Explainable Machine Learning for Public Policy: Use Cases, Gaps, and Research Directions

KASUN AMARASINGHE, KIT RODOLFA, HEMANK LAMBA, RAYID GHANI, Machine Learning Department and Heinz College of Information Systems and Public Policy, Carnegie Mellon University

In Machine Learning (ML) models used for supporting decisions in high-stakes domains such as public policy, explainability is crucial for adoption and effectiveness. While the field of explainable ML has expanded in recent years, much of this work does not take real-world needs into account. A majority of proposed methods use benchmark ML problems with *generic* explainability goals without clear use-cases or intended end-users. As a result, the effectiveness of this large body of theoretical and methodological work on real-world applications is unclear. This paper focuses on filling this void for the domain of public policy. We develop a taxonomy of explainability use-cases within public policy problems; for each use-case, we define the end-users of explanations and the specific goals explainability has to fulfil; third, we map existing work to these use-cases, identify gaps, and propose research directions to fill those gaps in order to have practical policy impact through ML.

CCS Concepts: • **Applied computing** → **Computing in government**; • **Human-centered computing** → *Interaction design*; • **Computing methodologies** → Machine learning.

Additional Key Words and Phrases: Explainability, Interpretability, Public Policy

ACM Reference Format:
Kasun Amarasinghe, Kit Rodolfa, Hemank Lamba, Rayid Ghani. 2020. Explainable Machine Learning for Public Policy: Use Cases, Gaps, and Research Directions. 1, 1 (October 2020), 14 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

The proliferation of Machine Learning (ML) applications has extended to decision-support systems in public policy areas such as criminal justice, education, healthcare, and social services [7, 11, 33, 38, 47]. As users of these methods have grown beyond ML experts and the research community, the desire to better interpret and understand them has grown as well, particularly in the context of high-stakes decisions that affect individuals' health or well-being [22, 23, 39]. Likewise, new legal frameworks reflecting these needs are beginning to emerge, such as the *right to explanation* in the European Union's General Data Protection Regulation [15].

Against this background, research into *explainability/ interpretability*[1] of ML models has experienced rapid expansion and innovation in recent years. Several methods have been developed, broadly falling into two categories: 1) directly interpretable models [11, 22, 39, 45], and 2) post-hoc methods for explaining (opaque) complex models and/or their predictions [5, 28–30, 36, 37]. Despite the extensive and promising body of methodological work, we argue here that further effort is necessary to connect these methods to settings in which they will be deployed. Methods are often developed with a broad and loosely-defined goal of "explainability" rather than to meet specific needs of real-world

[1]It is worth noting that in this paper, we do not distinguish between the two terms *interpretability* and *explainability*. We use both terms to refer to the ability to accompany ML models and their predictions with information about the model's decision-making process.

Author's address: Kasun Amarasinghe, Kit Rodolfa, Hemank Lamba, Rayid Ghani, {amarasinghek,krodolfa,hlamba,rayid}@cmu.edu, Machine Learning Department and Heinz College of Information Systems and Public Policy, Carnegie Mellon University.

use-cases (such as promoting trust and adoption versus better informing decision-makers acting on a model's output). Nor are these methods routinely evaluated to adequately reflect how they might be put into use in real-world settings. Barring a few exceptions [11, 30, 44], much of the existing work has been designed and developed for benchmark classification problems with synthetic data and validated with user studies limited to users in research settings such as Amazon Mechanical Turk [3, 5, 17, 29, 32, 36, 41, 48].

Here, we focus on applications of ML to public policy settings and seek to provide a framework that can help bridge the gap between methodological work in explainable ML and real-world applications, guiding future research to better build and evaluate these methods. When developing explainable ML systems for a particular domain, it is crucial to define the explanation goals and the targeted users to identify the requirements that explanations should satisfy to be effective in meeting the said goals [8, 46]. Moreover, to bridge the gap between existing work and the needs of the domain, it is vital to identify the extent to which the available body of work in explainable ML matches the needs of the problem-domain and propose directions for future work. To that end, this paper offers the following contributions:

(1) Identifying a taxonomy and specific use-cases for explainability in public policy applications of ML
(2) Identifying the explanation goals, the end-users, and the explanation needs, for each use-case
(3) Identifying research gaps by comparing the existing body of work to the needs of the domain
(4) Proposing research directions to develop effective explainable ML systems for improved decisions in public policy settings

2 USE OF MACHINE LEARNING IN PUBLIC POLICY

To illustrate the applicability of ML to policy problems, we focus on the common task of early warning systems. In an early warning system, the ML model is used to identify entities for some intervention, based on a predicted risk of some (often adverse) outcome, such as an individual being booked into jail in the next year, a student not graduating on time, or a child getting lead poisoning in the next year [7, 38, 47]. While there are several other policy problem templates that ML is used for, such as inspection targeting, scheduling, routing, and policy evaluation, we use early warning systems to illustrate our ideas in this paper.

2.1 Characteristics of ML applications in public policy

Several characteristics of typical public policy problems set them apart from standard benchmark ML problems:

Non-stationary environments. In a policy context, ML models use data about historical events to predict the likelihood of either the occurrence of an event in the future or the existence of a present need, and the context around the problem changes over time. This non-stationary nature in the data introduces strong temporal dependencies that should be considered throughout the modeling pipeline and makes these models susceptible to errors such as data leakage. For instance, standard k-fold cross-validation might create training sets with information from the future, which would not have been available at model training time.

Evaluation metrics reflect real-world resource constraints. The mental health outreach in [7] was limited by staffing capacity to intervene on only 200 individuals at a time, and the rental inspections team in [47] could only inspect around 300 buildings per month. Resource constraints such as these are inherent in policy contexts, and the metrics used to evaluate and select models should reflect the deployment context. As such, these applications fall into the *top-k* setting, where the task involves selecting exactly k instances as the "positive" class [24]. In such a setting, we are concerned with selecting models that work well for precision in the top $k\%$ of predicted scores [9] rather than optimizing accuracy or AUC-ROC (as in "standard" classification problems), which would be sub-optimal.

Heterogeneous data sources with strong spatiotemporal patterns. Developing a feature set that adequately represents individuals in policy applications typically entails combining several heterogeneous data sources, often introducing complex correlation structures to the feature space not usually encountered in ML problems used in research settings. For instance, in [7], the ML model combines data sources such as criminal justice data (jail bookings), emergency medical services data (ambulance dispatches), and mental health data (electronic case files) to gain a meaningful picture of an individual's state. Additionally, temporal patterns in the data are often particularly instructive, requiring further expansion of the feature space to capture the variability of features across time (number of jail bookings in the last six months, 12 months, and five years). Together, the combination of features across a range of domains, geographies, and time frames yields a large (and densely-populated) feature space compared to typical structural data based ML problems we encounter in research settings.

2.2 Socio-technical systems

Typical ML supported public policy decision making systems have at least four types of users that interact with ML models in public policy applications:

(1) **ML system developers** who build the ML components of the system.
(2) **High-level decision-makers/regulators** who determine whether to use/incorporate the ML models in the decision making process or are responsible for auditing the ML models to ensure intended policy outcomes.
(3) **Action-takers** (e.g., social workers, health workers, employment counselors) who act and intervene based on the recommendation of the model. Most policy applications of ML do not involve fully automated decision-making, but rather a combined system of ML model and action-taker that we consider as one decision making entity here. Action-takers often make two types of decisions: deciding whether to accept/override the model prediction for a given entity *(whether to intervene)*, and deciding which intervention to select in each case *(how to intervene)*.
(4) **Affected individuals** that are impacted by the decisions made by the combined human-ML system.

3 WHAT ROLE CAN EXPLAINABLE ML PLAY IN PUBLIC POLICY APPLICATIONS?

Based on our experience working in hundreds of such problems, we identify five main use-cases for explainable ML in a public policy decision-making process (see Table 1). For each use-case, we identify the end-user(s) of the explanations, the goal the explanations need to achieve, and the desired characteristics of the explanations to reach that goal. To better illustrate the use-cases, we will make use of concrete application drawn from our work (preventing adverse interactions between police and the public) to serve as a running example. Numerous applied ML contexts share a similar structure, such as: supporting child welfare screening decisions [12], allocating mental health interventions to reduce recidivism [7, 38], intervening in hospital environments to reduce future complications or readmission [35], recommending training programs to reduce risk of long term unemployment [49].

Illustrative example: Adverse incidents between the public and police officers, such as unjustified use of force or misconduct, can result in deadly harm to citizens, decaying trust in police, and less safety in affected communities. To proactively identify officers at risk for involvement in adverse incidents and prioritize preventative interventions (e.g., counseling, training, adjustments to duties), many police departments make use of Early Intervention Systems (EIS),

Table 1. Use-cases of ML explainability in public policy applications

Use-case	End-user	How the explanation would be used
Model debugging	ML System Developer	Uncover errors/bugs in the ML pipeline/model such as leakage or biases by understanding what patterns the model learned
Trust & adoption	Policymakers, Regulators, & Action takers	Help users understand how the model makes decisions, evaluate its reasonableness, and trust its recommendations
Whether to intervene	Action taker	Help action takers identify correct and unreliable predictions by explaining how the model arrived at individual risk scores
Improving intervention assignments	Action taker	Help action takers select appropriate interventions by understanding factors that contribute to risk
Recourse	Affected individuals	Help affected individuals take action to improve their outcomes in the future or appeal decisions based on inaccurate data

including several ML-based systems [10]. The prediction task of the EIS is to identify k currently active officers who are most at risk of an adverse incident in a given period in the future (in the next 12 months), where intervention capacity of the police department determines k. The EIS uses a combination of data sources such as officer dispatch events; citizen reports of crimes; citations, traffic stops, and arrests; and employee records to represent individual officers and generates labels using their history of adverse incidents [10].

3.1 Use Case 1: Model Debugging

ML model building workflow is often iterative: ML developers build models, analyze them , fix any errors, improve the models, and iterate until they are satisfied with the models and their performance. One critical piece of this workflow is doing continuous sanity checks on the model(s) to see if they *make sense*. A key goal of explanations, at this early stage, is to help the system developer identify and correct errors in their models. Common errors such as data leakage (the model having access to information at training/building time that it would not have at test/deployment/prediction time that is accidentally being used in the training data) [19], and spurious correlations/biases (that exist in training data but do not reflect the deployment context of the model) are often found by observing model explanations and finding predictors that should not be useful showing up as extremely predictive [11, 36].

E.g. In the EIS, an adverse incident gets determined to be *unjustified* quite a long time after the incident date. When training an ML model with the entire incident record, accidentally using the future *determination state* of the incident can introduce data leakage. In this case, explanations could uncover that the feature *case state* is deemed important by the model when it takes a value related to the determination state, and can point the ML practitioner and/or a domain expert to recognize that information has leaked from the future.

3.2 Use Case 2: Building Trust

Decision-makers have to sufficiently trust the ML model to adopt and use them in their processes. Trust, in general, is a common theme behind explainable ML [23, 28, 36]. In our experience, it takes two forms: 1) trust by high-level decision-makers that leads to its adoption in the process, and 2) trust by the action-taker in the model's predictions that leads to individual actions/interventions. This use case focuses on the former, where the goal of explanations is to help users (policymakers) understand and trust the model's overall decision-making process.[2]

The role of the explanation in this case is to both help the users understand what factors are affecting the model predictions, as well as characteristics of individuals that are being scored as high or low risk. Since the user in this instance is not an ML expert but has expertise in the domain being tackled, communicating the explanation in a way that increases the chances of creating trust is critical.

E.g. In the EIS, the explanations should inform the ranking officer at the PD—who acts as the regulator—of the factors that lead to increasing/decreasing a police officer's risk score [10]. In that instance, *"A high number of investigations in the last 15 years"* is an interpretable indicator while *"positive first principal component of arrest data"* is not.

3.3 Use Case 3: Deciding Whether to Intervene

No ML model makes perfect predictions, particularly when dealing with rare events. For example, consider an ML model that predicts children and homes at risk of lead hazards for inspection resource allocation. If only 5% of households have lead hazards, a model that identifies these hazards with a 30% success rate would provide significant improvement over a strategy of performing random inspections, but would still be wrong 70% of the time. In the ideal case, the action-taker would know when to agree with and act on the model's recommendation, and when to override it, and end up with a improved list of k entities. This is closely related to the notion of trust that we discussed in the above use-case, but at the level of individual predictions and with the end-user being the action-taker as opposed to a high level regulator.

Effective explanations can potentially help users, combined with their domain expertise, determine when the model is wrong and improve the overall decisions made by the combined Human-ML system. Therefore, the goal of explanations in this use case is to help the action-taker make the decision of *whether to intervene* by detecting unreliable model predictions so that the performance of the overall system—precision@k in the example above—would improve. For instance, if the explanation indicates that the model is basing the prediction on seemingly non-related factors, they may override that prediction. As the end-users are domain experts, the same user-interpretability requirement from the above use-case holds for the explanations.

E.g. In the EIS, if an explanation exists for each officer in the top-k, that explains *why* they are at risk of an adverse incident, the internal affairs division, who decides *whether to intervene*, can use those explanations to determine the reliability of the model's recommendation in order to act on it or override it.

3.4 Use Case 4: Improving Intervention Selection

While ML models may help identifying entities that need intervention, they provide little to no guidance on how to select from one of many interventions. For instance, consider a model that predicts students' risk of not graduating high school on time. A student might be at risk due to a number of reasons, such as: struggling with a specific course, bullying, transportation issues, health issues, or family obligations. Each of those reasons would require a different type of assistive intervention.

Here, the goal is to help the action-taker determine *how to intervene*. While explanations are not truly causal, the factors deemed important by the ML model can provide valuable information in choosing interventions. As above, the end-users here are domain experts and the explanations should be mapped to the problem domain. As domain expertise

[2]It is important to note that explainability is not the only aspect that affects user trust. In a policy context, factors such as: 1) stability of predictions, 2) training users have received, and 3) user involvement in the modeling process, also impacts user trust [1].

is extremely valuable in understanding causal links between the explanations and possible interventions, combined efforts between the domain experts and ML practitioners are necessary to map explanations to interventions.

E.g. Consider an officer flagged by the EIS for whom the explanation indicates the model is prioritizing features related to the type of dispatches the officer was assigned to in the last few months. Upon further inspection of the data, it can be seen that the officer had been dispatched to high-stress situations on a regular basis. In this instance, a possible intervention is reassigning of duties or putting them on low-stress dispatches after a series of high-stress dispatches.

3.5 Use Case 5: Recourse

When individuals are negatively impacted by ML aided decisions, providing them with a concrete set of actionable changes that would lead to a different decision is critical. This ability of an individual to affect model outcomes through actionable changes is called recourse [43]. While recourse has been studied independently from explainable ML [43], ML explanations have the potential to help individuals seek recourse in public policy applications.

In this use-case, there are two explanation goals: 1) help the user understand the reasons behind the decision that enable them discover any inaccuracies in the model and /or data and dispute the decision, and 2) help the user identify the set of actionable changes that would lead to an improved decision in the future. As the user in this use-case is the affected individual, the explanations that indicate reasons behind the decisions should be mapped to a domain that is understandable by the individual. Furthermore, the explanations that indicate changes should contain actionable changes (e.g. reducing age by 10 years vs reducing debt).

E.g. In the EIS, the affected individual is the flagged officer. If the officer is provided with explanations indicating the reasons behind the elevated risk score and actionable changes that could reduce their risk score, they could either point any inaccuracies or take measures themselves (in addition to the intervention by the PD) to reduce the risk score.

4 CURRENT STATE OF EXPLAINABLE ML

In this section, we summarize the existing approaches in explainable ML. The intention is not to provide a comprehensive literature review but rather a broad summary of existing approaches and how they apply to public policy settings. We refer readers to [2, 4, 8, 31] for more comprehensive reviews of existing work.

4.1 Existing work in explainable ML

Existing approaches fall into two broad categories: 1) directly interpretable ML models, and 2) post-hoc methods for explaining opaque ML models.[3] ML explanation takes two forms: 1) explaining individual predictions (local explanation), and 2) explaining overall behavior of the models (global explanation). Local explanations help users understand *why* the model arrived at the given prediction for a given instance, while global explanations explain *how* the model generally behaves [32]. Table 2 summarizes the existing approaches and how they fit in our use-case taxonomy.

4.1.1 Directly interpretable ML models. Directly (or inherently) interpretable ML models are designed such that an end-user could understand the decision-making process [22]. In a policy context, with an interpretable model, a user could: (a) understand how the model calculates a risk score (global explanation), and (b) for a given instance, understand what factors contributed to that risk score (local explanation). Several efforts have focused on developing directly

[3]Note that this opacity may either be a reflection of 1) the model being too complex to be comprehensible, or 2) the model is proprietary [39]. In this paper, we focus on opacity created through model complexity.

Table 2. Existing approaches for explainable ML

Method	Post-hoc methods				Interpretable Models	References
	Local		Global			
	Model agnostic	Model specific	Model agnostic	Model specific		
Sparse models					✓	[45], [44], [17]
Decision Rules/Lists/Sets					✓	[22]
Linear/additive models					✓	[11], [25]
Local surrogate models	✓	✓			✓	[36], [32]
Permutation (Shapley values)	✓	✓		✓		[29], [28], [30], [27]
Global rule extraction		✓	✓			[37], [42]
Gradient-propagation		✓				[5], [41], [6], [48]
Influence functions	✓					[21]
Counterfactual explanations	✓					[43], [34]

interpretable models, such as those for healthcare and criminal justice [11, 50]. These include sparse linear models [43, 45], sparse decision trees [17], generalized additive models [16, 25, 26], and interpretable decision sets [22].

Directly interpretable models often rely on carefully curated representations of data with meaningful input features [39], often through discretization or binary encodings [11, 22, 43]. While any ML model requires diligent feature engineering, distilling complex data spaces into a set of optimally discretized and meaningful features can entail extensive effort and optimization of its own. Doing so may prove particularly challenging with the complex and heterogeneous feature spaces typically found in policy settings.

4.1.2 Post-hoc methods for explaining black-box ML models. Post-hoc/post-modeling methods derive explanations from already trained black-box/opaque ML models. As post-hoc methods do not interfere with the model's training process, they enable the use of complex ML models to achieve explainability without risk of sacrificing performance. However, as black-box ML models are often too complex to be explained entirely, post-hoc methods typically derive an approximate explanation [14, 39], which makes ensuring the fidelity of the explanations to the model a key challenge in this work. Unlike directly interpretable models, local and global explanations for opaque complex ML models require different methods. For both types of explanations, both model-specific and model-agnostic methods exist in literature.

Post-hoc local explanations A local explanation in a typical public policy problem is used to understand which factors affected the predicted risk score for an individual entity. The most common format of local explanation is feature attribution–also known as feature importance or saliency—where each input feature is assigned an importance score that quantifies its contribution to the model prediction [6, 8]. Several approaches exist for deriving feature importance scores: training a directly interpretable surrogate model (linear classifier) around a local neighborhood of the instance in question (LIME, MAPLE) [32, 36]; feature perturbation based methods for approximating each feature's importance using game-theoretic shapely values (SHAP, TreeSHAP) [28, 29]; gradient-based techniques such as sensitivity analysis (SA) [48], deconvolution [41], layer-wise relevance propagation (LRP)[5]. Among these approaches, methods such as LIME, SHAP, influence functions, and SA are model-agnostic methods, whereas LRP, Deconvolution, and TreeSHAP are model specific methods. MAPLE stands out among these methods as it can act both as a directly interpretable model as well as a model-specific post-hoc local explainer [32].

Other approaches such as influence functions [21], nearest neighbors, prototypes, and criticisms [20, 32], make use of other instances, rather than features, to provide local explanations. A special form of example-based explanation is counterfactual explanations, which seek to answer the following question: *"what's the smallest change in data that would result in a different model outcome?"* [18, 31]. In a top-k setting, the *change in outcome* can be the inclusion vs. exclusion of the individual from the top-k list. Counterfactual explanations can provide insight on *how to act* to change the risk score, supplementing the feature attribution methods that explain *why* the model arrived at the risk score.

Post-hoc global explanations A global explanation in a typical policy problem would be a summary of factors/patterns that are generally associated with high-risk scores, often expressed as a set of rules [32, 37]. Global explanations should enable the users to accurately predict, sufficiently frequently, how the model would behave in a given instance. However, deriving global explanations of models that learn highly complex non-linear decision boundaries is very difficult [36]. As a result, the area of deriving post-hoc global explanations is not as fully developed as local explanation methods.

Some approaches for global explanations from black-box ML models include: 1) aggregation of local explanations [3, 27], 2) global surrogate models [13], and 3) rule extraction from trained models [42]. A noteworthy contribution to deriving globally faithful explanations is ANCHORS [37]. ANCHORS identifies feature behavior patterns that have high precision and coverage in terms of its contribution to the model predictions of a particular class. Methods proposed in [27, 37] are model-agnostic and methods presented in [3, 13, 42] are model-specific.

4.2 Mapping existing explainability methods to public policy use-cases

For each use-case, Table 3 ranks the capabilities of existing methods on a three-point scale:

> ★☆☆: Potentially applicable methods exist for the use-case. However, their efficacy in the use-case is not demonstrated through any form of evaluation.
>
> ★★☆: Some evidence of methods being effective in the use-case exists, but the efficacy is not empirically validated through a well-designed user-study.
>
> ★★★: Existing methods are empirically validated on their efficacy of helping users achieve better outcomes for the use-case.
>
> ✕ : Method group is not applicable to the use-case

The discussion below summarizes how existing work maps to each use-case and our assessment of the status of current work with respect to these applications. It should be noted that directly explainable models are potentially applicable to all the use-cases. Therefore, we focus on the post-hoc methods in the summaries below.

Model debugging: Methods for both local and global post-hoc explanations are potentially useful in this use-case. Global explanations could help identify errors in overall decision-making patterns (globally important features can help identify data leakage), and local explanations can help to uncover errors in individual predictions. Although some recent work lends evidence for the utility of explanations in discovering model errors [36], [11], the efficacy of these

Table 3. Applicability of existing methods to public policy use-cases

Use-case	Post-hoc Local	Post-hoc global	Interpretable Models	Potentially applicable approaches
Model debugging	★★☆	★★☆	★★☆	[4–6, 27–29, 36, 37, 43, 45]
Model trust and adoption	★☆☆	★☆☆	★☆☆	[5, 28–30, 32, 36, 37]
Decision making system performance	★☆☆	×	★☆☆	[5, 17, 22, 25, 28, 29, 32, 36, 44]
Intervention selection	★☆☆	×	★☆☆	[5, 17, 22, 25, 28, 29, 32, 36, 44]
Recourse	★★☆	×	★★☆	[5, 17, 22, 25, 28, 29, 32, 34, 36, 43, 44]

methods is not empirically validated through well-defined user trials in real-world applications. Likewise, evaluations have yet to be performed in the context of policy problems.

Trust and model adoption: As with model-debugging, both global and local explanation methods are potentially applicable. However, as the end-user is the domain expert, explanations will need to be extended beyond feature attribution while preserving fidelity to what the model has learned. While existing methods discuss user trust as a broad goal, to the best of our knowledge, their ability to help regulators or decision-makers adequately trust ML models is not demonstrated through well-defined evaluations or user-trials.

Unreliable prediction detection: Feature attribution based local explanations is potentially applicable to provide the necessary information to the user. However, feature attribution alone may not be sufficient. Users may need more contextual information such as *How does the instance fit into the training data distribution? How does the model behave for similar examples? and what factors did it rely on for those predictions?* To that end, there have been some efforts to present visual summaries of explanations to the user [29, 30, 37] which could potentially be useful in this use-case. Therefore, available local explanation methods do provide a good starting point. However, the effectiveness of those methods in generating contextual and user-interpretable explanations that help identify unreliable predictions is not evidenced through evaluations or well-defined user-trials

Intervention selection: As the intervention determinations are often individualized, local explanation methods are potentially applicable for generating the reasons behind the risk score. As with the above use-case, users may need more contextual information to supplement the local explanations such as: *how the instance fits into the training data distribution*, and *intervention history* for *similar—w.r.t* data and *w.r.t* explanation—individuals. To the best of our knowledge, there isn't evidence in the existing body of work on the efficacy of using these local explanation methods for informing intervention selection.

Recourse: Feature attribution based local explanations are potentially applicable for deriving *reasons* behind the decision, and counterfactual explanations are potentially useful in explaining how to improve the outcomes. However, simple counterfactual explanations do not guarantee explanations with actionable changes. There have been some efforts to deriving *actionable* counterfactual explanations [34, 44]. While there is some evidence of counterfactual explanations' potential for helping individuals seek recourse, empirical validation is still required to establish their efficacy.

5 GAPS AND PROPOSED RESEARCH DIRECTIONS

The last section mapped the applicability of existing methods to the use case taxonomy. In this section, we use that mapping to identify gaps in existing explainable ML research when compared to the needs of public policy, and propose a research agenda to fill those gaps. We believe that tackling these research gaps is critical for the machine learning discipline if we want to have a positive and lasting impact on society and public policy.

5.1 Gap 1: Existing methods have not been sufficiently and effectively evaluated in real-world contexts

The most pronounced gap in existing methods is the lack of effective evaluation to establish their efficacy in practical settings. A complete real-world evaluation needs three elements: 1) a real-world problem with a well-defined domain goal, 2) real-world data, and 3) a well-defined user-study. Most of the work in this area to date has focused on benchmark ML problems and data sets (e.g., image classification on MNIST data), with users in a lab setting (often Amazon Mechanical Turk) [5, 17, 26, 28, 29, 36, 40]. Benchmark problems have played a crucial role in the development and refinement of explainable ML methods by virtue of their convenience and availability to a wide range of researchers. However, we argue that these problems, data, and users are far removed from the actual deployment context and thus fail to provide convincing evidence of method effectiveness in informing the choices of domain experts in complex problem settings.

The relatively small number of explainable ML studies that have incorporated some aspects of practical evaluation have unfortunately consistently lacked at least one (and in many cases multiple) of the necessary elements to offer conclusive evidence of real-world efficacy. For instance, [11, 50] describe evaluations making use of real problems with a clear goal and real-world data. However, both studies failed to use real users to evaluate the usefulness of explanations. In the case of [27, 30], the authors applied methods to a real problem with clear goals, using real-world data and real end-users, but failed to conduct a well-defined user study for empirical evaluation. This gap is particularly acute in the context of public policy problems, given the characteristics that set them apart from other ML settings such as image classification.

Several research directions to better evaluate the extent to which existing explainable ML methods can meet the needs of real-world applications in the public policy domain:

5.1.1 *Identify real-world public policy test-cases.*

The first step is to identify test-cases for implementing these methods. A complete evaluation for public policy applications should involve the following components:

(1) **A real-world policy problem and goal**: Focusing on goals faced by practitioners will ensure that any evaluation reflects the ability of explanations to improve socially-relevant outcomes. Picking problems that represent a range of policy settings with different types of goals (resource allocation, early warning, impact analysis) would enable a more comprehensive assessment.

(2) **Real-world data**: To capture the nuances and characteristics of applying ML to a policy area in practice, the use of real data from the problem domain is essential. This is of particular importance with evaluating explainable ML methods, as simplified or synthetic data sets might provide an overly-optimistic evaluation of their ability to extract meaningful information.

(3) **Real users:** Although their time is often scarce, the domain users who will be acting on model outputs must be involved in the evaluation process to ensure it reflects the actual deployment scenario. Because interaction

between model predictions, explanations, and users' domain expertise will dictate the performance of the system, substituting inexperienced users (for instance, from Mechanical Turk) provides little insight into how well explanations will perform in practice.

5.1.2 Evaluate performance-explainability trade-off (if any) for directly explainable models.

As directly interpretable models rely on carefully curated input features, it is necessary to explore the trade-off between performance and scalability in practice. To that end, the models should be implemented on several real policy problems, evaluating: 1) the trade-off between feature preparation efforts and performance, and 2) their ability to generalize on future data under strong temporal dependencies. While the prospect of simple, directly-interpretable models certainly holds considerable appeal, their performance must be rigorously tested against more opaque models such as tree-ensembles across problem domains and applications to understand any potential trade-offs in practice and make implementation decisions.

5.1.3 Evaluate explanation methods on their ability to improve outcomes.

For each use-case, we need to define outcomes as well as other criteria by which explanation effectiveness should be evaluated. For instance, when informing decisions about whether (and how) to intervene, the outcome of interest is the *precision* of the list generated by the decision making-system, while other criteria—consistency/stability of explanations (e.g., *"Does SHAP/LIME yield the same feature attribution scores for repeated runs for the same instance and same model?"*) and how they apply to the specific problem context—will be informative of the utility of the explanations.

Once these evaluation criteria are defined, they can inform the design and implementation of user studies to directly validate existing explanation methods in the context of each use case. Most importantly, these experiments should focus on evaluating the usefulness of explanations to improve the relevant *policy outcomes* rather than more narrowly on end-user *perceptions* of the explanations. As such, any user studies should include the appropriate control and treatment group variants to rigorously assess how outcomes differ in the presence and absence of explanations.

5.2 Gap 2: Existing methods are not explicitly designed for specific use-cases

As discussed above, existing methods are developed with loosely defined or generic explainability goals (e.g., transparency) and without well-defined context-specific use-cases. As a result, methods are developed without understanding the specific requirements of a given domain, use-case, or user-base, resulting in a lack of adoption and sub-optimal outcomes.

While several existing methods may be applicable for each use-case, their effectiveness in real-world applications is not yet well-established, meaning this potential applicability may fail to result in practical impact. As more methods are rigorously evaluated in practical, applied settings as suggested above, gaps in their ability to meet the needs of these use-cases may become evident. For instance, model agnostic methods such as LIME [36] and SHAP [29] are capable of extracting input feature importance scores for individual predictions from otherwise opaque models. However, it is unclear whether they can address needs such as generating explanations that are well-contextualized and truly interpretable by less technical users without sacrificing fidelity (e.g., to help a domain expert identify unreliable model predictions or an affected individual seek recourse).

6 CONCLUSION

Despite the existence of a wide array of explainable ML methods, their efficacy in improving real-world decision-making systems has yet to be sufficiently explored. In this paper, we presented an initial step towards filling that void in the context of public policy applications by defining the scope of ML explainability in the domain. First, we identified a taxonomy of use-cases for ML model explanations in the ML aided public policy decision making pipeline: 1) model debugging, 2) regulator trust & model adoption, 3) unreliable prediction detection, 4) intervention selection, and 5) recourse. For each use-case, we defined the goals of an ML explanation and the intended end-user. Then, we summarized the existing approaches in explainable ML and identified the degree to which this work addresses the needs of the identified use-cases. We observed that, while the existing approaches are potentially applicable to the use-cases, their utility has not been thoroughly validated for any of the use-cases through well-designed empirical user-studies.

Two main gaps are evident in the design and evaluation of existing work: 1) methods are not sufficiently evaluated on real-world contexts, and 2) they are not designed and developed with target use-cases and well-defined explainability goals in mind. In response to these gaps, we proposed several research directions to systematically evaluate the existing methods with problems with real policy goals, real-world data, and domain experts.

We believe that these gaps are critical to fill if the promise of ML in the public sector will result in practical and long-lasting adoption and impact. We hope the gaps we identified here and the proposed research agenda will help the ML research community collaborate with the Policy and HCI communities to build on the promising body of methodological work to ensure that those methods are well-suited to meet the needs of the practitioners and end-users who will be applying them to the benefit of society.

ACKNOWLEDGMENTS

We would like to thank the Block Center for Technology and Society at Carnegie Mellon University for funding this work.

REFERENCES

[1] Klaus Ackermann, Hareem Naveed, Jason Bennett, Joe Walsh, Andrea Navarrete Rivera, Michael Defoe, Adolfo De Unánue, Sun Joo Lee, Crystal Cody, Lauren Haynes, and Rayid Ghani. 2018. Deploying machine learning models for public policy: A framework. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, 15–22.

[2] Amina Adadi and Mohammed Berrada. 2018. Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI). *IEEE Access* 6 (2018), 52138–52160.

[3] Kasun Amarasinghe and Milos Manic. 2019. Explaining What a Neural Network has Learned: Toward Transparent Classification. In *IEEE International Conference on Fuzzy Systems*, Vol. 2019-June. Institute of Electrical and Electronics Engineers Inc.

[4] Vijay Arya, Rachel K E Bellamy, Pin-Yu Chen, Amit Dhurandhar, Michael Hind, Samuel C Hoffman, Stephanie Houde, Q Vera Liao, Ronny Luss, Aleksandra Mojsilović, Sami Mourad, Pablo Pedemonte, Ramya Raghavendra, John Richards, Prasanna Sattigeri, Karthikeyan Shanmugam, Moninder Singh, Kush R Varshney, Dennis Wei, and Yunfeng Zhang. 2019. One Explanation Does Not Fit All: A Toolkit and Taxonomy of AI Explainability Techniques. (2019). arXiv:1909.03012

[5] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus Robert Müller, and Wojciech Samek. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE* 10, 7 (2015), 1–46.

[6] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. 2010. How to explain individual classification decisions. *The Journal of Machine Learning Research* 11 (2010), 1803–1831.

[7] Matthew J. Bauman, Erika Salomon, Joe Walsh, Robert Sullivan, Kate S. Boxer, Hareem Naveed, Jen Helsby, Chris Schneweis, Tzu Yun Lin, Lauren Haynes, Steve Yoder, and Rayid Ghani. 2018. Reducing incarceration through prioritized interventions. In *Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies, COMPASS 2018*, Vol. 18. Association for Computing Machinery, Inc, 1–8.

[8] Umang Bhatt, Alice Xiang, Shubham Sharma, Adrian Weller, Ankur Taly, Yunhan Jia, Joydeep Ghosh, Ruchir Puri, José M F Moura, and Peter Eckersley. 2020. Explainable Machine Learning in Deployment. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 648–657.

[9] Stephen Boyd, Corinna Cortes, Mehryar Mohri, and Ana Radovanovic. 2012. Accuracy at the Top. In *Advances in Neural Information Processing Systems 25*. 953–961.

[10] Samuel Carton, Jennifer Helsby, Kenneth Joseph, Ayesha Mahmud, Youngsoo Park, Joe Walsh, Crystal Cody, CPT Estella Patterson, Lauren Haynes, and Rayid Ghani. 2016. Identifying Police Officers at Risk of Adverse Events. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 67–76.

[11] Rich Caruana, Yin Lou, Johannes Gehrke Microsoft, Paul Koch, Marc Sturm, and Noémie Elhadad. 2015. Intelligible Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-day Readmission. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1721–1730.

[12] Alexandra Chouldechova, Emily Putnam-Hornstein, Suzanne Dworak-Peck, Diana Benavides-Prado, Oleksandr Fialko, Rhema Vaithianathan, Sorelle A Friedler, and Christo Wilson. 2018. A case study of algorithm-assisted decision making in child maltreatment hotline screening decisions. In *Proceedings of Machine Learning Research*, Vol. 81. PMLR, 1–15.

[13] Nicholas Frosst and Geoffrey Hinton. 2017. Distilling a Neural Network Into a Soft Decision Tree. In *Proceedings of the First International Workshop on Comprehensibility and Explanation in {AI} and {ML} 2017 co-located with 16th International Conference of the Italian Association for Artificial Intelligence (AI*IA 2017) Conference of the Italian Associat*.

[14] Leilani H. Gilpin, David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. 2018. Explaining Explanations: An Approach to Evaluating Interpretability of Machine Learning. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*. 80–89.

[15] Bryce Goodman and Seth Flaxman. 2017. European Union regulations on algorithmic decision-making and a "right to explanation". *AI magazine* 38 (2017), 50–57.

[16] Trevor J Hastie and Robert J Tibshirani. 1990. *Generalized additive models*. Vol. 43. CRC press.

[17] Xiyang Hu, Cynthia Rudin, and Margo Seltzer. 2019. Optimal Sparse Decision Trees. In *Advances in Neural Information Processing Systems 32*. 7267–7275.

[18] Amir-Hossein Karimi, Eth Zürich, Switzerland Bernhard Schölkopf, and Isabel Valera. 2020. Algorithmic Recourse: from Counterfactual Explanations to Interventions. arXiv:2002.06278v4

[19] Shachar Kaufman, Saharon Rosset, and Claudia Perlich. 2011. Leakage in Data Mining: Formulation, Detection, and Avoidance. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, 556–563.

[20] Been Kim, Rajiv Khanna, and Oluwasanmi Koyejo. 2016. Examples are not Enough, Learn to Criticize! Criticism for Interpretability. In *Advances in Neural Information Processing Systems 29*. 2280–2288.

[21] Pang Wei Koh and Percy Liang. 2017. Understanding Black-box Predictions via Influence Functions. In *Proceedings of the 34th International Conference on Machine Learning*. 1885–1894.

[22] Himabindu Lakkaraju, Stephen H Bach, and Jure Leskovec. 2016. Interpretable Decision Sets: A Joint Framework for Description and Prediction. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1675–1684.

[23] Zachary C. Lipton. 2018. The mythos of model interpretability. *Commun. ACM* 61 (2018). Issue 10.

[24] Li-Ping Liu, Thomas G Dietterich, Nan Li, and Zhi-Hua Zhou. 2016. Transductive Optimization of Top k Precision. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*. 1781–1787.

[25] Yin Lou, Rich Caruana, and Johannes Gehrke. 2012. Intelligible Models for Classification and Regression. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 150–158.

[26] Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. 2013. Accurate Intelligible Models with Pairwise Interactions. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. 623–631.

[27] Scott M Lundberg, Gabriel Erion, Hugh Chen, Alex Degrave, Jordan M Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. 2020. From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence* 2, 1 (2020), 56–67.

[28] Scott M Lundberg, Gabriel G Erion, and Su-In Lee. 2018. Consistent Individualized Feature Attribution for Tree Ensembles. arXiv:1802.03888

[29] Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Advances in neural information processing systems*. 4765–4774.

[30] Scott M. Lundberg, Bala Nair, Monica S. Vavilala, Mayumi Horibe, Michael J. Eisses, Trevor Adams, David E. Liston, Daniel King Wai Low, Shu Fang Newman, Jerry Kim, and Su In Lee. 2018. Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. *Nature Biomedical Engineering* 2, 10 (oct 2018), 749–760.

[31] Christoph Molnar. 2019. *Interpretable Machine Learning*. https://christophm.github.io/interpretable-ml-book/.

[32] Gregory Plumb, Denali Molitor, and Ameet Talwalkar. 2018. Model Agnostic Supervised Local Explanations. In *Advances in Neural Information Processing Systems*. 2515–2524.

[33] Eric Potash, Joe Brew, Alexander Loewi, Subhabrata Majumdar, Andrew Reece, Joe Walsh, Eric Rozier, Emile Jorgensen, Raed Mansour, and Rayid Ghani. 2015. Predictive modeling for public health: Preventing childhood lead poisoning. In *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Vol. 2015-August. Association for Computing Machinery, 2039–2047.

[34] Rafael Poyiadzi, Kacper Sokol, Raul Santos-Rodriguez, Tijl De Bie, and Peter Flach. 2020. FACE: feasible and actionable counterfactual explanations. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 344–350.

[35] Arthi Ramachandran, Avishek Kumar, Hannes Koenig, Adolfo De Unanue, Christina Sung, Joe Walsh, John Schneider, Rayid Ghani, and Jessica P Ridgway. 2020. predictive Analytics for Retention in care in an Urban HiV clinic. *Scientific reports* 10, 1 (2020), 1–10.

[36] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. " Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 1135–1144.

[37] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Anchors: High-Precision Model-Agnostic Explanations.. In *AAAI*, Vol. 18. 1527–1535.

[38] Kit T Rodolfa, Erika Salomon, Lauren Haynes, Jamie Larson, and Rayid Ghani. 2020. Case Study: Predictive Fairness to Reduce Misdemeanor Recidivism Through Social Service Interventions. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 142–153.

[39] Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence* 1, 5 (2019), 206–215.

[40] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. 2017. Learning Important Features Through Propagating Activation Differences. arXiv:1704.02685

[41] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2013. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. arXiv:1312.6034

[42] Hiroshi Tsukimoto. 2000. Extracting Rules from Trained Neural Networks. *Transactions on Neural Networks* 11, 2 (2000), 512–519.

[43] Berk Ustun and Cynthia Rudin. 2019. Learning Optimized Risk Scores. *Journal of Machine Learning Research* 20, 150 (2019), 1–75.

[44] Berk Ustun, Alexander Spangher, and Yang Liu. 2019. Actionable Recourse in Linear Classification. In *Proceedings of the 2019 Conference on Fairness, Accountability, and Transparency*. 10–19.

[45] Berk Ustun, Stefano Tracà, and Cynthia Rudin. 2013. Supersparse Linear Integer Models for Interpretable Classification. arXiv:1306.6677

[46] Adrian Weller. 2019. Transparency: motivations and challenges. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Springer, 23–40.

[47] Teng Ye, Rebecca Johnson, Samantha Fu, Jerica Copeny, Bridgit Donnelly, Alex Freeman, Mirian Lima, Joe Walsh, and Rayid Ghani. 2019. Using machine learning to help vulnerable tenants in New York City. In *COMPASS 2019 - Proceedings of the 2019 Conference on Computing and Sustainable Societies*. 248–258.

[48] Matthew D Zeiler and Rob Fergus. 2014. Visualizing and understanding convolutional networks. In *European conference on computer vision*. Springer, 818–833.

[49] Leid Zejnilović, Susana Lavado, Íñigo Martínez de Rituerto de Troya, Samantha Sim, and Andrew Bell. 2020. Algorithmic Long-Term Unemployment Risk Assessment in Use: Counselors' Perceptions and Use Practices. *Global Perspectives* 1, 1 (2020).

[50] Jiaming Zeng, Berk Ustun, and Cynthia Rudin. 2017. Interpretable classification models for recidivism prediction. *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 180, 3 (2017), 689–722.