 0.521, followed by `mixup`: $\alpha = 0.02$. However, none of these results are significant improvements over baseline BERT. For the less frequent classes (MF,JC), back translation outperforms baseline BERT by 1.5%. `mixup` does not offer a performance boost here. When it comes to the rarest classes (O,SM),

improvement is clearer: EDA, back translation (Spanish), and most settings of `mixup` can offer a boost in AUPRC. Among them, `mixup` ($\alpha = 4$) shows the biggest improvement in AUPRC by around 1.6%, which is statistically significant ($t(8) = 3.24, p\text{-}value = .012$). It is also notable that both GPT-2 based data augmentation methods decreased the performance of the base BERT model substantially (0.47 vs 0.52 for AD,C and 0.14 vs 0.21 for O,SM).

MentalBERT: When comparing results generated by MentalBERT with the ones from BERT, we observe improved performance for all classes, with the highest change for AD,C and MF,JC of 1.3%-1.8%. Similarly to BERT models, performance is highly related to class frequencies, with highest being 0.5359 for the most frequent class of AD,C, dropping to 0.385 for MF,JC then 0.217 for O,SM. This trend holds for different augmentation settings. For augmentation effects, the base model performs best for the more frequent classes of AD,C and MF,JC, as compared with augmented models. For rare class of O,SM, there is a small improvement from back translation (Spanish) of 0.5%. None of the `mixup` configurations provide a benefit over the base MentalBERT model.

model	AUPRC (high freq:AD,C)	AUPRC (medium freq:MF,JC)	AUPRC (low freq:O,SM)	macro-AUPRC†
BERT (no aug)	0.518 ± 0.0055	0.372 ± 0.0054	0.214 ± 0.0039	0.368 ± 0.0030
BERT (EDA)	0.517 ± 0.0062	0.378 ± 0.0071	0.228 ± 0.0091*	**0.374± 0.0067**
BERT (BT: German)	0.517	0.375	0.216	0.369
BERT (BT: Spanish)	**0.521**	**0.386**	0.222	0.376
BERT (GPT-2: contextual)	0.472	0.290	0.143	0.302
BERT (GPT-2: no context)	0.460	0.306	0.155	0.307
BERT (`mixup`: $\alpha = 0.02$)	0.519 ± 0.0013	0.372 ± 0.0026	0.218 ± 0.0078	0.370 ± 0.0041
BERT (`mixup`: $\alpha = 0.2$)	0.515 ± 0.0060	0.369 ± 0.0027	0.218 ± 0.0061	0.367 ± 0.0041
BERT (`mixup`: $\alpha = 0.5$)	0.510 ± 0.0058	0.367 ± 0.0058	0.213 ± 0.0034	0.363 ± 0.0033
BERT (`mixup`: $\alpha = 1$)	0.504 ± 0.0072	0.367 ± 0.0076	0.221 ± 0.0047	0.364 ± 0.0055
BERT (`mixup`: $\alpha = 2$)	0.505 ± 0.0043	0.366 ± 0.0046	0.222 ± 0.0054*	0.364 ± 0.0021
BERT (`mixup`: $\alpha = 4$)	0.505 ± 0.0048	0.367 ± 0.0027	**0.229 ± 0.0081***	0.367 ± 0.0038
BERT (`mixup`: $\alpha = 8$)	0.504 ± 0.0045	0.366 ± 0.0057	0.218 ± 0.0059	0.363 ± 0.0030
MentalBERT (no aug)	**0.536 ± 0.0029***	**0.385 ± 0.0059***	0.217 ± 0.0018	**0.379 ± 0.0032***
MentalBERT (BT: Spanish)	0.520	0.380	**0.222**	0.374
MentalBERT (`mixup`: $\alpha = 0.02$)	0.529 ± 0.0050*	0.379 ± 0.0031*	0.211 ± 0.0052	0.373 ± 0.0022*
MentalBERT (`mixup`: $\alpha = 0.2$)	0.523 ± 0.0033	0.382 ± 0.0049*	0.216 ± 0.0030	0.374 ± 0.0030*
MentalBERT (`mixup`: $\alpha = 1$)	0.520 ± 0.0064	0.381 ± 0.0056*	0.214 ± 0.0068	0.372 ± 0.0020*
MentalBERT (`mixup`: $\alpha = 4$)	0.515 ± 0.0028	0.379 ± 0.0021*	0.215 ± 0.0063	0.370 ± 0.0028
MentalBERT (`mixup`: $\alpha = 8$)	0.515 ± 0.0049	0.377 ± 0.0037	0.213 ± 0.0060	0.368 ± 0.0044

Table 6.3: AUPRC (mean ± std) for combined labels by frequency. *: significantly > BERT (no aug), unpaired t-test. †macro-AUPRC: macro-averaged AUPRC scores.

`mixup`: We explored an extensive range of the hyperparameter α with the BERT model. In Table 6.3, the best results typically come with a small α (0.02) for the dominant classes of AD,C and MF,JC. This best setting shows an increase of 1-2%. With an increasing α, the performance drops. For the rare classes of O,SM, a small α is no longer favored. The performance of AUPRC is not monotonic: with an increasing α, it first increases then drops, with its peak of 0.2285 at $\alpha = 4$. A similar trend is also observed for the MentalBERT model, although `mixup` did not perform best in this case, and improvements over the base BERT model may be attributable to in-domain pre-training.

The results discussed above show effectiveness in utilizing data augmentation techniques to improve prediction accuracy, especially on the rare classes. These techniques mainly focus on increasing data diversity in predictors (texts) while maintaining the label distribution. One exception is `mixup`, which implicitly changes the label distribution. It is hard to quantify the degree of this change since it introduces new target class labels that are no longer categorical. These implications for the problem of data imbalance are tested for their validity in the next section, under the DistMatch framework to adjust provenance shift.

6.6 Results on Classification under Provenance Shift with DistMatch Framework

6.6.1 DistMatch On Sentence-BERT and Binary Unigrams

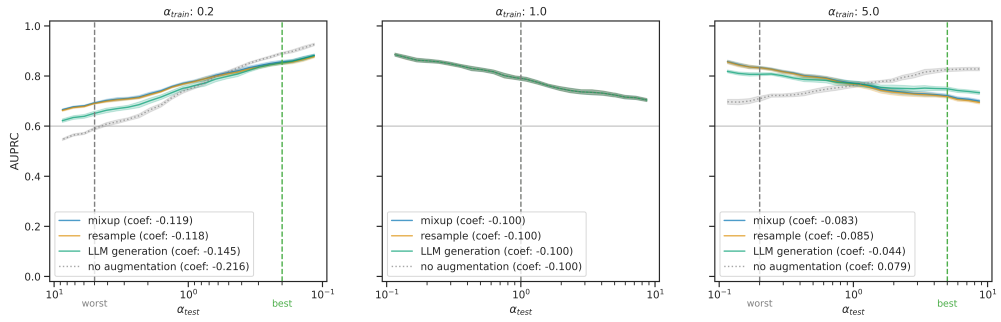
Using the DistMatch framework for provenance shift adjustment, results on all three datasets are reported with augmentation techniques (including LLM generation, if applicable). Figure 6.3 shows results based on models trained using Sentence-BERT embeddings, Figure 6.4 on models trained using binary unigrams.

Overall, models trained under the DistMatch framework show better robustness than the baseline model trained without any augmentation when there exists a provenance bias in the original training dataset, i.e., when $\alpha_{train} \neq 1$. This improvement in robustness is independent of the dataset and text representation method. This is shown in the leftmost and rightmost columns of Figure 6.3 for the Sentence-BERT embeddings and Figure 6.4 for binary

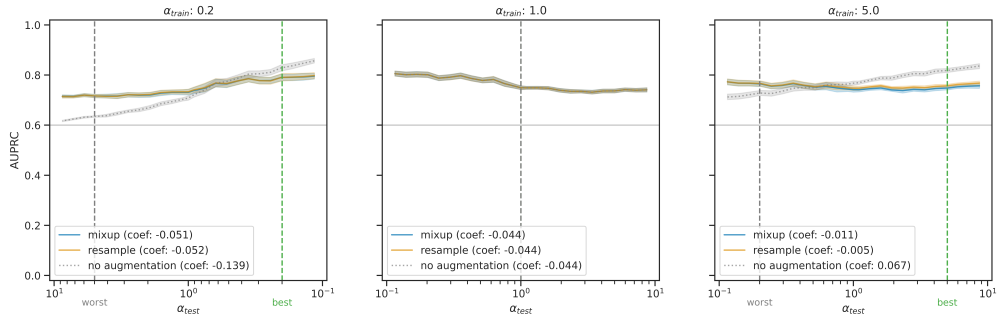
unigrams. For example, the model trained under the DistMatch framework with resampling can achieve a robustness measure of $|\beta| = 0.118$, and with `mixup` achieves $|\beta| = 0.119$, in comparison with the baseline model without augmentation with $|\beta| = 0.216$, when evaluated on the Cognitive Distortion set with $\alpha_{train} = 0.2$ in Figure 6.3a. When $\alpha_{train} = 1$, no augmentation mechanism was employed in the training process, because according to the DistMatch framework, $P_{train}(Y = 1|Z = z_1) = P_{train}(Y = 1|Z = z_2)$ thus there is no need for augmentation. When using LLM rephrasing for text generations, results on the Cognitive Distortion set show comparable robustness to other augmentation techniques (`mixup` and resampling). In some cases, the DistMatch framework with LLM generations can make models more robust than other techniques, such as with $|\beta| = 0.044$ (vs. 0.083 and 0.085 for `mixup` and resampling, respectively) when $\alpha_{train} = 5.0$ using Sentence-BERT embeddings.

There are two point estimates of interest: $AUPRC_{best}$ and $AUPRC_{worst}$, which corresponds to the model's performance when no provenance shift happens ($\alpha_{train} = \alpha_{test}$) and a "worst" case of shift ($\alpha_{train} = 1/\alpha_{test}$). Those two points are experimentally estimated at their respective α_{test} values, as indicated in Figure 6.3 and Figure 6.4 by a green vertical ("best") line and a gray vertical line ("worst"). In the results, the proposed DistMatch framework always leads to better worst-case performances and lower best-case performances, where the original training dataset is not balanced in terms of prevalence rates across provenances ($\alpha_{train} \neq 1$). This is observed across all three datasets regardless of text representation used. The improved worst-case performance indicates that a balanced training set (in terms of positive rate ratio α) can diminish or remove the potential provenance effect, leading to a model more "neutual" and robust thus more effective under worst-case scenarios. In comparison, when evaluated on the best-case scenarios, such purposeful balancing can make the training set shift away from the testing set. For best-case scenarios, imbalance is useful in way of maintaining the same distribution across training and test time. Thus, breaking this imbalance will hurt performance.

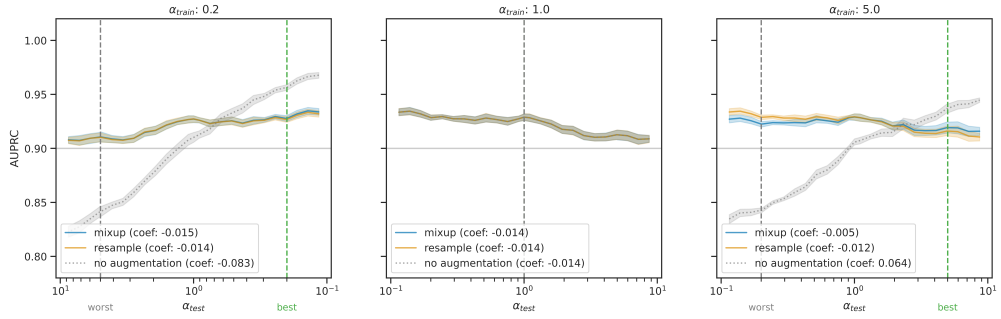
Detailed estimates for the worst-case performances on all three datasets are presented in Table 6.4 and Table 6.5 for Sentence-BERT embeddings and binary unigram representations



(a) CD

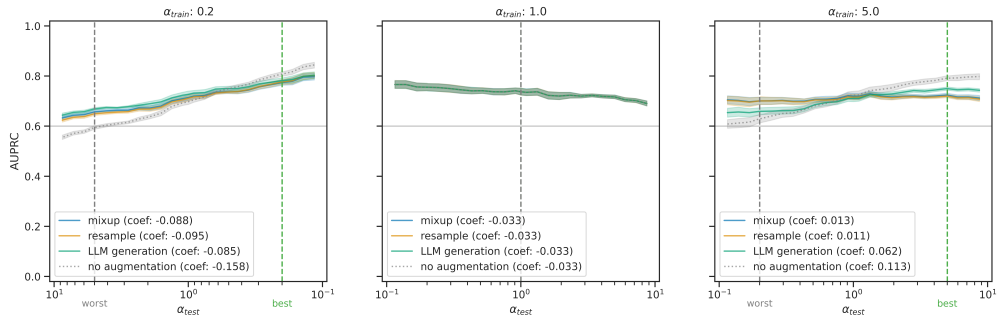


(b) HateSpeech

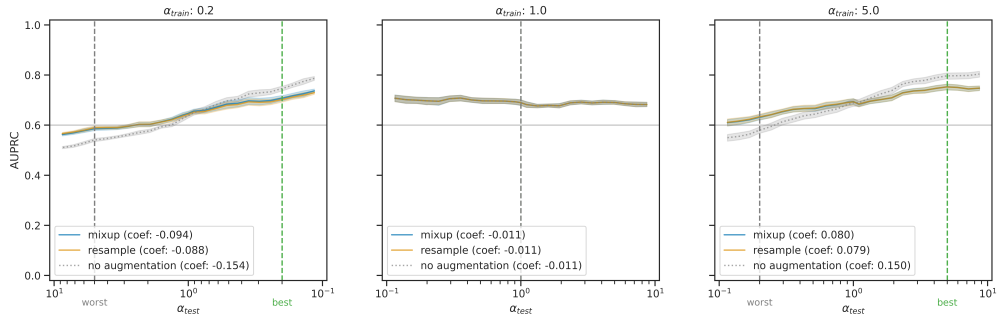


(c) SHAC

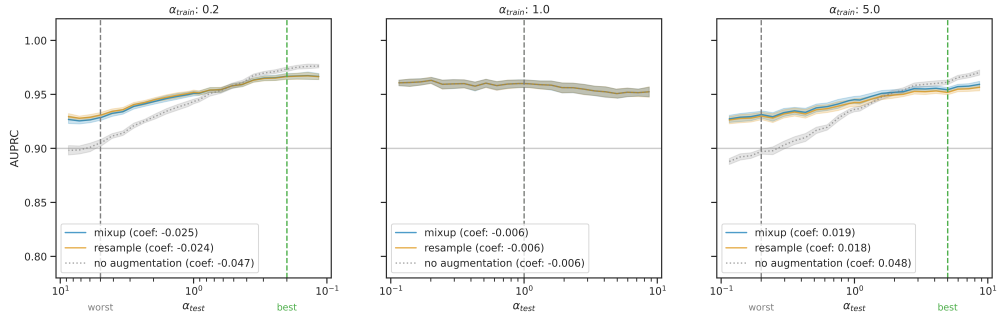
Figure 6.3: Results under the DistMatch framework for different α_{train} (in the column headers), using Sentence-BERT. (a) are results on the Cognitive Distortion dataset; (b) on the Hate Speech dataset; (d) on the SHAC dataset. Slopes of the fitted line (measured by coefficients) are reported for all settings.



(a) CD



(b) HateSpeech



(c) SHAC

Figure 6.4: Results under the DistMatch framework for different α_{train} (in the column headers), using binary unigrams. (a) are results on the Cognitive Distortion dataset; (b) on the Hate Speech dataset; (d) on the SHAC dataset. Slopes of the fitted line (measured by coefficients) are reported for all settings.

respectively. The improvements provided by the DistMatch framework over no augmentation are on the order of 0.01-0.10 increases in $AUPRC_{worst}$. The Cognitive Distortion and Hate Speech Detection datasets benefit more from DistMatch than the SHAC dataset. For example, models trained on the Cognitive Distortion dataset with $\alpha_{train} = 0.2$ have $AUPRC_{worst}$ of 0.69 ± 0.01 with `mixup`, an improvement over 0.59 ± 0.02 with no augmentation (Table 6.4). This trend holds for both methods of text representations. Even though, models trained on Sentence-BERT embeddings show higher $AUPRC_{worst}$ than those trained on binary unigrams. This gap is large for the Hate Speech Detection dataset, for example, 0.77 ± 0.04 vs. 0.63 ± 0.04 for corresponding representations when both models trained on the $\alpha_{train} = 5.0$ set using DistMatch with `mixup`.

With models trained under the DistMatch framework showing better robustness and worst-case performance, it can be observed from Figure 6.3 and Figure 6.4 that there is typically a cross point between baseline model and the augmented model. This point is not fixed, but usually lies around $\alpha_{test} = 1$. This cross shape also geometrically indicates a lower best-case performance given a higher worst-case performance, which is demonstrated in the figures. Usually the gap in $AUPRC_{best}$ is smaller than that in $AUPRC_{worst}$, further indicating benefits of improved robustness and worst-case $AUPRC$ with a relatively small sacrifice of best-case $AUPRC$.

α_{train}	Dataset	mixup	resample	LLM	no aug
0.2	CD	0.69 ± 0.01	0.69 ± 0.01	0.65 ± 0.03	0.59 ± 0.02
	Hate Speech	0.72 ± 0.02	0.72 ± 0.02	-	0.63 ± 0.01
	SHAC	0.91 ± 0.02	0.91 ± 0.01	-	0.84 ± 0.02
5.0	CD	0.83 ± 0.02	0.83 ± 0.02	0.81 ± 0.02	0.71 ± 0.04
	Hate Speech	0.77 ± 0.04	0.77 ± 0.04	-	0.73 ± 0.03
	SHAC	0.92 ± 0.01	0.93 ± 0.01	-	0.84 ± 0.01

Table 6.4: Worst-case performance, $AUPRC_{worst}$, under the DistMatch framework with different augmentation techniques. Results are from experiments using Sentence-BERT embeddings.

α_{train}	Dataset	mixup	resample	LLM	no aug
0.2	CD	0.66 ± 0.02	0.65 ± 0.02	0.67 ± 0.02	0.59 ± 0.02
	Hate Speech	0.59 ± 0.02	0.59 ± 0.02	-	0.54 ± 0.02
	SHAC	0.93 ± 0.01	0.93 ± 0.01	-	0.91 ± 0.01
5.0	CD	0.70 ± 0.05	0.70 ± 0.05	0.66 ± 0.05	0.63 ± 0.06
	Hate Speech	0.63 ± 0.04	0.63 ± 0.04	-	0.58 ± 0.04
	SHAC	0.93 ± 0.01	0.93 ± 0.01	-	0.90 ± 0.01

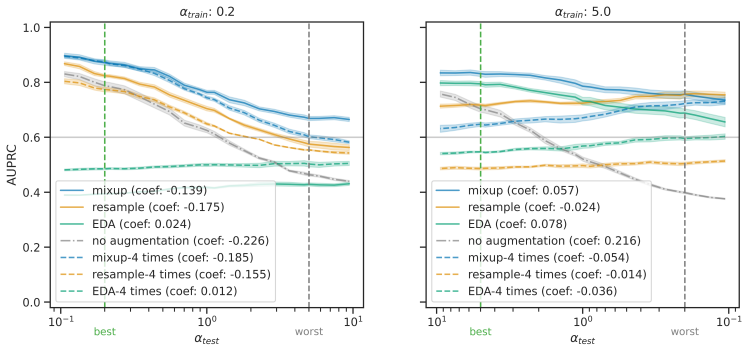
Table 6.5: Worst-case performance, $AUPRC_{worst}$, under the DistMatch framework with different augmentation techniques. Results are from experiments binary unigram representations.

6.6.2 DistMatch on Fine-tuning RoBERTa

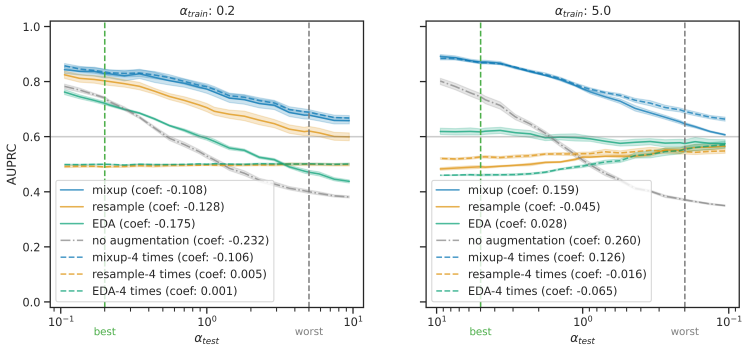
Results are shown in Figure 6.5. Two data augmentation techniques (`mixup` and EDA) and resampling were used in the DistMatch framework and they are the solid lines in the figure. For each technique, its baseline version where 4 new examples were generated per input sample is shown as the dotted line with the same color, which serves as a comparison. The gray dashed line is the RoBERTa model fine-tuned on the original training set where DistMatch was not applied.

Overall, the `mixup` technique under the DistMatch framework performs best in terms of both worst-case and best case scenarios in most settings. One exception for $AUPRC_{worst}$ is on the Hate Speech dataset when trained on $\alpha_{train} = 5.0$, in which setting the approach of `mixup` by 4 times is the best and `mixup` under DistMatch comes close after it. Another exception for $AUPRC_{best}$ on the SHAC dataset with $\alpha_{train} = 0.2$, where all models are worse than the baseline model. But in that area, models under the DistMatch framework are very close to the best performer of the baseline model and show great improvements over the augmentation or resampling techniques.

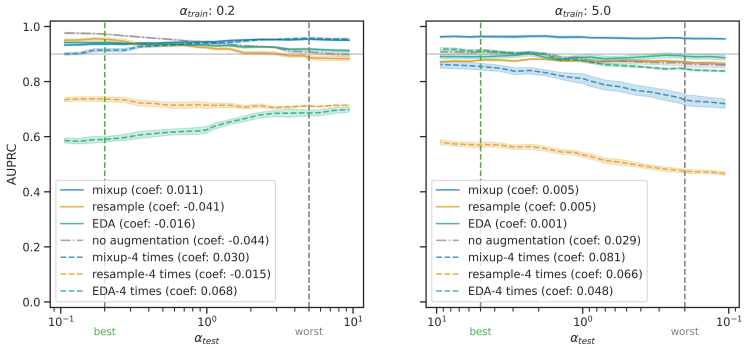
Across all degrees of provenance shift, models under the DistMatch framework consistently performs better than the baseline model (no DistMatch and augmentation), with the only exception in the region where α_{test} is less than around 1 on the SHAC dataset with $\alpha_{train} = 0.2$. As a comparison, the augmentation or resampling techniques without DistMatch perform poorly except the `mixup`. When it is used as augmentation technique without DistMatch, it outperforms other techniques, and can come close after the version under DistMatch or even outperform it on the Hate Speech dataset with $\alpha_{train} = 5.0$ around worst case scenario.



(a) CD



(b) HateSpeech



(c) SHAC

Figure 6.5: Results from RoBERTa under the DistMatch framework. (a) shows results on the Cognitive Distortion dataset; (b) on the Hate Speech dataset; (d) on the SHAC dataset. Each augmentation technique (solid line) is paired with its baseline (dotted line).

Fine-tuned RoBERTa model robustness is reported in Table 6.6. Each model under the DistMatch framework with specific augmentation or resampling technique is paired with the model using the same technique but no DistMatch framework (indicated by the column "Expand by 4").

		$\alpha_{train} = 0.2$		$\alpha_{train} = 5.0$	
Dataset	Adjustment	DistMatch	Expand by 4*	DistMatch	Expand by 4*
CD	no augmentation	-0.226		0.216	
	mixup	-0.139	-0.185	0.057	-0.054
	resample	-0.175	-0.155	-0.024	**-0.014**
	EDA	0.024	**0.012**	0.078	-0.036
Hate Speech	no augmentation	-0.232		0.260	
	mixup	-0.108	-0.106	0.159	0.126
		-0.128	0.005	-0.045	**-0.016**
	EDA	-0.175	**0.001**	0.028	0.065
SHAC	no augmentation	-0.044		0.029	
	mixup	**0.011**	0.030	0.005	0.081
	resample	-0.041	-0.015	0.005	0.066
	EDA	-0.016	0.068	**0.001**	0.048

Table 6.6: RoBERTa model robustness with and without the DistMatch framework. *Expand by 4: augmentation or resampling techniques are applied to each of input text to generate 4 new samples, to increase the training set size by 4 times as the original.

The DistMatch framework and augmentation or resampling techniques can both improve fine-tuned RoBERTa model's robustness across all settings on the three datasets. The DistMatch framework leads to the best robustness on the SHAC dataset. Augmentation and

resampling techniques show great robustness improvements compared to the baseline model and the models under the DistMatch framework. Those techniques can help to train a model that has a very small absolute coefficient value (smallest on the Cognitive Distortion and Hate Speech datasets), but the performance is usually poorer than with the counterparts under the DistMatch framework.

6.6.3 Why does LLM rephrasing not perform well?

This section focuses on the problem of why LLM rephrasing sometimes fails. Table 6.7 lists several common cases where the quality of the generated rephrasing is low or its outcomes are not usable. The most common failure scenario is that the Large Language Model (Llama-3.1 405B) refuses to generate any text. This happens when the input text for rephrasing contains harmful or biased contents, thus triggering safety precautions[8]. Such implementations can (most of the time) ensure that the open-sourced LLM will only perform in a helpful way, however, it prevents some distorted thinking patterns being generated (though not all of them).

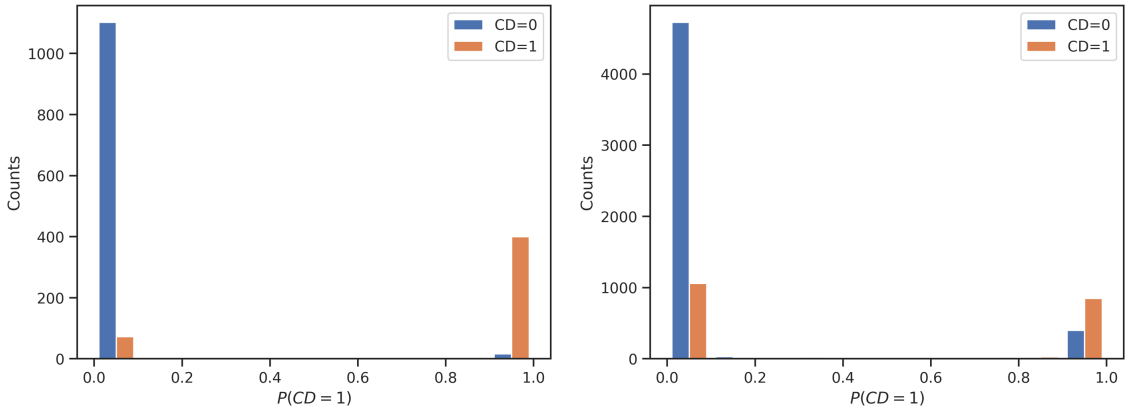
Categories for LLM Augmentation	N
"cannot create" or "cannot provide" or "cannot fulfill"	24
"no text"	12
"It seems like your sample text got cut off"	1
Successful text examples for LLM augmentation	763
Total	800

Table 6.7: One Training Set of Cognitive Distortion Detection for LLM Augmentation

We evaluated the quality of LLM rephrases, tested in two additional experiments. The

[8] https://ai.meta.com/blog/meta-llama-3-1-ai-responsibility/

first was to train a BERT model on the balanced Cognitive Distortion dataset with original samples for predicting existence of Any Distortion. Then this model was tested on the texts from the original dataset and the LLM generated dataset, separately, with the same positive rate of 50%. Figure 6.6 shows the results on two datasets, respectively. From the figure, we can find that both distributions of predicted probabilities for $CD = 0$ and $CD = 1$ subsets are more separated in the original dataset (on the left) in comparison with those in the LLM generated dataset (on the right). This is especially the case for the $CD = 1$ subset, where the trained model assigned near half positive cases with very low probabilities, potentially leading to higher false negative rates. This suggests the appropriate label for augmented text may differ from that for the source text that was augmented.



(a) original dataset

(b) LLM generated dataset

Figure 6.6: Results for distributions of predicted probabilities on the (a) original dataset vs (b) LLM generated dataset, separately. Subgroups based on Any Distortion conditions are reported separately in different colors.

The second experiment was to train a classifier for differentiating original texts vs LLM generated texts. This model was trained on a combination of those two sources, with same samples and same positive rates. Results in Figure 6.7 show that such a classifier can accu-

rately differentiate original texts from LLM generated texts, though the misclassification rate is slightly higher for LLM generated texts, indicating only in some cases that the generated texts are hard to detect.

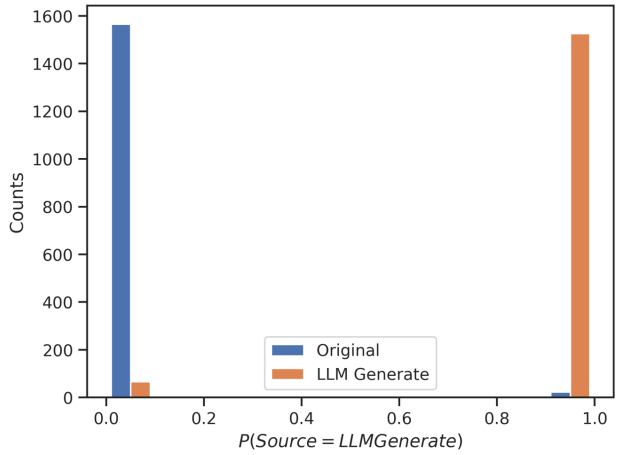


Figure 6.7: Results for distributions of predicted probabilities. Subgroups based on source (original texts vs LLM generated texts) are reported separately in different colors.

Combining the results from those two experiments, we observe that, even with 763 successful rephrasing cases out of 800, the quality of these generations varies. The generated text has two issues: (1) the original meaning of a text may not be fully retained; and (2) the generations are easy to identify, which may lead to another spurious correlation with the outcome, on top of confounding by provenance.

6.7 Discussion

6.7.1 Augmentation as Techniques

We examined several data augmentation methods and explored their applications in BERT and MentalBERT for detecting distorted thinking in a modestly-sized set of text-based therapy messages. Grouping distortion classes by frequency, we found that most of data augmentation methods do not improve performance for frequent classes (frequency: 8-25%).

For rare classes (3%), `mixup` significantly improved AUPRC results by 1.6%. In comparison, the domain-specific pretrained language model MentalBERT, offered the greated benefit for dominant classes. However, MentalBERT also performed relatively poorly with rare classes. This may be due to the limited number of training examples. Another reason might be the fact that our text messages sometimes represent general conversations related to case management (e.g. appointment reminders) rather than the specific mental health related concerns that predominate in mental-health-related subreddits.

We also explored different settings for the hyperparameter α for the `mixup` method. For dominant classes, `mixup` favors a small α, which is consistent with previous work [95]. This indicates the model performs better with limited mixing of two random samples, generating cases where only one example predominates. In comparison, a larger α is favored for rare classes. According to Supplementary Figure 6.1, this means the model tends toward mixes in which the influence of individual texts is diluted, a possible way to create more variation in this low-resource scenario for the model to learn from. However, progressing to more extreme values ($\alpha = 8$) harms performance, and this cutoff point may change in other settings. Taken together, our results suggest that `mixup` is helpful for rare classes, but may compromise performance on frequent classes. Future work with `mixup` should include increasing the number of training epochs, since Zhang, Cissé, Dauphin, *et al.* [95] suggest that errors may be further reduced with more iterations of training.

Contrary to expectations, GPT-2-based data augmentation harmed performance in this context. It appears that some GPT-2 generated texts (Table 6.8) do not express cognitive distortions as intended. This is likely because the data are not large enough to fully train a "distorted" GPT-2 model. Another reason may be that our prompts are not associated with distorted text by GPT-2. Designing better prompts may be a fruitful direction for future work.

label	Generated Text
JC	Yes you understand that it's incredibly frustrating and a lot of hard work but it's not at all stressful
C	They don't talk to me anymore

Table 6.8: Examples of GPT-2 generated texts. JC: Jumping to Conclusions. C: Catastrophizing

6.7.2 Augmentation as An Adjustment Method - DistMatch Framework

In this chapter, we proposed the DistMatch framework and showed its contribution in universal improvements of robustness when used alone on all three datasets in all settings, no matter what text representations were used (Sentence-BERT embeddings or binary unigrams). It is noted that different augmentation techniques may have varying effects. `mixup` is usually better than resampling, but both methods behave quite similarly and are stable. The DistMatch framework was also evaluated for fine-tuning RoBERTa models, using `mixup` and EDA as data augmentation techniques, and resampling. Results show the DistMatch framework can consistently be used to train models with best performance across all degrees of provenance shift, with the only exception being the Hate Speech dataset with $\alpha_{train} = 5.0$. The DistMatch framework can also generate more robust models than the baseline without DistMatch. This is consistent with both the logistic regression model and the RoBERTa model.

LLM generation, as an augmentation technique which was only evaluated on the Cognitive Distortion dataset, leads to less robust models, except for the training set with $\alpha_{train} = 5.0$ when using Sentence-BERT embeddings. Even this case, LLM generation under the DistMatch framework still resulted in a model that is more robust than the baseline (coefficients: -0.044 vs. 0.079). Overall, results demonstrated the effectiveness of using the DistMatch

framework for making models more robust. The relatively poor performance from LLM generations can be potentially due to safety guardrails built into the model training process, as reported for Llama-3.1[9]. One direct result is refusal to generate or reconstruction of sentiments (usually towards a more positive tone) when Llama-3.1 was asked to rephrase the original sentences. This is an undesired behavior for the rephrasing task for both contextual embedding methods (Sentence-BERT in our case) and distributional representations (binary unigrams in our case), making models less robust and unpredictable. By rephrasing, we expect the language model to keep the sentiment as close to the original as possible, instead of generating more "helpful" outputs, even though this safety guardrails are essential when released to the public. There are ways to bypass those safety constraints. One approach is "jailbreaking" the LLMs through Generative Techniques, Template Techniques, or Training Gaps Techniques [112]. For example, using an attacker LLM to automatically generate prompts exploiting Chain of Thought [113]; using techniques similar to SQL injection by integrating generative constraints and malevolent inquiries within specified templates [114]. Another approach is through editing a one-dimensional subspace that is found to elicit refusals such that safety guardrails could then be bypassed [115]. However, the research on that topic typically focuses on whether the resulting LLM can successfully generate harmful content, but not on rephrasing tasks, which are key to our framework. As such, it falls beyond the scope of current work.

This study shows the effectiveness of the proposed DistMatch framework in improving worst-case performance and robustness for the logistic regression and RoBERTa model. For the three datasets evaluated in the experiments a balanced training dataset, made available through the DistMatch framework, is important to model's robustness under provenance shifts. It should be noted that we evaluated the DistMatch framework with only four augmentation methods and two models. It remains as future work to test behaviors of different methods for augmentation and different models.

---

[9] `https://ai.meta.com/blog/meta-llama-3-1-ai-responsibility/`

Chapter 7

MANIPULATING HIDDEN SPACES FOR ADJUSTMENT

Building under the unified simulation and evaluation framework for provenance shift, previous chapters introduced statistical adjustment and distributional adjustment. In this chapter, I delve into neural networks (NN) and propose adjustment methods that are effective for them. Two approaches, one for directly manipulating the hidden spaces of NN (Provenance Effect Reduction), and another for indirectly interacting with hidden spaces (through robust learning), are the focus of this chapter. Both aim to eliminate influences of provenance on predictors (texts), through either direct arithmetic or forcing mixture of hidden spaces during the training process. This chapter provides a new perspective different from the statistical adjustment and distributional adjustment.

7.1 Introduction

Machine learning and deep learning models have been successfully applied for predictive modeling in many domains [1]–[3]. This includes their application to biomedical problems, including those at the molecular level, such as identifying potential new antibiotics [4] and predicting protein structure [5], [6]; at the patient level, such as electronic health record phenotyping [7], [8] and cancer diagnosis [9]; and at the institutional and policy levels, such as predicting demand at emergency departments [10]. A key requirement for successfully training and then deploying such systems is to use adequate amounts of diverse training data. Deep learning models, often with millions of parameters, require curated datasets large enough for the appropriate training procedures to achieve an accuracy adequate for deployment in practice [13], [116], [117]. Without such diversity, deep learning models tend to overfit on subtle institutional biases and may perform poorly when applied to data from

a new institution [118].

Training models on a diverse and balanced dataset to close the gap between traditionally underrepresented and overrepresented groups is a related research area, intended to address bias in AI. Daneshjou, Vodrahalli, Novoa, *et al.* [119] created the Diverse Dermatology Images (DDI) dataset to address disparities in dermatology AI performance. However, acquiring an adequately large and representative set of clinical data may exceed the resources of any single entity (a single study, a clinic/hospital, an institution). For example, researchers have utilized data modalities such as text messages [75], Magnetic Resonance Imaging (MRI) data [120], genomics data [121], symptom severity measures (MADRS, HRSD, BDI) [122], and multimodal smartphone-derived data [77] to study mental health conditions including cognitive distortion, depression, and auditory verbal hallucinations. Unlike studies using social media and electronic health records data where thousands and even millions of individuals have been involved [123], [124], the aforementioned studies involving advanced data modalities have relatively small sample sizes (number of patients), ranging from 39 to 793. These numbers may be adequate for the goals of the studies concerned, but are not sufficient to support the training of robust predictive models.

One feasible solution is to integrate data from multiple sources. In addition to supporting the development of robust models, this can also increase the diversity of the training population, improving generalizability. However, combining data from different sites may introduce bias and highly parameterized models may learn the source of a data element, and use this information to inform its predictions (which may well be correct within that particular dataset but are based on the wrong feature and therefore will not generalize to other data with a different distribution). For example, in natural language processing (NLP) - the focus of the current work - a model may learn to recognize section headers in clinical notes that are unique to a particular institution. This bias, where spurious associations between the source (provenance) and target (for prediction - e.g. assigning a diagnosis) distributions are learned by models trained on such datasets, is called *confounding by provenance*, and was identified in our previous work [17]–[20]. Such bias is detrimental when the composition

of datasets with differing provenance at the deployment time shifts away from that composition in the training set. This phenomenon has been demonstrated in our previous work [19] showing a 5% decrease in AUPRC score when using unadjusted compositions in training and test data from datasets with differing provenance. Though the focus of the current chapter is on this form of bias, we refer the interested reader to a broader discussion of distribution shift in medical data and its detrimental effects when left unadjusted [125], [126], and related problems with healthcare disparities and fairness [127]. Distribution shift in general is also an active field of research [128]–[130], but the specific problem of confounding by provenance is understudied by comparison.

In this chapter, we introduce two ways for manipulating hidden spaces of a given model: post-processing manipulation through Task Arithmetic and robust representation learning.

We propose **Task Arithmetic for Provenance Effect Reduction (TAPER)**, a method to attenuate effects from confounding by provenance while maintaining model performance. To develop TAPER, we adapt the idea of task vectors and its related task arithmetic introduced by Ilharco, Ribeiro, Wortsman, *et al.* [131] to the novel setting of *confounding by provenance*. We further develop the **Dominance-Aligned Polarized Provenance Effect Reduction** (DAPPER) approach, a task arithmetic tailored to cases in which positive examples from one source of data in the training set predominate. To efficiently apply our methods to large language models (LLMs), we utilize Low-Rank Adaptation (LoRA) for fine-tuning [132] and show equivalency to task vectors as originally proposed. A key benefit to using LoRA for generation of task vectors is that it preserves computational resources, and reduces the resource barrier to applying task vectors to LLMs.

Besides TAPER and DAPPER for post-processing of model weights, several commonly used techniques for representation learning are also evaluated in work described in this chapter: Maximum Mean Discrepancy (MMD) for Generalization, Group Distributionally Robust Optimization (GDRO), Domain-Adversarial Training.

The chapter proceeds as follows. First I revisit confounding by provenance and a simulation framework to evaluate robustness to it, developed in our previous work [19], [20]. Then

I introduce updates that make this framework more general, by removing some constraints. Next, I provide formal definitions and procedures for **TAPER** and **DAPPER**, and discuss their training settings (scaling hyperparameters). Robust learning methods are also introduced. Then, I present evaluations in which these methods were applied to two models of different architectures for NLP: RoBERTa and Llama. This is followed by detailed experiments on three datasets (two biomedical datasets and one general-domain dataset). The results suggest that our methods can improve model robustness to confounding by provenance, and improve performance at the extremes of distribution shift when **DAPPER** is applied. Finally, I present qualitative and error analyses, to inform the understanding of the effects of TAPER and DAPPER on provenance related confounding bias.

7.2 Background

7.2.1 Task Vectors

The task vector approach, proposed by Ilharco, Ribeiro, Wortsman, *et al.* [131], utilizes weight differences between a pretrained model and the target model with additional fine-tuning to characterize the changes that occur when a particular task is learned. Specifically, a pretrained model is used as the base model. Pretrained models have been trained on large datasets, and are expected to serve as foundations for fine-tuning towards specific downstream tasks. There are many pretrained models for different purposes, for example, BERT [79], FLAN-T5 [133], and Llama [134] for language in general, BioBERT [135] for biomedical language, Med-BERT [136] for structured electronic health records, ResNet [90], and ViT [137] for vision, to name just a few. Such pretraining requires considerable computational resources, so it typically only happens once, and the resulting model is expected to maintain some general knowledge of the training dataset (domain). Some extra work, if desired, is left for end users to further fine-tune such models, but at a significantly lower cost than the initial pretraining [138]–[140]. A key motivation for the task arithmetic approach is that the desired changes in performance may be either additive (e.g. improving performance on

a downstream task) or subtractive (e.g. constraining generation of toxic content). Task arithmetic can be defined as follows.

Task vectors are matrices containing differences in weights at corresponding positions (such as feed-forward layers, attentions, embedding, etc.). Following the original paper [131], we denote weights in such pretrained models as θ_{pre}. After fine-tuning towards a specific downstream task t, we denote the updated weights as θ^t_{ft}. The task vector for a specific task t, representing the changes that occurred during fine-tuning, is then defined as:

$$\tau_t := \theta^t_{ft} - \theta_{pre} \tag{7.1}$$

Task arithmetic allows for manipulation of task vectors, such that they are reoriented toward a desired direction in the weight space. Ilharco, Ribeiro, Wortsman, *et al.* [131] demonstrated three scenarios: (1) Forgetting via negation: $\tau_{new} = -\tau$; (2) Learning multiple tasks, here tasks A and B, via addition: $\tau_{new} = \tau_A + \tau_B$; (3) Task analogies: $\tau_{new} = \tau_C + (\tau_A - \tau_B)$. Pham, Marshall, Hegde, *et al.* [141] further used the idea of task vectors in text-to-image models, with an application resembling the first additive use case.

Neither deconfounding of classifiers in general nor the provenance shift problem in particular have been addressed using task vectors. They lie beyond the originally proposed use cases, requiring manipulation of two task vectors together, but differently from the multi-task learning scenario demonstrated in the original paper. In the case of the current work we are treating one task vector as primary (the behavior to retain: text classification) and the other as auxiliary (the behavior to remove: provenance recognition). In symbols, we intend to address provenance shift using $\tau_{new} = \tau_A - \lambda \times \tau_B$. Applying this idea to provenance effect reduction, where two task vectors are playing different roles (minuend vs subtrahend), controlled through the newly introduced variable λ, is a novel contribution of our work.

7.2.2 Robust Learning

In the field of domain generalization or out-of-distribution generalization, three categories of methods have been studied: data manipulation, representation learning, learning strat-

egy [142]. Optimization-based methods have aroused attention due to their data-agnostic and structure-agnostic features. Examples include Distributionally Robust Optimization (DRO) and Invariance-Based Optimization [130], which focus on controlling the worst-case prediction error among all subgroups. DRO defines the uncertainty set by constraints of f-divergence [143] or Wasserstein distance [144] from the training set.

Ben-David et al. theoretically showed that a good representation is the key to effective domain adaptation, and that this is one that achieves low training error and domain difference simultaneously [145]. Muandet, Balduzzi, and Schölkopf [146] also showed that, through a learning-theoretic analysis, "reducing dissimilarity [between different domains] improved the expected generalization ability of classifiers on new domains". Based on those findings, four lines of research around domain-invariant representation learning have been proposed, including kernel methods, feature (domain) alignment, domain adversarial learning, and invariant risk minimization [130], [142].

Domain Adversarial Training is a training strategy which promotes the emergence of features that are discriminative for the main learning task and indiscriminate for differentiating between domains. Domain Adversarial Neural Networks (DANN) under such strategy works by introducing a domain classifier and reversing its loss when performing backpropagation [23]. A simpler architecture proposed by Zhong and Ettinger [147] only introduces the additional domain classifier but not the reversed backpropagation.

Domain alignment, different from adversarial learning, learns domain-invariant representations through the alignment of features. Tzeng, Hoffman, Zhang, *et al.* [24] proposed deep domain confusion for maximizing domain invariance (or minimizing across-domain distance), measured by Maximum Mean Discrepancy (MMD). Other metrics for measuring distances between representations across domains have also been developed, including using Wasserstein distance [148] and the second order correlation (which is used in Deep CORAL [149]).

7.2.3 Provenance Shift

In Chapter 2, I define provenance shift as a type of distribution shift between the training and test sets. This is formally described through the inequality between distributions over the training and test set, as follows:

$$P_{train}(Y|Z) \neq P_{test}(Y|Z) \tag{7.2}$$

where Y is the variable of main interest, Z is the variable for provenance, e.g., subpopulations. This confounding effect of a provenance shift between the training and testing time is demonstrated in Figure 2.5.

In this chapter, I focus on the binary classification problem ($Y = \{0, 1\}$) and there are only two subpopulations ($Z = \{z_1, z_2\}$). I inherit the notions of this difference between two conditional distributions as the degrees of provenance shift, through two auxiliary variables, α_{train} and α_{test}. We use the unconstrained simulation framework proposed in Chapter 3, where conditions of (1) $P_{train}(Y) = P_{test}(Y)$ (the overall prevalence of the positive class is the same at training and test time); (2) $P_{train}(Z) = P_{test}(Z)$ (the relative contribution of examples from each site to the training and the test set is held constant), are dropped.

In the general domain of distribution shift, several methods have been proposed to make adjustments, including using importance weights for post-hoc adjustments [150], domain adaptation techniques [27], [28], and domain invariant representations [146], [151]. Post-hoc methods do not change model's behavior during the learning process, thereby posing limitations on their flexibility in more complicated scenarios. Domain adaptation and domain invariant representations involve generalizing training to unseen domains, which falls out of the scope of current work on confounding by provenance, where the provenance-specific positive rates of within-domain training data change at test or deployment time.

In Chapter 5 and previous work, we applied the Backdoor Adjustment method, developed by Landeiro and Culotta [15], in the context of our provenance shift simulation framework, and evaluated its effectiveness using different text representations with an adjusted logistic regression model [19].

In this chapter, we propose a different approach from statistical Backdoor Adjustment, using TAPER. This new procedure does not build on extracted text representations only, but rather fully fine-tunes a model towards different tasks, which can utilize language models to a deeper extent.

7.2.4 Evaluating Robustness to Provenance Shift

Models are tested under the unconstrained simulation framework described Chapter 3. For each test set, we use the Area Under the Precision Recall Curve (AUPRC) for its better discriminant ability in rare-case scenarios as compared with the Area under the Receiver Operating Characteristic curve [66]. In the simulation framework, which tests different degrees of provenance shift, one metric of evaluation is *robustness* [20]. This is quantified as the absolute value of the coefficient from the fitted regression line with AUPRC as the dependent variable and α_{test}, in the log scale, as the independent variable. Specifically, across all different testing scenarios, robustness is assessed as $|\beta|$ from:

$$AUPRC = intercept + \beta \times log_{10}(\alpha_{test}) \tag{7.3}$$

Intuitively, this measures the slope of the trend of a model's AUPRC values under different degrees of provenance shift. The flatter this trend is ($|\beta|$ is closer to 0), the more robust a model is, as this indicates that performance does not change with distribution shift.

Besides, a model's performance is evaluated under both best-case scenario and worst-case scenario, leading to two corresponding AURC's: $AUPRC_{best}$ and $AUPRC_{worst}$. They were described in detail in Chapter 3.

7.3 Methods for TAPER and DAPPER

To apply task arithmetic for provenance effect reduction, we follow the procedure introduced in the original work [131]. All of the weights of the model concerned - both before and after training - are flattened, and treated as a high-dimensional vectors, with the number of dimensions equal to the number of model weights. Like any vector, Task vectors permit

arithmetic operations. The three use cases (forgetting via negation, learning via addition, and task analogies) proposed by Ilharco, Ribeiro, Wortsman, *et al.* [131] demonstrated the flexibility of task arithmetic, and how to apply it to modify model behavior. We refer readers to the original paper for detailed explanations of each scenario. Though not originally proposed for this purpose, task vectors present an intuitive solution to the problem of debiasing models against confounding effects, where the goal is to preserve performance on a task of interest, while eliminating reactions to variables that induce spurious correlations.

Inspired by the flexible arithmetic mediated by task vectors, we adapt this method to the problem of provenance shift, where a new task vector (τ_t) is constructed by subtracting the task vector for provenance ($\tau_{provenance}$) from the task vector for the main target which is the main learning goal while spurious connections with provenance are present ($\tau_{t+provenance}$). The essential operation to mitigate confounding by provenance can be formally expressed as follows:

$$\tau_t = \tau_{t+provenance} - \tau_{provenance} \tag{7.4}$$

We proceed to explain how the task vectors concerned are constructed (Sections 7.3.1 and 7.3.2), and how they are combined using Task Arithmetic for Provenance Effect Reduction (TAPER) (Section 7.3.3). We then introduce Dominance-Aligned Polarized Provenance Effect Reduction (DAPPER), where the polarity of the task vector for provenance $\tau_{provenance}$ is aligned with the dominant class of Z (Section 7.3.4). Finally, we integrate Low-Rank Adaption (LoRA), a technique for efficiently fine-tuning large models, with Task Arithmetic for provenance effect reduction (Section 7.3.5).

7.3.1 Task Vectors for the Main Task

In order to obtain the task vector for the main classification task (e.g., identification of the disease corresponding to a piece of text in a clinical note), we use the primary classification goal for the main target (Y) to fine-tune a pretrained model. The dataset is a combination of data drawn from multiple sources, exemplifying the case in which all available data are

used for model training. Assume there are two provenance labels ($Z = \{z_1, z_2\}$), then the training set ($\mathcal{D}_{train}$) has the following composition:

$$\mathcal{D}_{train} = \{(X, Y, Z = z_1)_{i=1}^{N}, (X, Y, Z = z_2)_{i=1}^{M}\}$$

where N samples are drawn from $Z = z_1$ and M from $Z = z_2$, and at training time, predictors (X), primary task labels (Y) and provenance labels (Z) are all available, expressed as triples (X, Y, Z). Then, a classification function f can be learned through an empirical average loss function L, using the the optimization via:

$$\theta_{t+provenance} = \arg\min_{\theta} \frac{1}{N+M} \sum_{(\boldsymbol{X},Y)\in\mathcal{D}_{train}} L(f(\boldsymbol{X}|\theta), Y) \tag{7.5}$$

It is noted that in the presence of unequal class distributions across sources, a model trained on such a merged dataset, even with the main task as the primary goal, still has a tendency towards learning provenance information because this is associated with the probability of the primary outcome. Consequently, we assume the parameters learned above has combined information, thus the resulting parameters in Optimization Equation 7.5 are denoted as $\theta_{t+provenance}$. Following this, the task vector for the main task can be obtained as

$$\tau_{t+provenance} = \theta_{t+provenance} - \theta_{pre} \tag{7.6}$$

which represents the change in model weights during training for the primary task.

7.3.2 Task Vectors for the Provenance Task

For the auxiliary task of provenance classification (e.g., identifying the source of the data sample), we use the same training set $\mathcal{D}_{train}$ as for the main task. However, the prediction target now changes to provenance label Z. Following a similar procedure as above, the fine-tuned parameters for this task $\theta_{provenance}$ comes from the following optimization for the classification function g:

$$\theta_{provenance} = \arg\min_{\theta} \frac{1}{N+M} \sum_{(\boldsymbol{X},Z)\in\mathcal{D}_{train}} L(g(\boldsymbol{X}|\theta), Z) \tag{7.7}$$

Then, the task vector for the provenance task can be calculated as follows:

$$\tau_{provenance} = \theta_{provenance} - \theta_{pre} \tag{7.8}$$

7.3.3 Task Arithmetic for Provenance Effect Reduction (TAPER)

In TAPER, the selection of the positive class for the provenance task is arbitrary and we use $Z = z_2$ in the current work. Given the two task vectors for the main task and the provenance task, obtained from past sections, the adjusted task vector for provenance effect reduction can then be achieved using the procedure from Equation (7.4), while adding a scaling factor, λ:

$$\tau_t = \tau_{t+provenance} - \lambda \times \tau_{provenance} \tag{7.9}$$

The inclusion of a scaling factor can improve an arithmetically-adjusted model's performance [131], [141]. It also allows for generalization of this procedure, where the setting of $\{\lambda = 0\}$ represents model training without provenance effect reduction.

7.3.4 Dominance-Aligned Polarized Provenance Effect Reduction (DAPPER)

An added dimension of complexity to the problem of provenance shift concerns the positive class prevalence in data drawn from different sites, as outcome prevalence itself can have a significant effect on classifier performance [152], [153]. For the binary case, this happens when $\alpha_{train} \neq 1$, which indicates a difference in the proportion of the examples drawn from each site that fall into the positive class.

According to Definition (2.6), when $\alpha_{train} > 1$, we have $P_{train}(Y = 1|Z = z_2) > P_{train}(Y = 1|Z = z_1)$ and then define $Z = z_2$ as the *dominant* provenance class with a

higher positive rate (and vice versa for $\alpha_{train} < 1$ with $Z = z_1$ being the *dominant* provenance class).

Based on this observation, we further propose Dominance-Aligned Polarized Provenance Effect Reduction (DAPPER), where the $\tau_{provenance}$ model is trained by assigning the positive class label to the dominant provenance class. When $\alpha_{train} < 1$, the Optimization Goal (7.7) becomes:

$$\theta^{z_1}_{provenance} = \arg\min_{\theta} \frac{1}{N+M} \sum_{(\boldsymbol{X},Z)\in\mathcal{D}_{train}} L(g^{(z_1)}(\boldsymbol{X}|\theta), Z) \tag{7.10}$$

where the model is still trained on the same training set $\mathcal{D}_{train}$ but now uses z_1 as the primary provenance label. The distinction from Optimization Goal (7.7) is that the provenance classifier $g^{(z_i)}$ for each class z_i now predicts the probability of $Z = z_i$, the dominant class. When $\alpha_{train} > 1$, we have:

$$\theta^{z_2}_{provenance} = \arg\min_{\theta} \frac{1}{N+M} \sum_{(\boldsymbol{X},Z)\in\mathcal{D}_{train}} L(g^{(z_2)}(\boldsymbol{X}|\theta), Z) \tag{7.11}$$

where we now use z_2 as the primary provenance label.

Accordingly, the Dominance-Aligned task vectors for provenances are:

$$z_1 \text{ dominant} : \tau^{z_1}_{provenance} = \theta^{z_1}_{provenance} - \theta_{pre} \tag{7.12}$$

$$z_2 \text{ dominant} : \tau^{z_2}_{provenance} = \theta^{z_2}_{provenance} - \theta_{pre} \tag{7.13}$$

Then, with the task vectors for each dominant provenance class, the Dominance-Aligned Polarized Provenance Effect Reduction procedure uses Formula (7.9) to calculate the new task vectors, choosing the task vector for the dominant provenance class ($\tau^{z_1}_{provenance}$ or $\tau^{z_2}_{provenance}$) to substitute $\tau_{provenance}$ in the original formula. In summary, the Dominance-Aligned Polarized Provenance Effect Reduction procedure can be expressed as:

$$\tau_t = \begin{cases} \tau_{t+provenance} - \lambda \times \tau^{z_1}_{provenance}, & \alpha_{train} < 1 \\ \tau_{t+provenance} - \lambda \times \tau^{z_2}_{provenance}, & \alpha_{train} > 1 \end{cases} \tag{7.14}$$

The choice of which study site to label as $Z = z_2$ is arbitrary in TAPER. The key difference with DAPPER is that the task vector for provenance is oriented toward the dominant provenance class, which indicates the site Z that contributes the greatest number of positive examples to the training set.

7.3.5 LoRA Weights for Task Vectors

A recent trend in language modeling is an increase in model size. BERT is comparatively small with 108 million parameters (base version) [79]. RoBERTa has 125 million parameters [154]; the three versions of Llama 2 have 7, 13 billion, and 70 billion parameters, respectively [155]. This exponential increase in model size prohibits fine-tuning larger models for downstream tasks, given typical constraints on time and computing resources. Hu, shen, Wallis, *et al.* [132] proposed an efficient fine-tuning framework, Low-Rank Adaptation (LoRA), for models with high-dimensional weight update matrices. With LoRA, each weight update of dimension d is decomposed into two matrices with low ranks r, i.e., $\Delta W = BA$, where $\Delta W \in \mathbb{R}^{d\times d}$, $B \in \mathbb{R}^{d\times r}$, and $A \in \mathbb{R}^{r\times d}$. The changes during task-specific fine-tuning are confined to these LoRA matrices. With $r \ll d$ (usually $r = 2, 4, 8$ and $d = 512, 1024, 2048, ...$), the number of parameters stored through B and A is significantly less than that of the original weight updates (on the magnitude of $10^3 - 10^4$ depending on specifications of r and which matrices to decompose).

In the current work, we adapt LoRA weights (B and A) for task vector generation. From the original description of LoRA [132], it is apparent that for one specific weight matrix, at the end of fine-tuning, the final weights are:

$$W_{ft} = W_{pre} + \Delta W = W_{pre} + B_{ft}A_{ft} \tag{7.15}$$

Formula (7.15) has the form required for calculating task vectors in Formula (7.1), if we rearrange the terms as:

$$\tau_{ft} := B_{ft}A_{ft} = W_{ft} - W_{pre} \tag{7.16}$$

This observation suggests two ways of obtaining task vectors when LoRA is used: (1) apply matrix multiplication to the LoRA matrices, BA; (2) subtracting pretrained weights W_{pre} from fine-tuned weights W_{ft}. The first approach can substantially decrease resource usage during computation, and the second approach is straightforward to implement, as before.

The task vectors obtained through LoRA can then be used for TAPER and DAPPER with task vectors derived from LoRA weights, accordingly. For example, we can rewrite Formula (7.9) as follows:

$$\begin{aligned}\tau_t &= \tau_{t+provenance} - \lambda \times \tau_{provenance} \\ &= B_{t+provenance}A_{t+provenance} - \lambda \times B_{provenance}A_{provenance}\end{aligned} \tag{7.17}$$

This approach permits applying task arithmetic while using parameter-efficient fine-tuning, mediating its application to contemporary large language models within the typical computational constraints of an academic research institution.

7.3.6 Determining the Scaling Factors

Both TAPER and DAPPER have one scaling factor, λ, as shown in Formulae (7.9) and (7.14). This factor governs the resulting task vector's norm and direction, by controlling the magnitude of the subtracted task vector for provenance.

The targeted task vector's optimal norm and direction remain hard to determine beforehand. In this work, to examine the effects of scaling empirically, we varied λ from 0 to 2.0, with a step size of 0.2.

There are three point estimates of interest, λ_D (default), λ_O ("optimal"), and λ_E (estimated). We first define $\lambda_D = 1$ as the default setting, which is an arbitrarily chosen value. "optimal" value for λ is defined specifically for the worst-case performance for the model on each dataset, which means "optimal" λ could vary across models and datasets evaluated. For a specific model (RoBERTa or Llama-2), an estimation of λ values could be calculated through taking means of "optimal" values on other datasets. For example, the estimated λ_E

of RoBERTa model on Cognitive Distortion set could be calculated as:

$$\lambda_E^{RoBERTa,CD} = mean(\lambda_O^{RoBERTa,HateSpeech}, \lambda_O^{RoBERTa,SHAC}) \tag{7.18}$$

DAPPER models in accordance with different scaling factors are then labeled as DAPPER-D (default), DAPPER-O ("optimal"), and DAPPER-E (estimated).

7.4 Methods for Robust Learning

In this section, we focus on three representative methods for robust learning. They can be categorized based on different approaches for robust learning: through robust optimization (labeled as "GDRO"), through domain alignment (labeled as "MMD"), and through training strategy (Domain Adversarial Training, labeled as "DANN").

7.4.1 GDRO

Group Distributionally Robust Optimization (GDRO) Sagawa, Koh, Hashimoto, *et al.* [156] is an updated version of Distributionally Robust Optimization (DRO) [157], [158]. The goal of the traditional optimization process (Empirical Risk Minimization, ERM) of a model is to achieve a high accuracy *on average*, which can become detrimental when there exist spurious correlations across groups. DRO allows one to minimize the worst-case training loss over a set of pre-defined groups. Group DRO further applies regulations in the training process, and has achieved better performance than DRO.

With pre-specified groups $\mathcal{G}$ where spurious correlations are expected to be removed, the group DRO model minimizes the empirical worst-group risk $\hat{\mathcal{R}}(\theta)$ [156]:

$$\hat{\theta}_{DRO} := \arg\min_{\theta \in \Theta} \left\{ \hat{\mathcal{R}}(\theta) := \max_{g \in \mathcal{G}} \mathbb{E}_{(x,y)\sim \hat{P}_g}[\ell(\theta; (x, y))] \right\} \tag{7.19}$$

where $\hat{P}_g$ is the empirical distribution for a specific group; ℓ is the loss function. Instead of using batch optimization algorithms or stochastic optimization algorithms, Sagawa, Koh, Hashimoto, *et al.* [156] also propose an online optimization algorithm for the goal above, with convergence guarantees. We refer the interested readers to the original paper for the details of the algorithm and proof.

7.4.2 MMD

The second approach is through learning domain-invariant representations. Tzeng, Hoffman, Zhang, *et al.* [24] proposed a deep domain confusion loss, for maximizing domain invariance (or minimizing across-domain distance), measured by Maximum Mean Discrepancy (MMD). MMD is defined as:

$$MMD(X_S, X_T) = \left\| \frac{1}{|X_S|} \sum_{x_s \in X_S} \phi(x_s) - \frac{1}{|X_T|} \sum_{x_t \in X_T} \phi(x_t) \right\| \tag{7.20}$$

MMD measures the distance between generated feature representations ($\phi(\cdot)$) of the source set (X_S) and the target set (X_T). MMD can be directly integrated into the loss function through a weighting hyperparameter, together with the main classification loss [24]:

$$\mathcal{L} = \mathcal{L}_C(y_{true}, y_{pred}) + \lambda MMD^2(X_S, X_T) \tag{7.21}$$

For the problem of provenance shift, our simulation framework put models to the test under various degrees of the shift. To utilize this approach, we no longer use MMD to measure the distance between source and target, but instead focus on the representations from two provenances in the training process, i.e., $MMD(X^{z_1}_{train}, X^{z_2}_{train})$.

7.4.3 Domain Adversarial Training (DANN)

The third approach discussed is through domain adversarial training, proposed by Ganin, Ustinova, Ajakan, *et al.* [23]. It first introduces a domain classifier besides the primary target classifier and then inserts a gradient reversal layer beneath the domain classifier so that the representations learned through the training process can be domain-invariant while being effective for the primary target.

7.5 Experiments

7.5.1 Datasets

Three sets with data from different provenances were used in the experiments: Cognitive Distortion Detection (CD) dataset, the publicly-available Social History Annotation Corpus

(SHAC) dataset, and the Hate Speech (HateSpeech) Detection dataset. These three datasets are comprehensively described in Chapter 4, with characteristics and collection methodologies. Summary statistics for each of the datasets are presented in Table 4.1, 4.4, and 4.2.

7.5.2 Models

We used two models, RoBERTa [154] and Llama-2 [155] for our text classification tasks. These models exemplify two different commonly-used Transformer architectures: RoBERTa is an encoder-only model, and the Llama-2 is a decoder-only model. They also represent models of substantially different sizes, with Llama-2 seven billion parameter (7B) version being around 56 times larger than RoBERTa. The base version of RoBERTa is an extended version of the base BERT model [79], with 15 million more parameters (125 million parameters in total), modifications to the pretraining process, and a larger pretraining corpus. It is reported to outperform BERT in many tasks [154].

Llama-2 was published by Meta as an "open-source" Large Language Model (LLM) with publicly available weights [155][1]. Though it is reported to underperform the "closed source" LLM GPT-3.5 [155], Llama-2 weights are available to edit, enabling task arithmetic. There are three versions of Llama-2, with different parameter sizes: 7b, 13b, and 70b (b=billion). In this work, we chose the smallest version, 7b, to facilitate repeated experiments within available computational resources. As discussed previously, this version still has copious parameters. Therefore, we applied LoRA for parameter-efficient fine-tuning of this model, and used 8-bit quantization to further limit resource usage. Model quantization involves converting model parameters to a format with lower precision, to reduce resource usage and latency [159], [160]. For LoRA, we followed the original work [132] and only applied LoRA weights to the query (W_Q) and value (W_V) matrices. Other weights were frozen during fine-tuning, except for the classification layer. This leaves less than 140 million trainable parameters, which could be readily tuned within the constraints of available computational

[1]As the weights but not the training corpus were shared, "open-weight" may be a more apt description.

resources.

Table 7.1 provides a comparison between these two models, with details of the configurations used in our experiments.

	RoBERTa-base	Llama-2
Architecture	encoder	decoder
Number of Parameters	125 million	7 billion
Trainable parameters	125 million	4 million
LoRA Weights	NA	W_Q, W_V
Quantization	NA	8-bit

Table 7.1: Model specifications.

7.5.3 Training Settings

In order to isolate provenance-related confounding effects and preserve computational resources, we chose three specific configurations for the training set, as detailed in Table 7.2. Two training configurations ($\alpha_{train} = 0.2$ and 5) were selected to represent extreme provenance shifts (for positive rates) toward one source or the other. The final training set configuration ($\alpha_{train} = 1$) was selected to represent the case where there is no positive rate difference, and therefore no potential for confounding by provenance to manifest.

	$\alpha_{train} = 0.2$	$\alpha_{train} = 1$	$\alpha_{train} = 5$
# of Samples	800	800	800
from z_1	400	400	400
from z_2	400	400	400
# of Positive Samples	240	240	240
from z_1	200	120	40
from z_2	40	120	200
$P(Y = 1\|Z = z_1)$	0.5	0.3	0.1
$P(Y = 1\|Z = z_2)$	0.1	0.3	0.5
$P(Y = 1)$	0.3	0.3	0.3

Table 7.2: Three different training set compositions.

7.5.4 Model Interpretation

SHAP (SHapley Additive exPlanations) is a game theoretic approach to explain predictions from machine learning models [161]. The method assigns an importance value to each feature (in text classification, features are tokens) for a particular prediction. When aggregated, SHAP output indicates which features play important roles in a model's prediction behavior.

A held-out validation set of size 1,000 was randomly selected such that each source has 500 samples with 150 cases and 350 controls (a positive rate of 30%). Examples from the two provenances were evaluated separately, using the mean SHAP value over all 500 examples concerned. We selected a RoBERTa model trained on the set with $\alpha_{train} = 5$ and focused our analysis on the comparison between the baseline fine-tuned model ($\lambda = 0$) and the "default" scaled model using DAPPER ($\lambda = 1$).

7.5.5 Baseline approaches

Backdoor Adjustment [14], [15], as described in Section 7.2.3, was used as the baseline for comparison of the effectiveness of TAPER and DAPPER. For text representations, we used binary unigrams as predictors for Backdoor Adjustment with logistic regression models, as in our previous work [19]. Logistic regression models with and without Backdoor Adjustment were tested. Fine-tuned models without any provenance effect reduction ($\lambda = 0$) provide another baseline for comparison of models within the same model architecture (RoBERTa or Llama-2).

7.5.6 Robust Learning

For evaluating three robust learning discussed in Section 7.4, they were tested under our simulation and evaluation framework on all three datasets mentioned in Section 7.5.1. For GDRO, we used the default settings as in the paper and from the open-source codes[2]. For MMD, the default setting of $\lambda = 0.25$ was used as in [24].

All methods were tested using the RoBERTa model for text classification tasks on three datasets. All models were trained at 3 epochs, with the only exception of DANN on the SHAC dataset where 6 epochs were used to achieve comparable training loss.

7.6 Results for TAPER and DAPPER

7.6.1 Worst-case Performance

We first examine results showing the worst-case performance of models on each dataset, summarized in Table 7.3. In Section **??**, the "optimal" λ values for each model on each dataset are defined based on the worst-case scenarios and the recommended λ is thus defined on the basis of worst-case performance on other datasets. These λ settings result in an "default" model (DAPPER-D), an "optimal" model (DAPPER-O), and an estimated (recommended) model (DAPPER-E), respectively. Results from TAPER models with default

[2] `https://github.com/kohpangwei/group_DRO`

settings ($\lambda = 1$) are also included in the table. Of note, TAPER-D and DAPPER-D are the same model when $\alpha_{train} = 5.0$, as TAPER arbitrarily defines one polarization direction (in our experiments, z_2) for both $\alpha_{train} = 0.2$ and 5.0, leading to one shared setting with DAPPER. Thus, DAPPER-D and TAPER-D have the same results when $\alpha_{train} = 5.0$. Detailed results on how λ affects model's performance is in Section 7.6.3. The overall positive rate in the test is set to 0.5, and we further limit $P_{test}(Z = z_1) = P_{test}(Z = z_2) = 0.5$, to remove potential confounders.

Overall, Llama-2 with DAPPER provides best worst-case performance across all three datasets. Both DAPPER-E and DAPPER-O outperform baseline fine-tuned Llama-2 models by large margins. With DAPPER-O there is an increase of 0.2-0.3 in $AUPRC_{worst}$ on the Cognitive Distortion and Hate Speech datasets. Llama-2 models outperform RoBERTa models in every setting, except for baseline fine-tuning on the Cognitive Distortion dataset. Even in that case, Llama-2 outperforms RoBERTa when DAPPER is applied. The recommended λ settings show AUPRC very close, and even identical, to that in optimal settings, suggesting the validity of this simple way of calculating a λ value.

When trained using the RoBERTa model, DAPPER with the default setting (DAPPER-D) outperforms the baseline fine-tuned models by large margins. With the optimal λ's, $AUPRC_{worst}$ is further improved over, or as good as, the default setting. Under default settings, DAPPER outperforms TAPER, aside from with the RoBERTa model trained on SHAC with $\alpha_{train} = 0.2$ where the results are comparable. TAPER, with default settings, as the unpolarized version of DAPPER when $\alpha_{train} = 0.2$, performs slightly better than the baseline model when using RoBERTa and underperforms when using Llama-2, showing the added utility of DAPPER.

Logistic regression with Backdoor Adjustment provides a strong baseline in some cases. Consistent with results previously reported on the SHAC dataset [19], Backdoor Adjustment improves worst-case performance from AUPRC of 0.889 to 0.931 when $\alpha_{train} = 0.2$, and from 0.881 to 0.929 when $\alpha_{train} = 5.0$. The Cognitive Distortion dataset also benefits from Backdoor Adjustment but the Hate Speech set does not. With unadjusted deep neural

networks, there are performance boosts for the worst-case scenarios over unadjusted logistic regression (LR), typically across both pretrained models, with exceptions when $\alpha_{train} = 5$ for the Hate Speech and SHAC datasets using RoBERTa and the Cognitive Distortion dataset using Llama-2. These results indicate that while backdoor adjustment effectively remediates confounding by provenance, the additional representational capabilities of pretrained neural language models provide advantages over the performance of unadjusted models.

		$\alpha_{train} = 0.2$		$\alpha_{train} = 5.0$	
Dataset	Adjustment	RoBERTa	Llama-2	RoBERTa	Llama-2
CD	LR	0.552 ± 0.037		0.575 ± 0.027	
	LR+BA	0.658 ± 0.029		0.741 ± 0.030	
	baseline	0.575 ± 0.018	0.573 ± 0.036	0.618 ± 0.027	0.544 ± 0.012
	DAPPER-D	0.614 ± 0.015	0.629 ± 0.065	0.718 ± 0.041	0.788 ± 0.032
	DAPPER-O	0.619 ± 0.027	**0.723 ± 0.075**	0.734 ± 0.045	0.861 ± 0.028
	DAPPER-E	0.614 ± 0.015	**0.723 ± 0.075**	0.724 ± 0.047	**0.862 ± 0.020**
	TAPPER-D	0.603 ± 0.010	0.414 ± 0.017	0.718 ± 0.041	0.788 ± 0.032
Hate Speech	LR	0.511 ± 0.028		0.535 ± 0.036	
	LR+BA	0.442 ± 0.019		0.456 ± 0.037	
	baseline	0.580 ± 0.047	0.755 ± 0.045	0.491 ± 0.016	0.683 ± 0.014
	DAPPER-D	0.673 ± 0.046	0.728 ± 0.032	0.549 ± 0.017	0.813 ± 0.021
	DAPPER-O	0.673 ± 0.046	**0.864 ± 0.037**	0.549 ± 0.017	**0.874 ± 0.019**
	DAPPER-E	0.673 ± 0.046	**0.864 ± 0.037**	0.541 ± 0.016	0.856 ± 0.017
	TAPPER-D	0.609 ± 0.043	0.462 ± 0.014	0.549 ± 0.017	0.813 ± 0.021
SHAC	LR	0.889 ± 0.009		0.881 ± 0.022	
	LR+BA	0.931 ± 0.005		0.929 ± 0.020	
	baseline	0.909 ± 0.021	0.959 ± 0.006	0.818 ± 0.045	0.951 ± 0.013
	DAPPER-D	0.936 ± 0.020	0.568 ± 0.029	0.865 ± 0.033	0.707 ± 0.046
	DAPPER-O	0.939 ± 0.020	**0.967 ± 0.006**	0.867 ± 0.025	**0.970 ± 0.009**
	DAPPER-E	0.932 ± 0.020	0.966 ± 0.005	0.862 ± 0.025	0.951 ± 0.016
	TAPPER-D	0.940 ± 0.020	0.540 ± 0.029	0.865 ± 0.033	0.707 ± 0.046

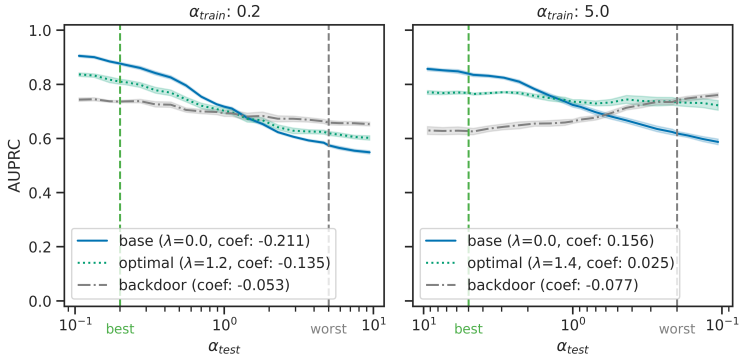
Table 7.3: "worst-case" performance ($AUPRC_{worst}$) using DAPPER with different λ values for all three datasets. Best performance is highlighted in each setting (row). LR: Logistic regression. LR+BA: Logistic regression with Backdoor Adjustment. baseline: baseline fine-tuned model, no TAPER nor DAPPER. DAPPER-D: default setting for DAPPER with $\lambda = 1$. DAPPER-O: "optimal" λ for DAPPER, where $AUPRC_{worst}$ is lowest. DAPPER-E: estimated λ from other datasets. TAPER-D: default setting for TAPER with $\lambda = 1$.

7.6.2 Robustness and Performance of DAPPER

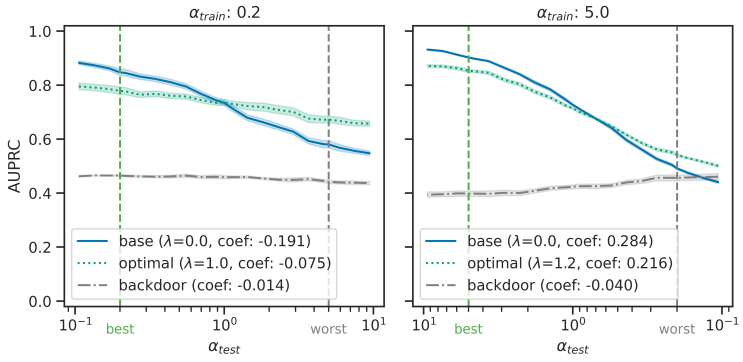
In Section 7.6.1, DAPPER was shown to perform better in the worst-case scenario as compared with logistic regression with Backdoor Adjustment and TAPER. As an updated version of TAPER with the targeted deletion of the auxiliary task vector generated on the dominant provenance label, DAPPER is the focus of the following analysis. Results on three datasets are shown in Figure 7.1 and 7.2 for models using RoBERTa and Llama-2, respectively. The results shown limit the overall positive rate in the test set to 0.5, and equal mixtures, $P_{test}(Z = z_1) = P_{test}(Z = z_2) = 0.5$, for removing potential confounders in the analysis. "base" refers to the baseline fine-tuning (primary task only), where $\lambda = 0$ (same as "baseline" in Table 7.3); "optimal", represents the λ value that leads to the best worst-case performance using DAPPER (same as DAPPER-O in Table 7.3). "backdoor" in gray dash-dotted lines refers to the Backdoor Adjustment method, as baselines (same as "LR+BA" in Table 7.3).

The figures can be interpreted as follows. In each figure, the solid curve represents the model without task arithmetic, and the dotted curve represents the model when task arithmetic is used to mitigate confounding effects. The left panels represent splits generated using an α_{train} of 0.2, indicating that positive examples from site $z = 1$ predominate in the training set. The right panels represent results from splits with an α_{train} of 5 (the reciprocal of 0.2), indicating that positive examples from site $z = 2$ predominate at training time. In all panels the y axis shows the AUPRC as a measure of performance, and the x axis indicates the α_{test} value. As α_{train} is fixed in each panel, the degree of confounding bias increases as the distance along this axis from the dashed vertical line increases, with the dotted vertical line indicating the "worst case" confounding bias, where $\alpha_{train} = 1/\alpha_{test}$.

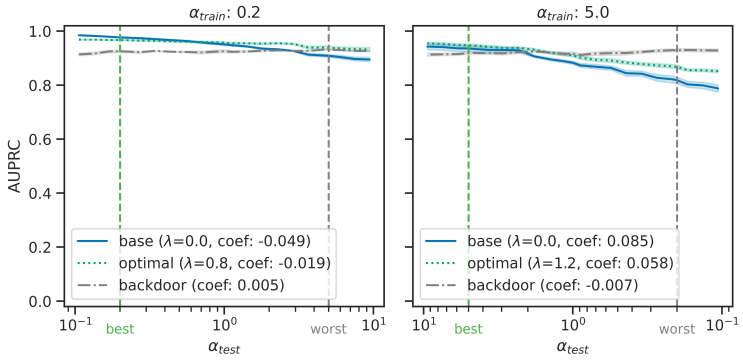
In each panel, the *slope* of the line indicates robustness to confounding by provenance. A model that is immune to this form of bias would present a flat horizontal line, because performance (AUPRC) does not change as confounding bias (distance between α_{test} and α_{train}) increases. Numerically, robustness is measured by the absolute value of the slope (or



(a) CD

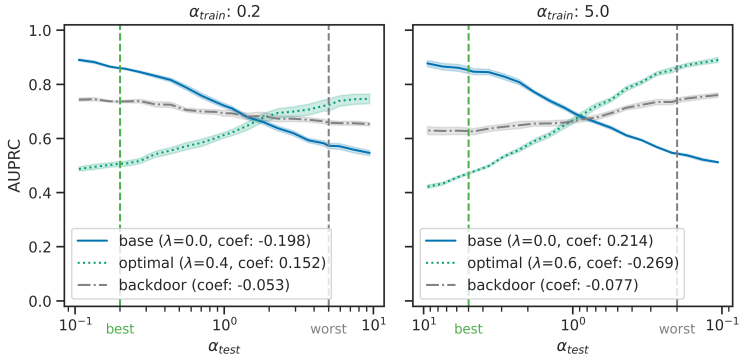


(b) HateSpeech

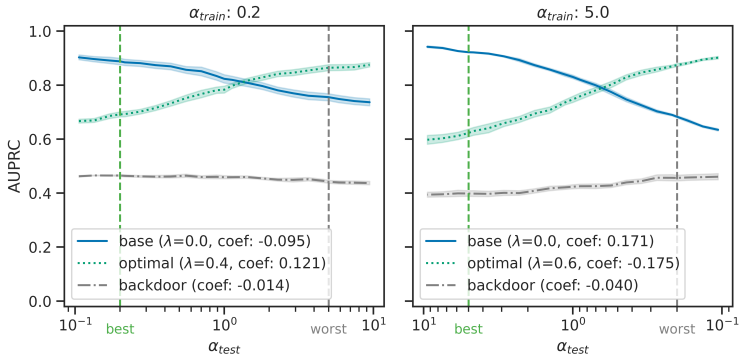


(c) SHAC

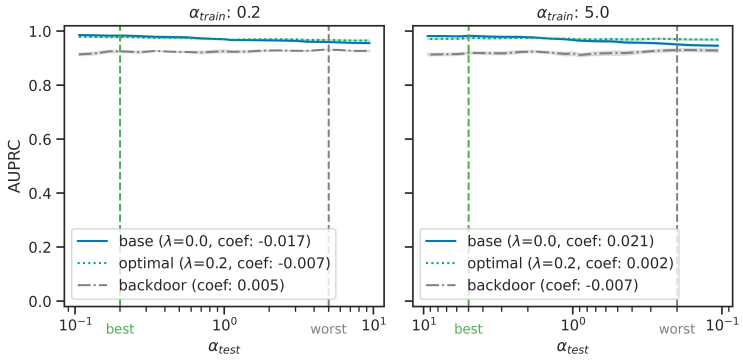
Figure 7.1: DAPPER on all three datasets for different α_{train}, using RoBERTa. (a) are results on Cognitive Distortion dataset; (b) on Hate Speech dataset; (d) on SHAC dataset. "base" refers to the baseline fine-tuning; "optimal", DAPPER with "optimal" λ.



(a) CD



(b) HateSpeech



(c) SHAC

Figure 7.2: DAPPER on all three datasets for different α_{train}, using Llama. (a) are results on Cognitive Distortion dataset; (b) on Hate Speech dataset; (d) on SHAC dataset. "base" refers to the baseline fine-tuning; "optimal", DAPPER with "optimal" λ.

coefficient). The smaller the absolute coefficient is, the more robust a model is to provenance-related confounding effects.

From the results, the absolute coefficients from RoBERTa models using DAPPER are smaller than the baseline for all datasets. For example, for the Cognitive Distortion dataset at $\alpha_{train} = 5$, the RoBERTa model after DAPPER shows a lower absolute coefficient of 0.025 than 0.156 for the model without DAPPER (Figure 7.1a). This improvement in coefficients is relatively small for the training set at $\alpha_{train} = 0.2$, with 0.135 vs 0.211 (Figure 7.1a). As for Llama-2, the robustness improves when using DAPPER on the SHAC dataset. This is not consistent with other two datasets, such as in Hate Speech Detection (Figure 7.2b).

As described in Section 7.2.4, performance at two points under the simulation framework are of particular interest, and are highlighted in graphs as green and gray vertical dashed lines (as in Figure 7.1 and 7.2). $AUPRC_{best}$ at $\alpha_{test} = \alpha_{train}$ (shown as the green dashed vertical lines on the left side within figures) and indicating the "best case" performance when provenance-related associations learned during training may help performance at test time; and $AUPRC_{worst}$ at $\alpha_{test} = 1/\alpha_{train}$ (shown as gray dashed vertical lines on the right side within figures), representing an extreme point of provenance-related confounding where the relative contribution of positive examples from each site at training time is inverted at test time. At the extreme of distribution shift, $AUPRC_{worst}$ from DAPPER is always better than that of the baseline model. For $AUPRC_{best}$, baseline models perform better, as one would anticipate when provenance-related biases in prediction are consistent with the provenance-specific class distribution at test time.

In comparison with the baseline Backdoor Adjustment, DAPPER typically generates less robust models for both RoBERTa and Llama-2. There are cases where DAPPER can result in better robustness than, or similar robustness to, Backdoor Adjustment, for example with absolute coefficients of 0.025 vs 0.077 (backdoor) for the Cognitive Distortion dataset and 0.004 vs 0.007 (backdoor) for SHAC set when $\alpha_{train} = 5.0$ using RoBERTa (Figure 7.1a). Backdoor Adjustment shows strong "worst-case" performances on the Cognitive Distortion and SHAC dataset over DAPPER with RoBERTa. However, DAPPER with Llama lead to

better $AUPRC_{worst}$ in those cases. On the Hate Speech dataset, fine-tuned models, with and without DAPPER, consistently outperforms Backdoor Adjustment by great margins.

Table 7.4 shows the coefficients, representing robustness to confounding by provenance, for baseline, TAPER (detailed results are reported in the Appendix A.3) and DAPPER models across all three datasets at $\alpha_{train} = 0.2$ for the RoBERTa model. While TAPER (middle column) does result in more robust models for three datasets, DAPPER (rightmost column) further improves robustness with smaller absolute coefficients for all of them. While Llama-2 model doesn't benefit from provenance effect reduction in comparison with the baseline fine-tuned model in terms of worst-case performance, as discussed in the previous section, DAPPER can still generate more robust models (or comparable) than the baseline. For example, the coefficient changed from -0.198 to 0.152 on the Cognitive Distortion set, from -0.017 to -0.007 on the SHAC set, when trained with $\alpha_{train} = 0.2$ (Figure 7.2).

	baseline	TAPER*	DAPPER**
Cognitive Distortion	-0.211	-0.182	-0.135
Hate Speech	-0.191	-0.137	-0.075
SHAC	-0.049	-0.025	-0.019

Table 7.4: RoBERTa model robustness using two provenance effect reduction procedures vs baseline, trained with $\alpha_{train} = 0.2$. *provenance effect reduction. **Dominance-Aligned Polarized Provenance Effect Reduction.

It should also be noted that improvements in model's robustness sometimes come with loss of performance in some regions and gain in other regions. For example, when comparing Llama-2 models on the Cognitive Distortion set when $\alpha_{train} = 0.2$ between two procedures (in Figure A.6a and Figure 7.2a), the Dominance-Aligned version shows poor performance in the left region of α_{test}, and improvement in the right region. This also means a worse $AUPRC_{best}$

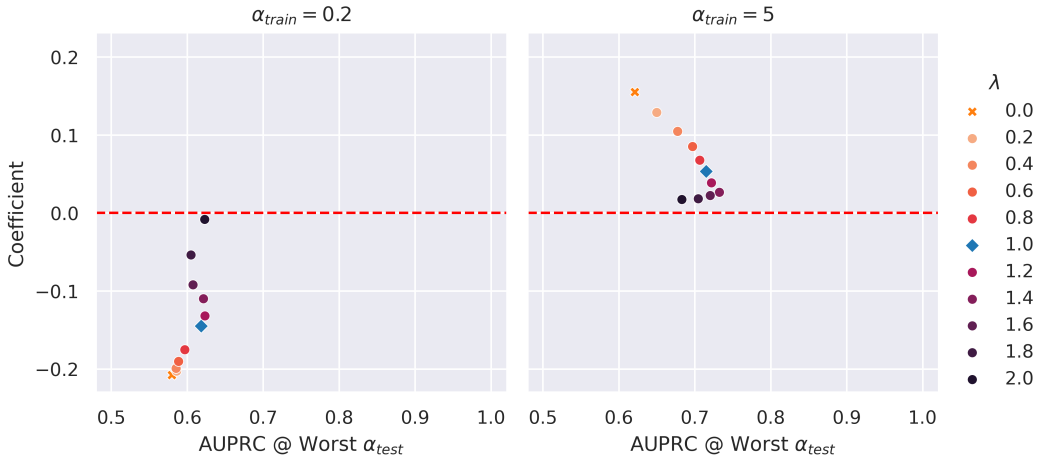
and better $AUPRC_{worst}$. This aligns well between two training settings ($\alpha_{train} = 0.2$ and 5). For RoBERTa model, such effect is subtle. When checking "worst-case" performances, DAPPER at $\alpha_{train} = 0.2$ setting outperform Backdoor Adjustment for both Cognitive Distortion and Hate Speech datasets, but not on SHAC dataset.

In summary, the findings indicate that DAPPER can effectively improve RoBERTa's robustness when there is an extreme imbalance in provenance-specific positive rates in the training set. It can also improve $AUPRC_{worst}$ over the baseline fine-tuned models. On the Hate Speech dataset, fine-tuned models are consistently better than Backdoor Adjustment. However, for $AUPRC_{best}$, DAPPER performs worse than the baseline level, presumably because the encoded biases help performance in this circumstance.

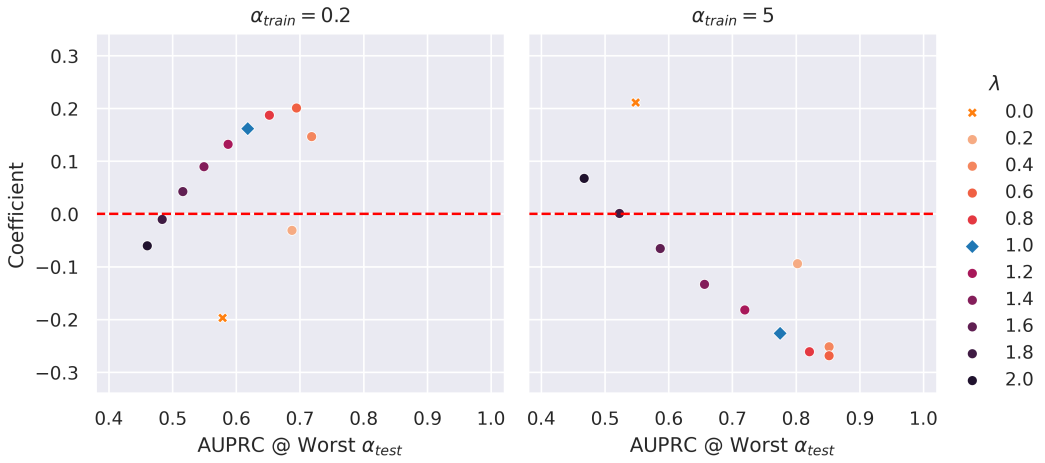
7.6.3 Scaling Factors for Dominance-Aligned Polarized Provenance Effect Reduction

Scaling is an important but previously unexplored factor (in Formula (7.9)). To generate the results shown in previous sections, we used the "default" setting of $\lambda = 1$. As shown in previous results, this setting does not necessarily generate models that are more robust than the baseline fine-tuned models. Thus, we empirically put the scaling into test, testing λ across a range from 0 to 2. To simplify analysis, we focus on the worst case scenario where there is extreme shift in the test set from the training set. Consequently, $AUPRC_{worst}$ was selected as the performance metric.

The results from DAPPER for Cognitive Distortion set are shown in Figure 7.3. Each point represents a model with a unique λ value. $\lambda = 0$ is the baseline fine-tuned model with no provenance effect reduction, $\lambda = 1$ provides a "default" scaling. This figure presents two strands of information. For model robustness, the closer a point is to the red horizontal line (meaning lower absolute coefficients), the more robust it is. For the worst case performance, a point positioned further to the right on the x-axis ($AUPRC_{worst}$ value) represents a model with a better worst-case performance. At the best hyper-parameter setting, DAPPER always improves over the baseline fine-tuned models (marked with a red x), in both robustness and worst-case performance. For both RoBERTa and Llama-2 models, when following the



(a) RoBERTa

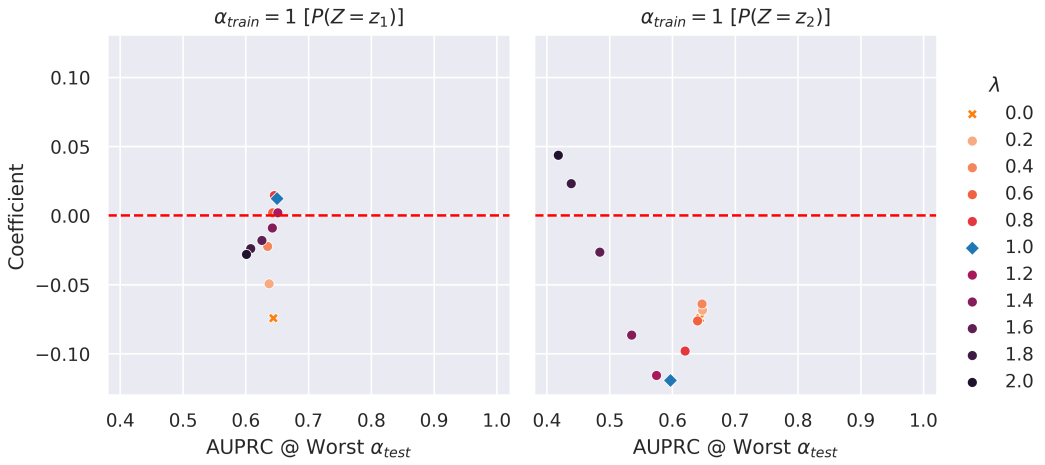


(b) Llama-2

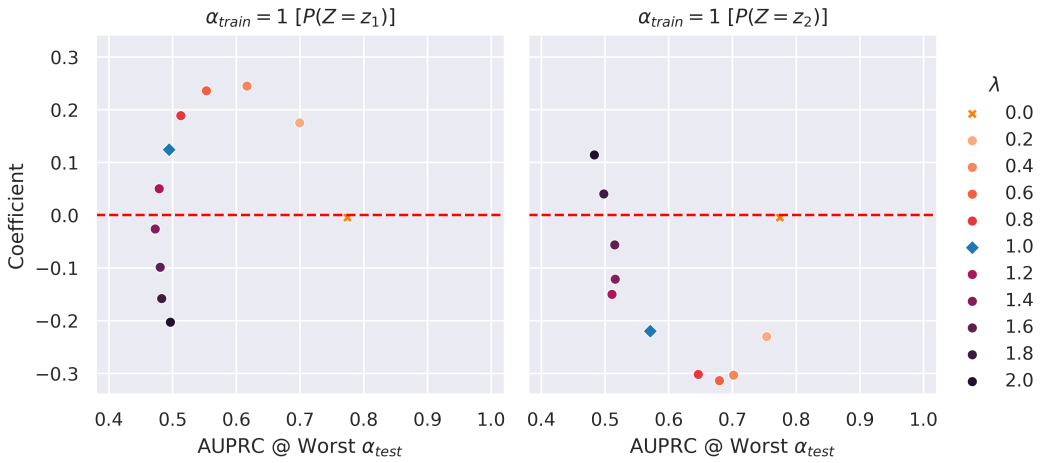
Figure 7.3: Scaling Factors for Dominance-Aligned Polarized Provenance Effect Reduction on Cognitive Distortion set. Positive rate for the training sets was set to 0.5. x-axis is the $AUPRC_{worst}$, and y-axis is coefficients (measure for robustness). Red horizontal line is the ideal anchor for robustness where the coefficient equals to 0. (a) are results on the RoBERTa model. (b) on the Llama-2 model. Slopes of the fitted line (measured by coefficients) are reported for all settings.

increase of λ value, there is a curved path in each figure. The first half of this curvature represents room where we can find a model with better robustness and performance by increasing the weight assigned to the subtraction of the task vector for provenance. After a specific point, the curve represents a trade-off between performance and robustness when stronger effects from $\tau_{provenance}$ are applied (as increasing λ), until after some point when model's robustness and performance both collapse. The "default" setting of $\lambda = 1$ presents a strong model across different datasets and models, even though sometimes it is not the best choice. In some settings, an optimal or near-optimal scaling factors could be found. For example, $\lambda = 1.4$ for RoBERTa on the Cognitive Distortion set with $\alpha_{train} = 5$ (Figure 7.3(a)), or $\lambda = 0.2$ for Llama-2 on the Hate Speech dataset with $\alpha_{train} = 0.2$ (Figure A.1(b)). But for cases in the RoBERTa model on Cognitive Distortion set with $\alpha_{train} = 0.2$ (Figure 7.3(a)), it is hard to decide which model is the best on those two scales (robustness and worst-case performance) together. It is also noted that even though DAPPER, when applied to different degrees, can improve model's robustness and performance, the specific scaling degree is highly dependent on the dataset. For example, with RoBERTa model at $\alpha_{train} = 0.2$, the Hate Speech dataset favors larger λ (1.0) than SHAC (0.8), but smaller than the Cognitive Distortion dataset (1.2). RoBERTa and Llama-2 models also show difference in their optimal scaling degrees. Llama-2 requires small scaling factors of around 0.2 or 0.4 in most cases across all three datasets.

For the case where no provenance class dominates (thereby eliminating the possibility of confounding by provenance), we evaluated the scaling factors in both directions in correspondence with two provenance classes as before (in Appendix A.2). However, it should be noted the direction is arbitrary in this case. Across all datasets, DAPPER shows no significant effect over the baseline fine-tuned models, no matter how much scaling was applied. This holds for both RoBERTa and Llama-2 models. These results confirm that provenance effect reduction is helpful only in the context of confounding by provenance.



(a) RoBERTa

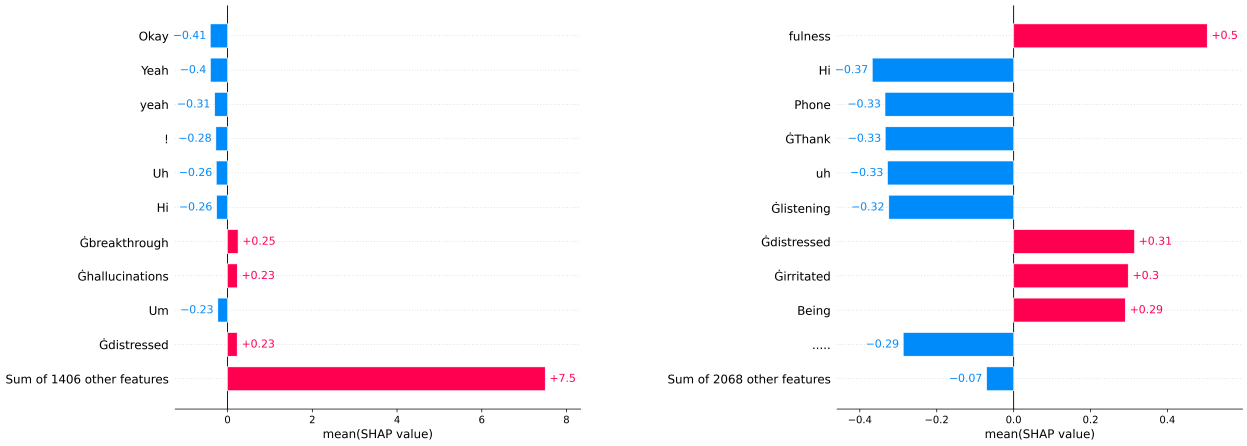


(b) Llama-2

Figure 7.4: Scaling Factors for Dominance-Aligned Polarized Provenance Effect Reduction applied on Cognitive Distortion set with $\alpha_{train} = 1$. Positive rate for the training sets was set to 0.5. x-axis is the $AUPRC_{worst}$, and y-axis is coefficients (measure for robustness). Red horizontal line is the ideal anchor for robustness where the coefficient equals to 0. (a) are results on the RoBERTa model. (b) on the Llama-2 model. Slopes of the fitted line (measured by coefficients) are reported for all settings.

7.6.4 Diagnosis: Token Usage from SHAP

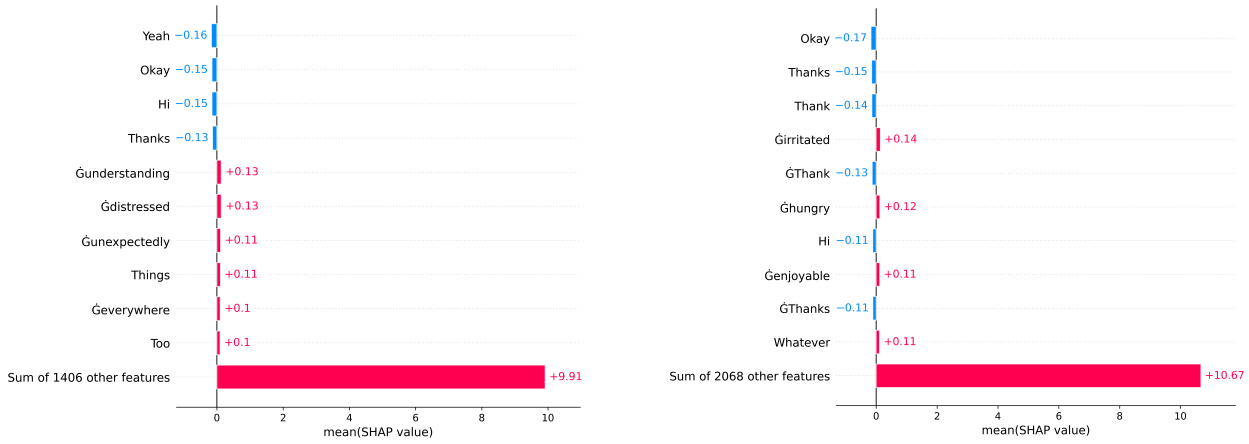
In this experiment, results on baseline fine-tuned model ($\lambda = 0$) are shown in Figure 7.5 and the "default" scaled model using DAPPER ($\lambda = 1$) in Figure 7.6. The figures include the top 12 tokens. For more details, Table 7.5 include top 30 tokens. The RoBERTa model uses a byte-level BPE (Byte-Pair Encoding) for tokenization, and it encodes white spaces using the special character Ġ, as is shown in the results. For SHAP values, positive values indicate positive effects on the prediction (Any Cognitive Distortion), and negative values indicate negative effects.



(a) AVH subset

(b) TM subset

Figure 7.5: Token importance using mean SHAP values for the baseline fine-tuned model, $\lambda = 0$. Tokens are ordered by their importance starting from the top. On the Cognitive Distortion set. (a) 500 validation examples from $Z = z_1$ (AVH subset) (b) 500 validation examples from $Z = z_2$ (TM subset).



(a) AVH subset

(b) TM subset

Figure 7.6: Token importance using mean SHAP values for the "default" DAPPER model, $\lambda = 1$. Tokens are ordered by their importance starting from the top. On the Cognitive Distortion set. (a) 500 validation examples from $Z = z_1$ (AVH subset) (b) 500 validation examples from $Z = z_2$ (TM subset).

Table 7.5: Top 30 most important tokens using mean SHAP values.

baseline*				DAPPER**			
AVH		TM		AVH		TM	
Okay	−0.41	fullness	+0.50	**Yeah**	−0.16	**Okay**	−0.17
Yeah	−0.40	**Hi**	−0.37	**Okay**	−0.15	**Thanks**	−0.15
yeah	−0.31	Phone	−0.33	**Hi**	−0.15	Thank	−0.14
!	−0.28	ĠThank	−0.33	**Thanks**	−0.13	Ġirritated	+0.14
Uh	−0.26	uh	−0.33	Ġunderstanding	+0.13	ĠThank	−0.13
Hi	−0.26	Ġlistening	−0.32	Ġdistressed	+0.13	Ġhungry	+0.12
Ġbreakthrough	+0.25	**Ġdistressed**	+0.31	Ġunexpectedly	+0.11	**Hi**	−0.11
Ġhallucinations	+0.23	Ġirritated	+0.30	Things	+0.11	Ġenjoyable	+0.11
Um	−0.23	Being	+0.29	Ġeverywhere	+0.10	ĠThanks	−0.11
Ġdistressed	+0.23	.....	−0.29	Too	+0.10	Whatever	+0.11
Ġmedicaid	−0.23	Feel	+0.29	Ġsurvive	+0.10	Ġsexually	+0.10
Ġstressed	+0.23	Ġ	−0.28	Ġstrange	+0.10	ashion	+0.10
Ġsurvive	+0.23	Thanks	−0.28	Ġembarrassing	+0.09	**Yeah**	−0.09
Ġregocnize	−0.22	ĠThanks	−0.27	Alright	−0.09	Being	+0.09
Ġaudible	−0.22	**Yeah**	−0.27	Can	+0.09	M	−0.09
Ġstupid	+0.22	els	+0.26	Ġstressful	+0.09	ages	+0.09
resses	+0.21	Ġsexually	+0.26	Ġstupid	+0.09	fulness	+0.09
Ġr	−0.21	Ġdisregard	+0.26	Ġfailure	+0.09	Hopefully	+0.08
Ġdistracted	+0.21	anti	+0.25	Ġdesperate	+0.09	Phone	−0.08
Ġcreating	+0.21	**Okay**	−0.25	Uh	−0.09	Ġrelationship	+0.08
_.	−0.21	Ġbegin	−0.24	_.	−0.08	Feel	+0.08
<	−0.20	Thank	−0.24	resses	+0.08	Ġsounds	+0.08

Continued on next page

Table 7.5 – continued from previous page

baseline*				DAPPER**			
AVH		TM		AVH		TM	
Ġsorry	−0.20	ĠMass	−0.24	ucker	+0.08	Ġincredible	+0.08
Ġunderstanding	+0.20	ocation	+0.23	Ġmiserable	+0.08	els	+0.08
Yes	−0.20	Today	−0.23	Ġpossible	+0.07	Yes	−0.08
Ġmiserable	+0.19	Ġrelationship	+0.23	Ġaudible	−0.07	ough	+0.08
Ġuseless	+0.19	Ġhungry	+0.23	Ġfaintly	+0.07	Ġbreakfast	+0.08
Ġ...	−0.19	Ġetc	−0.23	bred	+0.07	Sure	−0.08
Anyway	−0.19	**Yes**	−0.22	Ġhallucinations	+0.07	Ġstole	+0.08
Ġname	−0.19	able	+0.22	Ġintervention	+0.07	Ġdisrespect	+0.08

Tokens are ordered by their importance starting from the top.

Results are on the Cognitive Distortion dataset.

AVH: the Auditory Verbal Hallucinations subset.

TM: the text-message intervention subset.

*baseline fine-tuned model.

**DAPPER at the "default" setting of $\lambda = 1$.

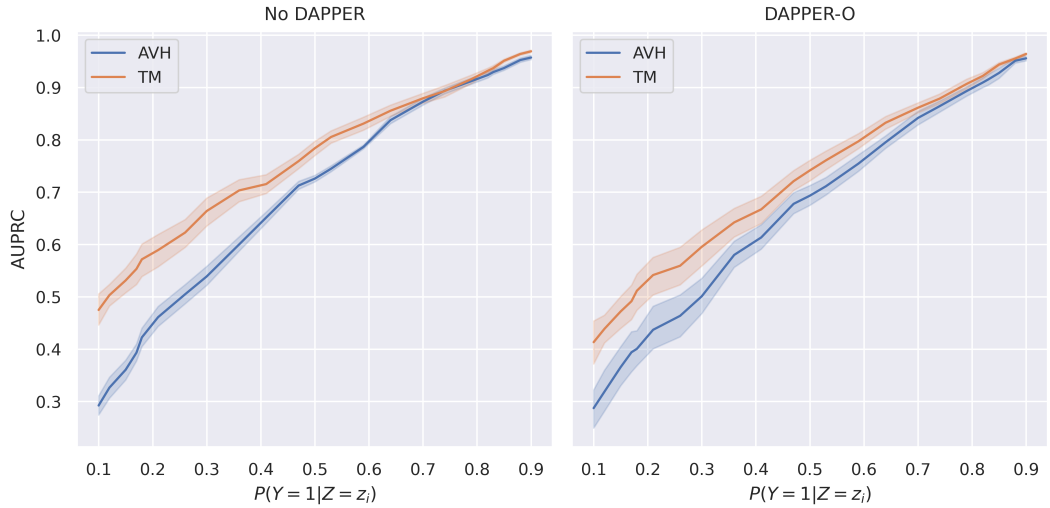
The results of SHAP value rankings in each source for the baseline fine-tuned model indicates that the models are utilizing different features to make "Any Distortion" predictions, with five overlapping tokens, **Okay**, **Yeah**, **Hi**, **Ġdistressed**, and **Yes**, across two sources. The model ranks **Okay** in very different places for two subpopulations, 1st vs 20th. The DAPPER model used four shared features across sources in the primary prediction task: **Yeah**, **Okay**, **Hi**, **Thanks**. In comparison, the rankings of these common tokens are closer in both subsets than those in the baseline model. When checking more broadly for tokens,

we can find more similar words for the DAPPER model, such as *Thank*, *ĠThank*, *Ġthanks*, in the shared list. Those different versions also show similar levels of importance, indicating the DAPPER can identify shared features more evenly.

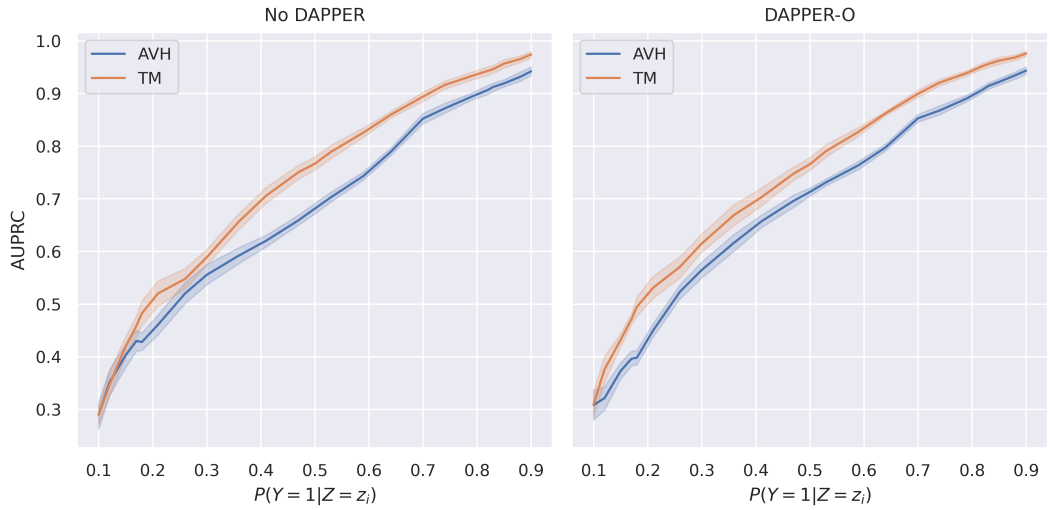
7.6.5 Diagnosis: Performance by Provenance

The concept of fairness in machine learning was proposed to ensure that models operate without bias across different groups [162], [163]. Commonly-applied measures of fairness consider differences (subtraction or taking the ratio) in the predicted probabilities of the primary outcome for each subgroup. For example, statistical parity considers the demographics between those who are classified as positve (or negative) and the whole population [164].In this section, we focus on fairness in predictive performance [165], by evaluating model's performance (AUPRC) on each subset, separately, under the simulation framework.

Figure 7.7 shows two models trained with $\alpha_{train} = 0.2$ (top) and $\alpha_{train} = 5$ (bottom). For simplicity, only models without DAPPER and with DAPPER-O are shown. Overall, performance using DAPPER-O, when trained with the same setting, aligns better on both subsets than models without DAPPER. This is indicated by the smaller gaps between AUPRC performances on two subsets for all levels of prevalence rates when using DAPPER-O. Models under $\alpha_{train} = 5.0$ (Figure 7.7b) are better aligned than those trained on $\alpha_{train} = 0.2$ (Figure 7.7a).



(a) Model trained with $\alpha_{train} = 0.2$



(b) Model trained with $\alpha_{train} = 5$

Figure 7.7: AUPRC by Provenance on the Cognitive Distortion set. The x-axis represents the positive rate within each provenance group in the test set. The y-axis represents the AUPRC. The overall (i.e. site-agnostic) positive rate for the primary task was fixed at 50% for the results shown. Sample sizes were balanced at 100 for each provenance. DAPPER-O is DAPPER with the "optimal" λ value.

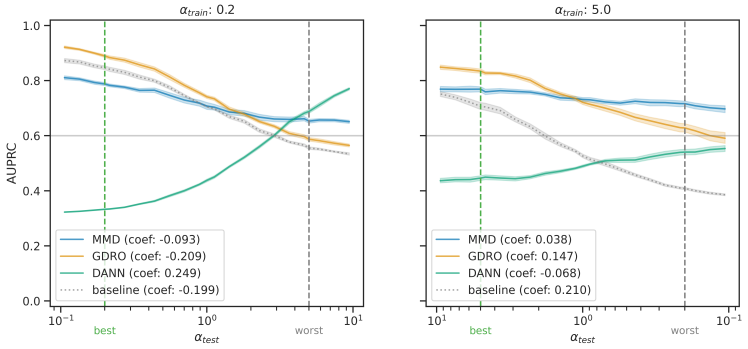
7.7 Results for Robust Learning

7.7.1 Performance under Provenance Shift

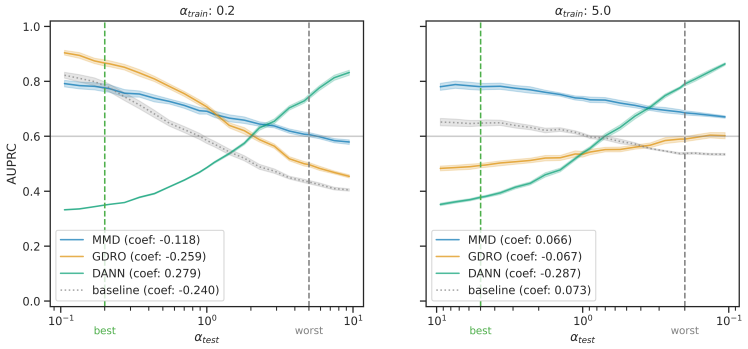
Performance of models on all three datasets under provenance shift is shown in Figure 7.8. MMD represents the domain alignment approach, using MMD as a measure of representation difference. GDRO represents group distributionally robust optimization. DANN represents domain adversarial training of neural networks. "baseline" refers to the standard training procedure with no modifications and using empirical risk minimization (ERM) which optimizes over the expected loss on the test set. The results are on the experiments with the overall positive rate in the test set to 0.5, and equal mixtures, $P_{test}(Z = z_1) = P_{test}(Z = z_2) = 0.5$, to remove potential confounders in the analysis.

Overall, models using MMD have the lowest absolute coefficients among all approaches. This indicates that MMD can help generate the most robust models. In general GDRO is worse than MMD in terms of robustness. However, in some cases, GDRO is comparable to MMD, such as on the Hate Speech dataset with $\alpha_{train} = 5.0$, showing $|\beta|$ of 0.067 vs 0.066 (MMD). In comparison, DANN made the models very unstable under provenance shift, with relative large, or even largest in many cases, $|\beta|$.

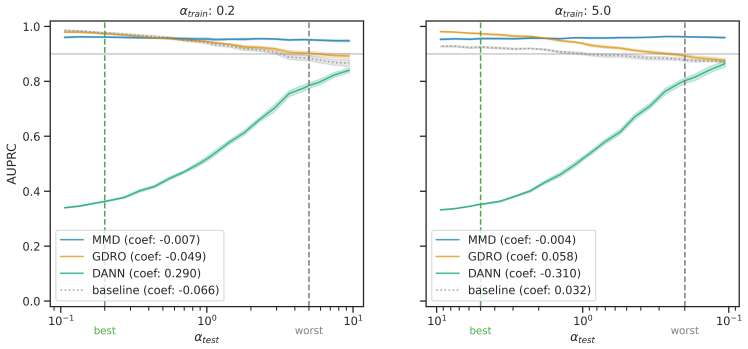
In terms of performances at the "worst-case" scenario ($AUPRC_{worst}$), MMD performs best in the SHAC dataset and the Cognitive Distortion dataset with $\alpha_{train} = 5.0$. On the Hate Speech dataset, DANN outperforms all other approaches by largely margins. GDRO generally performs poorly in the "worst-case" scenario for all datasets. However, it achieves the best performance at the "best-case" scenario ($AUPRC_{best}$), with the exception of the Hate Speech dataset when $\alpha_{train} = 5.0$.



(a) CD



(b) HateSpeech



(c) SHAC

Figure 7.8: Robust learning on all three datasets, using RoBERTa. Models were developed with different training set compositions, indicated by α_{train} in the column headers. (a) are results on Cognitive Distortion dataset; (b) on Hate Speech dataset; (d) on SHAC dataset. Slopes of the fitted line (measured by coefficients) are reported for all settings.

7.7.2 Diagnosis: Performance by Provenance

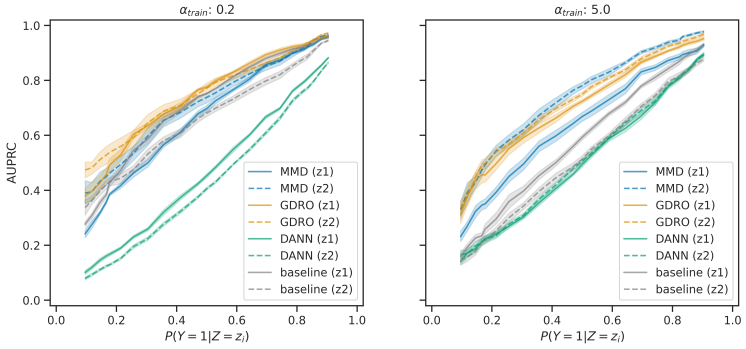
Following the procedures of examining fairness in predictive performance [165], we evaluated model's performance (AUPRC) on each subset, separately, under the simulation framework, as in Section 7.6.5. The results shown limit the overall positive rate in the test set to 0.5, and equal mixtures, $P_{test}(Z = z_1) = P_{test}(Z = z_2) = 0.5$, for removing potential confounders in the analysis.

Figure 7.9 shows all models trained with $\alpha_{train} = 0.2$ (left columns) and $\alpha_{train} = 5$ (right columns). When trained using GDRO and DANN, model performances between two provenances, are usually very close to each other. In comparison, MMD leads to a larger performance gap. Sometimes this gap is apparent, for example, on the Hate Speech dataset with $\alpha_{train} = 5.0$. In this case, however, the overall performance is good, as shown in Figure 7.8b, because the result is carried by the better performing models with $Z = z_2$.

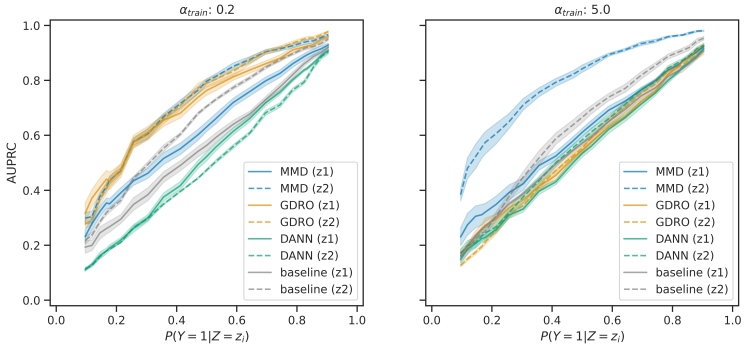
7.7.3 Diagnosis: Confusion of Hidden Space

Domain alignment through MMD and robust training strategies (e.g., DANN) both aim to train representations (or hidden spaces) that cannot discriminate between the provenances. In this section, we empirically evaluate how representations from two provenances confuse (or separate) with each other. Even though GDRO does not have that training goal, we evaluate it together with MMD and DANN. Baseline fine-tuned RoBERTa models are also included. In our implementation, both MMD and DANN operate on the outputs from the last hidden (encoder) layer (right before the classification) for the RoBERTa model, and only use the first token as input for the classification task. In this section, we therefore focus on the representation of the first token from the last encoder layer and visualize through t-SNE.

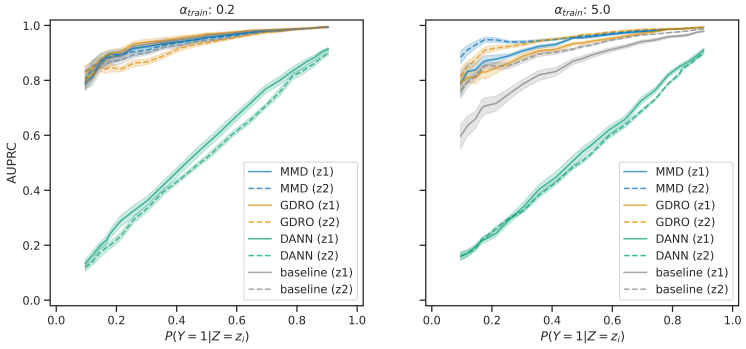
We used the same held-out validation set of randomly selected 1,000 samples as in Section 7.6.4. Among those, each source has 500 samples with 150 cases and 350 controls (a positive rate of 30%). Examples from two provenances are colored differently. Results on all three datasets are shown in Figure 7.10, 7.11, and 7.12, respectively.



(a) CD



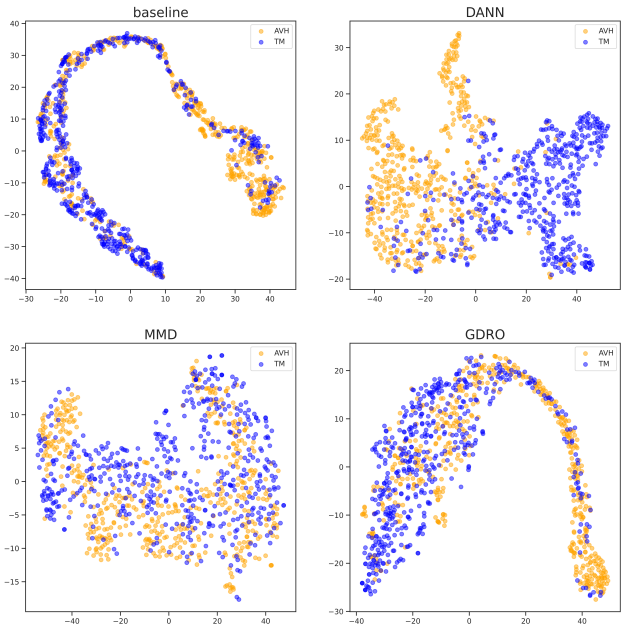
(b) HateSpeech



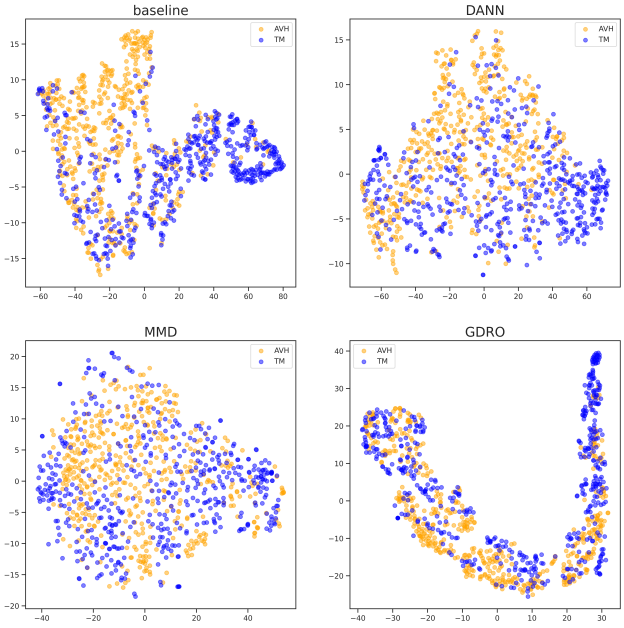
(c) SHAC

Figure 7.9: AUPRC by Provenance (z_1, z_2) on different models. The x-axis represents the positive rate within each provenance group in the test set. The y-axis represents the AUPRC.

Results overall show that more separated hidden spaces between two provenances correspond with a less robust model. For example, on the SHAC dataset, the RoBERTa trained using DANN is the least robust among all approaches (as shown in Figure 7.8c) for both $\alpha_{train} = 0.2$ and 5.0, indicated by highest absolute coefficients of 0.290 and 0.310, respectively. When checking Figure 7.12 for the representations of hidden spaces, we can find that the two groups under DANN are also the most separated, with a clear line in the middle. In comparison, the best performer of MMD in this case (with lowest absolute coefficients of 0.007 and 0.004 for $\alpha_{train} = 0.2$ and 5.0, respectively) show high similarity between the representations. A similar pattern can be found on other datasets as well.

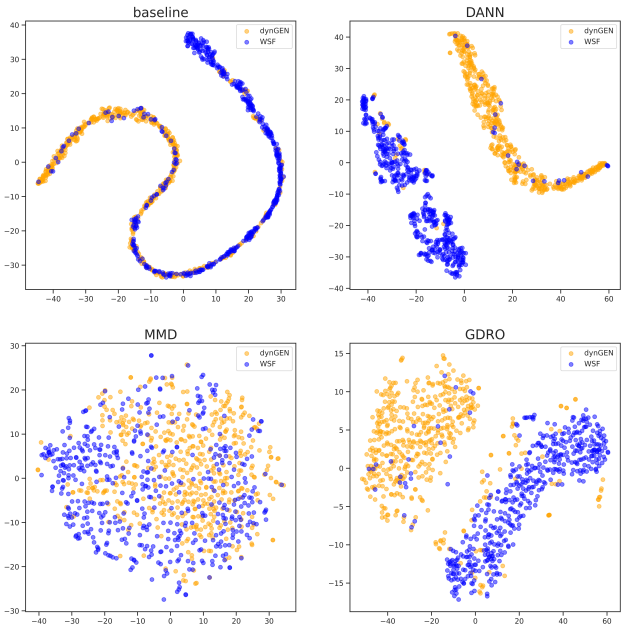


(a) $\alpha_{train} = 0.2$

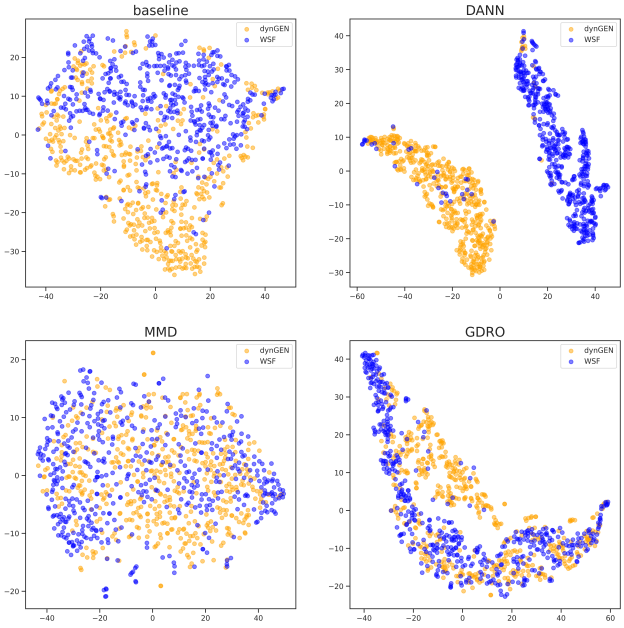


(b) $\alpha_{train} = 5.0$

Figure 7.10: Representations after t-SNE from two provenances under different robust learning approaches. On the Cognitive Distortion dataset.

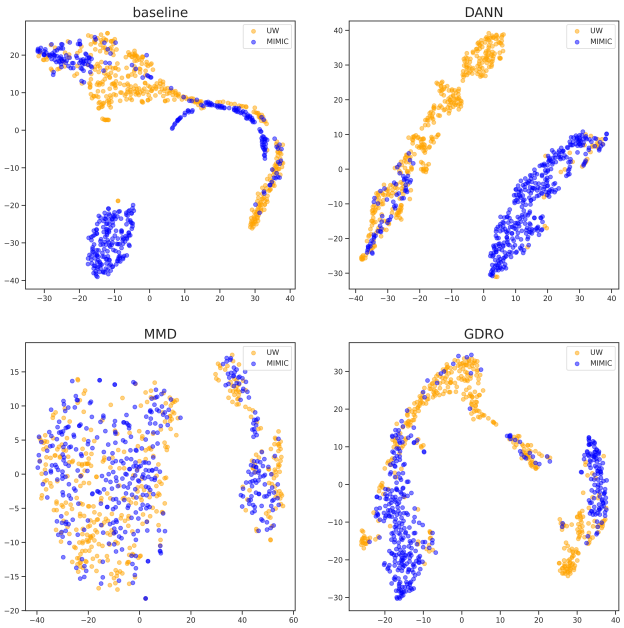


(a) $\alpha_{train} = 0.2$

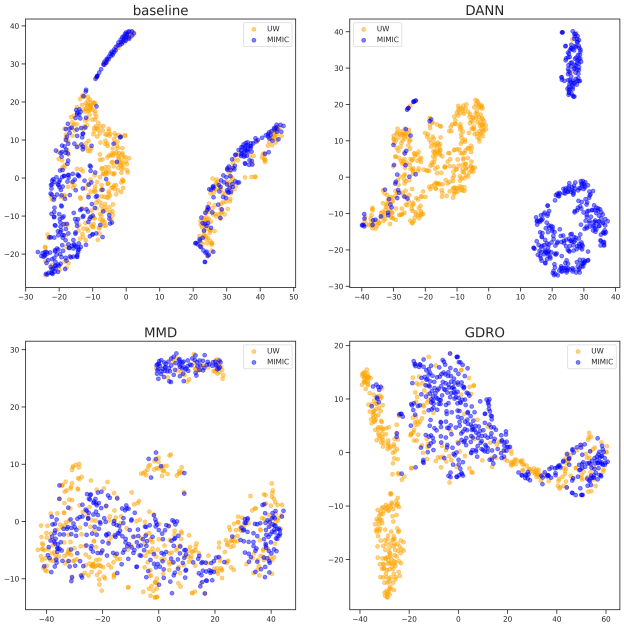


(b) $\alpha_{train} = 5.0$

Figure 7.11: Representations after t-SNE from two provenances under different robust learning approaches. On the Hate Speech dataset.



(a) $\alpha_{train} = 0.2$



(b) $\alpha_{train} = 5.0$

Figure 7.12: Representations after t-SNE from two provenances under different robust learning approaches. On the SHAC dataset.

7.8 Discussion

Confounding by provenance can occur when data are integrated from multiple sources, within which the distributions of predictors and targets differ. Provenance shift concerns cases in which the provenance-specific class distribution changes between the training set and test (or deployment) set. In this situation, any provenance-specific biases learned by the model would be expected to harm performance. In this work, we proposed a novel provenance effect reduction procedure, TAPER, and a dominance-aligned extension, DAPPER, both of which directly modify model weights through task arithmetic operating on task vectors. We evaluated the proposed methods on three datasets. Overall, results show improved model robustness with provenance effect reduction, especially with the dominance-aligned extension. With training dataset compositions favoring positive examples from each site ($\alpha_{train} = 0.2$ vs 5), DAPPER improved both model robustness and "worst-case" performance, indicating that it effectively mitigates confounding by provenance. These performance improvements are contingent upon provenance shift. In the circumstance in which provenance-specific class distribution at test time matches these distributions in the training set ("best-case" performance) performance with DAPPER deteriorates. This suggests that the biases learned by the uncorrected model are advantageous when they happen to align with the test set. The improved robustness suggests our methods shift the emphasis assigned to certain terms and semantic features, as demonstrated by the SHAP analysis. The methods are especially helpful when provenance shift is severe. The more shift there is, the worse the baseline model performs, and the better the adjusted model performs.

The "default" scaling of provenance effect reduction procedures is $\lambda = 1$, which gives task vectors representing the desired (primary classification objective) and undesired (provenance classification objective) model behaviors equal weight. However, as observed, this weighting may not be optimal. When the "optimal" λ value is selected based on worst-case performance, DAPPER always improves over baseline fine-tuned models in both robustness and AUPRC. We also established a simple way for recommending a λ value on a new dataset,

and showed that it can generate models with performance very close, if not identical, to the "optimal" setting. Detailed results indicate that a larger λ (around 1.2 or 1.4) may be a "sweet spot" for RoBERTa, and a smaller λ (around 0.2 or 0.4) works best with the Llama-2 model. However, the best setting varies across datasets. It is also hard to define an "optimal" scaling factor since the trade-off between robustness and performance is always present. Improvements in both robustness and $AUPRC_{worst}$ are observed when λ grows from 0, but after some turning point the trade-off is evident. This turning point differs across models and datasets.

DAPPER is only meaningful for a training set with different site-specific positive rates i.e., when $\alpha_{train} \neq 1$, to define the dominant provenance class (the class with a higher positives rate). For the case of $\alpha_{train} = 1$, which effectively eliminates the possibility of learning provenance-specific biases because the positive class rates are equal across sites, we nonetheless tested the method with both arbitrarily defined dominant provenances. As anticipated, results indicate that DAPPER harms model robustness and worst-case performance, as shown in A.2. This is likely due to the fact that no discrimination in background positive rates across provenances exists thus breaking the spurious link between predictors and provenance (sources).

Two models were used in this study, RoBERTa and Llama-2. Baseline fine-tuning of both models shows similar results, in both robustness and performance. However, they react differently to TAPER and DAPPER, in that Llama-2 has worse robustness than RoBERTa. When the provenance shift is small, the performance is close; when the shift is large, Llama-2 model falls behind RoBERTa by some margin across all three datasets. We can still see the improved robustness of the Llama-2 model, and its sensitivity to the scaling factor, which shows similar patterns to RoBERTa. It should be noted that even though the Llama-2 7b model has a larger parameter set than RoBERTa, the number of trainable parameters is smaller when using LoRA with further decreases in representational capacities from quantization. The similarity in results strengthens our argument for utilizing LoRA matrices for task arithmetic. This provides an alternative to generating full-scale task vectors on Large

Language Models (LLMs) for low-resource training.

Besides TAPER and DAPPER as direct post-hoc manipulations of hidden spaces, three robust learning methods were also explored in this chapter. Domain alignment through MMD generated the most robust models. It achieved the best "worst-case" performance as well on the SHAC dataset. DANN is the approach that led to the least robust models and best $AUPRC_{best}$ on the Hate Speech dataset. Analysis of the performance by provenance and hidden space separation show the varying effectiveness of these methods in promoting confusion in hidden spaces. The visualization analysis provides an explanation of why DANN is not as robust as others.

Several limitations should be noticed. For DAPPER, we empirically searched for an optimal scaling factor, but this was in the setting of fixing $\lambda \in [0, 2]$. This range does not guarantee a successful search for an optimal value. Theoretical guidance for a desired direction and norms of the resulting task vector is also lacking. Our straightforward way of estimating λ for new datasets from existing ones also has limitations when the scaling effect on one dataset is very different from others. For LoRA fine-tuning, only Query and Value matrices were used, greatly reducing the number of trainable parameters of the Llama-2 7b model. Even though similar patterns were observed with RoBERTa, it remains unknown how different LoRA settings affect provenance effect reduction methods. For robust learning, t-SNE visualization is a simplification of high-dimensional hidden spaces, thus could be misleading when the problem is complex. More causal explanations are needed. In this work, we only experimented with natural language. In the future, it would be of interest to take our methods to other data modalities, such as images.

7.9 Conclusions

In the work described in this chapter, I experimented with methods for making language models robust under provenance shift, while trying to preserve their performance. This chapter focused on hidden spaces, through two approaches: direct manipulation of hidden spaces and robust learning. TAPER and DAPPER are based on the usage of task vectors

and related task arithmetic. DAPPER showed improvements in robustness and worst-case performance, with careful choice of scaling factors. Furthermore, Low-Rank matrices (LoRA) proved effective for Dominance-Aligned Polarized Provenance Effect Reduction, which makes utilization of Large Language Models with LoRA feasible as an alternative to calculating task vectors directly. With DAPPER, task vectors can be used as tools for addressing confounding by provenance. With robust learning, worst-case performance is improved. MMD and GDRO can generate less separated hidden spaces from two provenances, leading to models with strong robustness. Consequently, it is recommended to use those approaches for building more reliable and robust applications for clinical NLP and healthcare.

Chapter 8

SUMMARY OF ADJUSTMENT METHODS

In the previous chapters, I introduced the problem of provenance shift and proposed a simulation and evaluation framework to systematically test any model with various degrees of shift. Several methods for adjustment were proposed and experimentally tested. They fall into three main categories: statistical adjustment (Backdoor Adjustment), distributional adjustment (the DistMatch framework), and adjustment of hidden spaces (including post-hoc editing of model weights and robust learning). In this chapter, I reflect on the previous results and summarize all methods, from three perspectives: robustness, worst-case performance ($AUPRC_{worst}$), and best-case performance ($AUPRC_{best}$). Detailed results are collected in this chapter. To help with a holistic view of methods, results are further aggregated by taking the mean over all three datasets on two training scenarios ($\alpha_{train} = 0.2$ and 5.0). Furthermore, improvement of each method over its corresponding baseline (either RoBERTa, Llama-2, logistic regression with Sentence-BERT, or logistic regression with binary unigrams) are calculated.

8.1 Robustness

Table 8.1 summarizes robustness across all methods. It should be noted that the robustness measurement from absolute coefficients of the fitted line is an indicator of the model's performance being consistent, but not necessarily good or bad, across different degrees of confounding shift.

From the averaged results, the Backdoor Adjustment on logistic regression models using binary unigrams achieves the best robustness when averaged over all three datasets and two training scenarios. On average, it is the Expand by 4 that achieves the highest improvement

when comparing with the baseline (77% for EDA), followed by Backdoor Adjustment (71%) and then DistMatch framework (68%).

When checking separately, the Backdoor Adjustment is very effective in training robust models on the SHAC and Hata Speech datasets when using logistic regression models with binary unigrams. When fine-tuning RoBERTa models, the DistMatch framework with `mixup` provides the most robust models on the SHAC dataset, while augmentation and resampling techniques work well on the Cognitive Distortion and Hate Speech datasets. DAPPER can generate models that approach or match the most robust ones. Robust learning methods do not perform better than other adjustment methods, but they still outperform the baseline models in most cases.

When comparing different models (logistic regression vs RoBERTa), it is not necessary for a model to have a larger set of parameters to be more robust. On the contrary, regression models with Backdoor Adjustment achieve best robustness on the SHAC dataset. This is not the case for more complicated datasets, such as the Hate Speech dataset with $\alpha_{train} = 0.2$. In these cases, we can observe large performance differences from different adjustment methods.

Model	Text representation	Methods	Specifications	CD		Hate Speech		SHAC		Avg.†	Inc. %‡
				$\alpha_{train} = 0.2$	$\alpha_{train} = 5.0$	$\alpha_{train} = 0.2$	$\alpha_{train} = 5.0$	$\alpha_{train} = 0.2$	$\alpha_{train} = 5.0$		
Fine-tuning RoBERTa*		baseline	RoBERTa**	-0.226	0.216	-0.232	0.260	-0.044	0.029	0.168	
			Llama-2 7b	-0.198	0.214	-0.095	0.171	-0.017	0.021	0.119	
		DistMatch	mixup	-0.139	0.057	-0.108	0.159	0.011	0.005	0.080	52.4%
			resample	-0.175	-0.024	-0.128	-0.045	-0.041	0.005	0.070	58.3%
			EDA	0.024	0.078	-0.175	0.028	-0.016	0.005	0.054	67.9%
		Expand by 4	mixup	-0.185	-0.054	-0.106	0.126	0.030	0.081	0.097	42.3%
			resample	-0.155	-0.014	0.005	-0.016	-0.015	0.066	0.045	73.2%
			EDA	**0.012**	-0.036	**0.001**	0.065	0.068	0.048	0.038	77.4%
		DAPPER-O	RoBERTa	-0.135	0.025	-0.075	0.216	-0.019	0.058	0.088	47.6%
			Llama-2 7b	0.152	-0.269	0.121	-0.175	-0.007	**0.002**	0.121	-1.7%
		MMD		-0.093	0.038	-0.118	0.066	-0.007	-0.004	0.054	67.9%
		GDRO		-0.209	0.147	-0.259	-0.067	-0.049	0.058	0.132	21.4%
		DANN		0.249	-0.068	0.279	-0.287	0.290	-0.310	0.247	-47.0%
Logistic Regression	Sentence-BERT	no aug		-0.216	0.079	-0.139	0.067	-0.083	0.064	0.108	
		Backdoor		-0.149	-0.173	-0.049	-0.044	-0.028	-0.035	0.080	25.9%
		DistMatch	mixup	-0.119	-0.083	-0.051	-0.011	-0.015	-0.005	0.047	56.5%
			resample	-0.118	-0.085	-0.052	**-0.005**	-0.014	-0.012	0.048	55.6%
			LLM generation	-0.145	-0.044					0.095	12.0%
	binary unigram	no aug		-0.158	0.113	-0.154	0.150	-0.047	0.048	0.112	
		Backdoor		-0.058	-0.058	-0.024	0.045	**-0.006**	**0.002**	**0.032**	71.4%
		DistMatch	mixup	-0.088	0.013	-0.094	0.080	-0.025	0.019	0.053	52.7%
			resample	-0.095	**0.011**	-0.088	0.079	-0.024	0.018	0.053	52.7%
			LLM generation	-0.085	0.062					0.074	33.9%

Table 8.1: Model robustness summary. Coefficients of the fitted line under the simulation framework are reported. *Fine-tuning RoBERTa: if not otherwise specified (training Llama-2 7b), the base model is RoBERTa. **RoBERTa: results are collected on the fine-tuned RoBERTa model through 20 epochs, in accordance with experiments in the DistMatch framework. Average†: mean robustness (absolute coefficients) from all settings across 3 datasets. Inc. %‡: improvements for mean absolute coefficients over baseline (either RoBERTa, Llama-2, logistic regression with Sentence-BERT, or logistic regression with binary unigrams).

8.2 Worst-case Performance

The worst-case performance, measured by AUPRC values for the testing set when $\alpha_{test} = 1/\alpha_{train}$, is referred as $AUPRC_{worst}$ throughout this work. In this section, a summary of $AUPRC_{worst}$ from all methods is listed in Table 8.2, with the best performance highlighted in bold.

Overall, DAPPER trained on Llama-2 7b shows the best worst-case performances across all settings, except on the Cognitive Distortion dataset when $\alpha_{train} = 5.0$. In that case, DAPPER with Llama-2 7b still achieved an AUPRC of 0.86, very close to the best of 0.87. The DistMatch framework, on average, shows the greatest improvements of 36.8% over the RoBERTa baseline for $AUPRC_{worst}$, followed by MMD of 33.3%. This indicates the effectiveness of the methods for improving in extreme shift scenarios. In comparison, the Llama-2 already shows a strong baseline performance of 0.74 on average, thus making improvement by best performing DAPPER-O appear relatively small (18.9%).

Results also show that for RoBERTa models, each augmentation and resampling technique under the DistMatch framework outperforms the corresponding one with Expand by 4. This indicates that both adjusting the label distribution and introducing some diversity for the predictors (texts) can contribute to great improvements in performance.

When checking each dataset separately, DAPPER with a fine-tuned RoBERTa model performs poorer than Llama-2 in $AUPRC_{worst}$, especially given it has larger trainable set of parameters than Llama-2 7b with LoRA and quantization. When comparing neural networks with logistic regression models, it is usually the case that the former provide better performance, showing the benefits coming from these larger models (RoBERTa and Llama-2 7b) when it comes to language modeling and prediction. It should be noted, however, that logistic regression with Sentence-BERT or binary unigrams still provides a good baseline. For example, when applied with Backdoor Adjustment, logistic regression with Sentence-BERT, when trained on the set with $\alpha_{train} = 5.0$, has $AUPRC_{worst}$ of 0.87 on the Cognitive Distortion dataset, which is the best across all models, and 0.81 on the Hate Speech dataset,

which is close to the best performance of 0.87.

The DistMatch framework stands out using logistic regression models on the Cognitive Distortion dataset with $\alpha_{train} = 5.0$. Other than this, the framework with augmentation and resampling techniques does not generate very strong worst-case performances. Robust learning approaches also does not perform well on $AUPRC_{worst}$.

Model	Text representation	Methods	Specifications	CD		Hate Speech		SHAC		Avg.†	Inc. %‡
				$\alpha_{train} = 0.2$	$\alpha_{train} = 5.0$	$\alpha_{train} = 0.2$	$\alpha_{train} = 5.0$	$\alpha_{train} = 0.2$	$\alpha_{train} = 5.0$		
Fine-tuning RoBERTa*		baseline	RoBERTa**	0.46 ± 0.02	0.40 ± 0.01	0.40 ± 0.02	0.37 ± 0.01	0.91 ± 0.02	0.87 ± 0.03	0.57	
			Llama-2 7b	0.57 ± 0.04	0.54 ± 0.01	0.76 ± 0.05	0.68 ± 0.01	0.96 ± 0.01	0.95 ± 0.01	0.74	
		DistMatch	mixup	0.67 ± 0.03	0.76 ± 0.05	0.68 ± 0.05	0.65 ± 0.02	0.95 ± 0.01	0.96 ± 0.01	0.78	36.8%
			resample	0.58 ± 0.04	0.76 ± 0.04	0.62 ± 0.06	0.56 ± 0.04	0.89 ± 0.01	0.87 ± 0.01	0.71	24.6%
			EDA	0.43 ± 0.02	0.69 ± 0.06	0.47 ± 0.02	0.58 ± 0.05	0.92 ± 0.01	0.89 ± 0.03	0.66	15.8%
		Expand by 4	mixup	0.60 ± 0.02	0.72 ± 0.04	0.69 ± 0.03	0.69 ± 0.03	0.96 ± 0.01	0.73 ± 0.07	0.73	28.1%
			resample	0.55 ± 0.01	0.50 ± 0.02	0.50 ± 0.02	0.55 ± 0.03	0.71 ± 0.01	0.47 ± 0.02	0.55	-3.5%
			EDA	0.50 ± 0.03	0.60 ± 0.02	0.50 ± 0.01	0.55 ± 0.04	0.69 ± 0.04	0.85 ± 0.01	0.62	8.8%
		DAPPER-O	RoBERTa	0.62 ± 0.03	0.73 ± 0.05	0.67 ± 0.05	0.50 ± 0.02	0.94 ± 0.02	0.87 ± 0.03	0.72	26.3%
			Llama-2 7b	**0.72 ± 0.08**	0.86 ± 0.03	**0.86 ± 0.04**	**0.87 ± 0.02**	**0.97 ± 0.01**	**0.97 ± 0.01**	**0.88**	18.9%
		MMD		0.65 ± 0.02	0.72 ± 0.03	0.61 ± 0.03	0.69 ± 0.03	0.95 ± 0.01	0.96 ± 0.01	0.76	33.3%
		GDRO		0.59 ± 0.02	0.63 ± 0.05	0.50 ± 0.03	0.59 ± 0.03	0.90 ± 0.03	0.89 ± 0.02	0.68	19.3%
		DANN		0.69 ± 0.02	0.54 ± 0.03	0.74 ± 0.03	0.79 ± 0.02	0.78 ± 0.04	0.80 ± 0.04	0.72	26.3%
Logistic Regression	Sentence-BERT	no aug		0.59 ± 0.02	0.71 ± 0.04	0.63 ± 0.01	0.73 ± 0.03	0.84 ± 0.02	0.84 ± 0.01	0.72	
		Backdoor		0.67 ± 0.02	**0.87 ± 0.01**	0.71 ± 0.01	0.81 ± 0.03	0.90 ± 0.02	0.94 ± 0.01	0.82	13.9%
		DistMatch	mixup	0.69 ± 0.01	0.83 ± 0.02	0.72 ± 0.02	0.77 ± 0.04	0.91 ± 0.02	0.92 ± 0.01	0.81	12.5%
			resample	0.69 ± 0.01	0.83 ± 0.02	0.72 ± 0.02	0.77 ± 0.04	0.91 ± 0.01	0.93 ± 0.01	0.81	12.5%
			LLM generation	0.65 ± 0.03	0.81 ± 0.02					0.73	1.4%
	binary unigram	no aug		0.59 ± 0.02	0.63 ± 0.06	0.54 ± 0.02	0.58 ± 0.04	0.91 ± 0.01	0.90 ± 0.01	0.69	
		Backdoor		0.68 ± 0.03	0.76 ± 0.05	0.63 ± 0.04	0.67 ± 0.03	0.94 ± 0.01	0.95 ± 0.01	0.77	11.6%
		DistMatch	mixup	0.66 ± 0.02	0.70 ± 0.05	0.59 ± 0.02	0.63 ± 0.04	0.93 ± 0.01	0.93 ± 0.01	0.74	7.2%
			resample	0.65 ± 0.02	0.70 ± 0.05	0.59 ± 0.02	0.63 ± 0.04	0.93 ± 0.01	0.93 ± 0.01	0.74	7.2%
			LLM generation	0.67 ± 0.02	0.66 ± 0.05					0.67	-2.9%

Table 8.2: Worst-case performance summary. Results correspond with $AUPRC_{worst}$. *Fine-tuning RoBERTa: if not otherwise specified (training Llama-2 7b), the base model is RoBERTa. **RoBERTa: results are collected on the fine-tuned RoBERTa model through 20 epochs, in accordance with experiments in the DistMatch framework. Average†: mean $AUPRC_{worst}$ from all settings across 3 datasets. Inc. %‡: mean $AUPRC_{worst}$ improvements over baseline (either RoBERTa, Llama-2, logistic regression with Sentence-BERT, or logistic regression with binary unigrams).

8.3 Best-case Performance

The best-case performance, measured by AUPRC values at the testing set when $\alpha_{test} = \alpha_{train}$, is referred as $AUPRC_{best}$ throughout this work. In this section, a summary of $AUPRC_{best}$ from all methods is listed in Table 8.3, with the best performance highlighted with bold.

Overall, the baseline models, when evaluated at $\alpha_{test} = \alpha_{train}$, provides the best AUPRC across all three datasets. Baseline Llama-2 7b models perform best with the only exception on the Cognitive Distortion dataset with $\alpha_{train} = 0.2$, and in that case, it is the baseline logistic regression model with Sentence-BERT embeddings is the best. This indicates that when no shift between the training and testing sets, the standard training procedure works very well. Generally, the performance from baseline models under the best case scenario indicates that a learned bias from the training process can be advantageous when the test set is also biased toward the same direction (i.e., when $\alpha_{test} = \alpha_{train}$), and any adjustment trying to remove such bias will then harm the performance for this specific scenario.

GDRO achieves best $AUPRC_{best}$ together with the baseline on the Cognitive Distortion dataset with $\alpha_{train} = 0.2$. In general, robust learning approaches perform worse than the best method by a small margin. For example, on the SHAC dataset, $AUPRC_{best}$ is 0.97 in comparison with the best of 0.98. MMD is more consistent across different training scenarios ($\alpha_{train} = 0.2$ adn 5.0) than GDRO. DANN performs worst for $AUPRC_{best}$. DAPPER, in this scenario, helps train Llama-2 7b to achieve best $AUPRC_{best}$ on the SHAC dataset. On the other two datasets, DAPPER makes models worse for $AUPRC_{best}$.

The DistMatch framework, in this scenario where no provenance shift is present, can still provide strong performance, with slightly lower $AUPRC_{best}$ than the best method by 0.02-0.06 across all three datasets. When using the logistic regression model, the gap between the DistMatch framework and baseline becomes larger on the Cognitive Distortion and Hate Speech datasets. On the SHAC dataset, however, the DistMatch framework improves upon the baseline.

Model	Text representation	Methods	Specifications	CD		Hate Speech		SHAC		Avg.†	Inc. %‡
				$\alpha_{train} = 0.2$	$\alpha_{train} = 5.0$	$\alpha_{train} = 0.2$	$\alpha_{train} = 5.0$	$\alpha_{train} = 0.2$	$\alpha_{train} = 5.0$		
Fine-tuning RoBERTa*		baseline	RoBERTa**	0.79 ± 0.05	0.70 ± 0.04	0.74 ± 0.01	0.75 ± 0.04	0.97 ± 0.01	0.91 ± 0.01	0.81	
			Llama-2 7b	0.81 ± 0.01	**0.85 ± 0.04**	**0.89 ± 0.04**	**0.92 ± 0.01**	**0.98 ± 0.01**	**0.98 ± 0.01**	**0.91**	
		DistMatch	mixup	0.87 ± 0.03	0.83 ± 0.03	0.83 ± 0.06	0.87 ± 0.02	0.94 ± 0.01	0.96 ± 0.01	0.88	8.6%
			resample	0.82 ± 0.02	0.72 ± 0.02	0.80 ± 0.05	0.49 ± 0.02	0.95 ± 0.02	0.88 ± 0.01	0.78	-3.7%
			EDA	0.39 ± 0.01	0.79 ± 0.03	0.72 ± 0.02	0.62 ± 0.04	0.94 ± 0.01	0.89 ± 0.03	0.73	-9.9%
		Expand by 4	mixup	0.87 ± 0.02	0.65 ± 0.03	0.84 ± 0.03	0.87 ± 0.01	0.91 ± 0.02	0.86 ± 0.04	0.83	2.5%
			resample	0.77 ± 0.03	0.49 ± 0.02	0.49 ± 0.01	0.53 ± 0.01	0.74 ± 0.03	0.57 ± 0.03	0.60	-25.9%
			EDA	0.49 ± 0.01	0.55 ± 0.01	0.50 ± 0.01	0.46 ± 0.01	0.59 ± 0.03	0.91 ± 0.02	0.58	-28.4%
		DAPPER-O	RoBERTa	0.81 ± 0.03	0.77 ± 0.01	0.78 ± 0.05	0.86 ± 0.01	0.97 ± 0.01	0.95 ± 0.02	0.86	6.2%
			Llama-2 7b	0.51 ± 0.03	0.47 ± 0.02	0.69 ± 0.03	0.62 ± 0.05	**0.98 ± 0.01**	0.97 ± 0.01	0.71	-22.0%
		MMD		0.79 ± 0.02	0.77 ± 0.03	0.78 ± 0.05	0.78 ± 0.04	0.96 ± 0.01	0.96 ± 0.01	0.84	3.7%
		GDRO		**0.89 ± 0.02**	0.83 ± 0.03	0.87 ± 0.04	0.49 ± 0.03	0.97 ± 0.01	0.97 ± 0.01	0.84	3.7%
		DANN		0.33 ± 0.01	0.45 ± 0.03	0.35 ± 0.01	0.38 ± 0.02	0.36 ± 0.01	0.35 ± 0.01	0.37	-54.3%
Logistic Regression	Sentence-BERT	no aug		**0.89 ± 0.02**	0.83 ± 0.03	0.83 ± 0.04	0.82 ± 0.03	0.96 ± 0.01	0.94 ± 0.01	0.88	
		Backdoor		0.88 ± 0.02	0.63 ± 0.03	0.78 ± 0.05	0.74 ± 0.03	0.93 ± 0.01	0.89 ± 0.02	0.81	-8.0%
		DistMatch	mixup	0.86 ± 0.02	0.72 ± 0.02	0.79 ± 0.05	0.75 ± 0.02	0.93 ± 0.01	0.92 ± 0.02	0.83	-5.7%
			resample	0.85 ± 0.02	0.72 ± 0.02	0.79 ± 0.05	0.76 ± 0.02	0.93 ± 0.01	0.92 ± 0.02	0.83	-5.7%
			LLM generation	0.85 ± 0.02	0.75 ± 0.03					0.80	-9.1%
	binary unigram	no aug		0.81 ± 0.03	0.79 ± 0.03	0.75 ± 0.03	0.80 ± 0.04	0.97 ± 0.01	0.96 ± 0.01	0.85	
		Backdoor		0.75 ± 0.04	0.68 ± 0.03	0.65 ± 0.04	0.74 ± 0.04	0.95 ± 0.01	0.95 ± 0.01	0.79	-7.1%
		DistMatch	mixup	0.77 ± 0.04	0.72 ± 0.03	0.71 ± 0.04	0.75 ± 0.04	0.97 ± 0.01	0.95 ± 0.01	0.81	-4.7%
			resample	0.78 ± 0.03	0.72 ± 0.02	0.70 ± 0.03	0.75 ± 0.04	0.97 ± 0.01	0.95 ± 0.01	0.81	-4.7%
			LLM generation	0.78 ± 0.04	0.75 ± 0.02					0.77	-9.4%

Table 8.3: Best-case performance summary. Results correspond with $AUPRC_{best}$. *Fine-tuning RoBERTa: if not otherwise specified (training Llama-2 7b), the base model is RoBERTa. **RoBERTa: results are collected on the fine-tuned RoBERTa model through 20 epochs, in accordance with experiments in the DistMatch framework. Average†: mean $AUPRC_{best}$ from all settings across 3 datasets. Inc. %‡: mean $AUPRC_{best}$ improvements over baseline (either RoBERTa, Llama-2, logistic regression with Sentence-BERT, or logistic regression with binary unigrams.)

8.4 Conclusions

In this chapter, I summarized all methods proposed and evaluated in this work from three perspectives: robustness, worst-case performance, and best-case performance.

Overall, the proposed methods improve model robustness. Baseline models show very poor robustness under provenance shift. Backdoor Adjustment and DistMatch framework can make models very robust across various degrees of provenance shift. However, evaluated at two specific scenarios: $\alpha_{test} = \alpha_{train}$ and $\alpha_{test} = 1/\alpha_{train}$, these two methods lead to mediocre performance and are not very competitive. The Distmatch framework works well on the Cognitive Distortion and SHAC dataset with strong $AUPRC_{worst}$ and $AUPRC_{best}$, and underperforms by around 0.02 in comparison with the best performing model.

Worst-case performance can be improved through adjustment methods. The baseline models, following the standard training procedure with the assumption of no shift between the training and test set, usually fail in this scenario with extreme provenance shift. DAPPER, as a post-hoc way of manipulating model weights, can lead to models with the best worst-case performance, especially when applied to the Llama-2 7b model, but it sacrifices some robustness and best-case performance ($AUPRC_{best}$). All other methods can also greatly improve $AUPRC_{worst}$ over their baseline counterparts.

When it comes to the best-case performance, baseline models achieve the best $AUPRC_{best}$ in many cases. This is likely due to that the setting of $\alpha_{test} = \alpha_{train}$ satisfies the common assumption in machine learning of the same or similar distribution from training to test time, thus requiring no adjustment. All adjustment methods discussed in this work tend to optimize for the scenario where there is shift, or even extreme shfit, and this can be harmful in this case. Robust learning methods, with training toward confused hidden spaces as shown in Chapter 7, can work well in the best-case scenario with no provenance shift. It should be noted that most of those methods are targeted for adapting models to other or new domains, but not specifically designed for the problem of provenance shift. Thus, at this single evaluation point ($\alpha_{test} = \alpha_{train}$), they work as expected.

From the training perspective, the logistic regression model is the easiest to train, implement, and deploy. Together with Backdoor Adjustment and the DistMatch framework, it can achieve great robustness. However, its worst-case performance is poorer by large margins than neural models (with some exceptions). Neural networks, especially the Llama-2 7b model, can provide best $AUPRC_{worst}$, but are relatively complicated to train within limitations of computing resources. DAPPER provides a simple framework for manipulating fine-tuned model weights to achieve robustness and good performance under provenance shift, making it possible to utilize neural networks of large sizes for the problem. DAPPER as adjustment method can lead to poor performance when there is no shift. However, the hyperparameter λ makes the DAPPER framework very flexible to change according to different needs. Robust learning methods show their effectiveness in generating shared hidden spaces, thus also helping to train more robust models than the baseline. When using the RoBERTa model, they can train models with similar, and sometimes better, robustness, $AUPRC_{worst}$, and $AUPRC_{best}$ to the DAPPER. DistMatch provides a training framework that can be applied to any model. It is effective in training robust models, but less effective in improving model performance compared with other methods, though still better than baseline for $AUPRC_{worst}$.

It should be noted that it is the best practice to consider all three perspectives to have a holistic view of the problem at hand when choosing between methods. Any single metric, when interpreted alone, can only illustrate part of the story. There is sometimes a trade-off between the three dimensions, that requires careful examination of the goal for successful application.

Chapter 9

CONCLUSIONS

This dissertation defines the problem of confounding by provenance and shows how the related shift in provenance can lead to a model that loses its accuracy when shift happens between training time and test/deployment time. As mitigation, several methodologies are proposed and evaluated in order to adjust for such shifts, ranging from statistical methods to adjustment in neural networks. I conclude with a summary of contributions, a discussion of limitations, and finally potential directions for future work.

9.1 Contributions

My primary contributions are organized around three specific aims: (1) building simulation and evaluation framework for provenance shift; (2) mitigation through distribution adjustment, which focuses on eliminating provenance-wide differences in class label distribution, thereby eliminating the causal link between site and target label prevalence that affects distributions directly; (3) mitigation through manipulation of hidden spaces, which focuses on eliminating the ability of models to perceive site-specific language differences, thereby eliminating the link between site and text representation.

For Aim 1, I developed and built a simulation framework for automatically test model's robustness under different degrees of provenance shift in experimental settings. The framework, with the core measurement of shifts being a positive rate ratios across two provenances, now targets on binary classification problem with two provenances. In terms of evaluation, we proposed measuring model's fitted performance across shifts and using the absolute value of the coefficient as surrogate for robustness. A smaller value is desired, indicating a more robust model. Model's performance is further split into two components, namely "best-case"

and "worst-case" performances. One goal is to maintain the "best-case" performance in the meanwhile increase "worst-case"performance. This simulation and evaluation framework allows us to systematically compare models for their robustness and performance.

As an evaluation of Backdoor Adjustment, we put it to test under this framework. New text representation methods are also introduced, including Sentence-BERT embeddings, which had not been used with Backdoor Adjustment previously. Backdoor Adjustment is flexible, building upon a logistic regression model. This setting provides a simple yet powerful adjustment method that can greatly increase model's robustness. On simple classification tasks, such as using SHAC for drug abuse prediction, its performance is acceptable. However, when tasks become complicated, the overall performance from logistic regressions is poor in comparison with neural networks, even though Backdoor Adjustment can still achieve robustness (with uniformly poor performance). Therefore, there is a clear need for adjustment methods compatible with current NLP approaches.

For Aim 2, the main goal is to adjust distributions. Several data augmentation techniques, as a way of making distributional adjustment, were explored. Augmentation through hidden spaces (`mixup`) and Easy Data Augmentation (EDA) showed improvements in accuracy for rare classes. Based on this, we further proposed the DistMatch framework, which aims to make positive rates across provenances equal. This framework, with `mixup`, EDA, and oversampling as adjustment techniques, showed improvements in robustness and "worst-case" performance. Large Language Model generations through rephrasing did not show strong effects in robustness, possibly due to the quality of generated texts.

For Aim 3, I focused on the manipulation of hidden spaces of neural networks for provenance shift adjustment. One approach of Provenance Effect Reduction (TAPER and DAPPER) is based on the work of task vectors. I extended this idea to the problem of provenance shift such that two task vectors are operated on, one serving as the main prediction task, other as the auxiliary task for provenance prediction. We introduced two methods of provenance effect reduction and showed the dominance-aligned version (DAPPER) performed better than the non-aligned version (TAPER). The scaling of those two task vectors is not

apparent, so a range of experiments were set in search of an "optimal" one. We notice that the RoBERTa and Llama-2 models, even though similar in behavior, favor different "optimal" scaling values. Based on this, an estimated value could be obtained, and we showed this results in robustness and performance very close to the "optimal" one that was determined through exhaustive search. Furthermore, we proved that by performing Low-rank adaptation (LoRA) during the fine-tuning process of a model, those obtained LoRA weights can be used as task vectors. This is time- and resource-conserving especially when the language model is of billions of parameters in size, which proved beneficial for our more challenging NLP tasks. The next adjustment came from robust learning. We discussed and evaluated three common approaches: domain alignment through MMD, group distributionally robust optimization, and domain-adversarial training. Those methods can effectively make hidden spaces from different provenances closer to each other, thus leading to more robust models.

Overall, through those three specific aims, a coherent framework, from simulation and evaluation, to mitigation methods was constructed. This framework and proposed methods allows one to evaluate potential effects of provenance shift and purposely adjust for it. Datasets from multiple institutions can then be combined while maintaining relatively good performance when things change at the deployment time. In biomedical research, this combined dataset with an increased size can include more diverse examples, thus helping with model training process and generalizability. The proposed methods focus on confounding by provenance and related shift. The provenance variable does not limit to only institutions. It can be other things like ethnic groups, genders, cultural backgrounds, and socioeconomic statuses. One application is model equity or fairness, which is a topic gaining more and more attention when AI models become pervasive in our daily lives and in the process of clinical decision making. This is especially concerning when algorithms make biased decisions based on learning spurious correlations. The proposed framework provides one way for examining such effects and making corrections so that a more reliable and trustworthy model can be built to facilitate day-to-day clinical decision support and improve the quality of healthcare.

9.2 Limitations and Future Work

The simulation framework is built upon two provenances for binary classification. Ideas on expending this binary framework are only briefly introduced. When there are multiple data sources and class labels, one first step is to use divergence as a measurement for provenance shift between the training and test/deployment time. In terms of evaluation, the technique of calculating performance (accuracy or AUPRC) in the manner of One-vs-One or One-vs-All naturally extends our current framework. However, implementation of these ideas falls outside the scope of this work.

The models used in specific experiments do not cover a wide range of machine learning or deep learning models, especially for the Backdoor Adjustment, where logistic regression models predominate. Future work will include extending the Backdoor Adjustment to neural networks. Datasets used in this work are two biomedical sets and one from the general domain. Future work should include more datasets to validate our framework. Besides, since the whole simulation framework and proposed adjustment methods are all model-agnostic, it remains untested how it performs on other data modalities, including structure data such as electronic health records or imaging data, and data from multiple modalities combined. It remains as future work to make our framework generalizable.

9.3 Concluding Remarks

Confounding by provenance is pervasive in clinical machine learning applications. Multi-institutional datasets are widely used for machine learning from clinical data, to increase dataset size and improve generalization. Models developed under such a composite dataset, when target prevalence rates among subgroups differ, may learn to recognize the source of a data element, for example, institution-specific section headers, or region-specific dialects in natural language processing. These differences in prevalence may reflect local conditions but may also be artifacts of a sampling procedure deliberately used to enrich the sample from one of the sites. Models trained without correction may fail to generalize beyond their

training data, and make inaccurate biased predictions at the point of care.

In this dissertation, I formally define the problem of confounding by provenance and provenance shift and characterize its detrimental effects when left unadjusted. Three major adjustment approaches are proposed, targeting different aspects of the relationship between the provenance variable, predictors (text), and labels. The dissertation describes evaluations of these approaches within a novel unified framework that simulates different degrees of provenance shift. Results show that most adjustment methods can improve model robustness. Backdoor Adjustment and the DistMatch framework are easy to implement, with moderate improvements. Provenance Effect Reduction on Llama-2 shows the best performance when the provenance shift is extreme, indicating its strong correction effectiveness.

The proposed methods were developed for provenance shift. However, other kinds of bias, if following a similar pattern where there exist prevalence differences across subgroups with effects on both predictors and labels, are amenable to adjustment using the proposed methods also. Some common sources of bias include gender, race, and socioeconomic status, all key concerns in research on machine learning model fairness. The proposed simulation framework and adjustment methods are flexible and can be generalized to these problems as well. As such, this work provides a generalizable framework to address confounding bias, to support the development of more reliable and robust applications of clinical NLP in healthcare and biomedical research.

BIBLIOGRAPHY

[1] F. Emmert-Streib, Z. Yang, H. Feng, S. Tripathi, and M. Dehmer, "An introductory review of deep learning for prediction models with big data," *Frontiers in Artificial Intelligence*, vol. 3, p. 4, 2020.

[2] L. Fregoso-Aparicio, J. Noguez, L. Montesinos, and J. A. García-García, "Machine learning and deep learning predictive models for type 2 diabetes: A systematic review," *Diabetology & metabolic syndrome*, vol. 13, no. 1, p. 148, 2021.

[3] Z. Han, J. Zhao, H. Leung, K. F. Ma, and W. Wang, "A review of deep learning models for time series prediction," *IEEE Sensors Journal*, vol. 21, no. 6, pp. 7833–7848, 2019.

[4] K. Swanson, G. Liu, D. B. Catacutan, A. Arnold, J. Zou, and J. M. Stokes, "Generative ai for designing and validating easily synthesizable and structurally novel antibiotics," *Nature Machine Intelligence*, vol. 6, no. 3, pp. 338–353, 2024.

[5] J. Jumper, R. Evans, A. Pritzel, *et al.*, "Highly accurate protein structure prediction with alphafold," *Nature*, vol. 596, no. 7873, pp. 583–589, 2021.

[6] M. Akdel, D. E. Pires, E. P. Pardo, *et al.*, "A structural biology community assessment of alphafold2 applications," *Nature Structural & Molecular Biology*, vol. 29, no. 11, pp. 1056–1067, 2022.

[7] S. Yang, P. Varghese, E. Stephenson, K. Tu, and J. Gronsbell, "Machine learning approaches for electronic health records phenotyping: A methodical review," *Journal of the American Medical Informatics Association*, vol. 30, no. 2, pp. 367–381, 2023.

[8] Z. Zeng, Y. Deng, X. Li, T. Naumann, and Y. Luo, "Natural language processing for ehr-based computational phenotyping," *IEEE/ACM transactions on computational biology and bioinformatics*, vol. 16, no. 1, pp. 139–153, 2018.

[9] K. Swanson, E. Wu, A. Zhang, A. A. Alizadeh, and J. Zou, "From patterns to patients: Advances in clinical machine learning for cancer diagnosis, prognosis, and treatment," *Cell*, 2023.

[10] M. A. Vollmer, B. Glampson, T. Mellan, *et al.*, "A unified machine learning approach to time series forecasting applied to demand at emergency departments," *BMC Emergency Medicine*, vol. 21, pp. 1–14, 2021.

[11] B. Percha, "Modern clinical text mining: A guide and review," *Annual review of biomedical data science*, vol. 4, no. 1, pp. 165–187, 2021.

[12] S. Wu, K. Roberts, S. Datta, *et al.*, "Deep learning in clinical natural language processing: A methodical review," *Journal of the American Medical Informatics Association*, vol. 27, no. 3, pp. 457–470, 2020.

[13] M. J. Sheller, B. Edwards, G. A. Reina, *et al.*, "Federated learning in medicine: Facilitating multi-institutional collaborations without sharing patient data," *Scientific reports*, vol. 10, no. 1, p. 12 598, 2020.

[14] V. Landeiro and A. Culotta, "Robust text classification in the presence of confounding bias," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, 2016.

[15] V. Landeiro and A. Culotta, "Robust text classification under confounding shift," *Journal of Artificial Intelligence Research*, vol. 63, pp. 391–419, 2018.

[16] M. L. Newman, C. J. Groom, L. D. Handelman, and J. W. Pennebaker, "Gender differences in language use: An analysis of 14,000 text samples," *Discourse processes*, vol. 45, no. 3, pp. 211–236, 2008.

[17] K. Howell, M. Barnes, J. Randall Curtis, *et al.*, "Controlling for confounding variables: Accounting for dataset bias in classifying patient-provider interactions," *Explainable AI in Healthcare and Medicine: Building a Culture of Transparency and Accountability*, pp. 271–282, 2021.

[18] Y. Guo, C. Li, C. Roan, S. Pakhomov, and T. Cohen, "Crossing the "cookie theft" corpus chasm: Applying what bert learns from outside data to the adress challenge dementia detection task," *Frontiers in Computer Science*, vol. 3, p. 642 517, 2021.

[19] X. Ding, Z. Sheng, M. Yetişgen, S. Pakhomov, and T. Cohen, "Backdoor adjustment of confounding by provenance for robust text classification of multi-institutional clinical notes," in *AMIA Annual Symposium Proceedings*, American Medical Informatics Association, vol. 2023, 2023, p. 923.

[20] X. Ding, Z. Sheng, B. Hur, F. Chen, S. V. Pakhomov, and T. Cohen, "Enhancing robustness of foundation model representations under provenance-related distribution shifts," *arXiv preprint arXiv:2312.05435*, 2023.

[21] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of big data*, vol. 6, no. 1, pp. 1–48, 2019.

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

[23] Y. Ganin, E. Ustinova, H. Ajakan, *et al.*, "Domain-adversarial training of neural networks," *The journal of machine learning research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[24] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," *arXiv preprint arXiv:1412.3474*, 2014.

[25] J. Quinonero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, *Dataset shift in machine learning*. Mit Press, 2008.

[26] A. Storkey *et al.*, "When training and test sets are different: Characterizing learning transfer," *Dataset shift in machine learning*, vol. 30, pp. 3–28, 2009.

[27] K. Zhang, B. Schölkopf, K. Muandet, and Z. Wang, "Domain adaptation under target and conditional shift," in *International conference on machine learning*, Pmlr, 2013, pp. 819–827.

[28] M. Gong, K. Zhang, T. Liu, D. Tao, C. Glymour, and B. Schölkopf, "Domain adaptation with conditional transferable components," in *International conference on machine learning*, PMLR, 2016, pp. 2839–2848.

[29] G. Widmer and M. Kubat, "Learning in the presence of concept drift and hidden contexts," *Machine learning*, vol. 23, pp. 69–101, 1996.

[30] A. Tsymbal, "The problem of concept drift: Definitions and related work," *Computer Science Department, Trinity College Dublin*, vol. 106, 2004.

[31] M. Sugiyama, M. Krauledat, and K.-R. Müller, "Covariate shift adaptation by importance weighted cross validation.," *Journal of Machine Learning Research*, vol. 8, no. 5, 2007.

[32] S. Bickel, M. Brückner, and T. Scheffer, "Discriminative learning under covariate shift.," *Journal of Machine Learning Research*, vol. 10, no. 9, 2009.

[33] X. Chen, M. Monfort, A. Liu, and B. D. Ziebart, "Robust covariate shift regression," in *Artificial Intelligence and Statistics*, PMLR, 2016, pp. 1270–1279.

[34] T. Walser, X. Cui, J. Yanagawa, *et al.*, "Smoking and lung cancer: The role of inflammation," *Proceedings of the American Thoracic Society*, vol. 5, no. 8, pp. 811–815, 2008.

[35] G. I. Webb and K. M. Ting, "On the application of roc analysis to predict classification performance under varying class distributions," *Machine learning*, vol. 58, pp. 25–32, 2005.

[36] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM computing surveys (CSUR)*, vol. 46, no. 4, pp. 1–37, 2014.

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

[38] Z. Xu, Y. Feng, Y. Li, *et al.*, "Predictive modeling of the risk of acute kidney injury in critical care: A systematic investigation of the class imbalance problem," *AMIA Summits on Translational Science Proceedings*, vol. 2019, p. 809, 2019.

[39] S. Makki, Z. Assaghir, Y. Taher, R. Haque, M.-S. Hacid, and H. Zeineddine, "An experimental study with imbalanced classification approaches for credit card fraud detection," *IEEE Access*, vol. 7, pp. 93 010–93 022, 2019.

[40] Y. Sun, A. K. Wong, and M. S. Kamel, "Classification of imbalanced data: A review," *International journal of pattern recognition and artificial intelligence*, vol. 23, no. 04, pp. 687–719, 2009.

[41] N. Japkowicz, "Concept-learning in the presence of between-class and within-class imbalances," in *Advances in Artificial Intelligence: 14th Biennial Conference of the Canadian Society for Computational Studies of Intelligence, AI 2001 Ottawa, Canada, June 7–9, 2001 Proceedings 14*, Springer, 2001, pp. 67–77.

[42] Y.-P. Zhang, L.-N. Zhang, and Y.-C. Wang, "Cluster-based majority under-sampling approaches for class imbalance learning," in *2010 2nd IEEE International Conference on Information and Financial Engineering*, IEEE, 2010, pp. 400–404.

[43] M. R. Smith, T. Martinez, and C. Giraud-Carrier, "An instance level analysis of data complexity," *Machine learning*, vol. 95, pp. 225–256, 2014.

[44] M. Kubat, S. Matwin, *et al.*, "Addressing the curse of imbalanced training sets: One-sided selection," in *Icml*, Citeseer, vol. 97, 1997, p. 179.

[45] D. L. Wilson, "Asymptotic properties of nearest neighbor rules using edited data," *IEEE Transactions on Systems, Man, and Cybernetics*, no. 3, pp. 408–421, 1972.

[46] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[47] M. E. Charlson, P. Pompei, K. L. Ales, and C. R. MacKenzie, "A new method of classifying prognostic comorbidity in longitudinal studies: Development and validation," *Journal of chronic diseases*, vol. 40, no. 5, pp. 373–383, 1987.

[48] K. Ng, J. Sun, J. Hu, and F. Wang, "Personalized predictive modeling and risk factor identification using patient similarity," *AMIA Summits on Translational Science Proceedings*, vol. 2015, p. 132, 2015.

[49] B. Zadrozny, J. Langford, and N. Abe, "Cost-sensitive learning by cost-proportionate example weighting," in *Third IEEE international conference on data mining*, IEEE, 2003, pp. 435–442.

[50] N. Japkowicz, "Supervised versus unsupervised binary-learning by feedforward neural networks," *Machine Learning*, vol. 42, no. 1-2, p. 97, 2001.

[51] L. M. Manevitz and M. Yousef, "One-class svms for document classification," *Journal of machine Learning research*, vol. 2, no. Dec, pp. 139–154, 2001.

[52] G. E. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD explorations newsletter*, vol. 6, no. 1, pp. 20–29, 2004.

[53] G. M. Weiss and F. Provost, "Learning when training data are costly: The effect of class distribution on tree induction," *Journal of artificial intelligence research*, vol. 19, pp. 315–354, 2003.

[54] A. M. O'Hare, A. I. Choi, D. Bertenthal, *et al.*, "Age affects outcomes in chronic kidney disease," *Journal of the American Society of Nephrology*, vol. 18, no. 10, pp. 2758–2765, 2007.

[55] R. Kazancioğlu, "Risk factors for chronic kidney disease: An update," *Kidney international supplements*, vol. 3, no. 4, pp. 368–371, 2013.

[56] D. N. Kyriacou and R. J. Lewis, "Confounding by indication in clinical research," *Jama*, vol. 316, no. 17, pp. 1818–1819, 2016.

[57] U. Pavalanathan and J. Eisenstein, "Confounds and consequences in geotagged twitter data," *arXiv preprint arXiv:1506.02275*, 2015.

[58] B. Hecht and M. Stephens, "A tale of cities: Urban biases in volunteered geographic information," in *proceedings of the international AAAI conference on web and social media*, vol. 8, 2014, pp. 197–205.

[59] M. A. Hernán and J. M. Robins, *Causal inference: What if*, 2020.

[60] N. Black, "Why we need observational studies to evaluate the effectiveness of health care," *Bmj*, vol. 312, no. 7040, pp. 1215–1218, 1996.

[61] J. S. Haukoos and R. J. Lewis, "The propensity score," *Jama*, vol. 314, no. 15, pp. 1637–1638, 2015.

[62] J. Pearl, *Causality: models, reasoning, and inference*, eng. Cambridge University Press, 2009, ISBN: 052189560X.

[63] W. K. Gray, A. V. Navaratnam, J. Day, J. Wendon, and T. W. Briggs, "Changes in covid-19 in-hospital mortality in hospitalised adults in england over the first seven months of the pandemic: An observational study using administrative data," *The Lancet Regional Health–Europe*, vol. 5, 2021.

[64] I. Goldenberg and G. I. Webb, "Survey of distance measures for quantifying concept drift and shift in numeric data," *Knowledge and Information Systems*, vol. 60, no. 2, pp. 591–615, 2019.

[65] C. Shui, Q. Chen, J. Wen, F. Zhou, C. Gagné, and B. Wang, "A novel domain adaptation theory with jensen–shannon divergence," *Knowledge-Based Systems*, vol. 257, p. 109 808, 2022.

[66] W. Qu, I. Balki, M. Mendez, J. Valen, J. Levman, and P. N. Tyrrell, "Assessing and mitigating the effects of class imbalance in machine learning with application to x-ray imaging," *International journal of computer assisted radiology and surgery*, vol. 15, pp. 2041–2048, 2020.

[67] R. Taori, A. Dave, V. Shankar, N. Carlini, B. Recht, and L. Schmidt, "Measuring robustness to natural distribution shifts in image classification," *Advances in Neural Information Processing Systems*, vol. 33, pp. 18 583–18 599, 2020.

[68] D. Ben-Zeev, B. Buck, S. Meller, W. J. Hudenko, and K. A. Hallgren, "Augmenting evidence-based care with a texting mobile interventionist: A pilot randomized controlled trial," *Psychiatric Services*, vol. 71, no. 12, pp. 1218–1224, 2020.

[69] K. Lybarger, M. Ostendorf, and M. Yetisgen, "Annotating social determinants of health using active learning, and characterizing determinants using neural event extraction," *Journal of Biomedical Informatics*, vol. 113, p. 103 631, 2021.

[70] K. Lybarger, M. Yetisgen, and Ö. Uzuner, "The 2022 n2c2/uw shared task on extracting social determinants of health," *Journal of the American Medical Informatics Association*, vol. 30, no. 8, pp. 1367–1378, 2023.

[71] B. Vidgen, T. Thrush, Z. Waseem, and D. Kiela, "Learning from the worst: Dynamically generated datasets to improve online hate detection," 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)*, Online: Association for Computational Linguistics, Aug. 2021, pp. 1667–1682. DOI: 10.18653/v1/2021.acl-long.132.

[72] O. de Gibert, N. Perez, A. García-Pablos, and M. Cuadros, "Hate speech dataset from a white supremacy forum," in *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 11–20. DOI: 10.18653/v1/W18-5102.

[73] D. D. Burns, *Feeling good*. Signet Book, 1981.

[74] J. S. Tauscher, K. Lybarger, X. Ding, *et al.*, "Automated detection of cognitive distortions in text exchanges between clinicians and people with serious mental illness," *Psychiatric services*, vol. 74, no. 4, pp. 407–410, 2023.

[75] X. Ding, K. Lybarger, J. Tauscher, and T. Cohen, "Improving classification of infrequent cognitive distortions: Domain-specific model vs. data augmentation," in *Proceedings of the 2022 conference of the North American chapter of the association for computational linguistics: human language technologies: Student Research Workshop*, 2022, pp. 68–75.

[76] K. Lybarger, J. Tauscher, X. Ding, D. Ben-Zeev, and T. Cohen, "Identifying distorted thinking in patient-therapist text message exchanges by leveraging dynamic multi-turn context," in *Proceedings of the Eighth Workshop on Computational Linguistics and Clinical Psychology (CLPsych)*, 2022, pp. 126–136.

[77] D. Ben-Zeev, B. Buck, A. Chander, *et al.*, "Mobile rdoc: Using smartphones to understand the relationship between auditory verbal hallucinations and need for care," *Schizophrenia Bulletin Open*, vol. 1, no. 1, sgaa060, 2020.

[78] P. M. Meddaugh and J. Kay, "Hate speech or "reasonable racism?" the other in stormfront," *Journal of Mass Media Ethics*, vol. 24, no. 4, pp. 251–268, 2009.

[79] J. Devlin, M.-W. Chang, K. Lee, and K. 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)*, Min-

neapolis, Minnesota: Association for Computational Linguistics, 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423.

[80] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Nov. 2019. [Online]. Available: https://arxiv.org/abs/1908.10084.

[81] J. Pearl, "Causal inference in the health sciences: A conceptual introduction," *Health services and outcomes research methodology*, vol. 2, pp. 189–220, 2001.

[82] R. Adib, P. Griffin, S. I. Ahamed, and M. Adibuzzaman, "A causally formulated hazard ratio estimation through backdoor adjustment on structural causal model," in *Machine Learning for Healthcare Conference*, PMLR, 2020, pp. 376–396.

[83] L. Keele, R. T. Stevenson, and F. Elwert, "The causal interpretation of estimated associations in regression models," *Political Science Research and Methods*, vol. 8, no. 1, pp. 1–13, 2020.

[84] X. Yang, H. Zhang, and J. Cai, "Deconfounded image captioning: A causal retrospect," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 11, pp. 12 996–13 010, 2021.

[85] C. Menni, K. Klaser, A. May, *et al.*, "Vaccine side-effects and sars-cov-2 infection after vaccination in users of the covid symptom study app in the uk: A prospective observational study," *The Lancet Infectious Diseases*, vol. 21, no. 7, pp. 939–949, 2021.

[86] B. Ozenne, F. Subtil, and D. Maucort-Boulch, "The precision–recall curve overcame the optimism of the receiver operating characteristic curve in rare diseases," *Journal of clinical epidemiology*, vol. 68, no. 8, pp. 855–859, 2015.

[87] C. Cortes, "Support-vector networks," *Machine Learning*, 1995.

[88] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[89] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[90] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[91] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[92] J. Wei and K. 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)*, Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 6382–6388. DOI: 10.18653/v1/D19-1670. [Online]. Available: https://aclanthology.org/D19-1670.

[93] R. Sennrich, B. Haddow, and A. Birch, "Improving neural machine translation models with monolingual data," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 86–96. DOI: 10.18653/v1/P16-1009. [Online]. Available: https://aclanthology.org/P16-1009.

[94] A. Anaby-Tavor, B. Carmeli, E. Goldbraich, *et al.*, "Do not have enough data? deep learning to the rescue!" In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 7383–7390.

[95] H. Zhang, M. Cissé, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, OpenReview.net, 2018. [Online]. Available: `https://openreview.net/forum?id=r1Ddp1-Rb`.

[96] V. Kumar, A. Choudhary, and E. Cho, "Data augmentation using pre-trained transformer models," *arXiv preprint arXiv:2003.02245*, 2020.

[97] X. Zhu, Y. Liu, J. Li, T. Wan, and Z. Qin, "Emotion classification with data augmentation using generative adversarial networks," in *Advances in Knowledge Discovery and Data Mining: 22nd Pacific-Asia Conference, PAKDD 2018, Melbourne, VIC, Australia, June 3-6, 2018, Proceedings, Part III 22*, Springer, 2018, pp. 349–360.

[98] R. Escobar Díaz Guerrero, L. Carvalho, T. Bocklitz, J. Popp, and J. L. Oliveira, "A data augmentation methodology to reduce the class imbalance in histopathology images," *Journal of Imaging Informatics in Medicine*, pp. 1–16, 2024.

[99] G. Ghiasi, Y. Cui, A. Srinivas, *et al.*, "Simple copy-paste is a strong data augmentation method for instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 2918–2928.

[100] Z. Liu, Y. Xu, Y. Xu, *et al.*, "An empirical study on distribution shift robustness from the perspective of pre-training and data augmentation," *arXiv preprint arXiv:2205.12753*, 2022.

[101] V. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 2013.

[102] A. Sugiyama and N. Yoshinaga, "Data augmentation using back-translation for context-aware neural machine translation," in *Proceedings of the fourth workshop on discourse in machine translation (DiscoMT 2019)*, 2019, pp. 35–44.

[103] J. Tiedemann and S. Thottingal, "Opus-mt–building open translation services for the world," in *Proceedings of the 22nd annual conference of the European Association for Machine Translation*, 2020, pp. 479–480.

[104] J. Tiedemann, M. Aulamo, D. Bakshandaeva, *et al.*, "Democratizing neural machine translation with opus-mt," *Language Resources and Evaluation*, vol. 58, no. 2, pp. 713–755, 2024.

[105] L. Sun, C. Xia, W. Yin, T. Liang, P. Yu, and L. He, "Mixup-transformer: Dynamic data augmentation for NLP tasks," in *Proceedings of the 28th International Conference on Computational Linguistics*, Barcelona, Spain (Online): International Committee on Computational Linguistics, Dec. 2020, pp. 3436–3440. DOI: 10.18653/v1/2020.coling-main.305. [Online]. Available: https://aclanthology.org/2020.coling-main.305.

[106] M. Zhang, G. Jiang, S. Liu, J. Chen, and M. Zhang, "Llm–assisted data augmentation for chinese dialogue–level dependency parsing," *Computational Linguistics*, pp. 1–24, 2024.

[107] S. V. Balkus and D. Yan, "Improving short text classification with augmented data using gpt-3," *Natural Language Engineering*, pp. 1–30, 2022.

[108] H. Dai, Z. Liu, W. Liao, *et al.*, "Auggpt: Leveraging chatgpt for text data augmentation," *arXiv preprint arXiv:2302.13007*, 2023.

[109] L. Hu, H. He, D. Wang, Z. Zhao, Y. Shao, and L. Nie, "Llm vs small model? large language model based text augmentation enhanced personality detection model," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, 2024, pp. 18 234–18 242.

[110] D. D. Burns, *Feeling Good: The New Mood Therapy*. New York, NY: Harper Collins, 1999.

[111] S. Ji, T. Zhang, L. Ansari, J. Fu, P. Tiwari, and E. Cambria, "Mentalbert: Publicly available pretrained language models for mental healthcare," *arXiv preprint arXiv:2110.15621*, 2021.

[112] Z. Xu, Y. Liu, G. Deng, Y. Li, and S. Picek, "A comprehensive study of jailbreak attack versus defense for large language models," in *Findings of the Association for Computational Linguistics ACL 2024*, 2024, pp. 7432–7449.

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

[114] G. Deng, Y. Liu, Y. Li, *et al.*, "Jailbreaker: Automated jailbreak across multiple large language model chatbots," *arXiv preprint arXiv:2307.08715*, 2023.

[115] A. Arditi, O. Obeso, A. Syed, *et al.*, "Refusal in language models is mediated by a single direction, 2024," *URL: https://arxiv. org/abs/2406.11717*,

[116] F. Wang, L. P. Casalino, and D. Khullar, "Deep learning in medicine—promise, progress, and challenges," *JAMA internal medicine*, vol. 179, no. 3, pp. 293–294, 2019.

[117] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 843–852.

[118] J. R. Zech, M. A. Badgeley, M. Liu, A. B. Costa, J. J. Titano, and E. K. Oermann, "Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study," *PLoS medicine*, vol. 15, no. 11, e1002683, 2018.

[119] R. Daneshjou, K. Vodrahalli, R. A. Novoa, *et al.*, "Disparities in dermatology ai performance on a diverse, curated clinical image set," *Science advances*, vol. 8, no. 31, 2022.

[120] D. Zhu, B. C. Riedel, N. Jahanshad, *et al.*, "Classification of major depressive disorder via multi-site weighted lasso model," in *Medical Image Computing and Computer Assisted Intervention- MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part III 20*, Springer, 2017, pp. 159–167.

[121] C. Fabbri, F. Corponi, D. Albani, *et al.*, "Pleiotropic genes in psychiatry: Calcium channels and the stress-related fkbp5 gene in antidepressant resistance," *Progress in Neuro-Psychopharmacology and Biological Psychiatry*, vol. 81, pp. 203–210, 2018.

[122] R. Iniesta, K. Malki, W. Maier, *et al.*, "Combining clinical variables to optimize prediction of antidepressant treatment outcomes," *Journal of psychiatric research*, vol. 78, pp. 94–102, 2016.

[123] G. Shen, J. Jia, L. Nie, *et al.*, "Depression detection via harvesting social media: A multimodal dictionary learning solution.," in *IJCAI*, 2017, pp. 3838–3844.

[124] N. Roysden and A. Wright, "Predicting health care utilization after behavioral health referral using natural language processing and machine learning," in *AMIA Annual Symposium Proceedings*, American Medical Informatics Association, vol. 2015, 2015, p. 2063.

[125] R. Tuwani and A. Beam, "Safe and reliable transport of prediction models to new healthcare settings without the need to collect new labeled data," *medRxiv*, 2023.

[126] E. Laparra, S. Bethard, and T. A. Miller, "Rethinking domain adaptation for machine learning over clinical language," *JAMIA open*, vol. 3, no. 2, pp. 146–150, 2020.

[127] R. J. Chen, J. J. Wang, D. F. Williamson, *et al.*, "Algorithmic fairness in artificial intelligence for medicine and healthcare," *Nature biomedical engineering*, vol. 7, no. 6, pp. 719–742, 2023.

[128] A. Subbaswamy, R. Adams, and S. Saria, "Evaluating model robustness and stability to dataset shift," in *International conference on artificial intelligence and statistics*, PMLR, 2021, pp. 2611–2619.

[129] N. Thams, M. Oberst, and D. Sontag, "Evaluating robustness to dataset shift via parametric robustness sets," *Advances in Neural Information Processing Systems*, vol. 35, pp. 16 877–16 889, 2022.

[130] J. Liu, Z. Shen, Y. He, *et al.*, "Towards out-of-distribution generalization: A survey," *arXiv preprint arXiv:2108.13624*, 2021.

[131] G. Ilharco, M. T. Ribeiro, M. Wortsman, *et al.*, "Editing models with task arithmetic," in *International Conference on Learning Representations (ICLR)*, 2023.

[132] E. J. Hu, yelong shen, P. Wallis, *et al.*, "LoRA: Low-rank adaptation of large language models," in *International Conference on Learning Representations (ICLR)*, 2022.

[133] H. W. Chung, L. Hou, S. Longpre, *et al.*, "Scaling instruction-finetuned language models," *Journal of Machine Learning Research*, vol. 25, no. 70, pp. 1–53, 2024.

[134] H. Touvron, L. Martin, K. Stone, *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[135] J. Lee, W. Yoon, S. Kim, *et al.*, "Biobert: A pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, 2020.

[136] L. Rasmy, Y. Xiang, Z. Xie, C. Tao, and D. Zhi, "Med-bert: Pretrained contextualized embeddings on large-scale structured electronic health records for disease prediction," *NPJ digital medicine*, vol. 4, no. 1, p. 86, 2021.

[137] A. Dosovitskiy, L. Beyer, A. Kolesnikov, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021.

[138] X. Han, Z. Zhang, N. Ding, *et al.*, "Pre-trained models: Past, present and future," *AI Open*, vol. 2, pp. 225–250, 2021.

[139] H. Wang, J. Li, H. Wu, E. Hovy, and Y. Sun, "Pre-trained language models and their applications," *Engineering*, 2022.

[140] X. Wang, G. Chen, G. Qian, *et al.*, "Large-scale multi-modal pre-trained models: A comprehensive survey," *Machine Intelligence Research*, vol. 20, no. 4, pp. 447–482, 2023.

[141] M. Pham, K. O. Marshall, C. Hegde, and N. Cohen, "Robust concept erasure using task vectors," *arXiv preprint arXiv:2404.03631*, 2024.

[142] J. Wang, C. Lan, C. Liu, *et al.*, "Generalizing to unseen domains: A survey on domain generalization," *IEEE transactions on knowledge and data engineering*, vol. 35, no. 8, pp. 8052–8072, 2022.

[143] J. C. Duchi and H. Namkoong, "Learning models with uniform performance via distributionally robust optimization," *The Annals of Statistics*, vol. 49, no. 3, pp. 1378–1406, 2021.

[144] P. Mohajerin Esfahani and D. Kuhn, "Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations," *Mathematical Programming*, vol. 171, no. 1-2, pp. 115–166, 2018.

[145] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, "Analysis of representations for domain adaptation," *Advances in neural information processing systems*, vol. 19, 2006.

[146] K. Muandet, D. Balduzzi, and B. Schölkopf, "Domain generalization via invariant feature representation," in *International conference on machine learning*, PMLR, 2013, pp. 10–18.

[147] Y. Zhong and G. Ettinger, "Enlightening deep neural networks with knowledge of confounding factors," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 1077–1086.

[148] J. Shen, Y. Qu, W. Zhang, and Y. Yu, "Wasserstein distance guided representation learning for domain adaptation," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, 2018.

[149] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14*, Springer, 2016, pp. 443–450.

[150] Z. Lipton, Y.-X. Wang, and A. Smola, "Detecting and correcting for label shift with black box predictors," in *International conference on machine learning*, PMLR, 2018, pp. 3122–3130.

[151] H. Zhao, R. T. Des Combes, K. Zhang, and G. Gordon, "On learning invariant representations for domain adaptation," in *International conference on machine learning*, PMLR, 2019, pp. 7523–7532.

[152] L. K. McKnight, A. Wilcox, and G. Hripcsak, "The effect of sample size and disease prevalence on supervised machine learning of narrative data.," in *Proceedings of the AMIA Symposium*, American Medical Informatics Association, 2002, p. 519.

[153] G. M. Weiss and F. Provost, "The effect of class distribution on classifier learning: An empirical study," 2001.

[154] Y. Liu, M. Ott, N. Goyal, *et al.*, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[155] H. Touvron, L. Martin, K. Stone, *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[156] S. Sagawa, P. W. Koh, T. B. Hashimoto, and P. Liang, "Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization," *arXiv preprint arXiv:1911.08731*, 2019.

[157] A. Ben-Tal, D. Den Hertog, A. De Waegenaere, B. Melenberg, and G. Rennen, "Robust solutions of optimization problems affected by uncertain probabilities," *Management Science*, vol. 59, no. 2, pp. 341–357, 2013.

[158] J. C. Duchi, P. W. Glynn, and H. Namkoong, "Statistics of robust optimization: A generalized empirical likelihood approach," *Mathematics of Operations Research*, vol. 46, no. 3, pp. 946–969, 2021.

[159] B. Jacob, S. Kligys, B. Chen, *et al.*, "Quantization and training of neural networks for efficient integer-arithmetic-only inference," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 2704–2713.

[160] T. Dettmers, M. Lewis, S. Shleifer, and L. Zettlemoyer, "8-bit optimizers via block-wise quantization," in *International Conference on Learning Representations (ICLR)*, 2021.

[161] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017.

[162] M. Hardt, E. Price, and N. Srebro, "Equality of opportunity in supervised learning," *Advances in neural information processing systems*, vol. 29, 2016.

[163] Y. Romano, S. Bates, and E. Candes, "Achieving equalized odds by resampling sensitive attributes," *Advances in neural information processing systems*, vol. 33, pp. 361–371, 2020.

[164] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel, "Fairness through awareness," in *Proceedings of the 3rd innovations in theoretical computer science conference*, 2012, pp. 214–226.

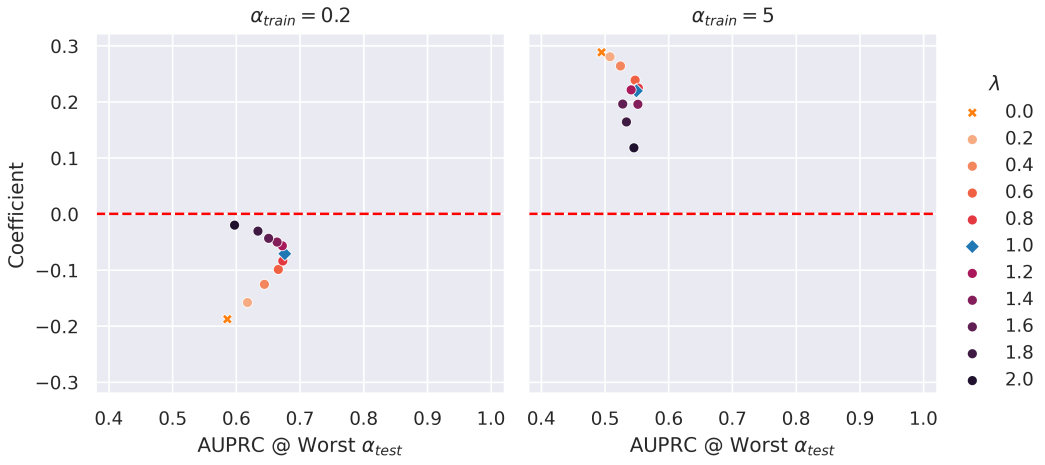
[165] C. Yuan, K. A. Linn, and R. A. Hubbard, "Algorithmic fairness of machine learning models for alzheimer disease progression," *JAMA Network Open*, vol. 6, no. 11, e2342203–e2342203, 2023.

Appendix A

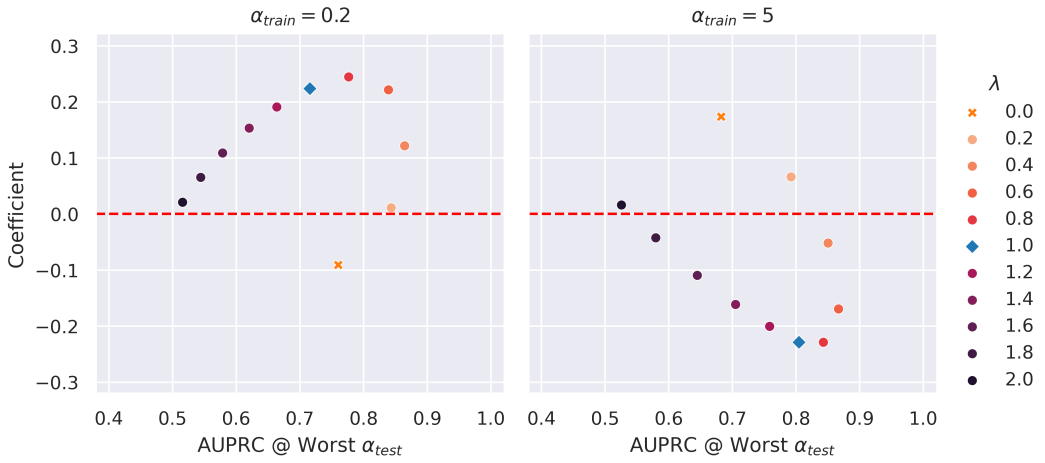
SUPPLEMENTAL MATERIALS FOR CHAPTER 7

A.1 Scaling Factors for DAPPER (trained on the imbalanced set)

Additional results checking the scaling factors for DAPPER on Hate Speech set (Figure A.1) and SHAC set (Figure A.2) are shown in this section. The results only concern the training set with $\alpha_{train} \neq 1$, when there is a dominant provenance class with a higher positive rate for the primary target.

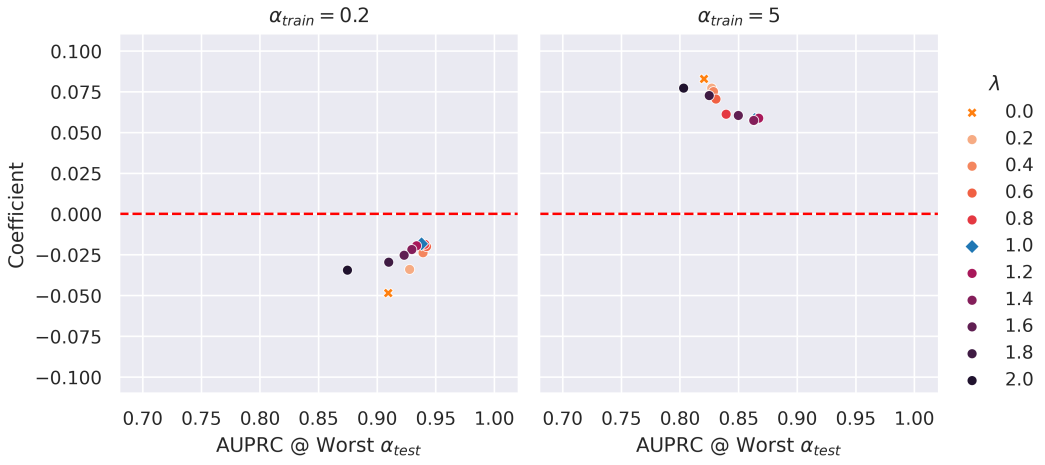


(a) RoBERTa

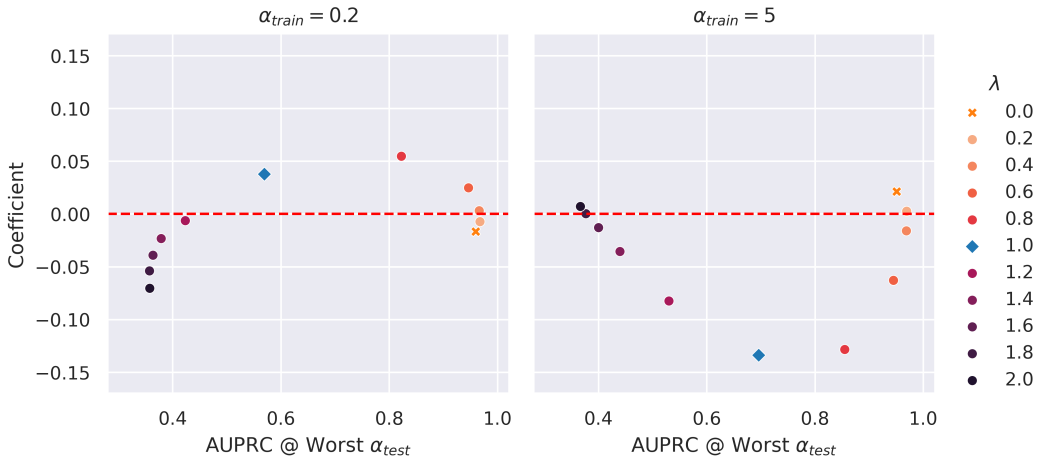


(b) Llama-2

Figure A.1: Scaling Factors for DAPPER on Hate Speech set. Positive rate for the training sets was set to 0.5. x-axis is the $AUPRC_{worst}$, and y-axis is coefficients (measure for robustness). Red horizontal line is the ideal anchor for robustness where the coefficient equals to 0. (a) are results on the RoBERTa model. (b) on the Llama-2 model. Slopes of the fitted line (measured by coefficients) are reported for all settings.



(a) RoBERTa

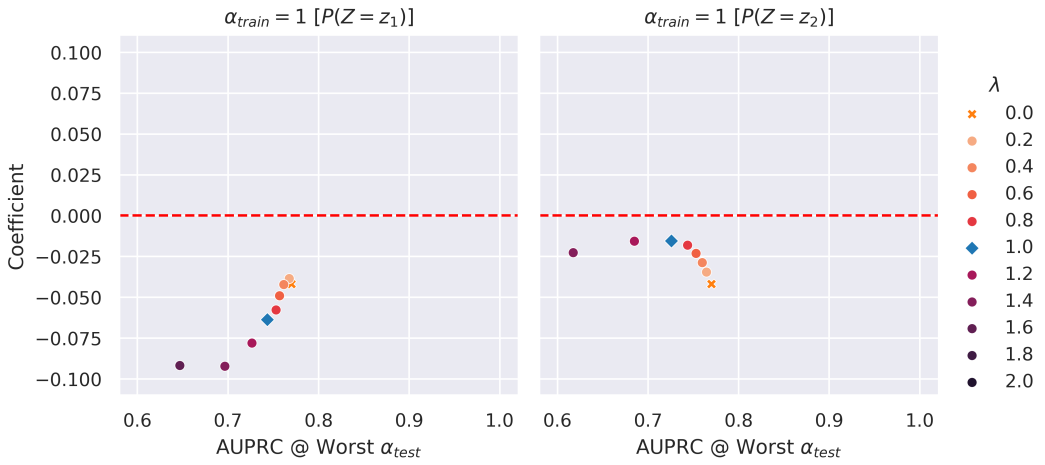


(b) Llama-2

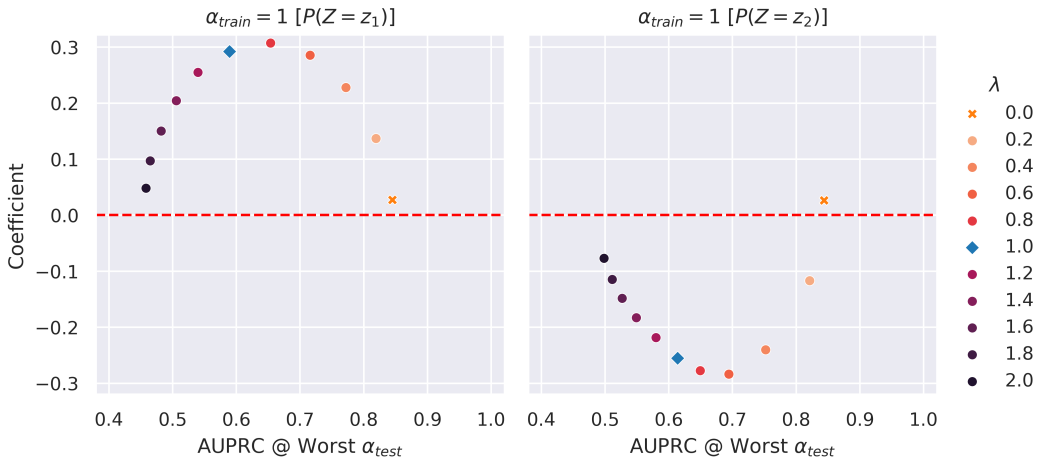
Figure A.2: Scaling Factors for DAPPER on SHAC set. Positive rate for the training sets was set to 0.5. x-axis is the $AUPRC_{worst}$, and y-axis is coefficients (measure for robustness). Red horizontal line is the ideal anchor for robustness where the coefficient equals to 0. (a) are results on the RoBERTa model. (b) on the Llama-2 model. Slopes of the fitted line (measured by coefficients) are reported for all settings.

A.2 Scaling Factors for DAPPER (trained on the balanced set)

Additional results checking the scaling factors for DAPPER on Hate Speech set (Figure A.1) and SHAC set (Figure A.2) are shown in this section. The results are on the training set with $\alpha_{train} = 1$, when there is no dominant provenance class. Each of two provenance classes ($Z = \{z_1, z_2\}$) was used as the "dominant" class, respectively, to check their effects in the arbitrarily chosen direction.

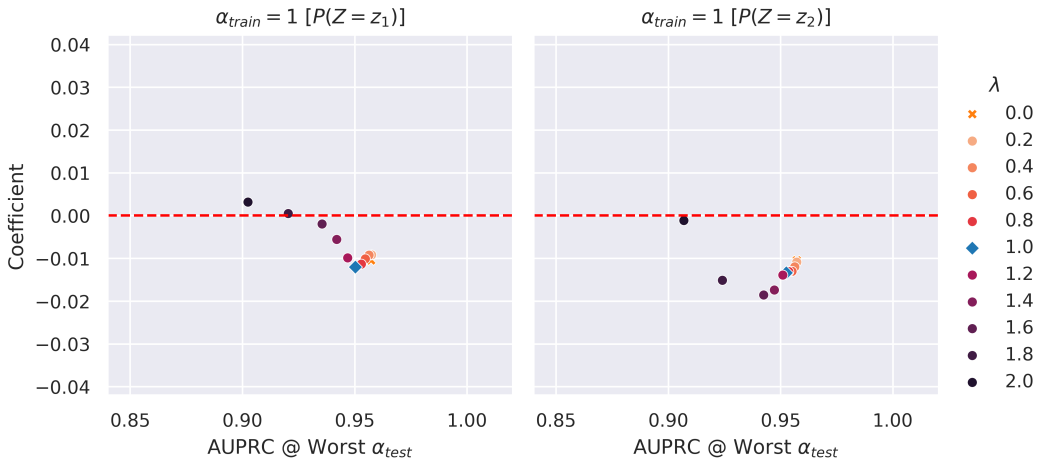


(a) RoBERTa

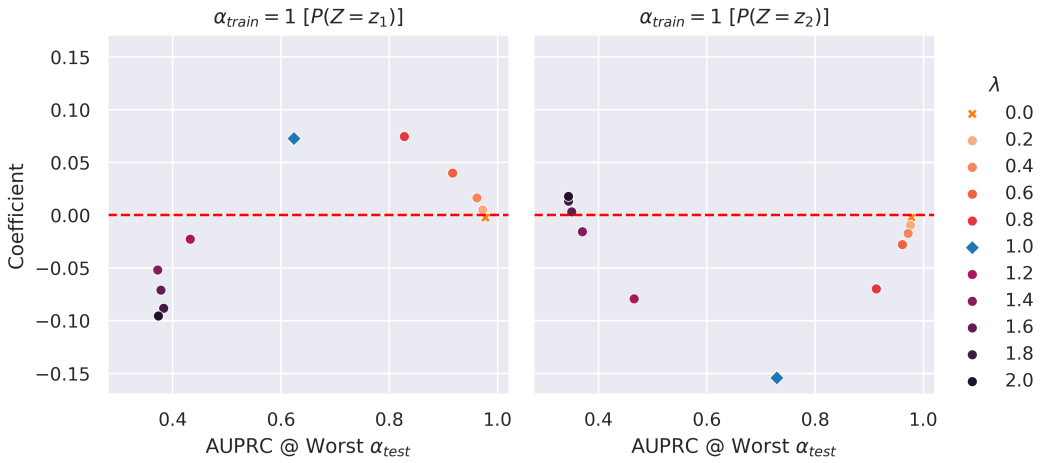


(b) Llama-2

Figure A.3: Scaling Factors for DAPPER applied on Hate Speech set with $\alpha_{train} = 1$. Positive rate for the training sets was set to 0.5. x-axis is the $AUPRC_{worst}$, and y-axis is coefficients (measure for robustness). Red horizontal line is the ideal anchor for robustness where the coefficient equals to 0. (a) are results on the RoBERTa model. (b) on the Llama-2 model. Slopes of the fitted line (measured by coefficients) are reported for all settings.



(a) RoBERTa

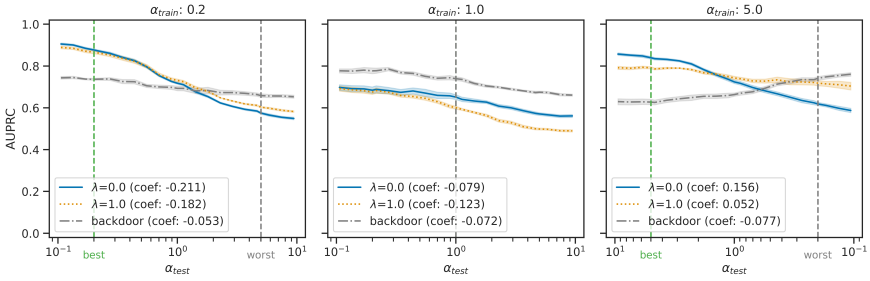


(b) Llama-2

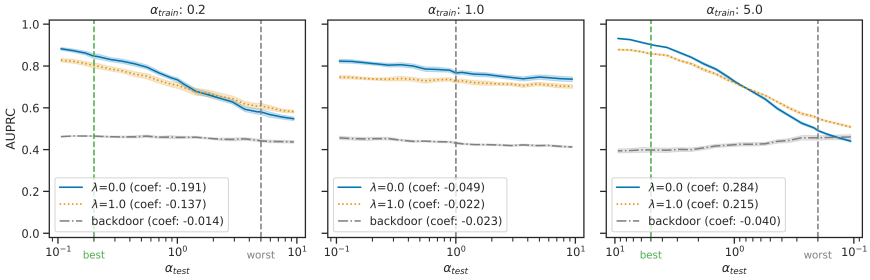
Figure A.4: Scaling Factors for DAPPER applied on SHAC set with $\alpha_{train} = 1$. Positive rate for the training sets was set to 0.5. x-axis is the $AUPRC_{worst}$, and y-axis is coefficients (measure for robustness). Red horizontal line is the ideal anchor for robustness where the coefficient equals to 0. (a) are results on the RoBERTa model. (b) on the Llama-2 model. Slopes of the fitted line (measured by coefficients) are reported for all settings.

A.3 TAPER

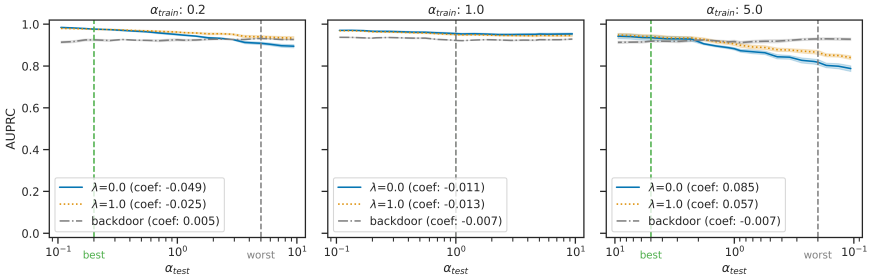
Results on three datasets are shown in Figure A.5 and A.6. The results shown limit the overall positive rate in the test set to 0.5, and equal mixtures, $P_{test}(Z = z_1) = P_{test}(Z = z_2) = 0.5$, for removing potential confounders in the analysis. "1.0-0.0" refers to the baseline fine-tuning (primary task only), where $\lambda = 0$; "1.0-1.0", represents the provenance effect reduction with a "default" setting of $\lambda = 1$ (assigning equal weight to the desired and undesired behaviors), in correspondence with Formula (7.9). "backdoor" in gray dash-dotted lines refers to the Backdoor Adjustment method, as baselines. The figures can be interpreted as follows. In each figure, the solid curve represents the model without task arithmetic, and the dotted curve represents the model when task arithmetic is used to mitigate confounding effects. The leftmost panels represent splits generated using an α_{train} of 0.2, indicating that positive examples from site $z = 1$ predominate in the training set. The middle panels represent results from models trained on data with an α_{train} of 1.0, which means positive examples from each site are equally represented. This eliminates the possibility of confounding by provenance, because the model cannot learn to associate increased prevalence of the outcome of interest with either of the sites. The rightmost panels represent results from splits with an α_{train} of 5 (the reciprocal of 0.2), indicating that positive examples from site $z = 2$ predominate at training time. In all panels the y axis shows the AUPRC as a measure of performance, and the x axis indicates the α_{test} value. As α_{train} is fixed in each panel, the degree of confounding bias increases as the distance along this axis from the dashed vertical line increases, with the dotted vertical line indicating the "worst case" confounding bias, where $\alpha_{train} = \frac{1}{\alpha_{test}}$.



(a) CD

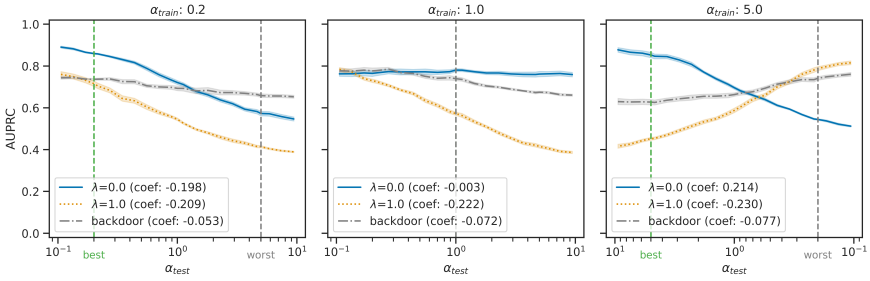


(b) HateSpeech

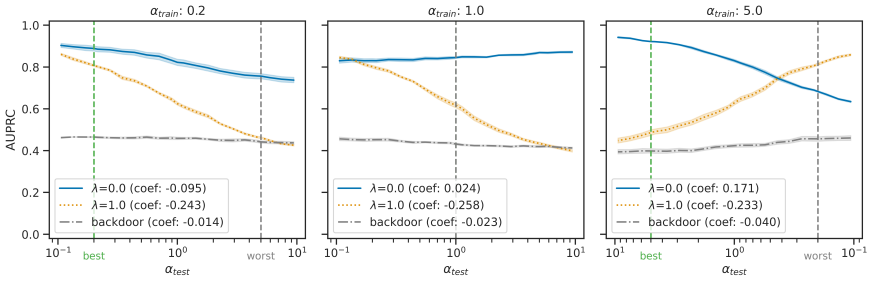


(c) SHAC

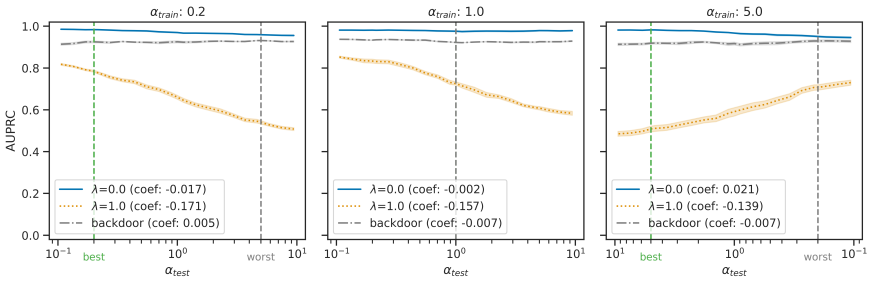
Figure A.5: Provenance Effect Reduction on all three datasets, using RoBERTa. Models were developed with different training set compositions, indicated by α_{train} in the column headers. (a) are results on Cognitive Distortion dataset; (b) on Hate Speech dataset; (d) on SHAC dataset. Slopes of the fitted line (measured by coefficients) are reported for all settings. "$\lambda = 0.0$" refers to the baseline fine-tuning; "$\lambda = 1.0$", the provenance effect reduction; "backdoor" the backdoor adjustment model.



(a) CD



(b) HateSpeech



(c) SHAC

Figure A.6: Provenance Effect Reduction on all three datasets, using Llama. Models were developed with different training set compositions, indicated by α_{train} in the column headers. (a) are results on Cognitive Distortion dataset; (b) on Hate Speech dataset; (d) on SHAC dataset. Slopes of the fitted line (measured by coefficients) are reported for all settings. "$\lambda = 0.0$" refers to the baseline fine-tuning; "$\lambda = 1.0$", the provenance effect reduction; "backdoor" the backdoor adjustment model.

In each panel, the *slope* of the line indicates robustness to confounding by provenance. A model that is immune to this form of bias would present a flat horizontal line, because performance (AUPRC) does not change as confounding bias (distance between α_{test} and α_{train}) increases.

Numerically, robustness is measured by the absolute value of the slope (or coefficient). The smaller the absolute coefficient is, the more robust a model is to provenance-related confounding effects. From the results, we can observe that the absolute coefficients from RoBERTa models trained on an imbalanced training set ($\alpha_{train} = 0.2$ and 5) are smaller after provenance effect reduction procedure for all three datasets, except for SHAC at $\alpha_{train} = 0.2$, where the baseline fine-tuned model already has a very flat slope ($|\beta|$=0.039 in Figure A.5c). For example, for the Cognitive Distortion dataset at $\alpha_{train} = 5$, RoBERTa models after provenance effect reduction shows a lower absolute coefficient of 0.06, in comparison with 0.147 for the model without the provenance effect reduction. This improvement in coefficients is relatively small for the training set at $\alpha_{train} = 0.2$, with 0.175 vs 0.206 (Figure A.5a). While checking the RoBERTa model trained on a balanced set ($\alpha_{train} = 1$, effectively eliminating the possibility of confounding by provenance), TAPER typically makes the model less robust, which can be observed for all three datasets, and is reflected in the relatively strong robustness of the baseline models (0.042 for Cognitive Distortion and 0.011 for Hate Speech).In comparison, the robustness of the Llama-2 model does not improve with TAPER. When the training set is biased, the baseline LoRA-tuned Llama-2 7b already shows good robustness in some cases, for example in Hate Speech Detection with $|\beta| = 0.095$ at $\alpha_{train} = 0.2$ (Figure A.6b) and in both cases for SHAC.

As one would anticipate, when confounding effects are prohibited by setting $\alpha_{train} = 1$, patterns are similar to those observed with the RoBERTa model. However, with Llama, the decrease in robustness observed when applying TAPER to the Cognitive Distortion set is (from 0.003 to 0.222) than with RoBERTa (from 0.042 to 0.102), as shown in Figure A.5 and A.6.

Besides robustness, it is also interesting to observe different directions of the trend on

model performance under opposite directions of the shift ($\alpha_{train} = 0.2$, indicating more positive examples from site $z = 1$ at training time vs $\alpha_{train} = 5$, indicating more positive examples from site $z = 2$ at training time). This can also be inferred from the signs of the coefficients. For $\alpha_{train} = 0.2$, both models across all three datasets show that the baseline fine-tuned model and TAPER model are always oriented toward the same direction, i.e., their coefficients always have the same sign (in our study setting, they are all negatives). This means that as the proportion of positive examples from the second site ($z = 2$) increases, performance deteriorates. This indicates that performance drops as more examples from the site ($z = 1$) that underrepresented at training time are included at test time.

When positive examples from the second site ($Z = z_2$) predominate at training time ($\alpha_{train} = 5$), the coefficients for the baseline fine-tuned model and model after TAPER generally have opposite signs, with the only exception being with RoBERTa on Hate Speech Detection set at $\alpha_{train} = 5$ (Figure A.5b), regardless of model architecture. This indicates that baseline model performance improves as more positive examples from the site that was over-represented at training time are included in the test set, while TAPER performance deteriorates.

As described in Section 7.2.4, performance at two points under the simulation framework are of particular interest, and are highlighted in graphs as green and gray vertical dashed lines (as in Figure A.5 and A.6). $AUPRC_{best}$ at $\alpha_{test} = \alpha_{train}$ (shown as the green dashed vertical lines on the left side within figures) and indicating the "best case" performance when provenance-related associations learned during training may help performance at test time; and $AUPRC_{worst}$ at $\alpha_{test} = 1/\alpha_{train}$ (shown as gray dashed vertical lines on the right side within figures), representing an extreme point of provenance-related confounding where the relative contribution of positive examples from each site at training time is inverted at test time.

With $\alpha_{train} = 5$ (rightmost panel),the coefficients from TAPER and baseline models have different signs, presenting an "X" shape in resulting simulation graphs. At the extreme of distribution shift $AUPRC_{worst}$ from provenance effect reduction is always better than that of

the baseline model. For $AUPRC_{best}$, baseline models perform better, as one would anticipate when provenance-related biases in prediction are consistent with the provenance-specific class distribution at test time.

In comparison with the baseline Backdoor Adjustment, TAPER usually generates models being less robust for both RoBERTa and Llama-2 7b model. There are cases where TAPER can result in better or similar robustness as Backdoor Adjustment, for example with absolute coefficients of 0.060 ($\lambda = 1$) vs 0.077 (backdoor) for Cognitive Distortion set and 0.038 ($\lambda = 1$) vs 0.040 (backdoor) for Hate Speech set when $\alpha_{train} = 5.0$ using RoBERTa (Figure A.5a). However, it is notable that the overall performance of Backdoor Adjustment, in many cases, fall far below than RoBERTa models, with or without TAPER (Figure A.5a), and in middle of Llama-2 7b with and without TAPER (Figure A.6a). For the "worst-case" scenarios, the point estimates from Backdoor Adjustment underperform those from TAPER for RoBERTa model except when $\alpha_{train} = 0.2$ for Cognitive Distortion and SHAC set.

In summary, the findings indicate that TAPER can effectively improve RoBERTa's robustness in most cases when there is an extreme imbalance in provenance-specific positive rates in the training set, i.e., when $\alpha_{train} \neq 1$. However, this advantage does not appear to extend to Llama-2 7b. For the training set with more positive examples from the second site ($z = 2$) ($\alpha_{train} = 5$), under the worst case scenario ($\alpha_{test} = 1/\alpha_{train} = 0.2$), both models' performance is boosted when the positive rate among provenance group z_2 is higher than that in the other group. This indicates that TAPER is only effective when the distribution shift in one direction. Since $\alpha_{train} = 0.2$ and $\alpha_{train} = 5$ indicate the same degree of positive rate bias and are arbitrarily chosen, this observation on different behaviors under different α_{train} provides motivation for the Dominance-Aligned Polarity Provenance Effect Reduction, results from which are shown in Section 7.6.2.