Privacy-Preserving Inductive Learning with Decision Trees

Stacey Truex, Ling Liu, Mehmet Emre Gursoy, Lei Yu
School of Computer Science, Georgia Institute of Technology, Atlanta, Georgia, USA

***Abstract*—With the continued explosion of digitized data, data mining and data collection have become more prevalent. With this growth, we have also seen increased concern over data privacy and intellectual property. Within this environment, an important question has emerged: *Can machine learning and data mining techniques be leveraged without compromising privacy*? This paper revisits the concepts and techniques of privacy-preserving decision tree learning, a fundamental model of inductive learning. We first examine different privacy risks during decision tree based inductive learning processes, including the sensitivity of private input data and potential privacy risks induced by inference over both the learning output and the intermediate results of inductive learning iterations. Then we review and compare the privacy notions and properties of three orthogonal and yet complimentary technical frameworks: randomization-based data obfuscation, differential privacy, and secure multiparty computation. We analyze their effectiveness and review representative approaches in each of these three frameworks. Finally, we highlight some of the open challenges to privacy-preserving solutions for decision tree learning.**

I. Introduction

Today, most industries are rapidly increasing the amount of data they collect while researchers continue to develop tools and techniques to mine valuable insights from this data. As more data is collected and mined by various machine learning tools and algorithms, data privacy awareness is increasing, and the need to protect personal privacy as well as organizational privacy is pushing to the forefront of many business and industry operations.

Inductive learning is a popular predictive analytic method that learns a target model through iterative inductions over the training sample set. This is done by finding an approximation to an optimal hypothesis by optimizing the objective learning function. A popular objective function is defined by minimizing a given loss function. Formally, a training dataset of n samples is denoted by $\{(x_1, c_1), \ldots, (x_n, c_n)\}$, in which c_i is the true label of x_i. We also assume a target function $c = \Gamma(x)$ for the training dataset, where c is the true label for the sample object x in the training set. An inductive learner produces a model $y = g(x)$ which approximates the true function $\Gamma(x)$ such that a given loss function $L(c, y)$ is minimized, when y is the label predicted by the model for the sample object x.

An optimal model is the one that minimizes the average loss defined by $L(c, y)$ for all samples in the training set, weighted by their *posterior* probability. The posterior probability, $P_c(y|x)$, is defined as the probability of class y being the label of sample x. For many application specific problems, $c = \Gamma(x)$ is a non-deterministic function. That is, if x is sampled repeatedly, different values of c may be given. In this case, the optimal choice of the label for sample object x among all candidate labels is the label y_{ml} that minimizes the expected loss for a given sample x, i.e., $\forall y = f(x), \exists y_{ml} = f(x), y_{ml} \neq y$, s.t. $E_c(L(c, y_{ml})) \leq E_c(L(c, y))$.

The decision tree is one of the most fundamental inductive learning models, with widespread deployment in big data services and applications in multiple industries, including healthcare cost prediction [1], disease diagnosis [2], [3], computer network analysis [4], and credit-risk assessment [5], [6]. Each of these domains has major privacy concerns from legal constraints around data privacy, e.g., the HIPAA rule for healthcare data [7], to business concerns over revealing their private organizational data to their competitors.

Privacy risks in decision tree learning. Consider a decision tree deployment scenario in the context of automated medical diagnosis service, in which a company that provides automated medical assessments will likely maintain risk profiles for various diseases, including the DNA profiles, medical history, and job-specific information of previously diagnosed individuals. The first privacy concern in such a scenario is the risk profiles, which are sensitive by themselves. The second is the decision tree model built from this sensitive risk profile data as the model often contains sensitive meta-data. Thus, revealing the model can compromise sensitive information and violate certain laws and regulations. The third privacy concern comes from the use of the decision tree model for evaluation. The consumers of such a decision tree learning service may submit either their own risk profile or, in the case of a medical provider, the risk profile of a patient for classification. Without proper protection, this service request leaks the sensitive data of consumers or their patients to the service provider. The classification result can also be highly sensitive. For example, revealing the classification result of a decision tree model for cancer diagnosis would compromise the consumer's privacy. In this scenario it is critical to protect (1) the training dataset, (2) the decision tree model, and (3) the evaluation and testing data and results. The first two properties are the core privacy notions for privacy-preserving decision tree training and the third property is the core privacy notion for privacy-preserving decision tree evaluation.

The notion of privacy-preserving decision tree learning was introduced in the seminal paper [8]. A number of privacy-preserving data mining methods have since been proposed [8], [9], [10], [11], [12], [13], [14], [15], [16], of which some leverage randomization techniques [8], [10], some use other statistical approaches [9], [11], [12], [13], and some leverage cryptographic mechanisms [14], [15], [16]. The cryptographic approaches ensure strong privacy and accuracy via secure multiparty computation (SMC), but typically suffer from poor performance due to high time and computational complexities. The statistical approaches are heuristics-driven and have high performance in comparison. However, existing research efforts in the literature for such statistical approaches demonstrate that the problem of privacy-preserving decision tree learning has yet to be thoroughly understood.

Scope and Contributions of the paper. Bearing in mind the above discussion, in this paper we review and revisit the concepts and techniques of privacy-preserving decision tree based inductive learning. This includes the sensitivity of private input data and the potential privacy risks induced by inference over both the learning output and the intermediate results of inductive learning iterations. We discuss state-of-the-art results in privacy-preserving decision tree learning across three broad categories: (1) randomization-based obfuscation, (2) differential privacy, and (3) secure multiparty computation. We review representative approaches in all three areas and highlight open challenges. We conjecture that our study will help developing and delivering privacy-preserving decision tree learning as a service for many privacy sensitive big data science and engineering applications.

II. Overview

The ultimate goal of privacy-preserving inductive learning is to protect the input data from unwanted and unauthorized leakages and risks that intrude individual privacy and organizational privacy during learning processes. This includes both the construction of decision trees (*privacy-preserving decision tree training*) by the inductive learner and the evaluation of the decision tree model by its authorized users and applications (*privacy-preserving decision tree evaluation*).

A. Privacy Notion and Privacy Risks

The notion of privacy with respect to decision tree learning should address three types of privacy risks in the life cycle of a decision tree classifier due to unwanted inference attacks: (1) the unwanted leakage of the private input data used in the training phase of a decision tree classifier, (2) the potential privacy risk of public release of the decision tree model, (3) the potential privacy risk involved in using the decision tree classifier for evaluating classification queries in the testing phase of the decision tree model. The first two types of privacy risks refer to the inference problems that may cause the leakage of private input data from the training dataset used for the decision tree model construction. The third type refers to information leakage during evaluation of decision tree classification queries when the private data is used as the classification query input and the decision tree classifier hosted at the server is similarly private.

Privacy Risk of Input Data. Among the potential privacy risks in privacy-preserving decision tree learning, the inference of the private input data samples is at the center of the problem. Existing proposals to combat this risk through statistical techniques have primarily centered on data obfuscation methods. The input data traditionally undergoes this data obfuscation and is then released as training data to the inductive learner. The two most prevalent obfuscation methods are randomization [8], [17], [18], widely used in the context of statistical disclosure control [19], [20], and differential privacy. Differential privacy as a privacy framework was initially introduced by Cynthia Dwork in [21] and has attracted a huge amount of attention in recent years [22], [23], [24], [25].

Alternatively, the cryptographic approaches proposed for privacy-preserving inductive learning are based on secure multiparty computation (SMC) protocols [26]. The SMC framework addresses the following question: How can m parties ($m \geq 2$) conduct a computation based on their private inputs when no party is willing to disclose its own input to any of the other parties? This problem is easily solved if one assumes the existence of a trusted third party, but such a trusted party is often not available in practice. The open challenge for SMC is how to design protocols that can support joint computations while protecting the participants' privacy in a dynamic and possibly malicious environment without a trusted third party. According to Goldreich in [27], although the general SMC problem is solvable in theory, the solutions derived from the general theory can be impractical in terms of efficiency for domain specific applications. Thus, special high performance solutions should be developed for these application-specific cases.

SMC is particularly suitable to the distributed and federated machine learning scenarios in which each organization wants to keep their own data private but all parties wish to jointly learn a decision tree model across their private datasets. The resulting decision tree may then be used internally by each organization for evaluation tasks.

Privacy Risks of Public Release of Decision Trees. In addition to the privacy risks of input data exposure discussed above, another subtle concern is that the public release of the learned decision tree model may be subject to inference attack, which can lead to the compromise of both the privacy of individual users represented in the training data and the privacy of organizations involved in distributed/federated learning. Recent study in [28] and [29] demonstrate the feasibility of model inversion attacks in the context of genomic privacy and image recognition privacy respectively. An adversarial client may be able to estimate aspects of the genotypes of some users involved in the training dataset simply using black-box access to the prediction models.

Similarly, an adversary may infer sensitive information with no or little false positives in the white-box setting, in which the inductive learning model and its associated statistical parameters are made publicly available.

These studies have demonstrated that confidence information in conjunction with access to the decision tree model can be exploited by adversarial clients to mount model inversion attacks. Two important lessons can be learned from these studies: (1) publishing decision tree models and associated statistical features can pose unwanted privacy risks for those contributing to the training data and (2) when the models built from inductive learning contain certain statistical features with high feature confidentiality, such models and associated statistical features should be carefully guarded against unwanted privacy risks.

Privacy Risk against Decision Tree Evaluation. The third type of privacy risk arises when the private decision tree model is hosted at a server which a client may query with her private input data. If the server is not properly guarded, the training data or query logs can be leveraged by insider attacks or exposure due to system compromises. We refer to this problem as the privacy-preserving decision tree evaluation problem. A decision tree evaluation is considered privacy preserving if and only if the server hosting a trained decision tree learns nothing about the input data or the classification result of its clients and any client who submits a classification query learns nothing about the decision tree model other than the classification result from the server. It is important to note that, under this construction, the knowledge gained by the client includes any information which can be directly inferred from the model output. While it is ideal to limit the amount of information a client should be able to infer, we will see that leakage in this regard varies depending on the privacy framework.

B. Privacy Protection: Generalization of Approaches

When considering privacy-preserving decision tree processes we must consider what is being protected, what level of protection is provided, and against whom this protection is guaranteed. Across the three frameworks for privacy-preserving machine learning, randomization-based obfuscation, differential privacy, and secure multiparty computation, each contribute different approaches to these considerations. Table I highlights some representative works for each of these approaches and the treatment of privacy, efficiency, and accuracy considerations within each work.

We can see at a high level that randomization and differential privacy techniques generally require access to the data during the training phase. Afterward, the evaluation phase is essentially trusted to be in the clear as the generated model is perturbed to protect the underlying training data. This leads to relatively fast approaches (statistical operations) but limits the resulting accuracy of the model. Comparatively, secure multiparty computation results usually protect all data elements throughout the life cycle of decision tree learning and allow for accuracy equivalent to that seen in privacy-free environments. These gains are made at significant efficiency costs as secure multiparty computation requires either large finite field operations or homomorphic encryptions.

III. Randomization based Privacy Models

Randomization based techniques have been popular in privacy preserving data mining research for more than a decade [8], [9], [10], [11], [12], [13], with randomized distortion being one of the dominant techniques. The privacy preserving inductive learning algorithms belonging to the randomization group work by first perturbing the training data using randomization techniques and then using the distorted data as input to extract patterns and models. The randomization methodology attempts to hide the sensitive information from data miners by randomly modifying the data values under certain model-specific constraints, e.g., preserving the underlying probabilistic properties.

The most straightforward randomization technique is to preserve data privacy by adding random noise while ideally maintaining data utility. On one hand, the random noise should be added in such a way that the individual data values are distorted. On the other hand, with added random noise, the perturbed dataset should still preserve the model-specific feature signals from the data, such underlying distribution properties, so that the model specific data utility patterns can still be "observed" and estimated with certain accuracy. An example of value distortion is additive noise based perturbation, which takes a training dataset of n samples $\{x_1, \dots, x_n\}$ and returns a perturbed dataset in which value x_i is replaced by $x_i + r_i$ ($i = 1, \dots, n$), where r_i is a random value drawn from some distribution. Alternatively, one can also use random data swapping (random shuffle) techniques to design a value distortion function such that a value returned from a field of a data sample record is still a true value in the training dataset, but from the same field in some other randomly chosen record [30].

However, randomization techniques may suffer from both low accuracy and inference attack vulnerability problems. With respect to accuracy, when the model-specific data utility is defined by latent and complex hidden statistical features, the accuracy of randomization methods will suffer significantly due to the lack of support for latent feature understanding and extraction. At the same time, several open challenges have been put forward with respect to the vulnerabilities of randomization techniques. As pointed out in [31], when the noise added can be statistically separated from the perturbed data, the input data privacy can be seriously compromised by inference over the model and its classification output. It is also true that when the dataset has significant skewedness, random data shuffling may no longer protect the individuals in the training dataset.

		random perturbation		differential privacy			secure multiparty computation		
comparison metric		Mining w/ Random Noise [8]	Random Decision Tree Framework [10]	SuLQ Framework [11]	ID3 w/ DP [12]	Random Forest w/ DP [13]	Decision Tree: Evaluate [14]	Decision Tree w/ TI [15]	Secure Tree Learn [16]
privacy metrics	input data during training	✓ perturbed	✓ original	✓ perturbed	✓ perturbed	✓ perturbed			✓ original
	input data during evaluation						✓ original	✓ original	
	learning model construction process	public	private	public	public	public	public	public	public
	decision tree produced by inductive learner		✓				✓	✓	
	output of learner						✓	✓	
computational complexity requirements		statistical operations	homomorphic encryptions	statistical operations	statistical operations	statistical operations	homomorphic encryptions	large finite field operations	large finite field operations
level of accuracy loss		$0 - 10\%$	$0 - 10\%$	$\geq 10\%$	$\geq 10\%$	$\geq 10\%$	none	none	none

Table I
HIGH LEVEL OVERVIEW OF PRIVACY-PRESERVING DECISION TREE TECHNIQUES

To address the additive random noise problem, multiplicative noise based random perturbation techniques are proposed, including geometric rotation based technique [32], [33] and random project-based approach [34].

In addition, we are also concerned with the unwanted exposure of statistical features that are repeatedly used for prediction. Random forest and random decision trees are candidate randomization techniques for providing baseline privacy protection of decision tree models. Random forests are an ensemble learning method that use multiple decision trees and outputs the class that is the mean prediction of the individual trees, aiming at correcting for decision trees which over-fit their training set [35]. The first algorithm for random decision forests was developed by Tin Kam Ho [36] who uses the random subspace method to implement the ”stochastic discrimination” approach to classification [37]. Random forests was later trademarked using an extension with bagging developed by Leo Breiman [38] and others.

Random decision trees represent another orthogonal dimension of efforts. A random decision tree is a variant of the traditional decision tree proposed by Fan et al. [39] where nodes split on randomly chosen features rather than features chosen using statistical measures. The main idea of random decision trees is to build the structure of N random decision trees solely based on the given feature set without using the training data samples. Each discrete feature can be used only once in a decision path from root to a leaf node. Continuous features can be either discretized and treated as discrete features or alternatively be divided by randomly picking a splitting point (i.e., the dividing value with less than and greater or equal to as the two branches) with a different splitting point chosen each time. After the trees are constructed, we only need to scan the dataset once to update the statistics of the leaf nodes in each tree. Random decision trees are shown to be efficient implementations of Bayes’ Optimal Classifier. With large datasets, random decision trees are shown to be as accurate as the single best decision tree carefully built using statistical criteria simply because a single decision tree will often either significantly over-estimate or under-estimate [39]. Interestingly, random decision trees can be much more robust against certain privacy risks of public release of decision tree models.

In summary, randomness and uncertainty may not be equivalent for all cases. If random events and their properties can be captured and analyzed by probabilistic theorems and if we can easily and intuitively interpret probabilistic characterization of random processes, then the randomness may be exploited to compromise privacy unless one pays extra-careful attention to the exposure of such structures.

IV. DIFFERENTIAL PRIVACY

In contrast to randomization based approaches, differential privacy approaches address the input data privacy of decision tree learning by using the “hiding in the cloud” principle, implying that every data sample is equally important to the inductive learner. Intuitively, an inductive learning algorithm K is considered *differentially private* if an adversary cannot differentiate between trees T and T' trained on datasets D and D' respectively, i.e., $K(D) = T$ and $K(D') = T'$, given D and D' differ by, at most, one training sample. Formally, Dwork [40] defines differential privacy, and specifically ε-differential privacy, as follows:

Definition 1: A randomized function K gives **ε-differential privacy** if for all datasets D and D' differing on at most one row, and all $S \subseteq Range(K)$,

$$Pr[K(D) \in S] \leq e^{\varepsilon} \times Pr[K(D') \in S],$$

where the probability space in each case is over the coin flips of K.

This definition shows that data privacy within the differential privacy framework is quantified by a value ε, where ε is typically small, e.g., $\varepsilon = 0.1$, and the smaller the value of ε, the higher the privacy guarantee.

In practice, ε-differential privacy is achieved by adding noise from the Laplace distribution per the following [40]:

Theorem 1: For $f : D \rightarrow \mathbb{R}^d$, the mechanism that adds independently generated noise with distribution $Lap(\Delta f / \varepsilon)$ to each of the d output terms enjoys ε-differential privacy.

Theorem 1 states that the higher the privacy guarantee is, i.e., the smaller the value of ε, the more noise is generally required to achieve differential privacy.

When using the differential privacy framework to support privacy-preserving decision tree learning, the choice of ε represents a significant trade-off. More noise is likely to decrease the accuracy and usability of the resulting decision tree, while less noise will decrease privacy. Additionally, one must consider the number of queries which are made against the data in creating any high-level structure [9]. While differential privacy allows the "release of coarse-grained information while keeping private the details" [40], there is a constant trade-off between the level of privacy and the usability of that information.

There exist multiple differential privacy solutions for decision tree learning including [11], [12], [9], [41], [42], [13] with each taking a slightly different approach to achieving a differentially private learning algorithm.

The first theoretical contribution made by Blum et al [11], built a differentially private ID3 decision tree through differentially private queries to the original dataset using the proposed SuLQ framework (SubLinear Queries). This approach, however, suffers from poor accuracy [12], [9]. This illustrates two important points: (1) the addition of noise in privacy-preserving decision tree training can dramatically impact the accuracy of the generated model and therefore solutions within the differential privacy framework should always be tested experimentally, and (2) for the resulting model to be usable, privacy must be guaranteed in the presence of reasonable query counts on the model structure.

Friedman et al. [12] extends the above approach [11] from two aspects: (1) it ensures that each step in the inductive learning process for decision tree training should be differentially private using mechanisms from [43], [44] and (2) it experimentally highlights the tension between privacy, accuracy, and dataset size.

To better address the trade-offs between privacy, accuracy, and dataset size, Jagannathan et al. leverage a random decision trees in an ensemble instead of relying on the traditional decision tree. As the structure of each tree is randomly chosen, no privacy protection is required for the tree structure. Thus, the noise traditionally incorporated in differential privacy solutions to the leaf nodes. Jagannathan et al. show that an ensemble of these random decision trees with noisy leaf nodes can perform well when compared to single decision trees trained without privacy.

Several threads of extensions have been made to the approach by Jagannathan et al.: Bojarski et al. similarly utilize the random decision tree structure with noise restricted to the class distributions within the leaf nodes in [41]. Patil and Singh [42] extend [9] by using multiple noisy trees to combat decreased accuracy. Rana et al. [13] use the step-wise differential privacy technique in [12] to train a random forest model by distributing the privacy parameter amongst the decision trees within the ensemble, thus making each individual tree less noisy while preserving the privacy of the entire ensemble.

V. Secure Multiparty Computation

The problem of secure multiparty computation relies on what is referred to as the simulation paradigm to allow multiple parties to perform computation across their private inputs without revealing them. The simulation paradigm formalizes the following idea: if every piece of information revealed to a certain party throughout the execution of a protocol can be computed using only that party's inputs and outputs than the protocol is secure.

Specifically, we consider an *ideal execution* of a given computation to which we compare the *real execution* of the proposed protocol. In the ideal execution, we assume each party can send its inputs over a perfectly private channel to a trusted party. This trusted party will then perform the desired computation and return the appropriate output. In this scenario, if we assume the existence of this completely trusted third party and perfectly private channels, privacy will hold because no party ever receives any message other than its intended output.

We now consider an adversary which attacks the real protocol execution. For any adversary which can be constructed for the real protocol execution, we want to construct adversary against the ideal execution such that the input and output distributions seen by both the adversary and the honest parties in the real protocol execution are equivalent to the distributions seen by the adversary and honest parties in the ideal execution. This is referred to as the *ideal/real simulation paradigm*. For further discussion on the ideal/real simulation paradigm, we direct the interested reader to [45].

When considering security based in the simulation paradigm with respect to privacy-preserving decision tree learning, we must keep in mind that security under the ideal/real simulation paradigm works in isolation. While one execution may be considered secure within the simulation paradigm and information leaked from one single output may be acceptable, repetitions can lead to greater information leakage. Consider the following example: Alice and Bob would like to know which one of them earns a higher salary, but neither wishes to reveal their exact salary. Alice and Bob can participate in a protocol π which allows each to provide their salaries as input and receive as output the name of the individual with the higher salary. We will assume that π is secure under the ideal/real simulation paradigm. Given this scenario, Alice can run π repeatedly, altering her input each time, to discover Bob's exact salary. Since Alice could lodge this same attack on the ideal execution, π remains a secure computation under ideal/real simulation paradigm. This weakness however leads to the leakage of Bob's private information. Researchers in privacy-preserving decision tree learning must consider the scenario where the tree structure is used to evaluate different instances many times over.

There are a number of building blocks which are commonly used in secure multiparty computation including oblivious transfer, Yao's garbled circuits, Shamir's secret sharing scheme, the trusted initializer in the commodity-based cryptography model, and additively homomorphic encryption schemes.

Private Multiparty Training of Decision Trees. The first to propose a privacy-preserving training solution for decision trees were Lindell and Pinkas in [46]. They proposed a secure two-party protocol for training an ID3 tree over horizontally partitioned data.

The most efficient secure multiparty protocol for the privacy-preserving training of decision trees is proposed by de Hoogh et al in [16]. This solution is implemented within the Virtual Ideal Functionality Framework (VIFF) using operations over large finite fields based on Shamir's secret sharing scheme [47] and provides provable security against semi-honest adversaries. As noted by [15] however, this result is limited to the consideration of categorical attributes and does not scale well for fine grained numerical attributes. Additionally, the complexity of the protocol increases exponentially on the bit-length representation of a category.

Private Evaluation of Decision Trees. There exist multiple secure multiparty computation solutions for decision tree evaluation including [14], [48], [15], [16]. Decision tree classification is traditionally a two-party scenario where there is a server which holds the machine learning structure, a decision tree, and a client which holds the classification instance. At the end of a secure evaluation protocol the ideal outcome would give the client the classification result of their instance based on the server's decision tree without revealing any information on the server's model while the server would learn nothing.

Three representative approaches have been studied. The first approach provides solutions for decision tree evaluation which use additive homomorphic encryption schemes and the oblivious transfer scheme of Naor and Pinkas [49] to allow for computation over the private data. [48] provide privacy against the weaker semi-honest adversarial model, and [14] offers solutions under both the semi-honest and malicious adversarial models. The second approach [15] uses Shamir's secret sharing scheme in the commodity-based model and provides unconditionally secure protocols for decision tree evaluation in the presence of semi-honest adversaries and are able to leverage the set-up phase with a trusted intializer to make evaluation more efficient. The third approach [50] uses trusted SGX processors for data-obivious decision tree evaluation.

VI. Concluding Remarks and Open Challenges

Risks of Reverse Engineering. We describe two scenarios of reverse engineering risks that can be detrimental even under the secure multiparty computation privacy framework.

Scenario 1: Two parties, Alice and Bob, jointly train a decision tree model with each providing their own private input data. Alice can then independently execute the decision tree training algorithm with the same parameters used in the joint training to see how Bob's dataset impacted the decision tree structure. Black-box access to Bob's data in the joint training can easily lead to the unwanted privacy leakage as Alice is able to infer detailed statistics with respect to Bob's private training data through careful reverse engineering.

Scenario 2: An adversarial client may query a decision tree repeatedly with cleverly engineered classification instances and, using only the classification results and a priori information on the decision tree algorithm, and may reverse-engineer the decision tree itself.

Beyond the loss of privacy for the decision tree structure, this leakage also reveals some amount of information about the underlying data on which the tree was trained. In some domains, this data leakage can be even more alarming than the model leakage. Solutions to privacy-preserving decision tree learning using secure multiparty computation technique often suffer from this pitfall of black-box access to computation results.

Computation Cost. Secure multiparty computation protocols routinely require expensive cryptographic operations, either with data encrypted with public key cryptography schemes [14], [48], [51], [52], or across a large finite fields [15], [53], [16]. Additionally, secure multiparty computation requires many back and forth exchanges between parties when compared to performing operations in the clear. Due to relatively high computation and communication costs, an open question is whether secure multiparty computation solutions are truly feasible for real world implementation. We argue that high performance SMC protocols can be a breakthrough for wide deployment of privacy preserving distributed decision tree ensemble learning.

Incorporating Different Trust and Sensitivity Levels. Currently, solutions across randomization, differential privacy, and secure multiparty computation models treat all data samples and all participating organizations equally as homogeneous entities. However, this is not the case in practice. Quite often, different organizations may want to add different levels of noise based upon different levels of trust or sensitivity. One party may only agree to share information which is an aggregation across the data of many parties to protect against revealing or intruding the privacy of any individual. Another party may only agree to share information about those less sensitive data elements or model parameters. Some parties may even abstain from sharing results to certain users. Each of these scenarios calls for variation in privacy preserving treatment in order to match the varying sensitivity of different data elements and the varying trust in participating parties.

Combining SMC with Differential Privacy. There is, to date, limited work in combining secure multiparty computation with differential privacy. We argue that if both techniques were combined, then some previously mentioned pitfalls may be minimized. There are two potential ways to combine SMC and differential privacy: (1) develop an SMC protocol that produces a differentially private decision tree model or (2) use differential privacy techniques to accomplish secure multiparty computation tasks for some data exchanges within the secure multiparty computation protocol where the addition of noise may not impact the accuracy of the resulting decision tree too strongly. Choosing to employ noise injection techniques from the differential privacy domain for some portions of a secure multiparty computation protocol may improve the computational complexity problem which plagues many existing secure multiparty computation solutions. We conjecture that a patchwork implementation working across different privacy models may have the potential to produce solutions that address a broader range of security and privacy issues.

Dynamic and Flexible Collaborative Learning. In a real world collaborative learning environment, not all parties can be or need to be online to participate in the entire work flow of the learning process. Whomever is available at any given time may share information or participate in some subset of protocols, representative of their level of trust in the participating parties or the level of privacy demanded by the kind of input data being used. We argue that enabling such a dynamic collaborative learning environment, where each party may be a part of the learning process when it is convenient and to the level at which it is comfortable for them, though challenging, is a more realistic approach for wide deployment of privacy-preserving decision tree learning solutions for real world applications.

Acknowledgement. This research has been partially support by the National Science Foundation under Grants CNS-1115375, NSF 1547102, SaTC 1564097, and an RCN BD Fellowship, provided by the Research Coordination Network (RCN) on Big Data and Smart Cities. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the RCN or National Science Foundation.

References

[1] S. Sushmita, S. Newman, J. Marquardt, P. Ram, V. Prasad, M. D. Cock, and A. Teredesai, "Population cost prediction on public healthcare datasets," in *Proceedings of the 5th International Conference on Digital Health 2015*. ACM, 2015, pp. 87–94.

[2] A. T. Azar and S. M. El-Metwally, "Decision tree classifiers for automated medical diagnosis," *Neural Computing and Applications*, vol. 23, no. 7-8, pp. 2387–2403, 2013.

[3] A. Singh and J. V. Guttag, "A comparison of non-symmetric entropy-based classification trees and support vector machine for cardiovascular risk stratification," in *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*. IEEE, 2011, pp. 79–82.

[4] M. Antonakakis, R. Perdisci, Y. Nadji, N. Vasiloglou, S. Abu-Nimeh, W. Lee, and D. Dagon, "From throw-away traffic to bots: Detecting the rise of dga-based malware," in *Proceedings of the 21st USENIX Conference on Security Symposium*, ser. Security'12. Berkeley, CA, USA: USENIX Association, 2012, pp. 24–24.

[5] S. Y. Kim and A. Upneja, "Predicting restaurant financial distress using decision tree and adaboosted decision tree models," *Economic Modelling*, vol. 36, pp. 354–362, 2014.

[6] H. C. Koh, W. C. Tan, and C. P. Goh, "A two-step method to construct credit scoring models with data mining techniques," *International Journal of Business and Information*, vol. 1, no. 1, 2015.

[7] A. Act, "Health insurance portability and accountability act of 1996," *Public law*, vol. 104, p. 191, 1996.

[8] R. Agrawal and R. Srikant, "Privacy-preserving data mining," in *ACM Sigmod Record*, vol. 29, no. 2. ACM, 2000, pp. 439–450.

[9] G. Jagannathan, K. Pillaipakkamnatt, and R. N. Wright, "A practical differentially private random decision tree classifier," in *2009 IEEE International Conference on Data Mining Workshops*. IEEE, 2009, pp. 114–121.

[10] J. Vaidya, B. Shafiq, W. Fan, D. Mehmood, and D. Lorenzi, "A random decision tree framework for privacy-preserving data mining," *IEEE Transactions on Dependable and Secure Computing*, vol. 11, no. 5, pp. 399–411, 2014.

[11] A. Blum, C. Dwork, F. McSherry, and K. Nissim, "Practical privacy: the sulq framework," in *Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*. ACM, 2005, pp. 128–138.

[12] A. Friedman and A. Schuster, "Data mining with differential privacy," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2010, pp. 493–502.

[13] S. Rana, S. K. Gupta, and S. Venkatesh, "Differentially private random forest with high utility," in *Data Mining (ICDM), 2015 IEEE International Conference on*. IEEE, 2015, pp. 955–960.

[14] D. J. Wu, T. Feng, M. Naehrig, and K. Lauter, "Privately evaluating decision trees and random forests," *Proceedings on Privacy Enhancing Technologies*, vol. 4, pp. 1–21, 2016.

[15] M. D. Cock, R. Dowsley, C. Horst, R. Katti, A. Nascimento, W. S. Poon, and S. Truex, "Efficient and private scoring of decision trees, support vector machines and logistic regression models based on pre-computation," *IEEE Transactions on Dependable and Secure Computing*, vol. PP, no. 99, pp. 1–1, 2017.

[16] S. de Hoogh, B. Schoenmakers, P. Chen, and H. op den Akker, "Practical secure decision tree learning in a teletreatment application," in *International Conference on Financial*

Cryptography and Data Security. Springer, 2014, pp. 179–194.

[17] P. Samarati, "Protecting respondents identities in microdata release," *IEEE transactions on Knowledge and Data Engineering*, vol. 13, no. 6, pp. 1010–1027, 2001.

[18] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[19] S. L. Warner, "Randomized response: A survey technique for eliminating evasive answer bias," *Journal of the American Statistical Association*, vol. 60, no. 309, pp. 63–69, 1965.

[20] L. Willenborg and T. Waal, "Disclosure risk for tabular data," *Elements of Statistical Disclosure Control*, pp. 137–157, 2001.

[21] D. Cynthia, "Differential privacy," *Automata, languages and programming*, pp. 1–12, 2006.

[22] M. Naveed, E. Ayday, E. W. Clayton, J. Fellay, C. A. Gunter, J.-P. Hubaux, B. A. Malin, and X. Wang, "Privacy in the genomic era," *ACM Comput. Surv.*, vol. 48, no. 1, pp. 6:1–6:44, Aug. 2015.

[23] K. M. P. Shrivastva, M. A. Rizvi, and S. Singh, "Big data privacy based on differential privacy a hope for big data," in *2014 International Conference on Computational Intelligence and Communication Networks*, Nov 2014, pp. 776–781.

[24] G. Bello-Orgaz, J. J. Jung, and D. Camacho, "Social big data: Recent achievements and new challenges," *Information Fusion*, vol. 28, pp. 45 – 59, 2016.

[25] J. Singh, T. Pasquier, J. Bacon, H. Ko, and D. Eyers, "Twenty security considerations for cloud-supported internet of things," *IEEE Internet of Things Journal*, vol. 3, no. 3, pp. 269–284, June 2016.

[26] A. C. Yao, "Protocols for secure computations," in *Foundations of Computer Science, 1982. SFCS'08. 23rd Annual Symposium on*. IEEE, 1982, pp. 160–164.

[27] O. Goldreich, "Secure multi-party computation," *Manuscript. Preliminary version*, pp. 86–97, 1998.

[28] M. Fredrikson, E. Lantz, S. Jha, S. Lin, D. Page, and T. Ristenpart, "Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing." in *USENIX Security*, 2014, pp. 17–32.

[29] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2015, pp. 1322–1333.

[30] V. Estivill-Castro and L. Brankovic, "Data swapping: Balancing privacy against precision in mining for logic rules," in *International Conference on Data Warehousing and Knowledge Discovery*. Springer, 1999, pp. 389–398.

[31] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar, "On the privacy preserving properties of random data perturbation techniques," in *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*. IEEE, 2003, pp. 99–106.

[32] K. Chen and L. Liu, "Privacy preserving data classification with rotation perturbation," in *Data Mining, Fifth IEEE International Conference on*. IEEE, 2005, pp. 4–pp.

[33] ——, "A survey of multiplicative perturbation for privacy-preserving data mining," *Privacy-Preserving Data Mining*, pp. 157–181, 2008.

[34] K. Liu, H. Kargupta, and J. Ryan, "Random projection-based multiplicative data perturbation for privacy preserving distributed data mining," *IEEE Trans. on Knowl. and Data Eng.*, vol. 18, no. 1, pp. 92–106, Jan. 2006.

[35] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Springer series in statistics Springer, Berlin, 2001, vol. 1.

[36] T. K. Ho, "Random decision forests," in *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on*, vol. 1. IEEE, 1995, pp. 278–282.

[37] E. Kleinberg *et al.*, "An overtraining-resistant stochastic modeling method for pattern recognition," *The annals of statistics*, vol. 24, no. 6, pp. 2319–2349, 1996.

[38] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[39] W. Fan, "On the optimality of probability estimation by random decision trees," in *AAAI*, vol. 2004, 2004, pp. 336–341.

[40] C. Dwork, "Differential privacy," in *Encyclopedia of Cryptography and Security*. Springer, 2011, pp. 338–340.

[41] M. Bojarski, A. Choromanska, K. Choromanski, and Y. LeCun, "Differentially-and non-differentially-private random decision trees," *arXiv preprint arXiv:1410.6973*, 2014.

[42] A. Patil and S. Singh, "Differential private random forest," in *Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on*. IEEE, 2014, pp. 2623–2630.

[43] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography Conference*. Springer, 2006, pp. 265–284.

[44] F. McSherry and K. Talwar, "Mechanism design via differential privacy," in *Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on*. IEEE, 2007, pp. 94–103.

[45] C. Hazay and Y. Lindell, *Efficient secure two-party protocols: Techniques and constructions*. Springer Science & Business Media, 2010.

[46] Y. Lindell and B. Pinkas, "Privacy preserving data mining," in *Annual International Cryptology Conference*. Springer, 2000, pp. 36–54.

[47] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[48] R. Bost, R. A. Popa, S. Tu, and S. Goldwasser, "Machine learning classification over encrypted data." in *NDSS*, 2015.

[49] M. Naor and B. Pinkas, "Oblivious transfer and polynomial evaluation," in *Proceedings of the thirty-first annual ACM symposium on Theory of computing*. ACM, 1999, pp. 245–254.

[50] O. Ohrimenko, F. Schuster, C. Fournet, A. Mehta, S. Nowozin, K. Vaswani, and M. Costa, "Oblivious multi-party machine learning on trusted processors," in *USENIX Security*, vol. 16, 2016, pp. 619–636.

[51] M.-J. Xiao, L.-S. Huang, Y.-L. Luo, and H. Shen, "Privacy preserving id3 algorithm over horizontally partitioned data," in *Parallel and Distributed Computing, Applications and Technologies, 2005. PDCAT 2005. Sixth International Conference on*. IEEE, 2005, pp. 239–243.

[52] S. Samet and A. Miri, "Privacy preserving id3 using gini index over horizontally partitioned data," in *Computer Systems and Applications, 2008. AICCSA 2008. IEEE/ACS International Conference on*. IEEE, 2008, pp. 645–651.

[53] Q. Ma and P. Deng, "Secure multi-party protocols for privacy preserving data mining," in *International Conference on Wireless Algorithms, Systems, and Applications*. Springer, 2008, pp. 526–537.