Designing Ethics Automation Projects to Increase the Odds of Success

The Second Workshop on Implementing Machine Ethics
University College Dublin
Steven Greidinger, MS, MPM
June 30th, 2020
Seeking collaborations
Steve’s Experience

Dot-com data manager

Headquarters staff, US National Science Foundation and National Institutes of Health

Machine Intelligence Research Institute

Stanford/SRI International: AI ethics and other projects

Enjoys teaching, designing in teams
What is Ethics Automation?

Development of software systems that perform ethical reasoning in a potentially generalizable way

A subset of the study of machine ethics

Eventually, ethics automation technology will be used in decision aids or as part of software agents, chatbots, and autonomous systems.
What Ethics Automation is Not

A solution to tragedies caused by error (such as a police shooting by reflex)
Algorithmic debiasing
Case-by-case hard coding or one-off models
Building a general intelligence that itself has ethical rights
Utilitarian Ethics Automation Approach

Utilitarianism’s non-hedonic, public policy formulation successors

   Rule utilitarianism for many, less important decisions

   Act utilitarianism for individual, important decisions
Ethics Automation Task Types

Framework set-up, development and administration

Encoding descriptions of the world and entities deserving ethical treatment

Embedding ethical reasoning in human decision aids, ethics audit or autonomous systems
Community and Collaboration

Setting an education mission may sometimes better justify the budget than the research itself.

Sharing cloud AI infrastructure with other related or unrelated groups may reduce or eliminate costs.
Is There a Sponsor or Client?

Do you accept enough of their goals and approach to ethics?

Technical success is not practical success.

Practical success is not gaining long-term adoption.

Clients may be “internal champions.” Likelihood they stay?

Negotiate a performance metric and show regular progress.
Project Management Methods
Requirements spec, GANTT chart, change management, test
Budget, task dependencies, resources, success rates
Software Development Lifecycle, Agile/Scrum?
Quickly create something to show, then another
3-5 year development blocks
Tools and Environment

Professional scientific software development environment:

- Shared cloud for geographically dispersed teams
- Python, Git, integration software
- Databases (RDBMS, GraphDB, NoSQL)
- AI Resources (Theorem Provers, Optimization, Bayes’, ML)

Concept encoding method
Getting uncorrupted ethics automation used at all is the most important problem.

Steven Greidinger

4sjgcombined@gmail.com

1-202-445-4247 (US)