Thesis Thoughts/Intro/Ideas

The current state of AI/technology indicates the need to imbue machines with the ability to make moral decisions. Self driving cars provide an excellent domain to study – they are already on our streets, yet we as a society have not taken any significant steps towards deciding how they will handle moral dilemmas, such as the "trolley car" problem.

There are many ethical and moral issues associated with programming self driving cars to make moral decisions, the obvious one being "what decisions should they make when"? However, there are also some non-obvious yet non-trivial ethical issues here.

- Who gets to program the cars? Is it an individual, a private company, a government?
- It is well-known that algorithms can exhibit biases (for example, racial). Should/how will algorithms account for this bias?
- What degree of ownership will society have over the programs/decision-making algorithms in these cars?
- Should people have a right to voice their inputs for self-driving car (and other moral AI) programs? Should these programs be designed by researchers/governments to maximize social utility, or, should people be able to"vote" on the moral preferences?

One way to implement a moral AI would be to try to somehow abstract or understand moral features/principles, and build an algorithm that uses this information to reason morally. For instance, you might survey people on which moral features they feel indicate whether a decision is right or wrong, and then build an algorithm that, using those features, makes decisions given new inputs. There are some potential problems with this approach:

- Voter/citizen distrust – people are less likely to support/adopt a system they do not understand (think: machines taking over the world), and a system that abstracts moral values may be harder to understand.
- Consistency issues – say we survey people on a problem X and use the results of that survey to build a new system, which then decides on problem Y. Is that the same as the majority vote decision the people would have made if they had originally been asked about problem Y?

One potential solution to this problem is via voting. Consider this: what if, every time a moral decision needed to be made, we conducted a vote, asking each person what action should be taken, and using a voting algorithm to decide what to do. This would seem to solve both problems – people are (generally) likely to trust systems in which votes make decisions (our current government) and the consistency issue is avoided. This also seems to be grounded in current democratic philosophy in which votes are used to make decisions.

Of course, it is not possible to conduct a vote each time a decision needs to be made (due to potentially > millions of machines making many decisions all the time). We can, however, conceive of programs that can cast votes for people. For instance, imagine a moral problem with well-defined features. We could have a classifier that, for each person, learns their moral preferences and predicts their view on the moral problem. Then, when we need to make a decision, we could query all the classifiers, treat them as votes, and aggregate them into one decision.

Already this is an interesting problem for its applications to moral AI decision making and society-in-the-loop computing (as discussed by Iyad Rahwan). However, it is not clear that an algorithm will always have the time to evaluate all classifiers. Situations on the road are dynamic – it seems that sometimes, a self-driving car will have very little time with which to make a decision. It seems that we need to develop methods by which computers can either efficiently evaluate all classifiers, or approximate the result of evaluating all classifiers with high probability. A first attempt to formalize this problem is as follows:

First, we need some way of defining what a moral problem is.

Definition 0.1. Moral Problem A moral problem $\mathfrak{m}$ consists of a domain in which in which a problem can occur, a set of features $\mathfrak{F}$ on which the problem can be evaluated, and a set of decisions $\mathfrak{D}$ that can be made to resolve the problem.

Definition 0.2. Moral Problem Space A moral problem space $\mathcal{M}$ is the set of all moral problems that share domains, feature sets, and decision sets. Since this is the case we can refer to feature set and decision set of a moral problem space as well.

Definition 0.3. Decision Function Let $\mathcal{M}$ be a moral problem space. Then $F_M : \mathfrak{F} \to \mathfrak{D}$ is called a decision function and maps from the feature set of $\mathcal{M}$ to the decision set of $\mathcal{M}$.

Let's flesh this out by looking at an example of self-driving cars choosing whether to drive straight or swerve to avoid something in the road. In this instance, we can call our domain "self-driving cars", and $\mathfrak{D}$ consists of two options: drive straight, or swerve. What does $\mathfrak{F}$ consist of? This is highly dependent on what information we are able to gather. In one instance, $\mathfrak{F}$ might just consist of the number of people in front of the car, and the number of people to the side of the car. On the other hand, we could envision a more complex scenario where $\mathfrak{F}$ contains

the number of people who will be hit if a car goes straight, the number of people who will be hit if a car swerves, probabilities of hitting the groups of people, estimated age, estimated race, culpability details (did someone dash into traffic), etc. We could take this one step further – already, half of U.S. citizens have their faces in law enforcement facial recognition databases. If these could be used to lookup details, we can imagine a situation where a car is able to evaluate a decision based on all sorts of information – criminal background, disease profile, religion, income, job – the possible feature set would be almost limitless.

Since the definition of $\mathcal{M}$ depends on the feature set, we have that each feature set leads to a different type of moral problem. Thus, the question of whether a car should swerve given knowledge of potential victims' ages is different than the question of whether a car should swerve not given knowledge of potential victims' ages.

Now we can turn our attention back to the moral classifier aggregation problem. The problem is as follows:

Definition 0.4. Moral Aggregation Problem Let $\mathcal{F}_M$ be the set of all decision functions on $\mathcal{M}$. The goal in the Moral Aggregation Problem is to develop a procedure that somehow aggegrates the results of applying $f \subset \mathcal{F}_M$ to some problem $m \in \mathcal{M}$ according to some set of evaluation measures.

Notes: Need to take into account how certain they are, how certain we are of them.

Use nearest neighbors?

Think about when people just don't know, or do know, or have conflicts. There are many ways this could happen:

- If all decision functions in $\mathcal{F}_M$ are of the same form (ie all decision trees, all SVM, all hyperplanes), then we could exploit the structure via preprocessing to quickly query all functions.
- We could use random sampling to
- More narrowly define problems, such as random sampling
- bias variance tradeoffs in sampling
- time tradeoffs in sampling as well
- if you know how long each algo will take, you can plan / knapsack problem
- probability you will get the wrong decision, minimize that

- look at binary decision.
- distribution over time that an algo will take, you may not know how long (but maybe postpone this)
- incentives to speed up classifier? does it make it less expressive?
- correlations between speed and accuracy
- MAKE ASSUMPTIONS ABOUT WHAT SEems reasonable, make a model/objective from there.
- tradeoff between strategy and speed.
- strategy only makes sense when people code up their own algorithms.
- don't want to bias towards people with simplistic moral tendencies.
- what to do when there is a correlation between speed and correctness? No correlation – query all fast functions. If you don't know, maybe query some mostly fast ones, and a few slow ones?

1 Probabilistic Framework

Let Y represent the function that, given a situation, represents the correct moral decision to make. I assume we can predict Y via an aggregation of decision functions, so $Y = f(X_q) + \epsilon(X)$. In this model, I do not assume each voting function has the same noise value.