THE GEORGE WASHINGTON UNIVERSITY

UNDERGRADUATE THESIS

Heuristic-Based Weak Learning for Moral Decision-Making

Author:
Ryan Steed

Comittee:
Dr. Benjamin Williams (Chair)
Dr. Rahul Simha
Dr. Brian Wright

A thesis submitted in fulfillment of the requirements for the degree of Bachelor of Science in Computational Economics

at the

Columbian College of Arts & Sciences

April 17, 2020

Abstract

As automation proliferates and algorithms become increasingly responsible for high-stakes decision-making, AI agents face moral dilemmas in fields ranging from market design to robots. For instance, should a self-driving car swerve into a barrier, endangering its passengers, to avoid colliding with a jaywalker? Technology companies, governments, and all AI practitioners must build and maintain autonomous systems that make responsible moral decisions.

Prior approaches to automated moral decision-making utilize either rules-based game theoretic models or machine learning models trained on crowd-sourced data. But rules-based systems are difficult to adapt to new moral dilemmas and data, and sourcing high quality, representative, hand-labeled data for machine learning is costly and even harmful if the labels are biased. To lower the barrier to training moral agents, I develop a heuristic-based weak learning approach to moral decision-making.

My approach synthesizes potentially conflicting legal, philosophical, and domain-specific heuristics to inexpensively and automatically label training data for moral dilemmas. Rather than attempting to survey a representative sample of users who may be unable to make informed decisions about complex dilemmas, this approach relies on a smaller sample of domain experts. By writing heuristic functions over the dataset, these experts efficiently specify ethical principles for technical dilemmas. Weak learning paves the way to a ubiquitous, transparent method for instilling moral decision-making in the machine learning pipeline.

As a proof-of-concept, I test this approach in two case studies for which there is publicly available data on people's moral preferences: 1) the Moral Machine trolley problem, in which an autonomous vehicle must choose to save only one group of characters; 2) a kidney exchange, in which a market clearing algorithm must choose between two potential matches for a donor kidney. I show that in these domains, heuristic-based weak learning is quicker and easier than fully supervised learning and achieves comparable performance, especially for specialized domains. I also identify patterns of disagreement between heuristics and individual respondents.

Contents

1 Introduction 3

2 Problem Statement 4

3 Related Work 6

4 Approach 8

4.1 Pipeline 9

4.2 Heuristic Functions 12

4.3 Generative Model 13

5 Experiments 15

5.1 Autonomous Vehicle Trolley Problem 16

5.1.1 Data 17

5.1.2 Heuristics 18

5.1.3 Label Model 19

5.1.4 Discriminative Model 22

5.2 Kidney Exchange 26

5.2.1 Data 26

5.2.2 Heuristics & Label Model 27

5.2.3 Discriminative Model 29

5.2.4 Ranked-Choice Heuristics 29

6 Discussion 32

7 Conclusion 35

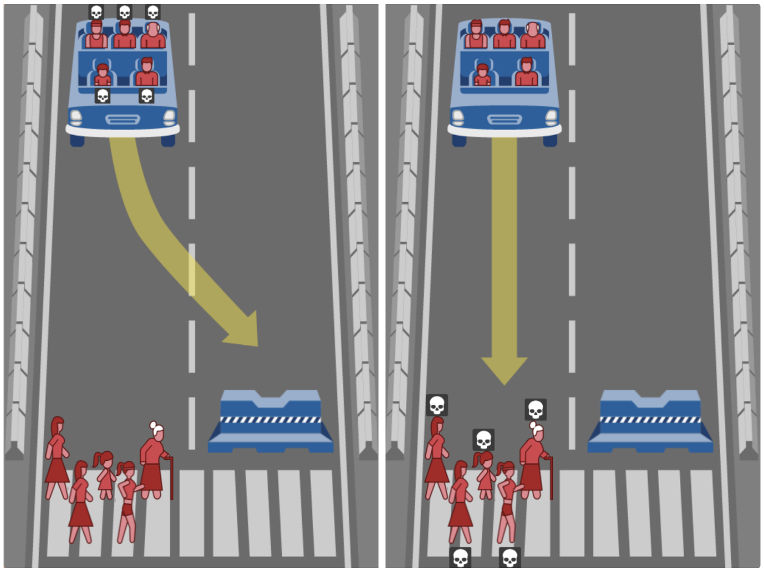

Figure 1: What should the self-driving car do? An example moral dilemma from the Moral Machine interface [Mor]. Staying on course would result in the death of two women, a female athlete, a young girl, and an elderly woman. Swerving would result in the death of two men, a male athlete, a young boy, and an elderly man. This scenario was designed to elicit moral preferences about gender.

1 Introduction

As the widespread application of artificial intelligence (AI) systems grows, so does the discovery of serious fairness and bias issues in AI applications from face recognition to the hiring process. Algorithms are increasingly faced with morally ambiguous decisions: for instance, should a self-driving car should swerve into a barrier, killing its passengers, to avoid killing a jaywalker (Figure 1)? In the kidney exchange market, should a clearing-house algorithm allocate kidneys to patients who drink less, all else equal? AI is already used to make life-and-death decisions in the kidney exchange, and autonomous vehicles are already being tested in cities. Technology companies, governments, and all AI practitioners are currently faced with the problem of easily building and maintaining algorithms that make moral decisions, given the wide variety of applications and moral considerations that exist in the wild.

Prior approaches to moral AI leverage crowdsourcing to develop ethical models that mimic popular moral preferences. For example, the Moral Machine project collected human judgments of autonomous vehicle (AV) behavior in trolley dilemmas, including the jaywalking example [Awad et al., 2018]. Noothigattu et al. [2018] developed a computational model for profiling the moral preferences of these respondents and constructing a voting system for making collective moral decisions at runtime. Survey results are good for measuring moral preferences in specific cases, but may not provide useful ethical guidance in complicated domains and often run into selection bias.

To improve the process of eliciting and operationalizing moral principles, I develop a framework for quickly generating training data for moral dilemmas based on a set of adjustable moral heuristics, as determined by *a priori* ethical or legal principles. Rather than attempting to survey a representative sample of users who are potentially unable to make informed decisions about the domain in question, my approach seeks to collect decision-making heuristics from a smaller sample of domain experts. With empirical data for two use cases (the autonomous vehicle trolley problem and the kidney exchange) I show that constructing heuristics is cheaper, quicker, and easier than collecting votes or rankings over individual sets of alternatives but provides comparable performance. I also show that if experts tend to share heuristics, a ranked-choice vote can be used to weight heuristics by their popularity; in the kidney exchange, this approach outperforms a fully-supervised approach.

Section 2 details major problems for automated ethical decision-making, and Section 3 describes various attempts to solve them. Section 4 presents my approach, which I test with two different case studies: the autonomous vehicle trolley dilemma and the kidney exchange dilemma (Section 5). The remaining sections conclude and point to important future work.

2 Problem Statement

The field of algorithmic ethics has a long history and a broad set of problems [Moor, 1985]. This paper deals in particular with moral dilemmas, scenarios in which an agent faced with multiple alternatives is morally compelled to choose each, but can only choose one [Sinnott-Armstrong, 1988, Yu et al., 2018]. For instance, an autonomous vehicle in the Moral Machine problem is morally compelled to save both the passengers and the pedestrians, but due to brake failure or some other physical constraint must choose which group to save. Formally, let $\mathcal{A}$ be a finite set of possible alternatives, where each alternative is represented by a vector of relevant moral or situational features. An instance of a moral dilemma consists of a set of alternatives $X \in \mathcal{A}$ of size K such that a moral agent is compelled to choose (not choose) each alternative but must choose only one. An answer to the dilemma takes the form of an ordinal ranking y over X which ranks the moral appeal of each alternative.

In addition to the trolley dilemma for autonomous vehicles, there are many scenarios in which an automated agent or algorithm may face a moral dilemma, especially when interacting with human beings: when two matches for a donor kidney are found, which patient should receive the kidney? Is a repentant convict more deserving of parole than an unrepentant one? Should an advertising algorithm consider the moral impacts of particular advertisements on particular users? Moral dilemmas are usually reserved for human judgment, but as algorithms are endowed with more and more responsibility in human affairs, they will inevitably face moral dilemmas that are either too immediate, too minute, or too complicated to be left to human judgment. As a result, some researchers have called for a general framework for automated moral decision-making [Conitzer et al., 2017].

There are several important problems to solve when automating ethical decision-making. (1) To specify a moral dilemma, researchers must choose a representation of all relevant considerations, or features, for each alternative. In the Moral Machine experiment, features include inter alia age, gender, and class status. Features must be specific enough to convey meaningful information about the moral factors at play, but general enough to be decipherable by a domain expert or a layperson for coding or training the model. Oversimplification of the problem into only a few moral features, as is often necessary in experiments intended to measure moral preferences, may miss crucial factors that should influence decision-making. (2) The moral agent must learn or operate according to semantic knowledge of moral principles or preferences. (3) In the absense of clear moral consensus, the agent must be pluralistic; that is, it must reconcile multiple moral principles or preferences. (4) The moral agent's decisions should be interpretable. If the agent has moral responsibility, there must be a way to account for its decisions to affected users, for ethical and regulatory reasons. For this reason, the use of black box models for moral decision-making is somewhat unappealing unless the model is made sufficiently interpretable [Du et al., 2018]. Section 3 reviews some contemporary solutions to these problems and their shortcomings.

3 Related Work

The problem of moral AI has been approached in two primary ways. In the "top-down" approach, moral principles are encoded in the algorithmic agent, generally with a short or extensive form game [Dehghani et al., 2008a, Anderson and Anderson, 2014, Blass and Forbus, 2015]. In the "bottom-up" approach, the agent learns to distinguish moral and immoral behavior from user data [Kim et al., 2018, Noothigattu et al., 2018, Kahng et al., 2019, Freedman et al., 2020]. In their call for a general framework for algorithmic ethical decision-making, Conitzer et al. [2017] suggest that by abstracting moral principles from individual moral preferences, whether through a manually encoded decision-making framework or learned preferences, AI models may result in a more consistent system than that of any individual. But to achieve consistency, moral algorithms must generalize or aggregate many, sometimes conflicting, moral principles.

There is disagreement about what kinds of moral principles should be encoded in agents [Shulman et al., 2009], and moral principles tend to vary from culture to culture [Dehghani et al., 2008b]. In ethics, moral principles usually include three dimensions: consequentialist ethics, in which an agent weighs utilitarian consequences across all alternatives; deontological ethics, in which an agent acts in accordance with established norms or duties; and virtue ethics, in which an agent attempts to embody intrinsic moral values such as fairness [Cointe et al., 2016]. The "top-down" solution to this problem is to construct a self-consistent moral framework under one of these views, or to allow one particular moral principle to supersede the rest. Some combination of these elements of ethical behavior may be combined to form an extensive form game [Conitzer et al., 2017, Cointe et al., 2016]. Even if one consistent approach is selected, or ethical principles are arranged such that conflicts are resolved by referencing a hierarchy, the same ethical principle may give rise to conflicting alternatives, creating a symmetrical dilemma as in Sophie's Choice [Sinnott-Armstrong, 1988, Greenspan, 1983]. A few researchers now advocate for models with built-in moral uncertainty, in which multiple moral decision-making frameworks operate simultaneously, either disjunctively or probabilistically Bogosian [2017], Martinho et al. [2020]. In the “bottom-up” (crowd-sourcing) paradigm, uncertainty is baked in: observed or simulated dilemmas are presented to human annotators, who choose the morally correct alternative according to their own sense of ethics. Usually, social computational choice models [Noothigattu et al., 2018], or another form of general preference measurement [Kim et al., 2018, Freedman et al., 2020], are applied to combine these responses according to some egalitarian voting rule. Machine learning models generalize beyond the training set with accuracy, while a top-down approach might struggle to adapt to new variations on a dilemma. The “bottom-up approach” takes the view that in the absence of a consensus philosophical theory, using psychological studies to measure “folk” intuitions may be the best way to ethically constrain algorithms [Bello and Bringsjord, 2013].

In fact, both the “top-down” and “bottom-up” paradigms rely on empirical data in practice, either to measure the prevailing opinion amongst experts or the collective preferences of regular users. Moral dilemmas are often used by psychologists to measure subjects’ preferences with respect to some through pairwise comparisons [Awad et al., 2020, Bonnefon et al., 2016]. In “bottom-up” frameworks, these moral preferences can be easily converted to a decision-making rule through a simple ranking system or a more complex hierarchical model [Freedman et al., 2020, Kim et al., 2018]. In “top-down” frameworks, abstract moral principles can be combined to create an *a priori* “meta-ethical” framework for decision-making. The mix of principles may be determined by either the preponderance of philosophers or domain experts sharing a particular moral precept or based on the special applicability of a particular moral precept to a given task [Macaskill, 2016, Bogosian, 2017]. For top-down approaches which attempt to create meta-ethical frameworks, empirical measurement of expert opinion becomes an important issue. Though empirical strategies like these take promising steps toward artificially intelligent moral decision-making, they require a democratized approach to ethics that comes with distinct limitations.

But in the absence of high quality, crowd-sourced data on moral preferences, practitioners often rely on Amazon Mechanical Turk or another less desirable survey method [Freedman et al., 2020]. Social computational choice approaches attempt to aggregate these individual votes into a general model. In many cases, surveyed voters are not likely to representative data: approximately 70% of Moral Machine respondents were male college graduates, and most were from Western countries [Awad et al., 2018]. On Mechanical Turk, there are selection biases towards females, lower-income individuals, though these biases may be less aggravated than in traditional university studies [Paolacci et al., 2010]. Further, since moral preferences tend to vary across cultures, cross-country data are sometimes necessary [Awad et al., 2018]. Worse, the representations used for these empirical studies are necessarily simple; a highly complex or contingent representation of the moral dilemma may be difficult for human subjects to parse, and simplifying the problem runs the risk of eliminating important interaction effects between features. Poor survey design might lead respondents to emphasize moral features they would not consider in a real-world scenario.

4 Approach

To address current shortcomings in moral AI, I turn to *weakly supervised* machine learning. Section 3 details the difficult issues that face machine learning approaches to ethical frameworks: namely, obtaining enough high-quality ground truth data from sufficiently qualified individuals. Rather than attempt to measure moral preferences directly, I suggest collecting moral principles directly from domain experts in the form of heuristic functions over a set of example dilemmas. Heuristic functions are practical, usually simplistic, rules for determining the right moral decision for a given dilemma. Heuristics may come from the experience of a domain expert, or they may be sourced from legal or ethical principles by an expert in law or philosophy. For example, the German Ethics Comission on Automated and Connected Driving published a set of ethical rules that places the protection of human life above the protection of other animal life [Luetge, 2017]. This law presents a clear heuristic for guiding ethical action in autonomous vehicles: "always choose to protect human life over animal life."[1] Fairness metrics in equitable machine learning also tend to formalize and optimize for simple rules about

[1] It should be noted that the ethical commission also recommended banning distinctions on the basis of personal features such as age Luetge [2017]. The selection of eligible features to represent a moral dilemma is an important problem that is not solved in this paper.

fairness, such as statistical parity [Dwork et al., 2011]. The advantage of this approach is that it does not require experts to hand-label thousands of data points; instead, experts need only write a sufficient number of heuristic labeling functions to represent their moral knowledge about the domain in question.

My central hypothesis is that most relevant moral principles can be represented by heuristic functions and that these heuristics can be used to label training data quickly and efficiently to automate ethical decisions in complex domains. Under this assumption, good moral decision-making depends on de-noising and aggregating each heuristic labeling function. I will leverage the open-source “data programming” framework Snorkel [Ratner et al., 2017]. Data programming is the programmatic creation of datasets based on weak supervision strategies, including heuristic labeling and alignment with external knowledge bases (distant supervision) [Bach et al., 2017]. Snorkel has been used to achieve significant gains in classifier performance by high-profile users from Google and IBM to Stanford Medicine and the National Institutes of Health [Ratner et al., 2017]. By producing training labels from noisy moral heuristics, I will avoid the data collection barrier to moral AI while retaining the predictive advantages of machine learning. The following section defines the key components of the method.

Note that this approach does not solve the representation model for moral dilemmas. Each moral alternative must be represented in a format that experts can understand, so granularity and dimensionality are somewhat limited. However, so long as an expert has semantic knowledge of at least some of the features, they can still provide a useful heuristic on a particular subset of all the features considered. It is therefore possible to use a complex or hierarchical representation so long as enough experts are consulted to provide meaningful heuristics for a sufficient area of the feature space.

4.1 Pipeline

As a simple example, take the classic lifeboat dilemma: a ship is sinking, and its K passengers must escape using a lifeboat which can only hold $K - 1$ people. Suppose the captain must choose who should stay behind with the ship. To choose, he consults the ship manifest, which contains the age, gender, occupation and ticket class of each passenger.

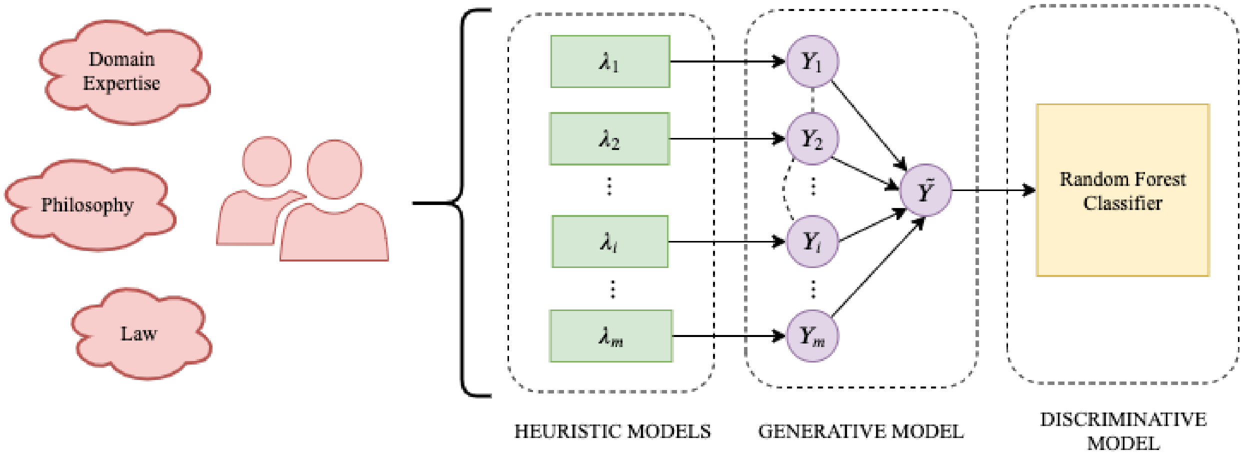

Figure 2: A data programming pipeline for training a random forest classifier to make moral decisions. Experts write heuristic functions based on domain knowledge which are used to produce labels. Labels are synthesized with a generative model and used to train a classifier.

Let $\mathcal{X}$ be the set of scenarios in which there are K moral alternatives. ($\mathcal{X}$ is the set of all K-combinations of the possible alternatives $\mathcal{A}$.) If $X_i \in \mathcal{X}$ is one such scenario, let $X_{i,k}$ be the feature vector representing the kth moral alternative available to the agent in X_i. In the lifeboat dilemma, X_i might be a voyage where all the passengers in the manifest are elderly ladies except for the captain; $X_{i,k}$ might represent the alternative where the captain chooses to sacrifice himself.

In crowd-sourcing, survey respondents are asked to solve various instances of moral problems like the lifeboat dilemma. Suppose survey respondents are presented with a sample of N scenarios $\mathbb{X} \subseteq \mathcal{X}$ from the set of all possible scenarios. Rankings are collected from each respondent for some or all of the sample scenarios; for instance, the jth survey respondent is presented with the moral features the ship's manifest X_i for a voyage and asked to provide a moral ranking $y_{i,j}$ over X_i. In the lifeboat dilemma, $y_{i,j}$ takes the form of a list of passengers in order of how morally appropriate it would be for each to stay behind. Then, a *fully* supervised classifier $\mathbb{F}$ is learned from the training set $\mathbb{X}$ and respondent rankings $\mathbb{Y} = \{y_{i,j}\}$.

In *weak* supervision, these crowd-sourced rankings are replaced by experts' heuristics. Rather than using a set of crowd-sourced rankings on individual scenarios to train $\mathbb{F}$, I use the open-source Snorkel framework [Sno] to generate probabilistic training labels $\tilde{\mathbb{Y}}$ for a training set without access to any "ground truth" rankings from experts (Figure 2):

1. Define the heuristic moral principles as a set of M labeling functions $\Lambda \subseteq \{\lambda \mid \lambda : \mathcal{X} \to \mathcal{Y}\}$, where $\mathcal{Y}$ is the set of all possible rankings over scenarios in $\mathcal{X}$. Each heuristic λ_m takes a scenario X_i as input and outputs a *heuristic* ranking $\hat{y}_{i,m}$ over the alternatives. As an example, an honorable captain might use the heuristic "leave behind passengers in order of descending rank, starting with the captain." (For more on the process of constructing heuristic labeling functions, see Section 4.2.) Then $\Lambda(\mathbb{X})_{N\times M}$ is a matrix of heuristic labels such that $\Lambda_{i,m} = \lambda_m(X_i) = \hat{y}_{i,m}\ \forall\ X_i \in \mathbb{X},\ \lambda_m \in \Lambda$.

2. Estimate the accuracies, correlations, and inter-dependencies of the black-box (for the purposes of estimation) labeling functions Λ by learning a *generative* model to produce a single probabilistic label for each scenario X_i. Since training labels are probabilistic, let $\tilde{\mathbb{Y}} = \{\tilde{y}_{i,k}\}$ be a set of *probabilistic* rankings denoting the true probability of choosing any alternative $X_{i,k}$ given X_i (with the stipulation that $\sum_{k=1}^{K} \tilde{y}_{i,k} = 1$). In this paper, we consider only the case where $K = 2$ (there are only two alternatives), so $\tilde{\mathbb{Y}}$ is just the set of probabilities $\{\tilde{y}_{i,1}\}$ of selecting the first of two alternatives in each X_i. In the lifeboat dilemma, $K = 2$ implies there are only two passengers; $\tilde{\mathbb{Y}}$ then contains the probabilities of choosing the first passenger in the manifest for all the possible voyages in X. The generative model aggregates the label matrix $\Lambda(\mathbb{X})$ into probabilistic labels $\tilde{\mathbb{Y}} \mid \mathbb{X}$ by estimating weights for each of the heuristic labeling functions. The structure of this model is described in Section 4.3.

3. Train a *discriminative* model $\mathbb{F}$ to predict $\tilde{\mathcal{Y}} \mid \mathcal{X}$. A wide variety of traditional classifiers, including deep learning models, may be used for $\mathbb{F}$. The choice of classifier depends on the domain.

Crucially, the true accuracies of each labeling function *are not known in advance*. (In the lifeboat example, there is no guarantee that the captain's strategy of sacrificing himself will always be the morally appropriate choice; on a prison boat, he may choose to sacrifice a convicted murderer instead.) Unless there is exogenous data about the success of each heuristic, their accuracies must be learned, unsupervised, in Step 2. In some lucky cases, there may be data about the quality of a given heuristic. For example, heuristics built to express the view-points of the various broad ethical theories (deontology, consequentialism, virtue ethics) may be specified by a study of the share of ethicists who hold each viewpoint, with some adjustment to account for abstentions and correlations [Bourget and Chalmers, 2014]. (A study may find that 90% of ethicists agree with the captain's self-sacrificing heuristic; then this heuristic may be *a priori* assigned a higher weight than a more controversial heuristic.) I explore a similar approach with the kidney exchange example in Section 5.2.

4.2 Heuristic Functions

Definition

A heuristic, or labeling, function λ is a simplistic rule for making a moral decision given a set of alternatives X_i. When faced with only two alternatives, the heuristic function is just a black box classifier which takes the concatenation of $X_{i,1}$ and $X_{i,2}$ as its input and outputs a binary classification $\hat{y}_i$. Some example labeling functions are provided in Section 5. This approach treats each labeling function as an individual "voter" voting on each alternative in X_i. Since heuristics express only incomplete strategies for decision-making, labeling functions have the option to abstain from voting. The *coverage* of a labeling function is the proportion of scenarios for which the labeling function does not abstain; its *polarity* is the frequency at which it outputs each label (some heuristics never output a particular alternative). If there are ground-truth labels available to form a development set $\mathbb{D} \subseteq \mathbb{X}$, then a heuristic's *accuracy* is the proportion of true positives in $\lambda(\mathbb{D})$ based on the crowd-sourced labels $\mathbb{Y}$. Other metrics (e.g. F1 score) should be used if the frequencies of choosing various alternatives are unbalanced.

Workflow

I used the following workflow to construct heuristic functions for the use cases in Section 5:

1. Review philosophical literature, legal regulations, and example scenarios to create a list of potential heuristics.
2. Write a prototype version of the heuristic function.

3. Assess polarity, coverage and accuracy for each labeling function. If absolutely no ground-truth labels are available to assess accuracy, create and label just a few unit tests by hand. Examine false positives and false negatives to check for bugs and edge cases.
4. Refine the heuristic function and repeat 3 until the function adequately expresses the abstract heuristic.

Implementation

In this study, I use Python functions to code heuristics, but any platform may implement a heuristic. Though this study is only a proof-of-concept, it is important to note that in practice, the use of a programming language may limit the representativeness of heuristics collected; if the only experts consulted are those who know Python, the heuristics obtained will likely be skewed. Likewise, if the interface for providing heuristics is exceedingly complicated or requires English language skills, individuals without a formal education or who are not native speakers may not be qualified as experts, not by virtue of their moral expertise, but by virtue of their backgrounds. Some level of abstraction and translation, manual or automated, is necessary to collect moral heuristics from experts of all backgrounds.

Creating heuristics also requires a minimal process of tuning and refining to achieve good results (Steps 3 & 4 in the heurstic development workflow). Ratner et al. [2017] demonstrate the importance of training experts to write and evaluate their heuristic functions and were able to train several experts with education levels ranging from B.S. to Ph.D. and prior coding experience to write heuristic functions in a two-day workshop. For a classification problem in the field of bioinformatics, these users achieved better accuracy scores than hand-labelers on Mechanical Turk with only around 10 heuristic functions [Ratner et al., 2017].

4.3 Generative Model

To de-noise experts' moral heuristics, we use the generative model proposed by Bach et al. [2017]. The true preferred alternative y is a latent variable in a probabilistic model, where the "votes" of each heuristic λ_m are noisy signals. The generative model, a factor graph for estimating the heuristic weights w, is defined as follows:

$$p_w(\Lambda, Y) = Z_w^{-1} \exp\left(\sum_{i=1}^{N} w^T \phi_i(\Lambda, y_i)\right) \tag{1}$$

where Z_w is a normalizing constant and Λ is the heuristic label matrix. $\phi_i(\Lambda, Y)$ is a concatenated vector containing factors for labeling propensity, accuracy, and pairwise correlations:

$$phi_{i,j}^{Lab} = \mathbb{1}\{\Lambda_{i,j} \neq \emptyset\} \tag{2}$$

$$phi_{i,j}^{Acc} = \mathbb{1}\{\Lambda_{i,j} = y_i\} \tag{3}$$

$$phi_{i,j,k}^{Corr} = \mathbb{1}\{\Lambda_{i,j} = \Lambda_{i,k}\}\ (j,k) \in C \tag{4}$$

Label propensity is the estimated likelihood that a heuristic provides a label for any given data point; label accuracy is the estimated likelihood that its label matches the ground-truth label; label correlations model the dependencies between labeling functions, which are not necessarily independent. C is the set of potential correlations (pairs of labeling functions). For m labeling functions, $w \in \mathbb{R}^{2m+|C|}$.

The objective function for unsupervised learning minimizes negative log *marginal* likelihood $p_w(\Lambda)$ conditional on Λ:

$$\hat{w} = \underset{w}{\operatorname{argmin}} \left(-\log \sum_{Y} p_w(\Lambda, Y) \right) \tag{5}$$

which yields predictions $\tilde{Y} = p_{\hat{w}}(Y|\Lambda)$. Since y is latent, only the marginal likelihood $p_w(\Lambda)$ can be used to estimate the weights $\hat{w}$. For computational efficiency, the objective function can be expressed as the marginal *pseudolikelihood* of a single labeling function Λ_j conditioned on the outputs of the others $\Lambda_{\neg j}$, with l_1 regularization:

$$\hat{w} = \underset{w}{\operatorname{argmin}} \left(-\log p_w(\Lambda_j | \Lambda_{\neg j}) + \epsilon ||w||_1 \right) \tag{6}$$

$$= \underset{w}{\operatorname{argmin}} \left(-\sum_{i=1}^{m} \log \sum_{y_i} p_w(\Lambda_{i,j}, y_i | \Lambda_{i,\neg j}) + \epsilon ||w||_1 \right) \tag{7}$$

where $\epsilon > 0$. Snorkel minimizes by interleaving stochastic gradient descent steps and Gibbs sampling steps [Ratner et al., 2017], an approach similar to contrastive divergence [Hinton, 2002].

There are some cases in which a simple majority voter is better specified for label generation than this model. (In a majority labeling model, the output of each heuristic is a "vote" for that moral alternative. The alternative with the most votes is the final label; ties are broken randomly.) When label density (average coverage across all labeling functions) is sufficiently high or sufficiently low, a majority voter performs just as well as the generative model. In low density settings (few data points have multiple votes), the number of conflicts between heuristics is lower and an egalitarian voting schema is negligibly worse. In high-density settings, assuming average labeling function accuracy is better than random, Ratner et al. [2017] prove that majority voting converges exponentially to an optimal solution with label density. In practical terms, the generative model is most appropriate when the domain is not hyper-specific (experts only have very specific moral expertise) and not hyper-general (experts provide heuristics with very high coverage).

5 Experiments

I construct a proof-of-concept implementation of this approach to demonstrate its application to real-world ethical dilemmas. I evaluate my approach on pairwise ($K = 2$) moral preference data from surveys conducted in two domains: autonomous vehicles (Section 5.1) and the kidney exchange (Section 5.2). To approximate a real-world data programming workflow, I assumed the surveyed users to be domain experts and used the survey results to write a set of heuristic functions for each domain. Heuristics are combined into a single label for each data point using either the *generative* labeler or a *majority voting* labeler (Section 4.3). I then compare the performance of a machine learning model trained on the labeler output (a *weakly supervised* classifier) to a model trained on the ground-truth user survey data (a *fully supervised* classifier) and benchmark my approach against prior models. All heuristic functions, data, and code used to produce the figures in this paper are available at `https://github.com/ryansteed/hmm`.

5.1 Autonomous Vehicle Trolley Problem

In the autonomous vehicle domain, Awad et al. [2018] explore the classic trolley problem in a modern context. Through the Moral Machine website [Mor], the authors collected 40 million moral decisions from 233 countries and territories for the following scenario: imagine an autonomous vehicle suffers brake failure just before a crosswalk and must choose whether to collide with the pedestrians or swerve into a barrier and crash (Figure 1). What should the self-driving car do? In the Moral Machine interface, each respondent is presented a set of 13 unavoidable accident dilemmas with only two possible actions: to stay on course or to swerve. Each dilemma presents a set of characters, pedestrians or passengers, designed to test moral preferences across the following dimensions: saving humans (versus pets), staying on course (versus swerving), saving passengers (versus pedestrians), saving more lives (versus fewer lives), saving men (versus women), saving young people (versus the elderly), saving law-abiding pedestrians (versus jaywalkers), saving the fit, and saving those with higher social status. Additional characters include criminals, pregnant women, and doctors. Some dilemmas isolate a particular feature (e.g. gender) and hold all other factors constant, as in Figure 1. Other dilemmas contain a random mix of moral decision-making factors. For each participant, all the alternatives presented are randomly generated and are nearly unique; each unique alternative tested is included an average of 2.3 respondent surveys.

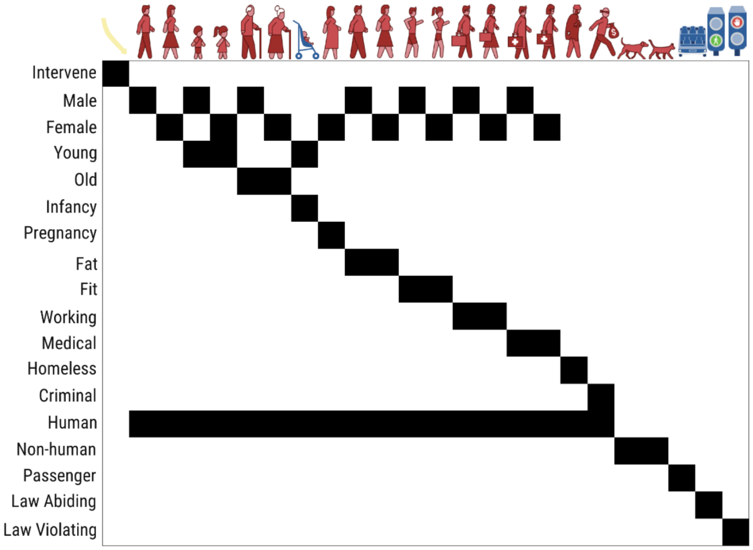

Figure 3: A binary matrix for decomposing Moral Machine characters into abstract moral features. Black squares indicates a positive mapping from character to abstract moral feature. Figure from Kim et al. [2018].

5.1.1 Data

Awad et al. [2018] published a set of over 18 million pairwise comparisons obtained from over 1.3 million respondents. So that the data collected will be balanced over each moral dimension and the responses of users fully explored, this experiment considers only votes from complete 13-dilemma sessions. This subset includes 1,544,920 moral decisions from 51,211 unique respondents. Respondents are concentrated mostly in the United States and Europe.

Formally, each moral alternative in a any scenario X can be represented as a vector of integer features Φ. The vector contains an integer representing the quantity of each character saved by choosing this alternative, along with several other features describing the alternative: a binary variable denoting whether the car is swerving (that is, whether an algorithmic intervention has occurred); a binary variable indicating whether the pedestrians have a red light, a green light, or no crossing signal whatsoever; and a binary variable indicating whether the characters saved in this alternative are passengers. To simplify the problem, I follow Kim et al. [2018] in decomposing the full feature vector Φ into the simplified *morally abstracted* vector Θ with a linear mapping $F : \Phi \to \Theta$. $F(\Phi) = B\Phi$, where B is the binary matrix shown in Figure 3. For the Moral Machine dilemma, each scenario X_i is represented by a pair of

```python
@labeling_function()
def utilitarian(x):
    """Save the most human lives."""
    saved_by_int = x['intervention']['Human']
    saved_by_no_int = x['no_intervention']['Human']
    return argmax([saved_by_int, saved_by_no_int])
```

Figure 4: A simple utilitarian heuristic in Python using the Snorkel labeling function interface. The function takes as input a dataframe with abstract feature vectors for each alternative (intervention or no intervention by the moral AV) and chooses the alternative that saves the most human lives.

abstract moral feature vectors $(\Phi_0, \Phi_1)_i$. Let Φ_0 be the alternative that results from the autonomous vehicle staying on course. Thus the j-th survey respondent's moral decision $y_{i,j}$ is a binary variable, 0 for Φ_0 and 1 for Φ_1.

5.1.2 Heuristics

The goal of the remainder of this section is to provide a set of heuristics for determining $\tilde{\mathbb{Y}}$ and to compare decision-making models trained on $\tilde{\mathbb{Y}}$ to models trained on the "ground-truth" labels $\mathbb{Y}$. For the sake of comparison, I assume that pairwise comparisons collected through the Moral Machine website reflect expert opinions about morality in the Moral Machine problem. Mirroring the cross-cultural preferences estimated by Awad et al. [2018], I wrote a set of 16 functions expressing heuristics for statistically significant global moral preferences (e.g. "save doctors" and "do not hit the pedestrians if they are crossing legally"). An example labeling function expressing a utilitarian principle is listed in Figure 4. Heuristics were debugged using a held-out development partition of 25,527 (20%) responses from the test set.

Heuristic Accuracy. To evaluate the heuristic functions, I partitioned the training set of Moral Machine responses again to obtain a validation set with 106,105 responses, 20% of the training set. It takes only seconds to label or abstain from every data point in the validation set. When do survey respondents tend to agree with each heuristic? Henceforth, I will call this measure "accuracy," since the human responses are treated as ground-truth for the sake of comparison. Figure 5 shows each heuristic's individual accuracy in each scenario type tested by the Moral Machine website. As expected, heuristics pertaining directly to the given scenario tend to perform best, such as the “save youth” heuristic in scenarios where the agent is asked to choose between a group of young and old people. In scenarios where the characters are generated randomly, the “sacrifice criminals” and “sacrifice pets” heuristics received a notably higher consensus than other heuristics, but less so for “sacrifice the homeless.”

5.1.3 Label Model

The next step is to aggregate the heuristic labels into a single predicted label for each scenario. This particular use case is relatively high density (Figure 7), so we can expect the majority voting model and the generative model to perform equally well on a large number of data points. In fact, I find that where majority voting labeler agrees with Moral Machine respondents 67.9% of the time, the generative model agrees only 63.0% of the time.

Weight Estimation. Figure 6 shows the rate of agreement between each labeling function and the pairwise preferences expressed by Moral Machine respondents. There is a clear tradeoff between coverage and accuracy; labeling functions that are more specific tend to perform better (e.g. the heuristic “if an alternative saves only pets, choose the other”). However, the “save females” function is a stand-out success, suggesting it may be a popular, widely applied heuristic among respondents. Most importantly, Figure 6 reports the weights w estimated by the generative model - there is a clear correlation between the coverage and accuracy of a heuristic and its estimated weighting in the label synthesizer. It appears that the generative model is capable of recognizing specific, accurate heuristics for this use case without access to ground-truth data.

Generative Model Accuracy. In addition to heuristic-specific accuracy scores, Figure 5 also shows the accuracy of the aggregate decision produced by the generative label model. Human respondents tend to agree with the generative model most when the dilemma is between pets and humans (“Species”) and when the dilemma is between saving more lives and saving fewer lives (“Utilitarian”). Notably, the accuracy of the heuristic model is highest for those scenarios with the highest effect sizes, as measured in the Moral Machine experiment [Awad et al., 2018]. In other words, the heuristic model tends to agree with human respondents when moral preferences about the scenario in question are strong. One other important observation

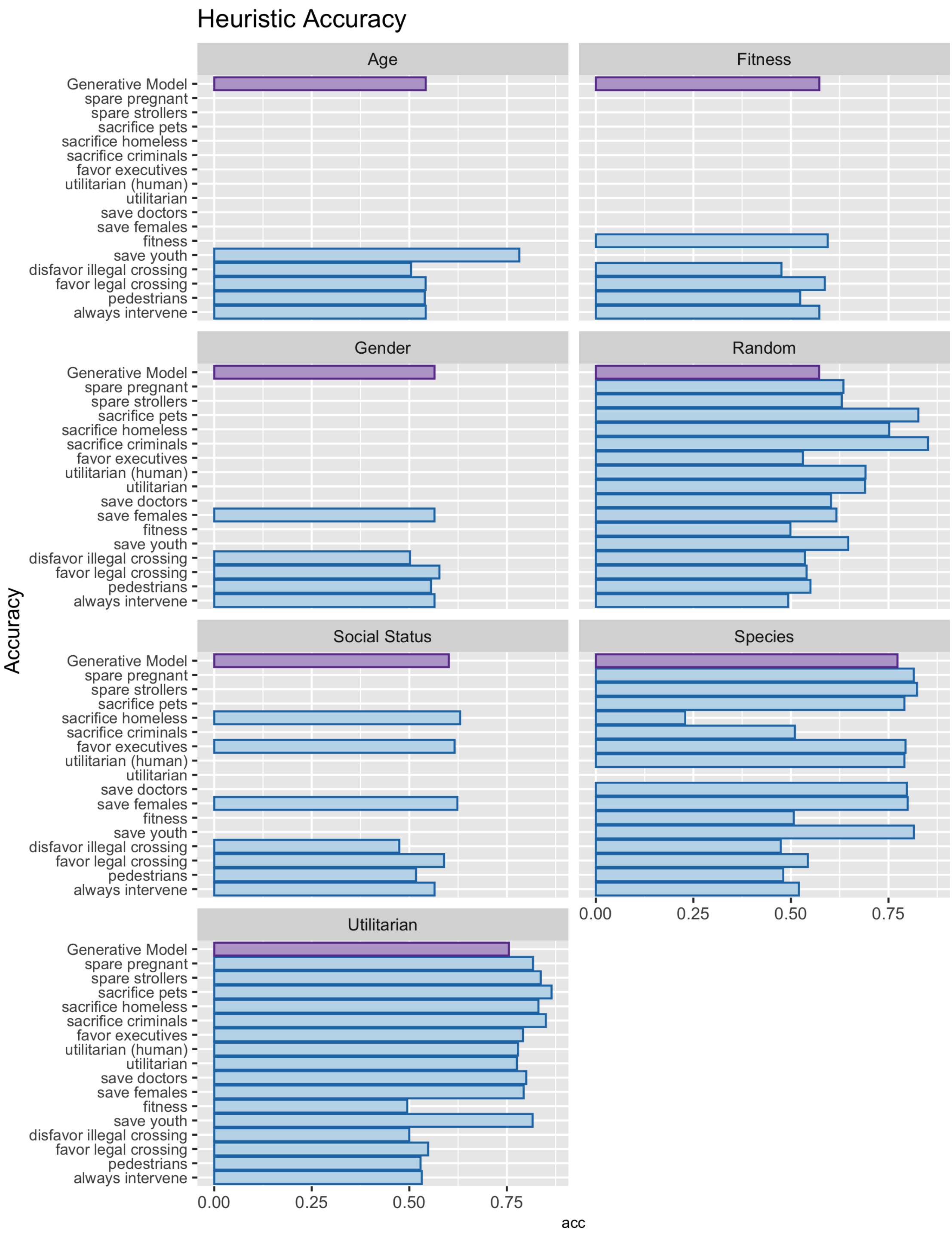

Figure 5: Accuracy by heuristics for each scenario type in the Moral Machine dataset. Scenario types describe scenarios experimentally designed to isolate a single moral factor (e.g. age) by holding every other factor constant and randomly varying the free factor. Scenarios that do not isolate a single factor are "Random." Note that some heuristics do not have coverage in certain scenario types; no bar is displayed for these cases.

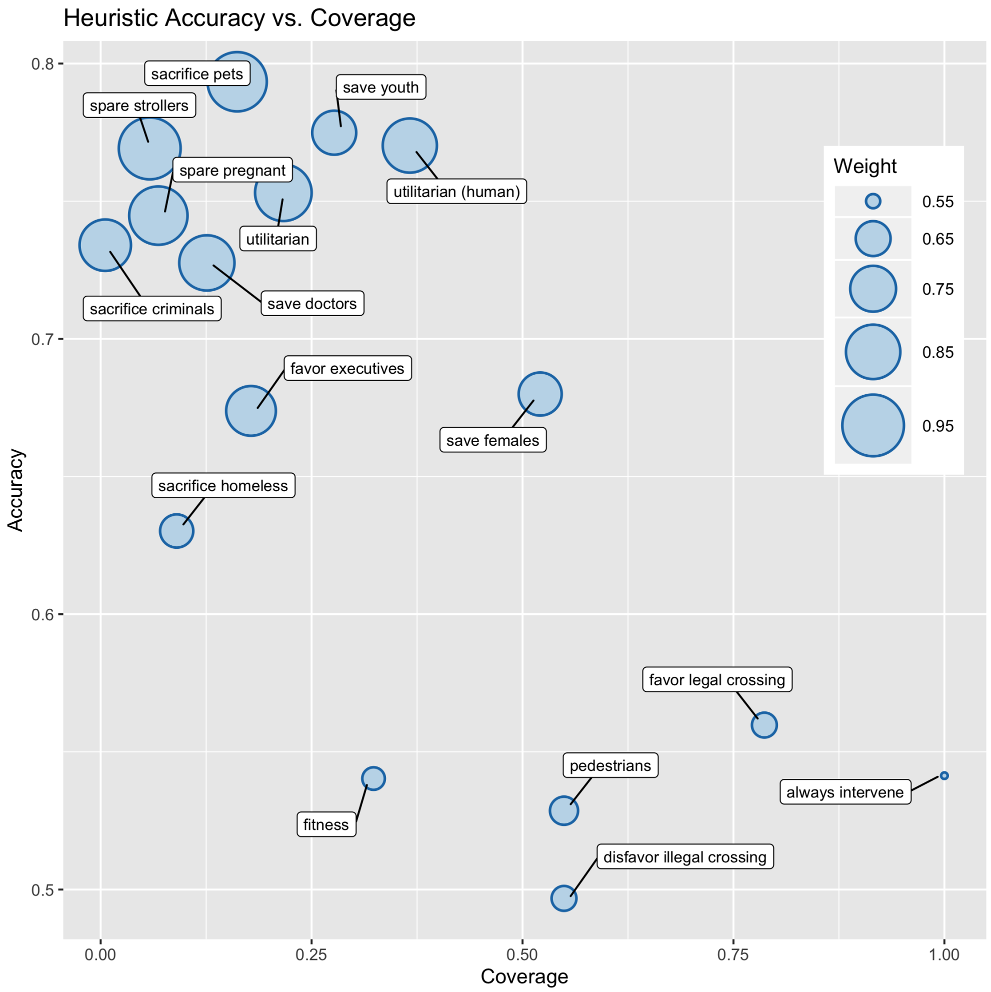

Figure 6: Rate of agreement between heuristic functions and Moral Machine respondents in the validation set, sized by estimated weight. Accuracy is the proportion of responses for which the heuristic (indicated by labels in the graph) chose the same alternative as the human respondent. Coverage is the proportion of scenarios for which the heuristic did not abstain. Estimated heuristic weights are computed without access to ground-truth.

about the performance of the heuristic functions is that for scenarios where many heuristics abstained (“Age,” “Fitness,” “Gender,” “Social Status”), the resulting heuristic label matched human responses less often. There are two likely explanations for this phenomenon: first, the fact that accuracy tends to decrease with label density in general; second, the fact that fewer moral factors were involved in these scenarios, putting the burden of decision-making on only a few heuristics. The heuristic generative model tends to match human respondents more when a diverse set of heuristics are applicable.

To assess the relative impact of each heuristic on predicted label accuracy and the robust-

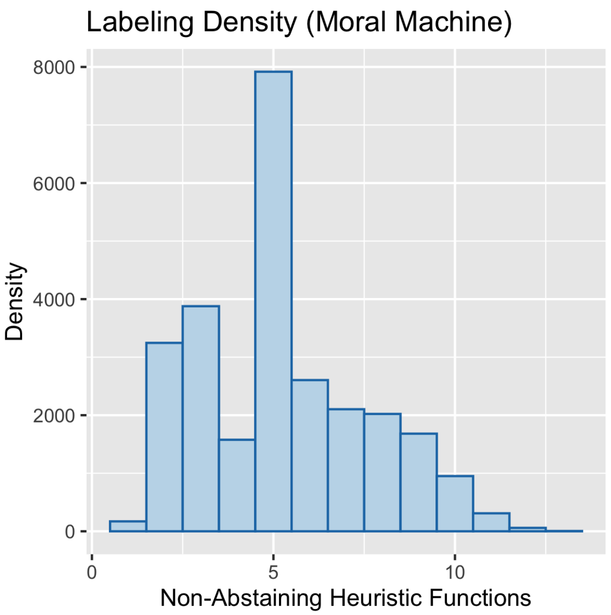

Figure 7: Label density in the validation set, smoothed with a multiplicative bandwidth adjustment. The label density is the number of non-abstaining heuristic functions for a given moral scenario.

ness of the label model to heuristic inclusion, I iteratively removed each heuristic function from the model and compared the accuracy of the perturbed model to the baseline model with all heuristic functions included (Figure 8). The accuracy gains are all relatively marginal, though the stand-out heuristics with especially high accuracy and coverage from Figure 6 seem to add the most predictive value. These heuristics also tend to match the effect sizes of respondents' moral preferences [Awad et al., 2018].

5.1.4 Discriminative Model

After tuning the generative model, I trained a discriminative model to generalize beyond the training to new dilemmas an autonomous vehicle might potentially encounter in the wild. The discriminative model is a random forest binary classifier with 100 estimators, Gini split criterion, no maximum depth, a minimum of two samples per split, and all remaining features considered at each split. All classifiers in the following section were trained on the training partition (424,419 examples) and tested on a separate test partition of 106,105 dilemmas presented to Moral Machine respondents.

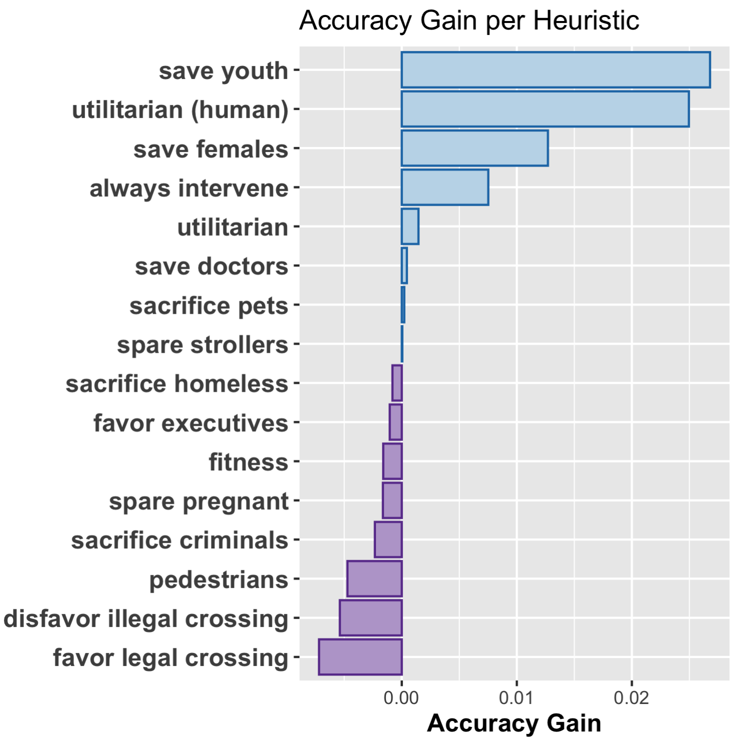

Figure 8: Negative accuracy loss after re-constructing the generative model without the given heuristic. Accuracy gain is equivalent to the baseline model accuracy, with all heuristics included, minus the perturbed model with the given heuristic removed.

Discriminative Model Accuracy. A baseline classifier trained on the ground-truth moral decisions from human respondents $\mathbb{Y}$ achieved 69.6% accuracy. (Accuracy is an appropriate measure of performance since the binary label is balanced.) To train a heuristic-based classifier on the results from the generative model $\tilde{\mathbb{Y}}$, I imputed the median for approximately 140,000 dilemmas with missing feature values. Additionally, I transformed the probabilistic labels generated by the discriminative labels into binary labels by choosing the label with the highest probability. In the case of a tie between labels, the predicted label is chosen randomly. This transformation is lossy - a classifier which can interpret probabilistic targets is preferred (for example, using a cross entropy loss function). Trained on the rounded labels, the heuristic-based classifier achieves only 66.6% accuracy.

Accuracy Gain from Additional Respondents. There is no current benchmark for aggregated accuracy on this dataset: Noothigattu et al. [2018] measure the correspondence of their method with a voting-based outcome for a set of synthetic respondents, but not for the Moral

Machine respondents because the dilemmas are randomly generated and responses cannot be grouped. Kim et al. [2018] measure approximately 75% out-of-sample prediction accuracy for their hierarchical Bayesian approach to learning moral preferences, predicting 128 respondents' final five decisions using a model fitted on their first eight. Figure 9 displays accuracy measurements under the same experimental conditions as Kim et al. [2018], finding that despite not accounting for individual variations in moral preference, the baseline classifier and achieves only a marginally lower comparable accuracy (70.0%) trained on responses from 128 voters. This result is comparable with the accuracy of Kim et al. [2018]'s *naive* benchmark, which does not account for group values. When trained on heuristic labels for the same scenarios presented to those 128 voters, the classifier learns at the same rate, but scores approximately 5 points lower.

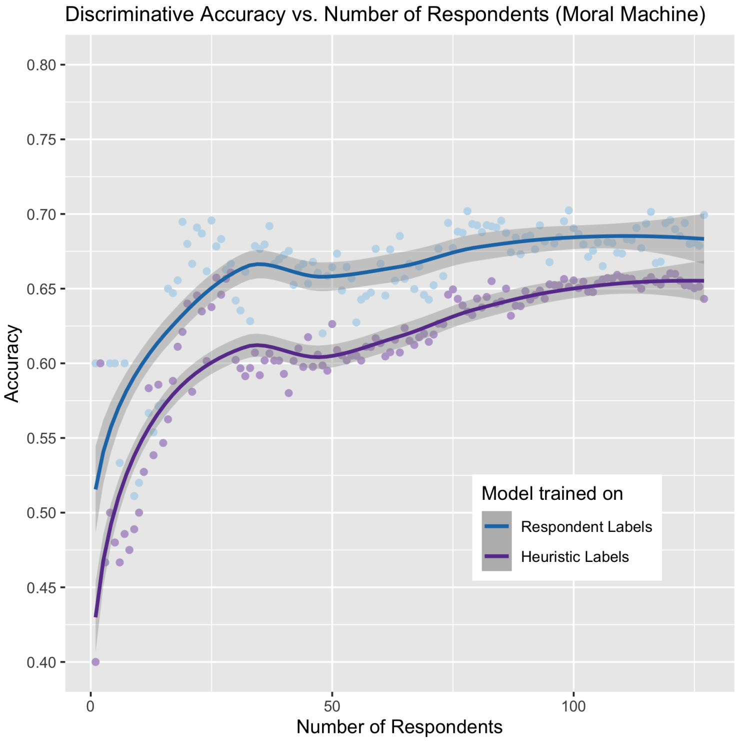

Figure 9: Discriminative model accuracy per number of respondents included in training, fitted with a regression on the square root of the accuracy. Both models are trained on respondents' first 8 scenarios and tested on their last 5 scenarios. Results are averaged across 10 trials; the 95% confidence interval is extremely small and not shown. Smoothed fit line is a Loess regression of accuracy on the training set size.

Learning Curve. How does each model perform when data is more scarce? I apply the full pipeline (running the heuristic functions over the training set, fitting the generative model, and fitting the discriminative model) over a 5-fold shuffled partition of the entire dataset, scoring the performance of a supervised classifier trained on ground-truth labels and a semi-supervised classifier trained on the heuristic labels. Figure 10 shows the cross-validated accuracy scores for each model plotted against the size of the training set. When human-labeled data is very scarce, the two approaches perform nearly equally well. It is only after this point that the fully supervised model begins to perform significantly better than the heuristic approach, suggesting that the gains in accuracy from a fully-supervised approach only come into effect after a heavy investment in manual labeling.

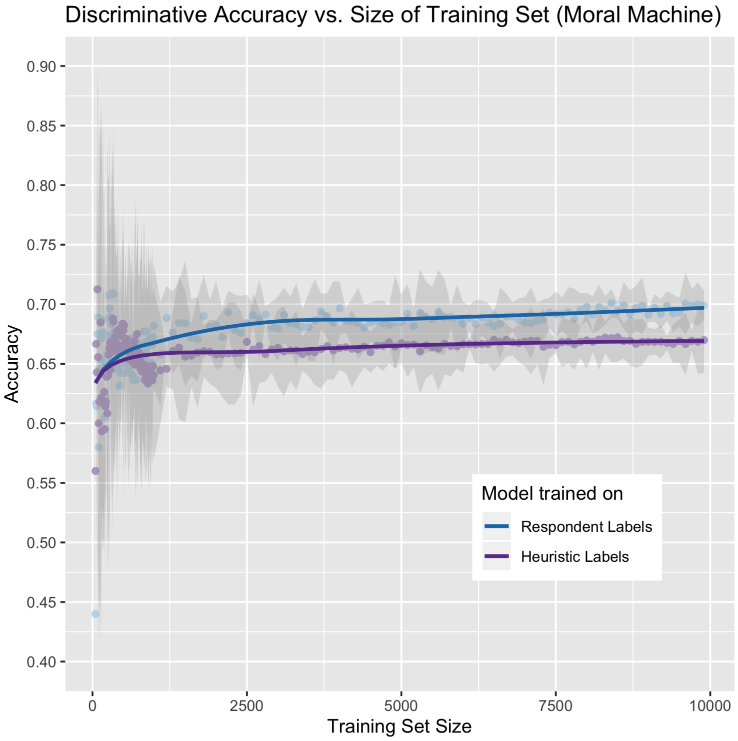

Figure 10: Discriminative model accuracy increase as the size of the training set is increased. Accuracy is measured as the mean across a 5-fold cross-validation, where the generative model and discriminative model are fitted on a training partition without access to a held-out test set. Grey ribbons report the 95% confidence intervals for the two discriminative models measured, supervised and semi-supervised (heuristic). Smoothed fit line is a Loess regression of accuracy on the square root of the training set size.

5.2 Kidney Exchange

Another domain in which an algorithm may be asked to make life-or-death moral decisions is in market mechanism design, particularly for scarce resources. In kidney exchanges, a central market clearing algorithm matches kidney donors to patients in need of an organ. Patients may be prioritized according to a mixture of medical and moral criteria, in addition to the logistical considerations of matching donors to kidneys.

5.2.1 Data

Freedman et al. [2020] develop an end-to-end method for estimating the moral weighting of patient profiles for tie-breaking in a normal kidney exchange. To estimate the moral value of a static set of 8 patient profiles, they survey 289 Amazon Mechanical Turk (MTurk) users, who are asked to allocate a kidney in a series of pairwise dilemmas. Each fictional patient is 30 or 60 years old, drinks alcohol rarely or frequently, and has no other health problems or has skin cancer in remission. Each respondent was presented with all 28 pairwise contests, for a total of 8,092 pairwise comparisons. There were no missing values. Each moral feature (age, drinking, and health) was coded as a binary variable, where 0 represents lower age, infrequent drinking, or no prior health conditions.

To assess the validity of a heuristic approach to moral decision-making in a different domain with fewer moral factors, I compare Freedman et al. [2020]'s approach with a model trained on three simple heuristics: give kidneys to younger patients, patients who drink infrequently, and patients who do not have skin cancer in remission. These heuristics are sourced from actual heuristics reported by the MTurk respondents, who were asked to explain the strategy they used to decide which patient should receive the kidney. Once again, for the sake of validation, I assume that the MTurk respondents are domain experts, though the advantage of this approach is that only one or two experts may write heuristic functions that capture the same moral knowledge as a large group of surveyed laypersons.

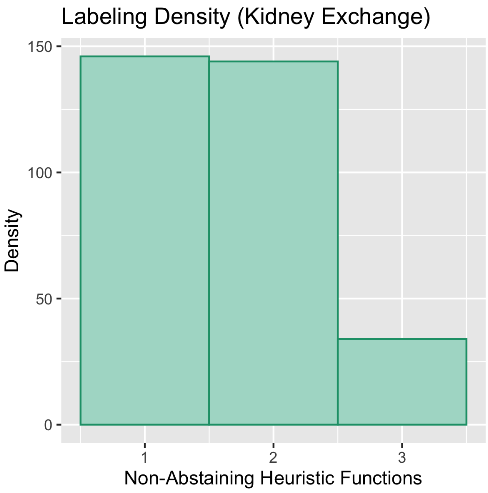

Figure 11: Label density in the validation set. The label density is the number of non-abstaining heuristic functions for a given moral scenario.

5.2.2 Heuristics & Label Model

Weight Estimation. Table 1 reports the coverage, accuracy, and estimated weight for each heuristic. All three heuristics cover a similar majority of the scenarios, since each variable was varied with equal frequency in the patient comparisons. Conflicts between the heuristics are relatively common; each heuristic conflicts with another about 25% of the time. The density of these labeling functions on these dilemmas is reported in Figure 11; nearly every point has one or two votes. Notably, the generative model does little to discriminate between the three heuristics - the estimated weights are nearly identical (Table 1). As a result, a majority vote labeler performs just as well as the generative model does, both models agreeing with the MTurk respondents 80.2% of the time.

Heuristic	% Coverage	% Accuracy	Estimated Weight
Choose younger patient	60.5	83.2	0.602
Choose patient who drinks less	56.8	78.8	0.594
Choose patient with no other health issues	56.4	61.0	0.600

Table 1: Coverage, accuracy, and estimated weight parameters for a simple set of moral heuristics for the kidney exchange. Accuracy refers to the heuristic's agreement with surveyed MTurk respondents.

Generative Model Accuracy. As displayed in Figure 12, the generative model agrees with human respondents for nearly all scenarios with only one isolated moral factor, but disagrees more frequently about interactions between two or more variables. Notably, the heuristic "choose the patient who drinks less" suffers very little loss in accuracy when applied to situations where only level of drinking is varied versus situations where both prior health conditions and drinking are varied. In random scenarios, "choose patient with no other health issues" performs barely better than a coin flip, but in scenarios where only health is varied performs vary well. This may suggest that participants only resort to heuristics about prior health conditions when no other differences are present and a choice must be made.

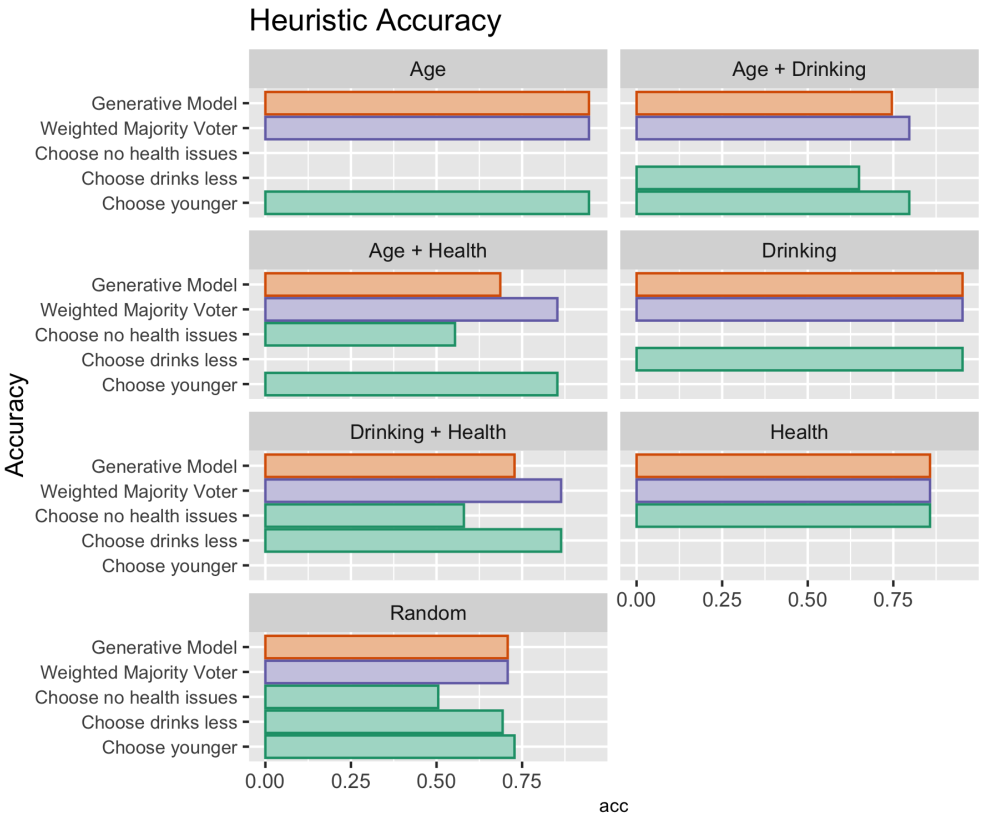

Figure 12: Accuracy by heuristics for each comparison type in the kidney exchange case study. Scenario types describe scenarios experimentally designed to isolate a single moral factor (e.g. age) by holding every other factor constant and randomly varying the free factor. Some scenarios isolate multiple factors, or all the factors at once ("Random" scenarios). All tested scenarios fall into one of these types. Note that some heuristics do not have coverage in certain scenario types; no bar is displayed for these cases.

5.2.3 Discriminative Model

Discriminative Model Accuracy. Using the survey data to estimate a moral preferences for each possible patient profile (combination of moral factor levels) , Freedman et al. [2020] adjust the kidney exchange algorithm to simply choose the patient with the higher normalized profile preference in the case of a tie. This strategy is not generalizable; Freedman et al. [2020] do not specify a way to calculate moral weights for new patient profiles or new moral factors. Further, their strategy requires a large number of responses for every combination of patient profiles, on the order of n^2 comparisons if n is the number of possible patient profiles in the exchange pool. It would be difficult to gather enough survey data to estimate moral preferences for that many pairwise comparisons, especially if more moral factors or more factor levels are added and the number of possible patient profiles grows. But for a small problem space like the one in this example, Freedman et al. [2020]'s strategy (choosing the patient profile with higher estimated preference) agrees with the MTurk respondents just as often as a supervised classifier, about 86% of the time (Figure 13). My weakly supervised approach also performs remarkably well, agreeing with respondents 81.1% of the time. Figure 13 shows the learning curves for each method.

Accuracy Gain from Additional Respondents. In the kidney exchange example, the model tends to learn less from additional respondents than in the Moral Machine example (14); in fact, the learning curve looks very similar to the learning curve for additional training data. Perhaps this is evidence that the variance of individual moral preferences is lower in the kidney exchange use case, either because respondents naturally agreed more about morality for each example or because the number of moral features is fewer.

5.2.4 Ranked-Choice Heuristics

Because the kidney exchange data included free-form responses from respondents about the strategies they used to choose kidney recipients, I conducted an additional experiment in which the presence of heuristics in respondents' reported strategies was used inform the heuristics' weight in the generative step. In this way, additional information about the accuracy of each

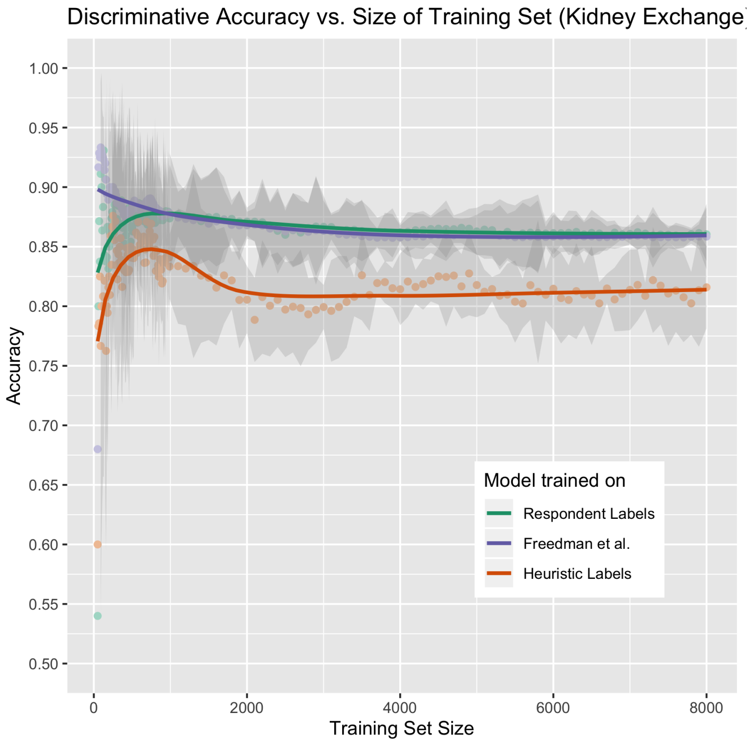

Figure 13: Discriminative model accuracy increase as the size of the training set is increased. Accuracy is measured as the mean across a 10-fold cross-validation, where the generative model and discriminative model are fitted on a training partition without access to a held-out test set. Grey ribbons report the 95% confidence intervals for the two discriminative models measured, supervised and semi-supervised (heuristic). Note that the weights are not recalculated for each training set - rather, the baseline is a fixed set of weights. Smoothed fit line is a Loess regression of accuracy on the square root of the training set size.

heuristic can be used to augment or supplant the weights estimated by the generative model.

Heuristic Rankings. At the end of the kidney exchange experiment, users were asked to describe in words the reasoning behind their moral choices. Most respondents' strategies can be categorized as one of the three heuristics specified in my model or its direct opposite. Each user tended to respond in a ranked fashion: e.g., "I always chose the younger patient; if both patients were the same age, then I chose the one who drank less." I manually coded each response into a ranking of heuristics (ties allowed); for the example above, the coded ranking is 1) choose younger patient; 2) choose patient who drinks less; 3) all other strategies and their opposites. I ignored respondents who did not provide a strategy or whose strategies did not directly match or directly contradict one of the three heuristics used in the generative model (there were 35 such respondents). Table 2 reports the Borda counts for these coded strategies;

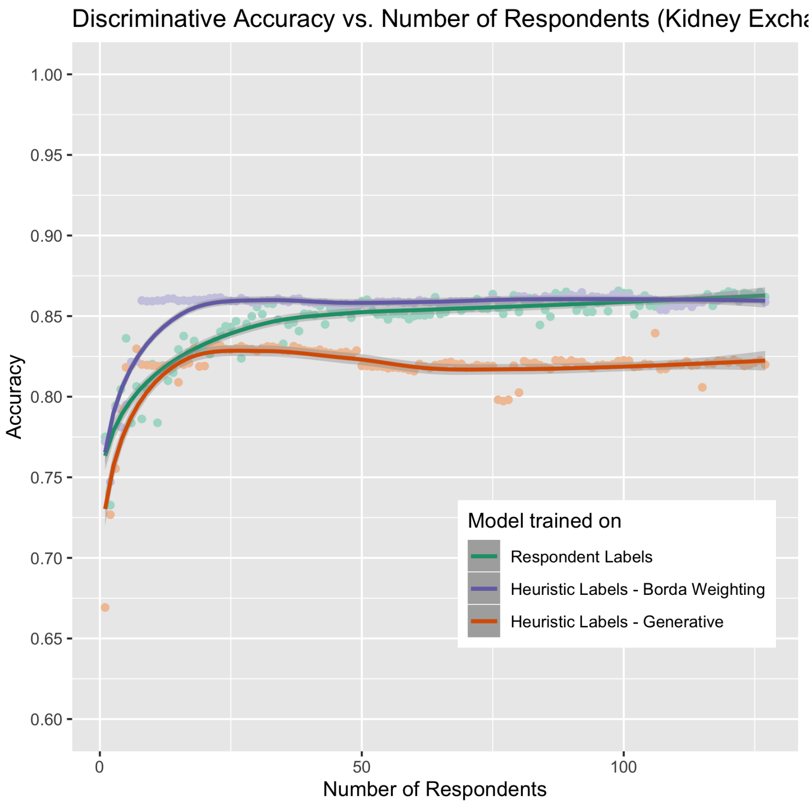

Figure 14: Discriminative model accuracy increase as the size of the training set is increased. Accuracy is measured as the mean across a 10-fold cross-validation, where the generative model, discriminative model, and Borda weights are fitted on a training partition without access to a held-out test set. Grey ribbons report the 95% confidence intervals for the two discriminative models measured, supervised and semi-supervised (heuristic). Note that the weights are not re-calculated for each training set - rather, the baseline is a fixed set of weights. Smoothed fit line is a Loess regression of accuracy on the square root of the training set size.

Heuristic	Borda Count
Choose older patient	0.11
Choose younger patient	3.42
Choose patient who drinks more	0.04
Choose patient who drinks less	2.71
Choose patient with other health issues	0.19
Choose patient with no other health issues	2.10

Table 2: Mean Borda count for each heuristic and its contradiction. Borda counts are calculated from manual ranked choice coding of text responses. (The Borda count for a given alternative is equal to the number of other alternatives ranked below it in a participant's survey response.) Ties are permitted.

the Borda counts can be interpreted as a popularity ranking of the strategies among the survey respondents.

Weighted Majority Model Accuracy. Imagining that the MTurk respondents are domain experts, the popularity of a given heuristic strategy may be used to inform which heuristics are given priority when their outputs are aggregated into a single label. As a simple proof-of-concept, I modify the majority voting model to use weights when tallying the votes from each heuristic function. (In other words, each heuristic function is allocated a certain number of votes according to its popularity amongst the survey respondents. The labels produced by more popular heuristics have greater influence over the final, aggregate label produced by the weighted majority voter.) The voting weights are just the Borda counts scaled from 0 to 1.

This weighted majority voting model, which boosts popular heuristics, agrees with MTurk respondents a remarkable 5.1% more often than either the generative model (Section 4.3) or an unweighted majority vote model (all heuristics have a single vote). Figure 12 also shows the performance of the weighted majority voter per comparison type; all of the gain in accuracy comes from scenarios where two moral factors are varied, such as Age & Health.

Weighted Majority Learning Curve. Figure 14 shows the learning curve for weak supervision with each label model type (weighted majority, unweighted majority, or generative) alongside a directly supervised classifier. Here, weights were calculated only using the mean Borda count of the n respondents included in the training set. The approach learns at a much quicker rate per number of respondents in the training set than even the fully-supervised classifier, and achieves the same equilibrium accuracy. While this strategy may not work for a use case where heuristics are too complicated or contingent to be easily ranked, it is especially effective for simple uses cases like this one. Rather than providing 28 pairwise choices, experts could simply vote on a set of candidate heuristics.

6 Discussion

Broadly, weakly supervised learning combines the "top-down" and "bottom-up" approaches to moral decision-making. This approach significantly reduces the costs associated with obtaining large sets of high-quality labels while retaining out-of-sample accuracy. Sourcing moral principles from experts provides advantages for customizing domain-specific algorithms without requiring the rigid, disjunctive set of rules need to code a game-theoretic model; it is unreasonable to assume that a set of non-conflicting moral principles can be obtained for any sufficiently complex moral problem, and heuristic functions are expected to overlap and conflict. My approach does not attempt to create a unified moral theory; rather, experts can provide just as much heuristic information as they have about the domain without attempting to generalize beyond their contingent experience; in this way, moral expertise from multiple specialists can be combined to create an efficient moral decision-maker across the entire domain. Additionally, in the fully-supervised case, domain experts have no way to express uncertainty in their moral decisions. In semi-supervision, probabilistic labels confer the relative uncertainty of the heuristics about a given moral decision to the discriminative model, which makes the final, discrete decision based on generalizations across multiple, uncertain labels.

My approach also has the logistical advantage of interfacing with machine learning at the training phase. Rather than adding a new suite of moral models to their machine learning pipelines, practitioners can use existing classification techniques trained on the combined labels provided by experts' moral heuristics. For the purposes of interpretability, the output of the label model can be accounted for by examining the individual votes of each heuristic function. Interpretations at the discriminative step must be obtained using existing interpretability methods, or a naturally interpretable classifier (such as a decision tree) [Du et al., 2018].

The experiments performed in Section 5 show that while heuristics derived from measured moral preferences do not always agree with survey respondents, a weakly-supervised machine learning approach can make comparable moral decisions with no crowdsourcing whatsoever. Though this paper uses crowd-sourced data to prove the efficacy of my method, this approach does not require massive data collection. In a real-world example, a set of experts need only provide a set of heuristics and a small development set of hand-labeled dilemmas for validation. For simpler domains, I show that finding a small set of popular heuristics and asking respondents to rank those heuristics may be a viable alternative to collecting a full set of pairwise comparisons, and may in fact capture more information about the relative certainty of respondents' moral decisions. Surveys may not be even necessary; one potential application of this approach is to search an external knowledge base, such as a body of law, for heuristics relevant to a given moral dilemma and write heuristics using those legal principles. Ethical recommendations, commonly carefully constructed and released by watchdogs, also constitute a fruitful source of moral heuristics [Luetge, 2017].

Theoretically, this approach is not limited to cheaply replicating the preferences of experts, scholars, lawyers, and crowds. Heuristics are especially useful for breaking down complex problems into sub-problems that can be solved with shortcuts or approximations. As a proof-of-concept, this investigation was limited to domains where moral preference data already exists. I hope that future work will attempt to elicit and validate heuristics from domain experts to solve a real-world moral dilemma in more complex domains without easily measured moral preferences. In these domains, the heuristic approach may be the only viable framework for making moral decisions.

However, there are still important problems to solve before moral decision-making algorithms are truly reliable. First, the problem of representation is ignored; heuristic functions can only operate on a set of measurable, observable features selected by domain engineers to have moral status. This framework mitigates the representation problem by relying on experts, who may be informed enough to handle extremely detailed feature sets, but representations may still aggressively reduce the dilemma at hand. Hierarchical feature representations might provide a useful interface for combining the expertise of individuals in different fields of a particular domain; an expert might be able to specify very granular heuristics for one set of features, but may only be able to express broad heuristics for less-familiar areas of the feature space.

Second, experts must be selected carefully; issues of selection bias and small sample sizes could create poorly specified heuristic functions or systematically disadvantage underrepresented groups. For example, the best-performing heuristic writers in a Snorkel workshop were individuals with an M.S. or a Ph.D. and strong Python coding skills [Ratner et al., 2017]. To incorporate multiple perspectives, some level of translation between domain experts without a high level of education and coding skill is required. Oversight is also important for accountability; future work should focus on designing human-in-the-loop workflows for interpreting moral decisions and adjusting heuristics accordingly. In general, the risk of systematic error increases as fewer perspectives are consulted; but crowd-sourcing is not the only way to solve this problem, as I demonstrate with the ranked-choice experiment. Crucially, this framework depends wholly on the moral expertise of the heuristic writers and their ability to represent the interests of stakeholders in the domain. In other words, if a crowdsourcing approach is democratic and a "top-down" approach is totalitarian, then this approach relies on a representative democracy to reach acceptable moral outcomes, with all the ensuing trade-offs.

Perhaps the most useful avenue of future work is the incorporation of moral heuristics with amoral or domain-specific optimization objectives. This investigation deals only with isolated moral dilemmas, in which the moral decision has been isolated from other optimizations in the problem. (For instance, in the kidney exchange, moral decisions are made only in the case of a tie; moral factors are not considered in the rest of the matching algorithm.) But because this framework operates at the labeling stage, it could be integrated as a secondary or superseding label ground-truth during training. With a set of moral heuristics, an autonomous vehicle might act differently when it encounters a deer in the road than it would if it encounters some inanimate obstacle.

7 Conclusion

Heuristics provide a means to specify ethical positions for especially complex, high-dimensional dilemmas and allow analysis of more complicated quandaries. By lowering costs and adding domain expertise, this framework dramatically lowers the barriers to incorporating ethical principles in practical applications. Moreover, weak supervision paves the way to a ubiquitous method for instilling ethical principles in learning algorithms.

References

Moral Machine. URL http://moralmachine.mit.edu/.

Snorkel. URL https://www.snorkel.org/.

M. Anderson and S. L. Anderson. GenEth: A General Ethical Dilemma Analyzer. In *Proceedings of the National Conference on Artificial Intelligence*, 2014. URL www.aaai.org.

E. Awad, S. Dsouza, R. Kim, J. Schulz, J. Henrich, A. Shariff, J. F. Bonnefon, and I. Rahwan. The Moral Machine experiment. *Nature*, 563(7729):59–64, 11 2018. ISSN 14764687. doi: 10.1038/s41586-018-0637-6.

E. Awad, S. Dsouza, A. Shariff, I. Rahwan, and J.-F. Bonnefon. Universals and variations in moral decisions made in 42 countries by 70,000 participants. *Proceedings of the National Academy of Sciences*, 117(5):2332–2337, 2020. ISSN 0027-8424. doi: 10.1073/pnas.1911517117. URL https://www.pnas.org/content/117/5/2332.

S. H. Bach, B. He, A. Ratner, and C. Ré. Learning the Structure of Generative Models without Labeled Data. *34th International Conference on Machine Learning, ICML 2017*, 1:434–449, 3 2017. URL http://arxiv.org/abs/1703.00854.

P. Bello and S. Bringsjord. On How to Build a Moral Machine. *Topoi*, 32(2):251–266, 10 2013. ISSN 01677411. doi: 10.1007/s11245-012-9129-8.

J. A. Blass and K. D. Forbus. Moral Decision-Making by Analogy: Generalizations vs. Exemplars. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pages 501–507, Austin, 2015. URL www.aaai.org.

K. Bogosian. Implementation of Moral Uncertainty in Intelligent Machines. *Minds and Machines*, 27, 2017. doi: 10.1007/s11023-017-9448-z. URL https://doi.org/10.1007/s11023-017-9448-z.

J. F. Bonnefon, A. Shariff, and I. Rahwan. The social dilemma of autonomous vehicles. *Science*, 352(6293):1573–1576, 6 2016. ISSN 10959203. doi: 10.1126/science.aaf2654.

D. Bourget and D. J. Chalmers. What do philosophers believe? *Philosophical Studies*, 170(3): 465–500, 2014. ISSN 15730883. doi: 10.1007/s11098-013-0259-7.

N. Cointe, G. Bonnet, and O. Boissier. Ethical Judgment of Agents' Behaviors in Multi-Agent Systems. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, pages 1106–1114, 2016. URL `www.ifaamas.org`.

V. Conitzer, W. Sinnott-Armstrong, J. S. Borg, Y. Deng, and M. Kramer. Moral Decision Making Frameworks for Artificial Intelligence. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, pages 4831–4835. AAAI Press, 2017.

M. Dehghani, E. Tomai, K. Forbus, R. Iliev, and M. Klenk. MoralDM: A Computational Modal of Moral Decision-Making. In *Proceedings of the 30th Annual Conference of the Cognitive Science Society*, 2008a.

M. Dehghani, E. Tomai, K. Forbus, and M. Klenk. An Integrated Reasoning Approach to Moral Decision-Making. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*, pages 1280–1286, 2008b. URL `www.aaai.org`.

M. Du, N. Liu, and X. Hu. Techniques for Interpretable Machine Learning. *Communications of the ACM*, 63(1):68–77, 7 2018. URL `http://arxiv.org/abs/1808.00033`.

C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel. Fairness Through Awareness. *ITCS 2012 - Innovations in Theoretical Computer Science Conference*, pages 214–226, 4 2011. URL `http://arxiv.org/abs/1104.3913`.

R. Freedman, J. S. Borg, W. Sinnott-Armstrong, J. P. Dickerson, and V. Conitzer. Adapting a kidney exchange algorithm to align with human values. *Artificial Intelligence*, page 103261, 2020.

P. S. Greenspan. Moral Dilemmas and Guilt. *Philosophical Studies: An International Journal for Philosophy in the Analytic Tradition*, 43(1):117–125, 1983. URL `https://www.jstor.org/stable/4319577`.

G. E. Hinton. Training Products of Experts by Minimizing Contrastive Divergence. *Neural Computation*, 14(8):1771–1800, 2002.

A. Kahng, M. K. Lee, R. Noothigattu, A. Procaccia, and C.-A. Psomas. Statistical Foundations of Virtual Democracy. In *International Conference on Machine Learning*, pages 3173–3182, 2019.

R. Kim, M. Kleiman-Weiner, A. Abeliuk, E. Awad, S. Dsouza, J. B. Tenenbaum, and I. Rahwan. A Computational Model of Commonsense Moral Decision Making. In *AIES 2018 - Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 197–203. Association for Computing Machinery, Inc, 12 2018. ISBN 9781450360128. doi: 10.1145/3278721.3278770.

C. Luetge. The German Ethics Code for Automated and Connected Driving, 12 2017. ISSN 22105441.

W. Macaskill. Normative Uncertainty as a Voting Problem. *Mind*, 125(500):967–1004, 2016. doi: 10.1093/mind/fzv169. URL `https://academic.oup.com/mind/article-abstract/125/500/967/2277457`.

A. Martinho, M. Kroesen, and C. Chorus. An Empirical Approach to Capture Moral Uncertainty in AI. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, page 101, 2020. ISBN 9781450371100. doi: 10.1145/3375627.3375805. URL `https://doi.org/10.1145/3375627.3375805`.

J. H. Moor. What is Computer Ethics? *Metaphilosophy*, 16(4):266–275, 1985. ISSN 14679973. doi: 10.1111/j.1467-9973.1985.tb00173.x.

R. Noothigattu, S. S. Gaikwad, E. Awad, S. Dsouza, I. Rahwan, P. Ravikumar, and A. D. Procaccia. A voting-based system for ethical decision making. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

G. Paolacci, J. Chandler, P. G. Ipeirotis, and L. N. Stern. Running experiments on Amazon Mechanical Turk. *Judgment and Decision Making*, 5(5):XX–XX, 2010.

A. Ratner, S. H. Bach, H. Ehrenberg, J. Fries, S. Wu, and C. Ré. Snorkel: Rapid training data creation with weak supervision. In *Proceedings of the VLDB Endowment*, volume 11, pages 269–282. Association for Computing Machinery, 11 2017. doi: 10.14778/3157794.3157797.

C. Shulman, H. Jonsson, and N. Tarleton. Which Consequentialism? Machine Ethics and Moral Divergence. In C. Reynolds and A. Cassinelli, editors, *AP-CAP 2009: The Fifth Asia-Pacific Computing and Philosophy Conference, October 1st-2nd, University of Tokyo, Japan, Proceedings*, pages 23–25. AP-CAP, 2009. URL `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.363.2419`.

W. Sinnott-Armstrong. *Moral Dilemmas*. Philosophical Theory. Basil Blackwell, Oxford, 1988. ISBN 0631157085.

H. Yu, Z. Shen, C. Miao, C. Leung, V. R. Lesser, and Q. Yang. Building Ethics into Artificial Intelligence. 12 2018. URL `http://arxiv.org/abs/1812.02953`.