1 DESIGN IMPLICATIONS

We've discussed the demonetization of videos, the management of crowd work, and the bias of algorithmic justice, but the same undercurrent moves all of these cases. The inability of artificial intelligences to *reflect* on their goals appears as the myriad symptoms of AI's underlying problem. A human has the capacity to recognize the substantive injustice of a judicial system that targets and disenfranchises people of color; an algorithm can only see a pattern. Not a good or bad pattern — just a pattern. And even the goodness or badness of that pattern must itself be taught.

What should designers of street-level algorithms do? The question we hope to answer is how to find and identify circumstances for which algorithmic systems would not yield problematic outcomes. The defining goal, then, should be to identify unique or special cases requiring discretion and flexibility. In some cases that will be easy — some classification's confidence will be low, or the result will be ambiguous in some predictable, measurable way. In much the way we already do, we should divert those cases to human bureaucrats. However, often the system performs these erroneous classifications with high confidence, because it does not recognize that the uniqueness of the input is different than other unique tokens or inputs.

Lipsky argues that street–level bureaucrats must exercise reflexivity, recognizing the underlying purpose of the tasks at hand, to be effective. If this is the substantive goal of designing effective street–level algorithms, then we need to figure out how to get there. Today's most advanced AIs cannot reflect on the purpose or meaning of the tasks for which they're optimizing results. We turn, then, to Lipsky's work yet again, where he argues that appeals and recourse — and the ability for people to recognize the uniqueness of a situation when it's explained to them — are necessary features of street–level bureaucrats who fail to recognize the marginality of a case at hand the first time around. Developing more robust mechanisms for recourse may be the path to sufficient, if not yet effective, street–level algorithms.

We argue that system–designers need to develop ways for people to get *recourse* if the system makes a mistake, much like citizens of a bureaucratic institution can mitigate harm due to mistakes. A necessary assumption is that designers want to create a prosocial, fair system. Our theory suggests to look to best practices in bureaucratic design as inspiration. Crafting a fair appeals process is one common lever: for example ensuring that the person reviewing any appeal does not have conflicting interests or misaligned incentives—or in this case, perhaps not the same software developers or machine learning model as the system that made the original decision. Another approach is a predefined process for recourse, for example compensating lost income. Finally, since bureaucracies can be as opaque as algorithms, many bureaucracies are required by law or design to publish materials describing peoples rights in plain language.

Recourse and appeals require grounds for the individual to understand precisely where and how the system made a mistake. How, for instance, can a person prove that they have been misjudged by the algorithm? One solution might be to represent the embeddings generated in classifying that case by showing similar points in the embedding space, suitably anonymized. If, for instance, the classification system figured that the current case was very similar to a number of other cases, presenting the user's case in the context of some of those closely–aligned cases can give the user sufficient context to articulate why their situation is marginal relative to the projection the system has of it.

For example, YouTube's demonetization system could communicate its judgments about videos to YouTubers, giving those performers an opportunity to assert that they've been misjudged. If a YouTuber uploads a video discussing LGBTQ issues, and the system thinks that content is sexually explicit content, it might present a handful of similar videos nearby in the embedding space. By comparing their video to these comparators, a YouTuber can identify whether they're being misjudged as similar to videos from which they're substantively different. This kind of information is crucial for systems such as these to develop depth in the sense that they understand the subtle difference between "sexually explicit" and "explicitly sexual" content, for instance.

Bail recommendation systems could offer similar insights to stakeholders and help both judges and defendants better understand the intuition the algorithmic system has developed. A judge might see that the embeddings generated for a defendant are categorically wrong in some way that the algorithm either doesn't understand or can't measure. In this kind of scenario, information about the other defendants might be sensitive for various reasons, but some representation of the present case and its neighbors in the embedding space can substantively reveal whether a categorical error has been made.

These examples may prove untenable as a result of any number of circumstances, but by sparking a conversation along this dimension — one that recognizes that the decisions of designers manifest in their systems as features of an emergent bureaucracy which billions of people may have to navigate — we hope to encourage reasoning about these problems along the same general lines that political scientists have been thinking for the better part of half a century. In doing so, we may be able to find and leverage stopgap solutions and processes — developed and improved by social scientists over decades — that mitigate some of the harms done to marginalized communities.

The design implications to this point have focused on individual-level recourse—what about institution-level checks? In other words, how can designers of these algorithmic systems ensure that the systems will self-police? Our approach seeks grounding in the literature via the mechanisms that police bureaucracies to identify possible design directions. Bureaucracies experience oversight via a number of channels, including internal audits, external review, and the judicial system. These methods are variably effective, which is to say that public administration has no panacea to policing and oversight. However, these models have worked and likely will continue to work as ways that designers can build oversight into algorithms, for example peer juries of disruptive behavior online [1]. This approach provides us with a structure for reasoning about algorithmic systems with the added benefit of decades of theoretical refinement.

The more distant question, of course, is what needs to change before algorithms can reliably subsume the roles of street–level bureaucrats as Lipsky described. Lipsky argues that programs will never be able to take the roles of human beings: *"the essence of street–level bureaucrats is [that] they cannot be programmed"* [2], because they can't think deeply about the circumstances of the case in front of them or their role in the decision being rendered. That certainly is true today, but advances in AI since Lipsky's time may have surprised even him. It may be possible that algorithmically intelligent systems will reach a point where they can believably take up the roles of human bureaucrats; what they will need to demonstrate is some capacity to reflect on the novelty of novel or unusual cases, and what the implications of their decisions might be *before* a decision is made.

REFERENCES

[1] Yubo Kou, Xinning Gui, Shaozeng Zhang, and Bonnie Nardi. 2017. Managing Disruptive Behavior Through Non-Hierarchical Governance: Crowdsourcing in League of Legends and Weibo. *Proc. ACM Hum.-Comput. Interact.* 1, CSCW, Article 62 (Dec. 2017), 17 pages. https://doi.org/10.1145/3134697

[2] Michael Lipsky. 1983. *Street-Level Bureaucracy: The Dilemmas of the Individual in Public Service.* Russell Sage Foundation.