





# A Tool for Distributed Collaborative Causal Discovery

Alexey Tregubov<sup>(✉)</sup>, Jeremy Abramson, Stephen Schwab, and Jim Blythe

USC Information Sciences Institute, Marina del Rey, CA, USA  
{tregubov, abramson, schwab, blythe}@isi.edu

**Abstract.** The development of accurate causal models is crucial for achieving and explaining desired outcomes that require interventions. Building these models efficiently requires combining available data with expert causal knowledge. Often experts have unique data and model insights, but sharing them is challenging due to privacy or security concerns. Federated machine learning addresses similar issues by allowing multiple sites to collaborate on a common model without sharing private datasets. This paper introduces CCaT, a distributed causal discovery tool enabling collaborative development of a shared causal model while preserving local models and data privacy. CCaT allows each site to evaluate and refine the shared model using its private dataset, sharing only summary statistics or suggested new causal relations. The tool supports maintaining distinct local causal models, as analysts can choose to adopt or change parts of the shared model. CCaT enhances the accuracy of causal models by leveraging diverse expertise and data, achieving a generality and accuracy unattainable by individual sites. We present several common scenarios with the CCaT to demonstrate its effectiveness.

## 1 Introduction

An accurate causal model is essential when action must be taken to achieve some desired outcome [2, 8]. However, many domains lack a formal causal model. These can be built most efficiently by combining available data with elements of causal knowledge from human experts. Tools such as UPREVE [10] and CauseWorks [4] allow users to create and edit graphical causal models and provide an interface to evaluate models against data and suggest modifications. In many domains, several experts may each possess a distinct subset of the larger causal model and each may have access to a unique dataset. Each expert is motivated to share aspects of their causal model with others in order to improve all of their models, but may be unable to share their data in detail. In medical settings, for example, experts from different groups may not be able to share data due to privacy laws. In the intelligence community, security concerns may prohibit data sharing.

In this paper, we describe a distributed causal discovery tool that supports a collaboration between sites that pool their expertise to develop a shared causal model of their domain, without sharing any other information about their private data. This approach is analogous to sharing weights or gradients on a deep

network in the federated learning approach [3], which has received considerable attention in recent years, but does not support sharing causal knowledge. In our approach, each site maintains a private dataset that it uses to evaluate the shared model and optionally to mine new causal relationships. Each site shares summary statistics on a developing shared model based on their local dataset, without revealing that dataset, and may also introduce new variables and new causal links into the model with associated summary statistics. In contrast to most federated learning approaches, since causal discovery typically requires human input, the tool supports maintaining a local causal model at each site which may be different from the shared causal model, since analysts may choose whether to adopt or ignore any part of the shared model.

We present CCaT, (Collaborative Causal discovery Tool), an implemented collaborative causal discovery tool that supports groups in contributing to a shared causal model while maintaining a distinct local model, and evaluating both models with private data. We describe several scenarios in which CCaT supports several groups building a shared causal model with an accuracy that may not be achievable at any one site individually, under assumptions that we detail about the distribution of observational data and expert knowledge among the groups. In some cases, the shared model can achieve a level of generality that would not be possible at each individual site, allowing more rapid sharing of causal knowledge.

In the next section we define the collaborative causal discovery task that our tool is designed to solve. The following section introduces CCaT and its UI and analytical capabilities that support the task. We then describe several scenarios that illustrate the power of this tool, and finally discuss privacy considerations and future work.

## 2 Related Work

UPREVE offers both a GUI for causal discovery and a set of integrated algorithms and metrics [10]. However, it does not support combined human-machine discovery or collaborative causal discovery with private data. Stefano et al. introduce a paradigm for multi-agent collaborative learning of causal networks in cases of partial observability, in which agents may ask others to perform experiments on their behalf [6]. Meganck et al. propose an algorithm for distributed learning of multi-agent causal models, an extension of causal bayesian networks to a distributed domain [7]. However, neither of these approaches integrates human insight with algorithmic results via a GUI. Causeworks is a GUI that supports collaborative construction of causal models, overlaying analytic results on a shared model built by experts [4, 5]. Compared to CCaT, it does not support private data or private models; instead, all participants have access to all relevant data. CCaT is designed for a collaboration mode where local models and data remain private unless they are shared (partially or in full).

### 3 Problem Description

There are many domains (healthcare, financial institutions, education) where information is collected in a distributed manner and cannot be shared outside the local information system (for privacy, security and other reasons). When training data cannot be shared directly (or at all), we seek to share model parts to the extent it is feasible and gain common benefits for all distributed sites without breaking security and privacy constraints.

In this paper, we focus on an approach for collaborative causal discovery, designed for settings where distributed sites cannot share their private local data but can choose what model aspects to share. These aspects include: individual rules learned by each site, parts of a local causal model (e.g. partial DAGs), model parameters and aggregated data properties (e.g. best thresholds discovered by each site and/or summary evaluation statistics computed from their local data). Shared aspects (models, rules, thresholds) can be tested and incorporated by other sites and a global evaluation of the model can be approximated from individual site summary statistics. New findings can then again be shared with everybody in a continuous, iterative model improvement process. This approach allows distributed sites jointly to discover confounding bias and incorporate new features or more relevant features into the model, benefiting from the experience of each site while maintaining data privacy. We do not yet support results from explicit interventions by individual sites into our model but intend to do so in a future extension.

### 4 CCaT Overview

CCaT is a distributed causal discovery tool with a web interface. Each distributed site runs its own instance of a tool and shares selected models and model parameters with others. Shared models and parameters are distributed across all sites. The CCaT web interface provides access to local and shared causal models. These models can be edited, then fitted and tested on local historical data. CCaT allows sharing causal models as a whole or just their parts (fit data such as thresholds, weights, etc.).

In this paper we use a social media domain example to illustrate CCaT, in which causal models are developed to explain bans on a social media platform (e.g. censorship policy valuations on the platform or effective bans that are triggered by internet trolls or other actors). Each site has its own historical observational data on what actions social media influencers/users took in the past (posts, comments, likes etc.) and observed ban events. This historical data has information about several influencer/user accounts and their history of bans. In CCaT each causal model is called a policy, and consists of one or more rules combined together. Causal relationships among features and with the outcome variable (ban or no ban label) are reflected in the model's DAGs.

The home screen of the CCaT user interface provides access to three tabs: (1) local and shared policies viewer to view and compare policies; (2) rules library with all available rules; (3) editor for DAG editing and evaluation.

CCaT - Collaborative Causal Discovery Tool

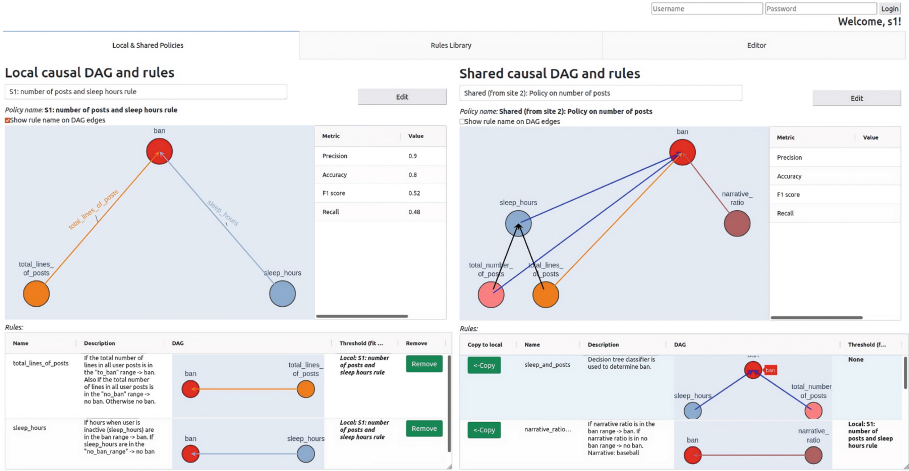


Fig. 1. CCaT Local and shared policies viewer tab of the CCaT.

The local and shared policies viewer tab (Fig. 1) allows experts to see details (DAGs, constituent rules, evaluation summary) about local (left panel) and shared policies (right panel). Model fragments from shared policies can be copied to local models. Policies can be opened in the editor tab.

Model fragments in the rules library (Fig. 2) are collected from both shared and local policies. For each fragment, all available thresholds (or model fit data) are listed in the “Available thresholds” column. In this social media example, there are three sets of general (or global) thresholds: local historical data fit from the current site and shared thresholds from two other sites. The list of thresholds also includes thresholds that were shared as part of some rule. Both of these groups of thresholds are displayed in the “Available thresholds” column.

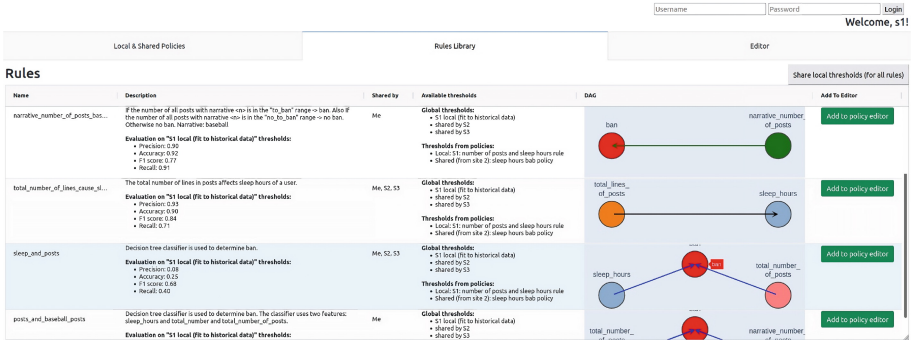
Once the historical data with labels is loaded in CCaT, it generates a set of simple model fragments (feature-to-outcome causal links). These fragments, which we refer to as rules, can be found in the rules library tab (Fig. 2). Each rule is associated with some classifier model (e.g. a simple threshold or decision tree classifier). These rules serve as building blocks for policy models. Each rule can be added to the model currently opened in the DAG editor tab.

Figure 3 shows DAG editor and policy/model evaluation interface. Rules can be added or removed from the policy currently in the editor. For each rule, the expert can choose what parameters to use (thresholds, fit data) in the Thresholds column. Once all rules have thresholds/fit data selected, the evaluation panel will immediately show how the current model scores on local historical data.

The evaluation section of the DAG editor tab shows precision, accuracy, recall and F1 score. The histogram on the right shows how many times each rule in the policy was triggered (multiple rules can be triggered simultaneously). If policy

uses rules with decision tree classifiers, the bar chart on the right also shows the importance of each feature using Gini impurity.

### CCaT - Collaborative Causal Discovery Tool



Name	Description	Shared by	Available thresholds	DAG	Add to Editor
narrative_number_of_posts_ban...	If the number of all posts with narrative_tag is like "no_ban" (stage <= 0) then the number of all posts with narrative_tag is in the "no_ban" range → no ban. Otherwise no ban. (Narrative based)	Me	Global thresholds: • S1 local (fit to historical data) • Shared by S2 • Shared by S3 Thresholds from policies: • Local: S1 number of posts and sleep hours rule • Shared from site 2: sleep hours bab policy		<a href="#">Add to policy editor</a>
total_number_of_lines_case_of...	The total number of lines in posts affects sleep hours of a user.	Me, S2, S3	Global thresholds: • S1 local (fit to historical data) • Shared by S2 • Shared by S3 Thresholds from policies: • Local: S1 number of posts and sleep hours rule • Shared from site 2: sleep hours bab policy		<a href="#">Add to policy editor</a>
sleep_and_posts	Decision tree classifier is used to determine ban.	Me, S2, S3	Global thresholds: • S1 local (fit to historical data) • Shared by S2 • Shared by S3 Thresholds from policies: • Local: S1 number of posts and sleep hours rule • Shared from site 2: sleep hours bab policy		<a href="#">Add to policy editor</a>
posts_and_banall_posts	Decision tree classifier is used to determine ban. The classifier uses two features: sleep_hours and total_number and total_number_of_posts.	Me	Global thresholds: • S1 local (fit to historical data) • Shared by S2 • Shared by S3		<a href="#">Add to policy editor</a>

Fig. 2. CCaT Rules library tab of the CCaT.

## 5 Scenarios

In this section we show several common scenarios with CCaT. For demonstration purposes we use a social media domain where a modeling expert develops causal models explaining user bans on the platform. We assume that the modeler has observational data on the activities of several social media influencers/users. Some of these activities triggered bans (or ban warnings) from the social media platform. Reasons for these bans could be very different depending on the circumstances (platform policy violation, censorship, targeted troll attacks) but all bans are caused by influencer/user actions.

In the scenarios below we assume that the initial feature engineering has been done, and each distributed site has a set of already precomputed features for its local data set. Each distributed site has observational historical local data for multiple users and features of these users and their activities. Ban labels are also included in the data set. Each site may have different feature sets, and CCaT can help modeling experts discover appropriate features and model parameters (e.g. thresholds for rules in a ban policy) as described in the subsections below.

### 5.1 Adopting Causal Variables with a Common Effect

Our first example shows how a site may change the thresholds in its existing model when incorporating a new causal relationship from the shared model (and therefore proposed by a different site). This can happen when the new relationship shares a common effect variable with the existing model, under the

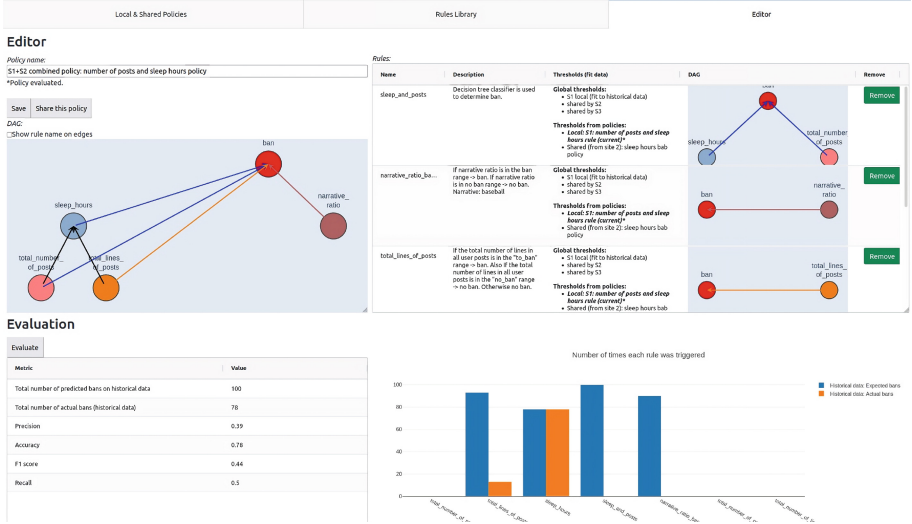


Fig. 3. CCaT DAG editor and policy evaluation UI of the CCaT.

assumption that the site chooses thresholds for the best fit to its dataset both before and after incorporating the new model fragment.

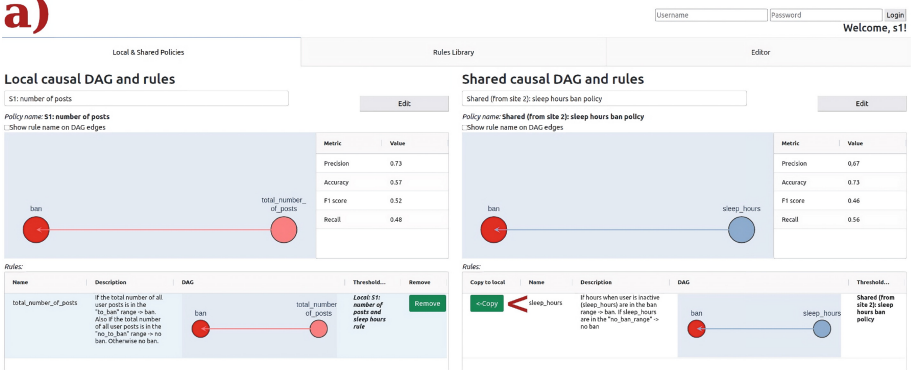
In this case, shown in Fig. 4, site 1 begins with a rule determining if a user will be attacked based solely on the total number of posts made, with a recall of 0.48 and accuracy of 0.57. This is shown in the top of the figure, along with a shared rule that uses only the number of hours the user has been asleep. The lower window shows the site’s local model after incorporating the rule based on sleep into its existing model. On fitting to local data, the new model has a recall of 0.84 and accuracy of 0.81.

Since the new rule shared an effect variable, which signifies whether the user will be attacked, this variable now has two incoming links in the model. In general, the rule that best fits the data might combine several thresholds for each variable, for example if it is generated by a decision tree algorithm [9]. Our tool supports this approach but allows the user to use any of a set of approaches to fit a rule to its dataset. One such example is a disjunctive set of single-variable rules that might be found by a rule-learning approach such as Ripper [1]. Even in this case, we note that the thresholds for each variable may be changed from the original thresholds on each individual rule, because each may have over-compensated for cases better handled by the other rule.

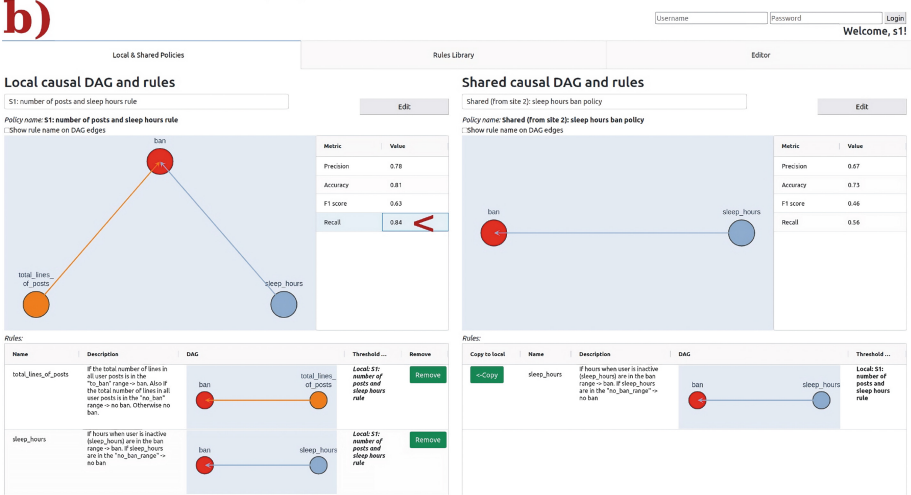
### 5.2 Jointly Discovering Confounding Bias

In our second scenario, adopting a variable from the shared model uncovers a confounding bias that leads a variable to be dropped from the model. In confounding bias, the new variable affects both the outcome of interest and the

CCaT - Collaborative Causal Discovery Tool



CCaT - Collaborative Causal Discovery Tool

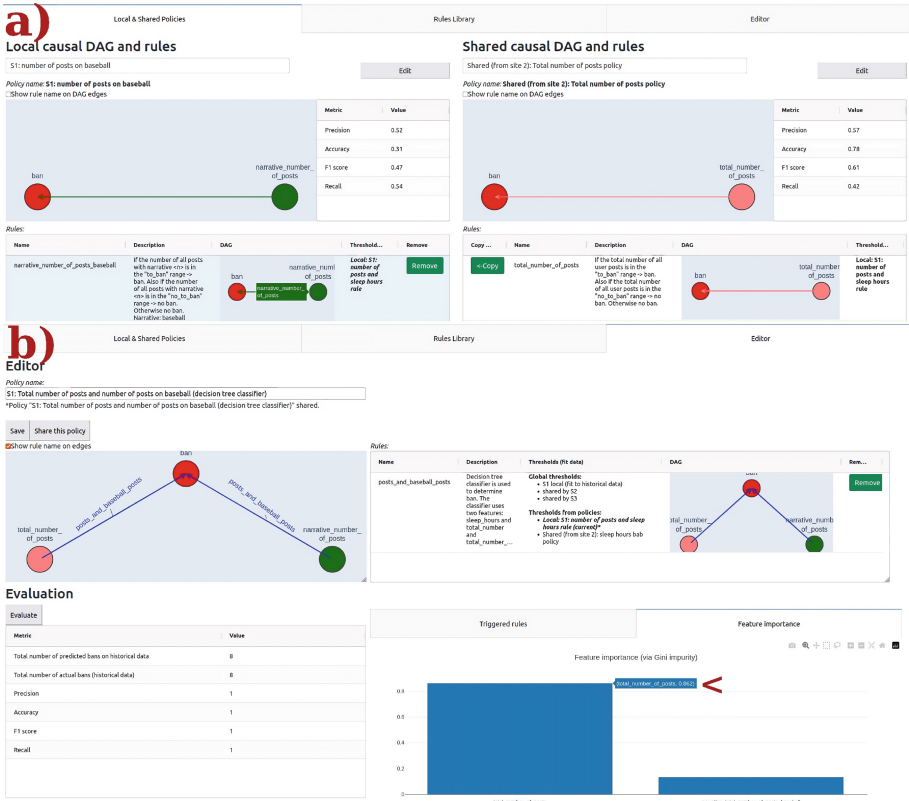


**Fig. 4.** Scenario 1: a) Site 1 observes that site 2 uses a different rule (the number of sleep hours model on the right) and copies that rule and threshold to local model (left). b) Site 1 uses a combined model (left), leading to improved recall and accuracy.

current treatment variable. One way to mitigate confounding bias is to stratify the model on the confounding variable. In this example, doing so would uncover the fact that a variable in the local site’s original model is effectively subsumed by the newly adopted variable, and has little to no effect on the outcome when conditioned on the new variable. Here, however, this situation is more effectively uncovered when the new model is fitted to the local data, and the local site’s original, subsumed variable can be safely dropped from the model.

Specifically, the local site has the initial model that being attacked is dependent on the number of posts made in a particular narrative, while the shared model has the (correct) rule that it is caused by the total number of posts in all narratives, as shown in Fig. 5. The site incorporates the shared rule into its local

model, then finds a best fit to its local data, increasing its F1 score from 0.47 to 1. The decision tree used to fit the data leans heavily on the total number of posts, as can be seen by the feature importance graph in the lower right corner of Fig. 5. On seeing this, the user at the local site chooses to delete the rule based on the number of posts in a particular narrative from the local model.



**Fig. 5.** Scenario 2: a) Site 1 observes that site 2 uses a different rule (the total number of posts, on the right) b) Site 1 uses a combined decision tree model with both features. A correlation analysis would reveal a confounding bias when both variables are included, however an analysis of feature importance shows that the original feature should be dropped from the model.

### 5.3 Collaborative Support for the Model

The previous scenarios focused on sharing new variables and causal links between sites, and updates made by local sites when fitting the emerging model against their private data. An important and complementary component of building a



shared model is aggregating statistics about the shared model’s support from each of the component sites. Since the variance of the mean of a set of  $n$  independent variables decreases in proportion to  $1/n$ , each site can combine summary statistics from all the sites to evaluate its model with lower variance than possible using only local data. The approach can also mitigate overfitting that may occur using only local data, and does not share private data from any site. This approach is analogous to sharing weights or gradients on a deep network in federated learning. The complementary component of sharing causal variables and links is typically not done in that setting.

We illustrate the use of sharing statistics from each site within the first scenario above. 10-fold cross-validation may yield a mean and variance for the model accuracy of  $\mu_1$  and  $\sigma_1^2$ , while the same method applied by site 2 may yield  $\mu_2$  and  $\sigma_2^2$  respectively. By combining these we can assign a mean and variance to samples drawn from the combined dataset. The mean is the mean of  $\mu_1$  and  $\mu_2$  while the variance is given by:  $\sigma_{1,2}^2 = 1/2(\sigma_1^2 + \sigma_2^2 + 2(\mu_1 - \mu_2)^2)$ . Thus, when  $\mu_1$  and  $\mu_2$  are relatively close, the sample mean of this combined dataset has roughly half the variance ( $\sigma_{1,2}^2/20$ ) of the sample mean for either site ( $\sigma_1^2, 2/10$ ).

Improving model estimates via shared summary statistics is most straightforward when all the sites adopt the shared model and find a good fit with a single shared threshold or decision rule at each link in the model. In this case, meaningful statistics can be shared for the entire model as well as individual links, allowing each site to estimate the global performance of the current model as well as regions where potential improvements according to local data can be shared to update the model. However, CCaT does not enforce this constraint in order to provide support even when individual sites may wish to deviate from the shared model. This may either be due to the local site having different goals from other sites, leading to different tradeoffs on the model, or because the sites have datasets that are not identically distributed due to the nature of the sites. For example, in our social media domain, different users might be working with a different mix of social media platforms, or might be engaging in topics or geographic locations that lead to systematic differences in the data they observe.

## 6 Discussion

We discussed several scenarios where CCaT addresses some of the challenges of collaborative causal discovery. In situations when sharing data or entire models is impossible, collaboration via partial sharing of models, rules, and summary statistics for models can help discover appropriate features, confounding biases and new causal links.

Our CCaT provides basic functionality for distributed collaborative causal discovery. It addresses some of the limitations of existing tools by providing a web user interface for distributed model development and sharing for human experts with algorithmic support for testing against local data and sharing summary statistics with collaborators. Modeling experts can choose what knowledge to share (parts of the model, individual rules and their parameters, statistics).

The current version of the CCaT was developed as a prototype and has several limitations. It does not support experimental interventions, which is an important tool in causal discovery. We plan to extend our tool to interventions in the future, though we also note it is not appropriate for some domains, for example where they may be too expensive or infeasible.

The user interface is currently limited to sharing small model fragments that use fixed relations between cause and effect (threshold rules, decision tree and disjunctive normal form), and model sharing is automatic between all sites (users cannot choose with whom to share). These limitations will be addressed in future versions of the CCaT with a decentralized peer-to-peer sharing system where pairwise model sharing is possible and users can choose with whom to share their models. We will also test CCaT on larger models from a broader set of domains for scalability and user experience, and incorporate more summary statistics, such as propensity-based methods [11].

The tool can be downloaded for testing by contacting the authors.

**Acknowledgements.** The authors are grateful to the Defense Advanced Research Projects Agency for their support and to Daryl Best and George Cooper for valuable discussion.

## References

1. Cohen, W.W., et al.: Learning rules that classify e-mail. In: AAAI Spring Symposium on Machine Learning in Information Access, vol. 18. Stanford, CA (1996)
2. Imbens, G.W., Rubin, D.B.: Causal Inference in Statistics, Social, and Biomedical Sciences. Cambridge University Press, Cambridge (2015)
3. Kairouz, P., et al.: Advances and open problems in federated learning. *Found. Trends Mach. Learn.* **14**(1–2) (2021)
4. Kapler, T., Gray, D., Vasquez, H., Wright, W.: Causeworks collaboration: simultaneous causal model construction and analysis. In: Extended Abstracts of the CHI Conference on Human Factors in Computing Systems. CHI EA '21 (2021)
5. Kapler, T., Gray, D.W.S., Vasquez, H., Wright, W.: Causeworks: a mixed initiative framework for causal modeling. *SN Comput. Sci.* **4**, 1–19 (2022)
6. Mariani, S., Roseti, P., Zambonelli, F.: Towards multi-agent learning of causal networks. In: International Conference on Autonomous Agents and Multiagent Systems (2023)
7. Meganck, S., Maes, S., Manderick, B., Leray, P.: Distributed learning of multi-agent causal models. In: International Conference on Intelligent Agent Technology (2005)
8. Pearl, J.: Causal inference in statistics: an overview. *Statistics Surveys* (2009)
9. Safavian, S.R., Landgrebe, D.: A survey of decision tree classifier methodology. *IEEE Trans. Syst. Man Cybern.* **21**(3), 660–674 (1991)
10. Unni, S.J., Sheth, P., Ding, K., Liu, H., Candan, K.S.: UPREVE: an end-to-end causal discovery benchmarking system. In: SBP-BRiMS Demo (2023)
11. Xiong, R., Koenecke, A., Powell, M., Shen, Z., Vogelstein, J.T., Athey, S.: Federated causal inference in heterogeneous observational data. *Stat. Med.* **42**(24) (2023)