Fairness & Abstraction in Sociotechnical Systems

CS598 Paper Presentation

Yuxin Xiao

04/12/2019

Introduction

- Abstraction: ML Systems – Black boxes
  - Defined by inputs, outputs & relationship between them
  - Internals of the system & provenance of the inputs and outputs are abstracted away
- Fairness-aware Machine Learning (fair-ML) without social context misses the broader picture
  - Social Context + Technical Tools
- 5 traps due to this abstraction error
- Science and Technology Studies (STS) – sociotechnical systems (a combination of technical and social components)
  - Solutions-oriented approach -> process-oriented approach
  - Draws the boundary of abstraction to include social actors, institutions, and interactions

Abstraction Trap 1 – Framing Trap

- Failure to model the entire system over which a social criterion, such as fairness, will be enforced
- Algorithmic Frame: representations of data & labeling of outcomes
  - Efficacy: properties of the output as related to the input
  - Abstraction taken as given, notion of “fairness” not defined
- Data Frame: algorithm + its inputs & outputs
  - Fair-ML algorithms investigate ways in which the choices of representations and labels might affect the resulting model
  - Seeks to approximate a socially desirable goal e.g. fairness
- Sociotechnical Frame: include human decisions within the abstraction boundary
  - ML model is part of a sociotechnical system, other components of the system needed to be modeled
  - System fairness analyzed as an end-to-end property
- Example: Risk Assessment Tool to aid in decision-making across the criminal justice pipeline
  - Failure to account for how judges respond to scores declines fairness guarantees

Abstraction Trap 1 – Framing Trap

- Adopt a “heterogeneous engineering” approach
  - Consider both human and machine activities at the same time
  - Draw the boundaries of abstraction to include people and social systems as well
  - Recognizing which parts of the sociotechnical system are in focus when evaluating for fairness is crucial for communicating the boundaries of the fairness guarantee

- Example: Cell phones
  - Satellites, wireless protocols, batteries, electrical outlets to companies like Apple, regulatory agencies like the FCC, standards setting organizations like the IEEE
  - Categorical Mistake: conceptually separating ML from the social context = company that designs a cell phone without knowledge of data plans, satellites, regulators and so on

Abstraction Trap 2 – Portability Trap

- Failure to understand how repurposing algorithmic solutions designed for one social context may be misleading, inaccurate, or otherwise do harm when applied to a different context
- Portability
  - The problem “enters” the system as data and exits as a prediction
  - Same “solution” appears to be applicable to problems in a variety of social settings regardless of the different social context around these questions
- Making assumptions specific to the social context makes the ML systems not portable between social contexts -> work around a programmer’s core programming
- Two additional notes:
  - Within the same domain, local fairness concerns may be different enough that systems do not transfer well between them
  - Frameworks like domain adaptation and transfer learning are not sufficiently expressive to capture the vast changes in social context between domains

Abstraction Trap 2 – Portability Trap

- Contextualizing user “scripts”
  - User "scripts" that dictate how technologies are supposed to be used only work if all the social and technical elements of a network are assembled properly
  - Example: light bulbs and generators, developed in France, failed in West Africa; engineers did not consider how generators might be shared in rural villages or how electricity was metered and paid for
- Concepts such as “fairness” not tied to specific objects but to specific social contexts
  - Attach the label “fair” to the code -> erroneously encourage others to appropriate this code without understanding how the script changes or is disrupted with a shift in social context

Abstraction Trap 3 – Formalism Trap

- Failure to account for the full meaning of social concepts such as fairness, which can be procedural, contextual, and contestable, and cannot be resolved through mathematical formalisms
- Limiting the question of fairness to a mathematical formulation cannot capture its full range of similar and overlapping notions in philosophical, legal, and sociological contexts
- First, no way to arbitrate between irreconcilably conflicting definitions using purely mathematical means
  - Automated resume screening: less concerned with FNs than FPs; Criminal justice context: concerned about equalizing FPs
- Second, no definition might be a valid way of describing fairness
  - Procedurality: fair-ML definitions are primarily outcome-based; disparate impact is procedural
  - Contextuality: discrimination can only be comprehended with access to situated cultural knowledge
  - Contestability: discrimination and fairness are politically contested and shifting

Abstraction Trap 3 – Formalism Trap

- Identifying “interpretive flexibility”, “relevant social groups”, and “closure”
- Social Construction of Technology program (SCOT)
  - How technology is developed, made sense of, and adopted in social contexts, with human users at the forefront
  - Interpretive flexibility experienced by relevant social groups, stabilization, and closure
- Social groups have the power to shape technological development
  - Different interpretations emerge, each advanced by a relevant social group: a group in society that has a specific idea of what problems the technology needs to solve
  - Groups who become "relevant" simply by means of their existing relationships to fair-ML researchers or by means of an existing voice in society
- Rhetorical closure
  - The relevant social groups describe the problem as solved, and move on
  - Problems associated with formalism are related to underlying assumptions about who can solve the problems of fairness and how, which other problems must be solved, and which social groups are deemed relevant in the process

Abstraction Trap 4 – Ripple Effect Trap

- Failure to understand how the insertion of technology into an existing social system changes the behaviors and embedded values of the pre-existing system

- Unintended consequences due to the insertion of technology:
  - How people respond to the technology and how technology interacts with a pre-existing social system
  - Example: how risk assessment scores are initially used might differ from what happens when judges see them frequently
  - The effects of the technology will change based on the particulars of the social context

- Technologies can also alter the underlying social values and incentives embedded in the social system
  - New tools may unconsciously privilege quantifiable metrics

Abstraction Trap 4 – Ripple Effect Trap

- Avoiding “reinforcement politics” and “reactivity”
- Awareness of several common changes avoids common pitfalls that may negatively affect the fairness of their proposed systems
- Reinforcement politics
  - Existing groups use the occasion of this new technology to reinforce or argue for power and position
  - Example: management purchases software for monitoring or otherwise controlling subordinate groups in an organization
- Reactivity behaviors
  - Alter the very social context that the original design was meant to support
- Heterogeneous engineers:
  - Once a technology is part of the social context, new relevant social groups can arise and radically reinterpret it, return it to a state of interpretive flexibility, and suggest new mechanisms for closure
  - Cannot completely eliminate unintended consequences, but considering key choices in a technology’s development can go a long way toward controlling ripple effects and even detecting trouble spots in advance

Abstraction Trap 5 – Solutionism Trap

- Failure to recognize the possibility that the best solution to a problem may not involve technology

- If you have a hammer, everything looks like a nail - by starting from the technology and working outwards, there is never an opportunity to evaluate whether the technology should be built in the first place

- Fairness definitions can be politically contested or shifting, a model may not be able to capture how it moves
- The modeling required could be so complex as to be computationally intractable

- Must understand the existing social system
  - When there is not enough information to understand everything that is important to a context, approximations are as likely to make things worse as better

Abstraction Trap 5 – Solutionism Trap

- Considering when to design
- Careful consideration of the complex sociotechnical system at play
- Cooperation between fair-ML researchers and domain experts
- Not all problems can or should be solved with technology

Takeaways

- When designing a new fair-ML solution, determine
  - If a technical solution is appropriate to the situation in the first place, which requires a nuanced understanding of the relevant social context and its politics (Solutionism)
    - whether developing a risk assessment is appropriate given the current societal goal of reducing pre-trial detention
    - compare a risk assessment proposal against not only other possible algorithmic solutions, but also the existing human processes
  - If a technical solution affects the social context in a predictable way such that the problem that the technology solves remains unchanged after its introduction (Ripple Effect)
    - understand that introducing the risk assessments may alter the embedded values
    - predicting dangerousness may lead to a focus away from other societal goals such as rehabilitation
    - attempt to account for unintended consequences

Takeaways

- When designing a new fair-ML solution, determine
  - If a technical solution can appropriately handle robust understandings of social requirements such as fairness, including the need for procedurality, contextuality, and contestability (Formalism)
    - Concerns about relevant social groups can be obviated by taking seriously the needs of people typically underrepresented in these processes
    - understand the power dynamics that prevent these voices from having influence in society to begin with
    - Work with advocacy organizations, social scientists, or the population in question
  - If a technical solution has appropriately modeled the social and technical requirements of the actual context in which it will be deployed (Portability)
    - When transferring an algorithm designed to predict good hires to the context of risk assessment, any assumptions built into the algorithm or the fairness definition used could render such an algorithm inappropriate to the risk assessment context
  - If a technical solution is heterogeneously framed so as to include the data and social actors relevant to the localized question of fairness (Framing)
    - What particular decision the fairness criteria apply to: detain-or-release decision VS dangerousness determination

Thank you!