A Utility-aware Visual Approach for Anonymizing Multi-attribute Tabular Data

Xumeng Wang, Jia-Kai Chou, Wei Chen*, Huihua Guan, Wenlong Chen, Tianyi Lao, and Kwan-Liu Ma

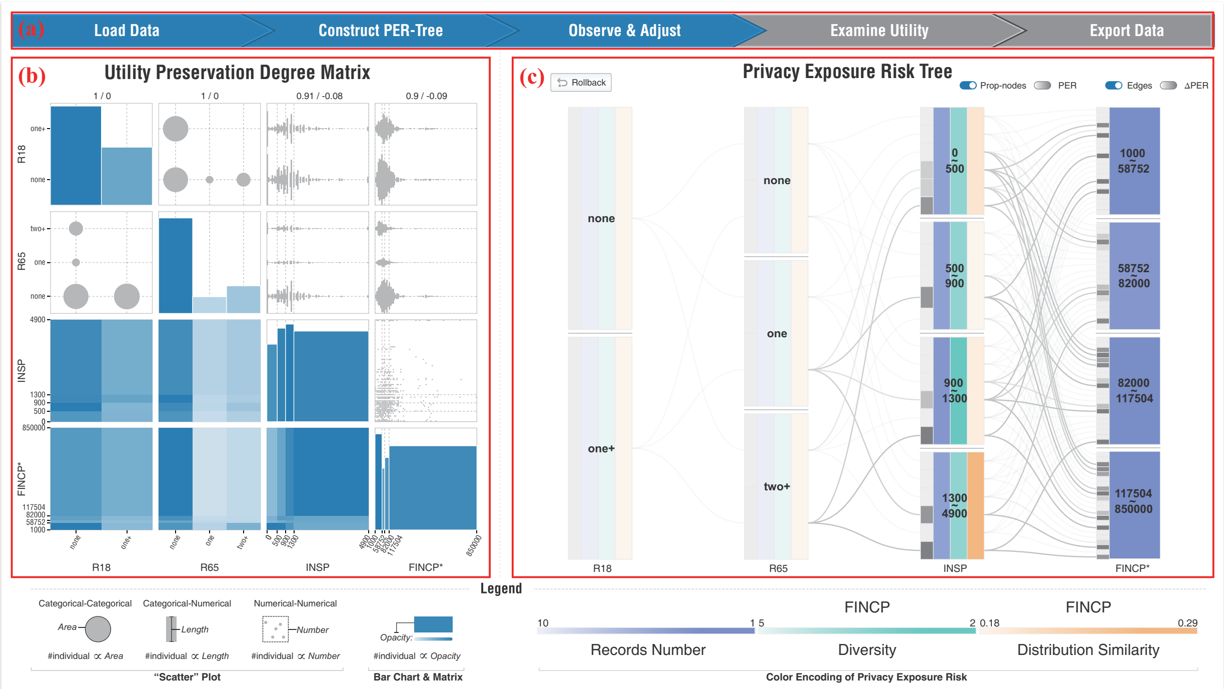


Fig. 1. Our utility-aware visual data-anonymizing process follows (a) a 5-step pipeline and is facilitated with two main visualization components: (b) utility preservation degree matrix (UPD-Matrix) and (c) privacy exposure risk tree (PER-Tree). The PER-Tree helps our users identify privacy issues in the underlying data and provides interactions to address the detected privacy issues. The UPD-Matrix presents the difference between the processed data and the original data. Users can use the chart to examine how utility of data changes during the anonymization process.

Abstract— Sharing data for public usage requires sanitization to prevent sensitive information from leaking. Previous studies have presented methods for creating privacy preserving visualizations. However, few of them provide sufficient feedback to users on how much utility is reduced (or preserved) during such a process. To address this, we design a visual interface along with a data manipulation pipeline that allows users to gauge utility loss while interactively and iteratively handling privacy issues in their data. Widely known and discussed types of privacy models, i.e., syntactic anonymity and differential privacy, are integrated and compared under different use case scenarios. Case study results on a variety of examples demonstrate the effectiveness of our approach.

Index Terms—Privacy preserving visualization, utility aware anonymization, syntactic anonymity, differential privacy

1 Introduction

As technology advances, organizations and corporations can easily collect vast amounts of data from their users/customers and store them as multi-attribute tables. Finding correlations between data attributes is one fundamental analytics task for this type of data as it can lead to better decision-making. For example, analyzing the medical records from a group of patients may help improve the accuracy of diagnosis and treatment. While making datasets publicly available or accessible to external users, such as collaborators, certainly has its benefits, the risk of potentially exposing sensitive information often deters a data owner from sharing without restriction.

Traditionally, attributes in a dataset that can be used to directly or uniquely specify an individual's identity, such as a person's name, are anonymized or removed for the purpose of privacy protection. Unfortunately, by utilizing side information and/or the categorical or schematic information of the data, sometimes targeted individuals can still be secluded or even re-identified from the data [35]. More advanced techniques have thus been developed for handling privacy issues under different types of assumptions.

In general, privacy preservation inevitably comes at the cost of data

- *Xumeng Wang, Wei Chen, Huihua Guan, Wenlong Chen and Tianyi Lao are with Zhejiang University. E-mail: wangxumeng@zju.edu.cn; chenwei@cad.zju.edu.cn; {ghh, chenwenlong, laotianyi}@zju.edu.cn. Wei Chen is corresponding author.*
- *Jia-Kai Chou and Kwan-Liu Ma are with University of California, Davis. E-mail: jkchou@ucdavis.edu, ma@cs.ucdavis.edu.*

Manuscript received 31 Mar. 2017; accepted 1 Aug. 2017.
Date of publication 28 Aug. 2017; date of current version 1 Oct. 2017.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TVCG.2017.2745129

utility. That is, when data is removed, obfuscated, or hidden, it becomes less useful for analysis, exploration, and discovery. Prior studies (such as [20, 24, 27, 31, 33]) have proposed various ways to measure utility. In some cases, it is crucial to inform a data owner how to augment a dataset because privacy affects its utility. That gives the motivation for our work, designing a system for balancing these considerations.

Our solution is a visual interface and privacy preservation pipeline which allows users to interactively and iteratively resolve the privacy issues while still taking data utility into account. In particular, we employ and incorporate commonly used syntactic anonymization models, namely k-anonymity [36], l-diversity [26], and t-closeness [23], as well as two differential privacy algorithms [6, 15], namely Laplace mechanism [13] and exponential mechanism [28], for detecting and handling the privacy issues in multi-attribute tabular datasets. In order to assist users to not only identify the privacy exposure risk in the data, but also address those issues with suitable techniques, we integrate the essence of these models into a Privacy Exposure Risk Tree (PER-Tree). Giving full play to the advantages of visual approaches, the PER-Tree further expands the processing approaches of the original models. In addition, we present a matching design called the Utility Preservation Degree Matrix (UPD-Matrix), which provides users with visual feedback on how utility is changed as privacy preserving operations are applied. We demonstrate the effectiveness of our system through several use case scenarios. Feedback from potential users of our system with domain expertise is also discussed.

2 Related Work

We review related work in the following aspects: (1) privacy preserving models, (2) trade-off between privacy and utility, and (3) visualizations that take privacy into consideration.

2.1 Privacy Models

Syntactic anonymity and differential privacy are two types of commonly used privacy models that address privacy issues in different perspectives.

k-anonymity [36] is one of the representative syntactic anonymity models. To satisfy *k-anonymity*, each data record should have at least $k-1$ other records that share the same set of values in the quasi-identifier fields [9], thus forming equivalence classes of at least size k. The concept of *k-anonymity* is useful when applied as an anonymization measure against identity disclosure. However, it does not take the diversity of the sensitive attribute into account. An attacker is still able to reveal certain individuals' sensitive information if those individuals who have the same quasi-identifier information also obtain similar or even the same sensitive attribute values. The *l-diversity* [26] model was then proposed to address such an issue. In *l-diversity* [26], the data records in the same equivalence class are required to obtain l different values in the sensitive attribute. *t-closeness* [23] is another model designed to extend *k-anonymity* with a slightly different strategy. Instead of enforcing the number of different sensitive values in an equivalence class, it targets on maintaining the distribution of sensitive values in each equivalence class to be similar enough (smaller than a threshold t) as compared to the global distribution of the sensitive attribute. There have been many other syntactic anonymity models, such as p-sensitive [37] and β-likeness [5], developed based on the three aforementioned models.

Unlike syntactic anonymity models, which are usually used for privacy preserving data publishing, differential privacy models are mainly applied to anonymize query responses [34]. In accordance with the definition of differential privacy, a function K is said to be differentially private if any subject's participation or absence of a dataset would not significantly affect the result of the output. Essentially, a differential privacy mechanism is achieved by adding random noise chosen from an appropriately determined distribution to the true query result.

2.2 Trade-off between Privacy and Utility

As performing privacy preserving operations to the data inevitably leads to some loss of utility, how to effectively maintain and measure the utility of data has been a challenging and widely studied problem.

Calculating the sum [38] or the average [22] of the interval size in the equivalence classes, reflecting the loss of information, is commonly adopted for measuring utility when syntactic anonymity models are applied. Using average interval size as the utility metric, Loukides and Shao [25] proposed a clustering algorithm of data records that finds an optimal trade-off between privacy and utility within the predefined parameter space. In some other cases [4, 32], utility is interpreted as how accurate/precise the results of the intended computation or data analysis algorithms can be carried out after data being anonymized.

In differential privacy, a majority of research considers utility as the distance between the queried results and their real values. Alvim et al. [2] proposed an information-theoretic framework to quantify both information leakage and utility. In [17], Ghosh et al. presented an approach of how an optimal (utility-maximizing) geometric mechanism can be found for answering fixed counting queries. In order to adapt to mathematical analysis, Kifer and Lin [21] presented formally defined axioms, called "Generality Axiom", for measuring the privacy and utility in the context of applying differential privacy. More recently, Hong et al. [18] proposed a framework that allows for collaborative search log generation while satisfying differential privacy and maintaining reasonable output utility.

2.3 Privacy-aware Visualizations

The concept of privacy preserving has been considered in the visualization community. In [11], Dasgupta and Kosara discuss the strategies of applying syntactic anonymization approaches, i.e., *k-anonymity* and *l-diversity*, when multi-dimensional data is presented with parallel coordinates. Particularly, their privacy preserving approach is called "Screen-Space Sanitization", which essentially introduces visual uncertainty to the parts of the visualization where privacy issues exist. While adding visual uncertainty can affect both privacy and utility, Dasgupta et al. [10] summarized a series of quantification methods for assessing the change of privacy and utility from the resulting visualizations. In contrast to addressing privacy issues in the visual space, our approach aims at providing greater flexibility and transparency in the process of balancing between privacy and utility at the data-level with the help of visualization. One other important difference of our method is that the resultant dataset processed by the tool can be exported and used for further analysis.

There also have been visualization studies considering privacy in various other types of applications/data. Oksanen et al. [30] developed a method for generating a privacy preserving heat map, which takes into account the diversity of users in the collected mobile sports tracking data. Andrienko et al. [3] devised a visual analytics approach for supporting privacy preserving analysis of mobility diaries collected from a massive population. In [7, 8], Chou et al. designed interactive visual interfaces for addressing privacy issues in different types of data, such as event sequence data [8] and graph [7]. Their systems provide users with necessary visualization assistance and the ability to not only examine privacy issues, but also to decide how to apply the desired privacy preserving operations under different circumstances. Our approach focuses on bridging the gap for users who need more awareness of how much data utility has been compromised while certain level of data privacy is guaranteed.

3 Privacy models and utility metrics

We introduce the privacy models and utility quantification approaches employed in our system.

3.1 Privacy Models

Syntactic anonymity and differential privacy are two types of models that address privacy issues in a different perspective.

3.1.1 Syntactic Anonymity models

We employ three common syntactic anonymity models for the purposes of detecting privacy issues and serving as quantifiers to indicate the degree of privacy exposure in each equivalence class.

k-anonymity [36] An equivalence class satisfies *k-anonymity* if it contains at least k data records. Conversely, an equivalence class is considered privacy-exposing if the number of data records it contains, n, is smaller than the user-defined threshold, k. The value of $k-n$ indicates its degree of privacy exposure with respect to *k-anonymity*.

l-diversity [26] An equivalence class satisfies *l-diversity* if it contains at least l different values for the sensitive attribute. Conversely, an equivalence class is considered privacy-exposing if the number of different sensitive values it has, s, is smaller than the user-defined threshold, l. The value of $l-s$ indicates its degree of privacy exposure with respect to *l-diversity*.

t-closeness [23] An equivalence class satisfies *t-closeness* if its distribution of the sensitive attribute is close to the distribution of the sensitive attribute in the entire dataset, i.e., the absolute distance between the two distributions is smaller than t. Conversely, an equivalence class is considered privacy-exposing if the absolute distance between the two distributions, d, is larger than the user-defined threshold, t. The value of $d-t$ indicates its degree of privacy exposure with respect to *t-closeness*.

While privacy issues are identified by the above syntactic anonymity models, we also employ common privacy preserving operations, such as aggregation or generalization, applied in these models.

3.1.2 Differential Privacy Models

Differential privacy models do not assume what the attacker's background knowledge is, thus making it not that suitable to be used as a means for privacy detection. We utilize their privacy preserving mechanisms to provide alternatives for addressing privacy needs.

By definition [12], a function $\mathcal{K}$ is ε-differentially private, if:

$$Pr[\mathcal{K}(D) \in S] \leq exp(\varepsilon) \times Pr[\mathcal{K}(D') \in S]$$

, where D and D', are two datasets differing in at most one row, and all $S \subseteq Range(\mathcal{K})$. As differential privacy models are normally designed to obfuscate query responses, we adaptively apply it to a static table by treating every involved attribute value as a response separately.

Numerous differential privacy approaches have been developed. Among them, the Laplace mechanism adds a random noise generated from a Laplace distribution $Lap(\Delta f_D(x)/\varepsilon)$ to the data, is particularly effective for protecting privacy of numerical query responses [15]. The term $f_D(x)$ represents the attribute value of a data point in the original dataset D while $\Delta f_D(x)$ is the range of the corresponding attribute. In our system, we define $\Delta f_D(x)$ as the range of the attribute values that are among the top 10% closest to $f_D(x)$. For each attribute value $f_D(x)$, we transform it to $f_{D'}(x)$ as follow:

$$\mathcal{M}_L(f_D(x)) = f_D(x) + Y, whereY \sim Lap(\Delta f_D(x)/\varepsilon).$$

For categorical data, the exponential mechanism [28] is considered. A function $\mathcal{M}_E(f_D(x), q, \mathcal{R})$ is ε-differentially private under the exponential mechanism if it outputs an element $f_{D'}(x) \subseteq \mathcal{R}$ ($\mathcal{R}$ represents the range of categorical attribute values) with the probability proportional to $exp(\varepsilon q(f_D(x), f_{D'}(x))/2\Delta q)$, where q is a function defined as:

$$q(f_D(x), f_{D'}(x)) = p_D(f_D(x))$$

, with Δq as the largest possible change in q [14, 28].

3.2 Utility Quantification

To evaluate the utility of an anonymized multi-attribute dataset, we measure the distance between the distribution of each data attribute in the original data and the sanitized data. For any of the attribute values, $f_D(x)$, from the original dataset, we first find its corresponding value, $f_{D'}(x)$, from the sanitized dataset, and exploit a variant of the Earth Movers Distance metric applied in [23] to decide the distance between them. In our approach, numerical data and categorical data are treated differently. In addition, categorical data can be further classified into two sub-categories depending on the presence or absence of the hierarchical structure.

Let P and Q denote the distributions of a numerical attribute in the original dataset and the sanitized dataset, respectively. If the attribute is sanitized by aggregating values into bins, we first transform each attribute value to the mean of its associated bin. Then, we combine all attribute values in P and Q together and sort them by ascending order: $\{v_1, v_2, \cdots, v_m\}$. After that, we calculate the utility for each attribute value as follows:

$$u(f_D(x), f_{D'}(x)) = 1 - \frac{|i-j|}{m-1}$$

, where i and j refer to the sorted index of $f_D(x)$ and $f_{D'}(x)$, respectively.

For a categorical attribute that does not have a pre-defined hierarchy, the utility of an attribute value is defined as a binary:

$$u(f_D(x), f_{D'}(x))_{noInfo} = \begin{cases} 1 & f_D(x) = f_{D'}(x) \\ 0 & f_D(x) \neq f_{D'}(x) \end{cases} .$$

If the hierarchy is provided, we apply the following metric:

$$u(f_D(x), f_{D'}(x))_{Info} = level(f_D(x), f_{D'}(x))/H$$

, where level($f_D(x)$, $f_{D'}(x)$) represents the lowest common ancestor of $f_D(x)$ and $f_{D'}(x)$ and H is the height of the hierarchical tree.

The utility score of the entire dataset (or a subset of data such as a specific subset of attributes or values) is then calculated as the average utility value of the data records involved:

$$U(D, D') = \frac{\sum_{i=1}^{n} u^*(f_D(x), f_{D'}(x))}{n}$$

, where u^* refers to the utility metric for the corresponding attribute type and n is the number of records.

4 System Overview and Privacy Preserving Pipeline

The primary goal of our system is to help users balance between privacy protection and utility of data. Our target users are data owners who want to keep the sensitive information of their data private while still obtaining the need to share data with others. For example, a senior marketing manager in a company may want a junior data analyst to help conduct analysis on data collected from customers. Due to privacy concern, the data has to be anonymized before handed to the data analyst. In addition, the manager would want that the distortion of data introduced during the anonymization process does not make the resultant data useless. We design a 5-step pipeline, as shown in Fig. 2, that allows users to iteratively and interactively realize their desired trade-off between privacy and utility.

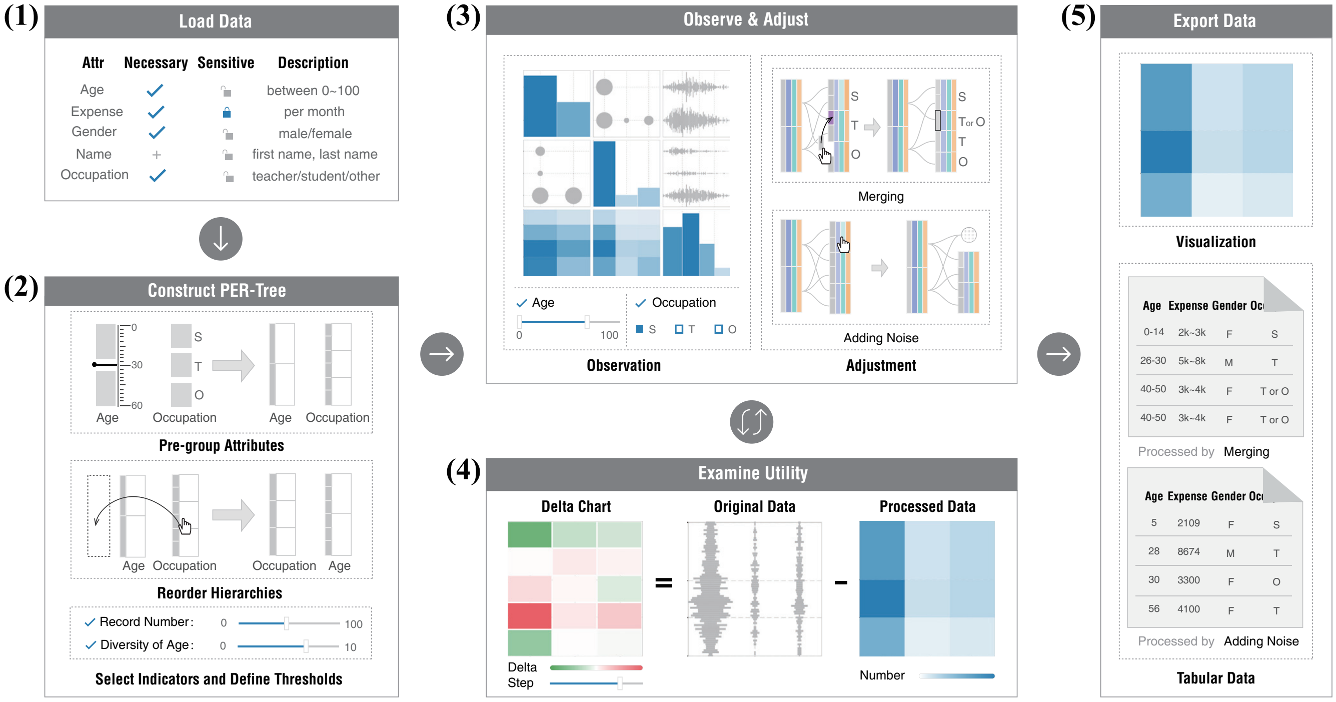
(1) Load Data

Upon loading a dataset, users first decide two things for each attribute in the dataset: 1) should it be involved in the analysis? 2) is it a sensitive attribute that requires privacy protection?

(2) Construct Privacy Exposure Risk Tree (PER-Tree)

Once data loading is completed, users can then go through three sub-steps to construct a Privacy Exposure Risk Tree (PER-Tree). First, users can decide how each dimension of the data should be categorized or aggregated for further analysis or exploration. The system provides assistance by displaying the distribution of each dimension from the original data on the diagonal of the Utility Preservation Degree Matrix (UPD-Matrix). Users can also make the decision based on his/her domain knowledge. While a finer granularity of aggregation is more likely to lead to a more precise analysis later, the possibility of revealing privacy information might also increase. This is a factor that users have to consider.

Next, users can freely switch the order of the attributes to be presented in the PER-Tree. Placing an attribute to a higher level of the



Attr	Necessary	Sensitive	Description
Age	✓	🔓	between 0~100
Expense	✓	🔒	per month
Gender	✓	🔓	male/female
Name	+	🔓	first name, last name
Occupation	✓	🔓	teacher/student/other

Age	Expense	Gender	Oc
0-14	2k~3k	F	S
26-30	5k~8k	M	T
40-50	3k~4k	F	T or O
40-50	3k~4k	F	T or O

Processed by Merging

Age	Expense	Gender	Oc
5	2109	F	S
28	8674	M	T
30	3300	F	O
56	4100	F	T

Processed by Adding Noise

Fig. 2. Our 5-step privacy preserving pipeline: (1) load dataset, then select attributes of interest and define which ones are sensitive; (2) construct the PER-Tree via three sub-steps; (3) review the UPD-Matrix to observe patterns in the data and see the change of utility as data being processed, and make necessary data manipulation for privacy preserving purposes, including merging nodes and adding noise, using the PER-tree; (4) compare and examine the difference of attribute values between the original data and the processed data; (5) export visualization result and/or its underlying data.

tree reduces the number of edges linked to it, and thus leading to less clutter. As a result, users can see more clearly that in what range or for which category does the attribute bear the most privacy issues. We recommend to always put the sensitive attributes to the lowest levels of the tree to avoid further confusion. Detailed explanation will be described in Section 5.1.2.

The last step to construct the PER-Tree is to set the criteria values for the syntactic privacy models so that privacy issues in each dimension and each level of the tree can be detected.

We use an example dataset, shown in Fig. 3(a), to demonstrate how to build up a PER-Tree. Assume the dataset has two attributes: gender and occupation, and their attribute values are [male, female] and [teacher, student, others], respectively. Then, we set the attribute order as gender followed by occupation. As can be seen in Fig. 3(b), the top level of the tree contains two nodes: male (M) and female (F). Each of the top level nodes has three edges link to the nodes at the second level because the second attribute has three different values: teacher (T), student (S), and other (O). Each node is further split into two type of sub-nodes. One is called “Prop-node”, which stores the propagated privacy information from its parent node. The other one is “Attr-node”, which stores attribute-specific privacy information. In Fig. 3(c), we use the “teacher” node as an example and highlight its associated Prop-nodes and Attr-node. The details on the color encoding of the nodes and interactions will be discussed in Section 5.1.2.

(3) Observe & Adjust

After the PER-Tree is constructed, the next step is to look at the data and make necessary adjustments to reach a better balance between the data privacy and utility. To do so, users can start by viewing the UPD-Matrix to find correlation between attributes. Users can also apply different aggregations on certain attributes through interactions on the PER-Tree. Changing the aggregation of attributes in the PER-Tree might result in the following effects: 1) patterns or correlation of the data might vary; 2) privacy and utility might change as well. Another possible interaction can be done on the PER-Tree is to apply differential privacy to address certain privacy issues if aggregation is not desired.

Our system provides a “rollback” function to revoke previously applied operation(s). With such a functionality, users can interactively and iteratively examine how different operations affect privacy and utility, thus obtaining the flexibility and transparency for pursing the most desired balance between privacy and utility.

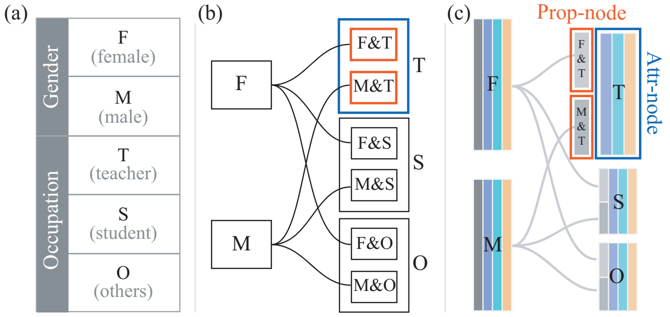


Fig. 3. Generating the hierarchy of the PER-Tree. (a) An example dataset containing two attributes: gender and occupation. (b) Forming the hierarchy of the tree by setting the attribute order as gender first, then followed by occupation. (c) Each node is split into two types of sub-nodes, “Prop-node” and “Attr-node”. “Prop-node” stores the propagated privacy information from its parent node and “Attr-node” stores attribute-specific privacy information. Details on the color encoding of the nodes are provided in Section 5.1.2.

(4) Examine Utility

Our metric introduced in Section 3.2 measures the utility at a data-aggregated level, it sometimes may not reflect to the exact change of information that users are interested in. We provide a detailed utility comparison view for one user-selected attribute at a time. In this view, users can examine the difference between the distributions of an attribute before and after data manipulation at the data level.

(5) Export Data

The last step of the pipeline is to export the generated visualization and/or its underlying anonymized data for future use.

5 Visualization and Interaction Design Details

We provide details on the visualization design and interaction of our system, which consists of two main components: a PER-Tree (Fig. 1(c)) and a UPD-Matrix (Fig. 1(b)).

5.1 Privacy Exposure Risk Tree (PER-Tree)

PER-Tree shows the privacy concerns in the data regarding the privacy criteria set by users and allows interactions to manipulate and anonymize the underlying data.

5.1.1 Tree Construction (Fig. 2(2))

The system initializes the PER-Tree by listing out all user-selected attributes and all possible values for each attribute. Then, operations can be applied to help gradually build up the tree. Users can start by deciding how the attributes should be organized and ordered. For a categorical attribute, users can perform operations, like filtering and aggregation, to organize the data. If needed, users can also aggregate attribute values with customized rules. For example, one can select several cities and aggregate them into one single province. For a numerical attribute, the system by default displays its maximum and minimum values. Users can create new splitting points to form multiple numerical ranges as bins by left clicking on any position of the representing bar/node or by right clicking on the bar/node to enter the exact number in a pop-up text box. Adjusting and deleting a splitting point can be done through mouse dragging and right clicking, respectively. To change the order of the attributes, a drag-and-drop interaction method is supported.

To finalize the construction of PER-Tree, users define the desired threshold values for the three syntactic anonymization models to detect privacy issues in the data. While satisfying *k-anonymity* limits the number of data records contained in an equivalence class, we treat it as a universal indicator. That is, the same k value is assigned to all sensitive attributes. On the other hand, *l-diversity* and *t-closeness* look at the variety of sensitive values in an equivalence class, we give users the flexibility to decide whether to set and how to set these two criteria for each individual sensitive attribute separately. In summary, if there are n_{sen} sensitive attributes, the maximum number of configurable threshold values is $(2n_{sen}+1)$.

5.1.2 Tree Encoding

We encode each node of the PER-Tree with the information of privacy leaks measured by the syntactic privacy models. As demonstrated in Fig. 3, each node represents an attribute value and is split into two types of sub-nodes: "Attr-node" and "Prop-node".

An Attr-node comprises three types of colored bars while the hue of a bar corresponds to one of the syntactic models: blue for *k-anonymity*, green for *l-diversity*, and orange for *t-closeness*. Because *k-anonymity* is a universal privacy indicator, every Attr-node contains exactly one blue bar. The number of green bars and orange bars, however, is dependent on how the privacy criteria are defined in each sensitive attribute. A *l-diversity* or *t-closeness* criterion set for a sensitive attribute creates one green or orange bar. The opacity of a bar indicates the maximum degree of privacy exposure considering all possible equivalence classes that involves the attribute value and the all its parent attributes.

Sensitive attributes, assumed to be unknown to the attackers in syntactic models, are not used to form equivalence classes. Applying *l-diversity* and *t-closeness* based on a sensitive attribute thus defies the assumption. However, we consider that exposing the value in one sensitive attribute may contribute to the exposure to other sensitive attributes. As a result, we derive at most $(2(n_{sen}-m)+1)$ privacy indicators, depending on how l and t are set in other sensitive attributes, for the m-th sensitive attribute.

On the other hand, a Prop-node is colored by gray. Its opacity shows the total amount of privacy risk that is propagated from its immediate parent and is caused by accounting for the current attribute value. The privacy risk that is caused by involving the current attribute value is also encoded on the edge that links to the Prop-node. By looking at the edges helps users more easily identify certain attribute values that contribute to significant privacy increase.

We consistently present the opacity of the nodes and the edges as the amount of privacy risk it represents: the more opaque, the more privacy risk involved. The opacity mappings are done by linearly normalizing the ranges of k, l, and t values between 0.1 and 1. To be more specific, the user-defined threshold values are assigned to 0.1. Values that are larger (for *k-anonymity* and *l-diversity*) or are smaller (for *t-closeness*) than the corresponding threshold values are truncated to 0.1, because they are considered satisfying the privacy criteria. For the values that are within the minimum (or maximum) possible value and the user-defined thresholds, we then linearly scaled them into the range between 0.1 and 1. The design choice made here to set the minimum opacity to 0.1 is to prevent the nodes from becoming not visible.

Fig. 4 shows how privacy are encoded in the PER-Tree using a 4-attribute dataset (with "Expense" being the sensitive attribute). When considering the first two attributes (Gender and Occupation), there is a small amount of privacy issues with respect to *t-closeness* observed on the "S" occupation node, while the privacy concern mostly has to do with the gender "M". If we include one more attribute (Age), the tree tells us that age group "6∼30" bears larger privacy leaks than age group "30∼60" as occupations "O" and "T" are the major contributors.

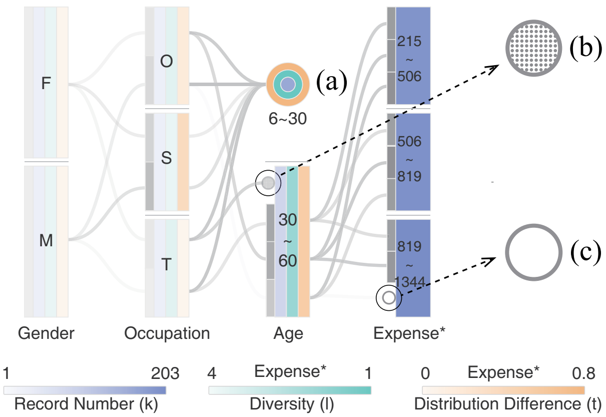


Fig. 4. Different ways to collapse a node. (a) Collapse a node to show only information contained in the Attr-node. (b) Collapse a Prop-node and transform it into a circle filled with dotted texture after differential privacy is applied. (c) Collapse a Prop-node and transform it into a plain circle if it satisfies all privacy criteria set by users. A collapsed node no longer propagates its carried privacy information to its child nodes.

As the number of hierarchies grows, the number of Prop-nodes and edges in each tree level increases in a geometric progression. Our system provides two methods to tackle this potentially information-overload and visual clutter problem:

Collapse nodes Our system automatically collapses a Prop-node under two conditions: 1) if differential privacy is applied to the Prop-node, then the shape of the node is transformed into a circle filled with dotted texture, as shown in Fig. 4(b); 2) if the Prop-node and its associated child nodes satisfy all privacy criteria defined by users, then it is transformed into a plain circle, as shown in Fig. 4(c). Users can also manually collapse a node to present only information contained in the Attr-node (see Fig. 4(a) for example). A collapsed node will not propagate its carried privacy information to its child nodes, thus reducing the number of edges displayed in the following tree levels.

Toggle display options Users can toggle to hide the Prop-nodes and/or the edges temporarily so the focus can be put on the Attr-nodes. Prop-nodes and/or edges can be recalled back whenever users need to see the details or interact with them.

5.1.3 Tree Adjustment

The PER-Tree supports two operations for addressing privacy issues. The first operation is node merging. Users can choose to merge either two Attr-nodes or two Prop-nodes. Attr-nodes can only be merged if they are at the same tree level, while Prop-nodes can only be merged if they have the same parent node. Merging two Attr-nodes reduces the number of attribute values by 1. On other hand, merging two Prop-nodes may create a new attribute value depending on whether the combination of the associated attribute values is pre-existed or not. Fig. 5 illustrates users' interaction of merging two Prop-nodes that leads to the creation of a new node in the PER-Tree.

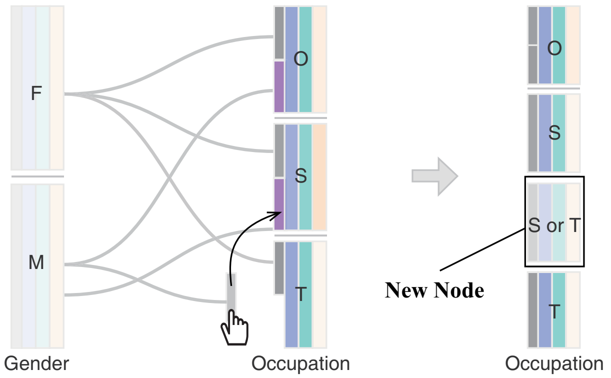


Fig. 5. The node merging operation. When users drag a node, a Prop-node here for example, candidate nodes that can be merged with are highlighted with purple color. After dropping the dragged node on any of the purple nodes, the structure of the tree will be changed and the privacy information will be updated accordingly.

The second operation is noise addition (by applying differential privacy). Users first need to identify which data records represented by a Prop-node or an Attr-node to apply differential privacy on. Then, by right clicking on the node-of-interest, a menu will pop out to let users input the value of ε for controlling the noise level. Next, users can decide to add noise to which attribute(s). This will allow users to achieve a better utility preservation for certain attribute(s).

After a privacy preserving operation is applied on the PER-Tree, the privacy information of the entire tree will be recalculated. The UPD-Matrix will also be updated accordingly to reflect the change on data aggregation or on individual data values.

5.2 Utility Preservation Degree Matrix (UPD-Matrix)

A UPD-Matrix, as an example shown in Fig. 1(b), consists of three parts: the diagonal, the lower triangle, and the upper triangle. The diagonal cells present the distribution of each user-selected attribute. The cells in the upper triangle and the lower triangle display the pairwise joint distribution of the user-selected attributes derived from the original data and the processed data, respectively.

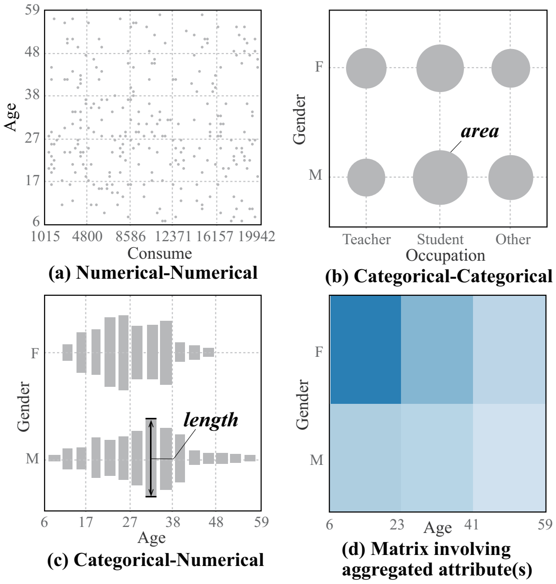
When any of the operation is applied to the data, i.e., filtering, aggregation, node merging, or noise addition, the cells in the lower triangle are also updated to reflect the change. In addition, at the top of each column, we display the utility value of each attribute in the processed data comparing to the original data as well as the amount of utility change caused by latest operation performed to that attribute. In this way, our users are able to keep track of the data utility more easily while trying to manipulate and anonymize data.

5.2.1 Joint Distribution Representation

Kay and Heer [19] proposed a model that considers both predictive accuracy and generalization. A conclusion was drawn that scatterplots yield unparalleled performance in identifying correlation of data. We therefore choose a scatterplot-based representation for presenting the joint distributions of data.

Fig. 6(a) to (c) show how we visualize scatterplots for different combinations of attribute-type pairs: (a) numerical-numerical, (b) categorical-categorical, and (c) categorical-numerical. Radius of a circle and height of a bar represent the data record count in a categorical-categorical scatterplot and a categorical-numerical scatterplot, respectively.

We treat an aggregated attribute as a variant of categorical data because the data values are transformed from exact values into fuzzy ranges/categories. We then employ a matrix representation, as seen in Fig. 6(d), for showing the joint distribution if at least one of the two attributes is aggregated. The opacity of a cell indicates the amount of data records contained in that cell. In addition, if both aggregation and noise addition are applied to an attribute, we overlay a dotted-texture, as in Fig. 4(b), onto the matrix cell for indication purpose.



(a) Numerical-Numerical
(b) Categorical-Categorical
(c) Categorical-Numerical
(d) Matrix involving aggregated attribute(s)

Fig. 6. Examples of scatterplots for different combinations of attribute-type pairs.

5.2.2 Brush and Highlight

Users can hover over or brush through any dot, circle, bar, or matrix cell in the UPD-Matrix to highlight data records with certain attribute values. It allows users to quickly identify correlation among multiple attributes. For example, assume there is a dataset containing 100 teachers, in which 20 are males, 80 are females. Among those female teachers, 20 of them are 35 years old and 60 of them are 55 years old. Brushing the "female-teacher" circle not only allows users to see the age distribution within female teachers, but also to reveal the proportion of female teachers in different ages comparing to male teachers.

5.2.3 Comparisons

To compare the joint distributions of the original data and the processed data, users can double-click on a cell to open a comparison pane. An example can be seen in Fig. 2(4). With the comparison pane, users can view the pair of related charts in more detail. In addition, we provide a "delta chart" which essentially shows the value differences between the original data and the processed data in each matrix cell. It offers a direct and intuitive visual comparison by encoding the value of "delta" using a red-to-green gradient color map, where green maps to positive values and red means negative values.

In order to be able to compare and to unify the visual expression, we transform all the data representation to be compared in a matrix form. In addition, all the matrices should be of the same granularity. Users can scroll the "Step" slider bar, shown in Fig. 2(4), to control the desired granularity. The smaller the "Step" value is, the finer the granularity to compare.

6 Evaluation

We present three possible use case scenarios of our system and summarize feedback collected from three potential expert users.

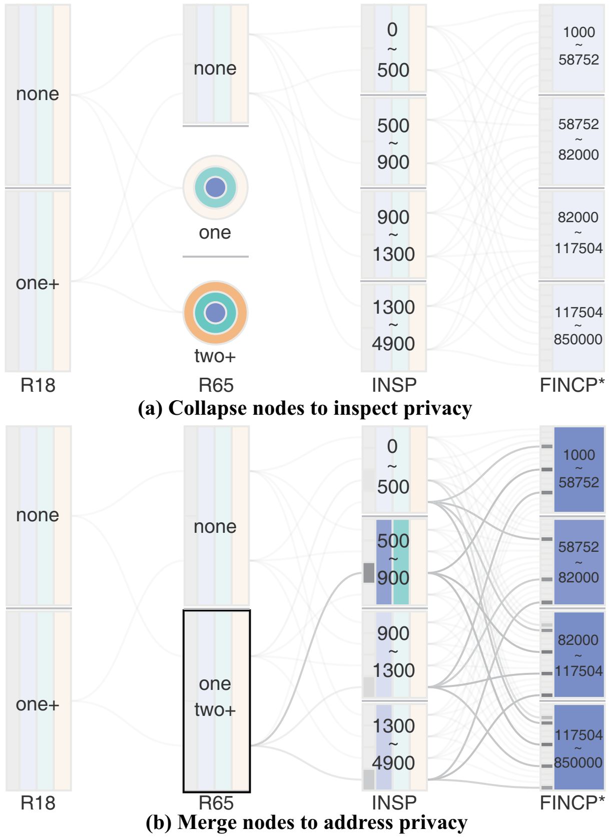
6.1 Household Income and Insurance Census Data

We sampled a subset of the Public Use Microdata Samples (PUMS) survey data [1] from Wyoming, USA in 2015. 1,233 records remain for our use after removing records with missing values in any of our interested attributes. With this dataset, we assume that an insurance company located in Wyoming, USA wants probe the potential of developing a local business in this area.

To conduct the analysis, we look at four attributes: *INSP*–fire/hazard/flood insurance (yearly amount), *FINCP*–family income (past 12 months), *R*18–presence of persons under 18 years old in household, and *R*65–presence of persons over 65 years old in household. *FINCP* is the sensitive attribute that requires protection.

With a very rough attribute value aggregation on each dimension, as shown in Fig. 1(b), we find that the household income has a positive correlation with the amount of money spent on insurance. We further aggregate *FINCP* based on distribution of the class division provided by [16], and display the joint distribution between *FINCP* and *R*18 in Fig. 8(a). One interesting pattern is that families with more children (one+) tend to spend more on their insurance (majority of them spend $900 - $1300 per year) as compared to those families without any child (majority of them spend $500 - $900 per year).

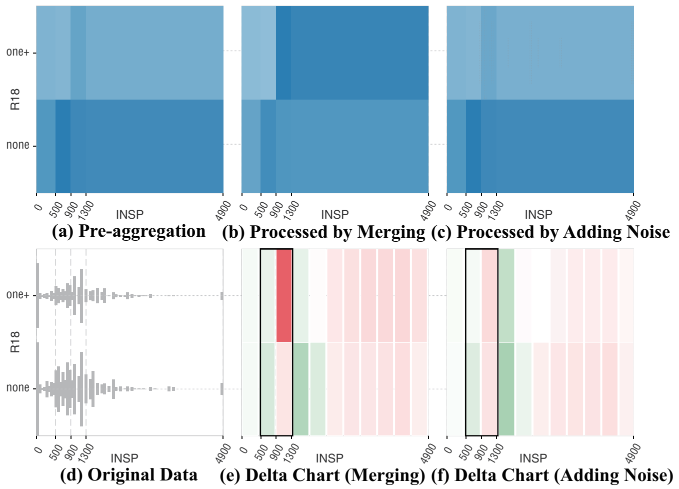
By observing the PER-Tree displayed in Fig. 1(c), we quickly identify that families with more than one elder (65+) adults are more vulnerable to privacy exposure. By collapsing the two *R*65 nodes: "*one*" and "*two*+", shown in Fig. 7(a), we find that families with two or more elder adults are actually having a relatively higher privacy risk than the families with only one elder adults. To address this, we simply merge the two nodes. Fig. 7(b) presents the resulting PER-Tree which suggests that subsequent operations should be made to protect the privacy of those elder adults' families that spend more $500 on insurance (especially for those who spend 500 - 900 per year).



(a) Collapse nodes to inspect privacy

(b) Merge nodes to address privacy

Fig. 7. (a) Collapsing the two 65 nodes ("one" and "two+") which cause the most privacy issues in the data. (b) Merging the two collapsed nodes resolves most privacy issues originally shown in Fig. 1.



(a) Pre-aggregation (b) Processed by Merging (c) Processed by Adding Noise

(d) Original Data (e) Delta Chart (Merging) (f) Delta Chart (Adding Noise)

Fig. 8. (a) - (c): joint distribution matrices between *R*18 and *INSP* for (a) before applying privacy preserving operations, (b) privacy preservation by merging, and (c) privacy preservation by noise addition. (d): the distribution chart of the original, non-processed data. (d) and (f): the delta charts for comparing the utility change of applying aggregation and noise addition.

One strategy is to keep merging nodes for privacy preservation. However, as suggested by the resultant joint distribution view, Fig. 8(b), and the delta chart, Fig. 8(e), the utility of data seems reduced quite a lot. Therefore, we decide to rollback to the previous step, and instead apply differential privacy techniques. Fig. 8(c) presents the corresponding joint distribution while Fig. 8(f) demonstrates its detailed distribution comparison with Fig. 8(a). We can see that this time the utility loss has been controlled at a more acceptable level.

6.2 Graduate Transition Data

This dataset is about 712 teenage students' career transition within a six-year span after graduation [29]. Analysts may wish to learn about how living environments can affect a student's development. We select attributes including *gen* (gender), *cat* (whether a Catholic believer or not), *res* (place of residence), *fue* (whether the student's father is unemployed), and *jol* (how many months have the student been jobless). The last two attributes are defined as sensitive. As longer period of unemployment may be referred as unsuccessful career development, using the UPD-Matirx we find that females, Catholic believers, residents of Belfast, and those whose father is unemployment are less likely to undergo a smooth transition, as shown in Fig. 9.

In accordance with the distribution of *jol*, we find that most of the students are employed within two months after graduation. The proportion of unemployment even dropped more significantly in the following 10 months. Based on this finding, we set the splitting points of the *jol* attribute as two months, one year, and two years. We set "*l*" and "*t*" for *jol* and only "*l*" for *fue* as their privacy indicators. The PER-Tree in Fig. 10 shows that the students who live in S.Eastern and the students whose father is unemployed are particularly of high privacy risks. Meanwhile, for students whose fathers are employed, we find that most of their Prop-nodes are not that privacy sensitive, therefore we consider noise addition as a better option in this case. After addressing privacy with noise addition, the total utility score drops from 94% to 71%. Nevertheless, as highlighted by the black box in Fig. 11, the pattern of students who live in Belfast are more likely to experience a negative career development is still well-preserved.

Adding noise to the "S.Eastern" (of the *res* attribute) and the "yes" (of the *fue* attribute) nodes resolves most of the privacy risks of the dataset, as shown in Fig. 12. There are still a few privacy risks remain

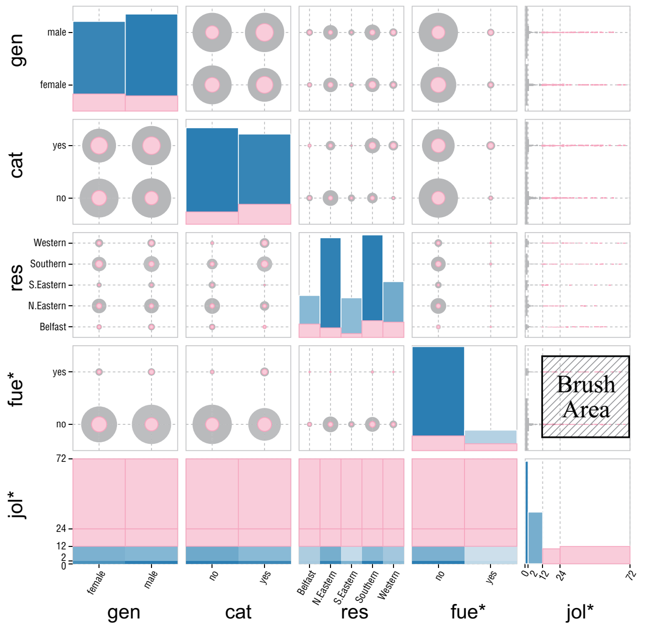


Fig. 9. The UPD-Matrix of the Graduate Transition Data after pre-aggregation. We highlight the graduates who had been unemployed for longer than a year by brushing.

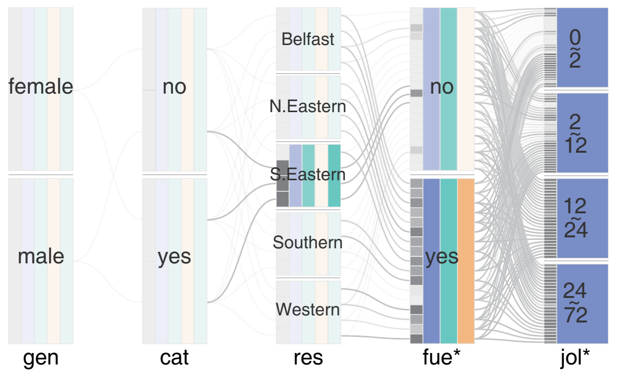


Fig. 10. The PER-Tree of the the Graduate Transition Data after pre-aggregation.

in the "jol" attribute even we further merge all Attr-nodes that represent student who had been unemployed for more than two month (see the rightmost, lowest level of the PER-Tree in Fig. 12). If users wish to remove all the privacy concerns and still keep the "jol" attribute meaningful. More differential privacy operations are needed.

6.3 Garment Industry Data

We extract 740 data records on garment processing enterprises from the enterprise survey data of 2004 provided by National Bureau of Statistics of China. We are particularly interested in understanding the typical wage and profit level of small-scale enterprises in East China. We thus select the following attributes: *reg* (region of China), *TA* (total asset of company), *TPR* (total profit, and *AWP* (average wage paid). *AWP* is set as the sensitive attribute and is pre-aggregated by adding split points at 4 and 6 so that the records can be equally divided into three groups. To focus only on the East China (EC) region, we collapse all other *reg* nodes and resolve all privacy issues in the intermediate levels, so that the PER-Tree can only shows the *k-anonymity* criteria involving the *AWP* attribute. As shown in Fig. 13, most privacy issues remain in the "0 ˜ 4" *AWP* node, while some others in "4 ˜ 6".

We opt to merge the two Attr-nodes, which then does not affect the utility much (approximately 1%). By viewing the delta chart, displayed in Fig. 14, however, we do see a change of distribution. We look at the data values closely and find that most East China companies offer *AWP* in the range of 3 ˜ 5, while their total assets (*TA*) and profit level (*TPR*) are also similar to each other. As utility values are calculated based on the aggregated data instead of reflecting the change of individual data values. In situations like this case, one single utility value itself may not a precise indicator. Other information, such as the delta chart, is needed for further confirmation.

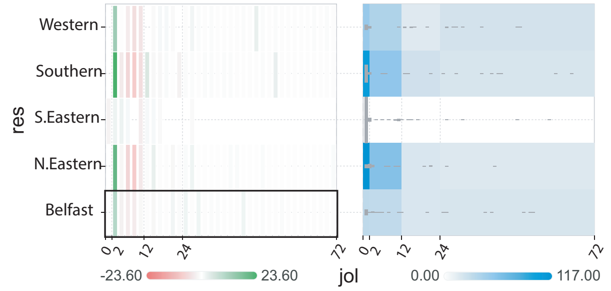


Fig. 11. Examining the pattern and utility after adding noise. Although the utility value of *jol* decreases from 94% to 71%, the pattern we concerned about Belfast is still well-maintained.

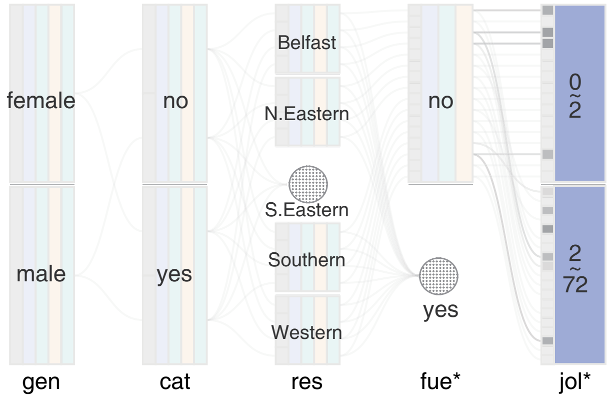


Fig. 12. The PER-Tree of the Graduate Transition Data after differential privacy is applied to the "S.Eastern" and "yes" nodes.

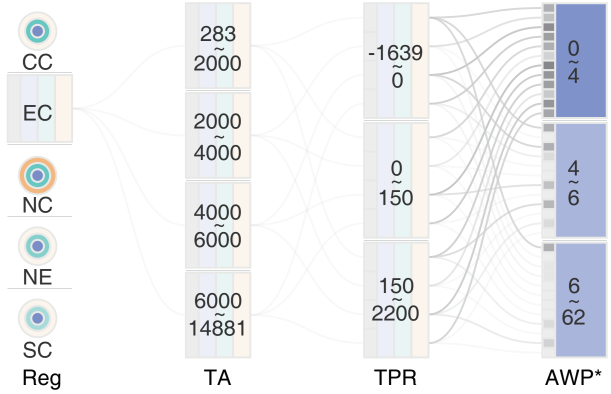
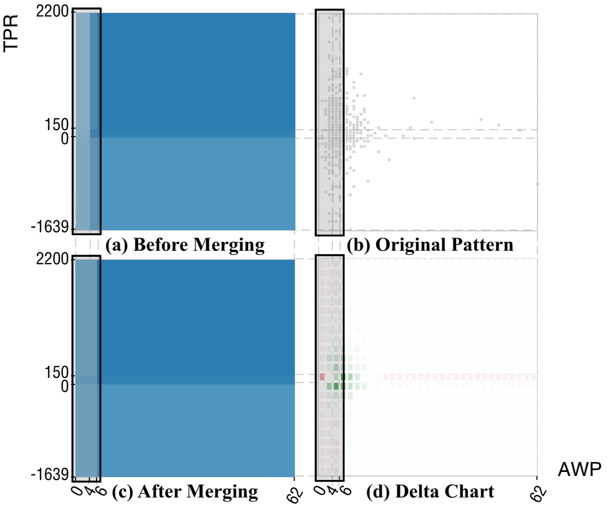


Fig. 13. The PER-Tree of the Garment Industry Data after collapsing all region (*reg*) nodes except East China (EC) and removing all privacy leaks involving *TA* (total asset) and *TPR* (total profit).

6.4 Domain User Reviews

Our system was reviewed by three domain experts who constitute potential users. Each works in a field of study where, after sensitive data is analyzed, the results (and/or datasets) must be sanitized before release or publication. Each user also has a basic understanding of the syntactic



(a) Before Merging

(b) Original Pattern

(c) After Merging

(d) Delta Chart

Fig. 14. After merging based on AWP, the effect on change of distribution can be observed in the delta chart.

models introduced in this paper, for example, two users regularly perform data anonymization via aggregation and generalization.

For each user, we introduced our system and explained the available functionalities as well as demonstrating some exemplar use cases. We then asked each to fill out a questionnaire about our system and how it would help their workflow. Each session took between 30-60 minutes.

Users had positive feedback for several system aspects. First, they liked the fact that we consider multiple privacy models and preservation actions. This provides flexibility in dealing with different dataset types and with different privacy scenarios. Providing real-time feedback on how data utility changes with respect to anonymization actions was an important feature. In comparison, their current practices typically consider data utility only after the entire anonymization process is complete. This makes it difficult (and inefficient) to identify which particular actions cause significant changes to utility.

Users also gave several comments about system design considerations, namely, dataset scalability. For example, as companies work with larger datasets, both in terms of number of records and data dimensions, efficiently detecting privacy leaks, fixing privacy leaks, and assessing data utility are high priority, non-trivial tasks. For our system to handle larger dataset, specifically time-complexity optimized privacy models and utility metrics may need to be considered to further improve the system performance. In addition, all three users suggested that a recommendation scheme can be designed to help them examine and then decide the appropriate actions to take in different scenarios.

7 Discussions

We discuss aspects worth considering when designing privacy preserving solutions.

7.1 Aggregation or Noise Addition?

While noise addition and data aggregation are both effective for privacy preservation, each comes with its own shortcomings. Aggregating numerical types of attributes unavoidably causes varying degrees of loss on granularity, which is decided by the split points. So the question is what the best split point is–there is no one-size-fits-all standard. It largely depends on user requirements as the underlying topology (i.e., sensitivity) of the data. Another problem is the curse of dimensionality. As attribute numbers scale up, the potential for aggregation does likewise at an exponential rate.

On the other hand, the addition of noise causes the uncertainty bounds in the data to become less exact and more probabilistic. As a result, correlations between attributes may be distorted or may produce patterns that do not actually exist. Boundary conditions are another concern–these have to take into account data type restrictions. For example, naively adding noise to an "age" attribute might change the value to a negative number.

7.2 Flexibility of our Visual Approach

Each of the privacy preserving techniques affects the users' understanding of the data (and the visualization) in a different way. We thus let users make their own decisions about which operations to apply on specifically assigned and/or chosen subsets of the data. The advantage of a flexible user-centered design like ours is that it allows for a more fine-grained and precise sanitization process as opposed to being a purely automated approach. One can easily introduce different amounts of noise or aggregate attributes at different granularities and then quickly observe and compare the pros and cons for each option before making a final decision. The main disadvantage of a visualization-assisted approach against a data-centric method is that it requires more time and engagement from users. It is thus not particularly suitable for batch processing. One potential and worth exploring direction, as also mentioned in our expert review, is to design a recommendation mechanism to help users quickly perform necessary privacy preserving operations without losing too much flexibility.

7.3 Potential limitations

While we consider our design effective for its purposes, there are still potential limitations that should be addressed. One limitation has to do with how utility is derived for a dataset. Normally, utility for individual points is calculated, however we need to present these at an aggregated level (either visually or literally), since displaying or releasing the utility at an individual level may greatly increase the possibility of revealing privacy. Given this constraint, utility metrics cannot always guarantee they will yield results that reflect the real situation comprehensively. Optimum ways to visually indicate and encode this uncertainty (for raising user awareness) is a topic suited for future research.

Another limitation is that our PER-Tree design may be restrained by the curse of dimensionality. As the number of attributes increases, the number of possible combinations in the lower level increases significantly. Although our tree pruning design can to some extent alleviate this issue, the loss of detailed information may cause certain important privacy leaks being neglected. One possible solution would be to allow further interactions to manipulate the display of the edges, such as bundle or filter edges based on their opacity values. Another alternative could be to provide some auxiliary view that organizes and presents the information in a different way. For example, a list-based table presentation with sorting and searching functionality can allow quicker identification of the most interested edges.

8 Conclusion

We have presented a visual system for interactively detecting and addressing privacy issues and subsequently inspecting the operational change in data utility. The underlying data manipulation pipeline guides users in iteratively finding the best balance between privacy and utility. Our system integrates several commonly used syntactic anonymization and differential privacy models, allowing greater flexibility in fulfilling the various privacy needs that different users may have.

As part of the interface, we introduce the PER-Tree representation. Its space-compact design and effective pruning scheme help users navigate through the high dimensional data space to quickly locate privacy issues. The UPD-Matrix provides additional necessary visual feedback, giving reference to how much utility is affected by the applied privacy preserving operations. We demonstrate how to emphasize privacy considerations in a multi-attribute tabular dataset in a utility-aware manner.

Acknowledgements

This research has been sponsored in part by the National 973 Program of China (2015CB352503), Major Program of National Natural Science Foundation of China (61232012), National Natural Science Foundation of China (61422211), and the U.S. National Science Foundation through grants IIS-1320229 and IIS-1528203.

References

[1] Acs pums data. https://www2.census.gov/programs-surveys/acs/.

[2] M. S. Alvim, M. E. Andrés, K. Chatzikokolakis, P. Degano, and C. Palamidessi. Differential privacy: on the trade-off between utility and information leakage. In *Proceedings of the International Workshop on Formal Aspects in Security and Trust*, pp. 39–54. Springer, 2011.

[3] N. Andrienko, G. Andrienko, G. Fuchs, and P. Jankowski. Scalable and privacy-respectful interactive discovery of place semantics from human mobility traces. *Information Visualization*, 15(2):117–153, 2016.

[4] J. Brickell and V. Shmatikov. The cost of privacy: Destruction of data-mining utility in anonymized data publishing. In *Proceedings of ACM SIGKDD 08'*, pp. 70–78, 2008. doi: 10.1145/1401890.1401904

[5] J. Cao and P. Karras. Publishing microdata with a robust privacy guarantee. *Proceedings of the VLDB Endowment*, 5(11):1388–1399, 2012.

[6] R. Chen, N. Mohammed, B. C. Fung, B. C. Desai, and L. Xiong. Publishing set-valued data via differential privacy. *Proceedings of the VLDB Endowment*, 4(11):1087–1098, 2011.

[7] J.-K. Chou, C. Bryan, and K.-L. Ma. Privacy preserving visualization for social network data with ontology information. In *IEEE Pacific Visualization Symposium*, 2017.

[8] J.-K. Chou, Y. Wang, and K.-L. Ma. Privacy preserving event sequence data visualization using a sankey diagram-like representation. In *Proceedings of the SIGGRAPH ASIA Symposium on Visualization*, 2016.

[9] T. Dalenius. Finding a needle in a haystack or identifying anonymous census records. *Journal of official statistics*, 2(3):329, 1986.

[10] A. Dasgupta, M. Chen, and R. Kosara. Measuring privacy and utility in privacy-preserving visualization. In *Proceedings of the Computer Graphics Forum*, vol. 32, pp. 35–47. Wiley Online Library, 2013.

[11] A. Dasgupta and R. Kosara. Adaptive privacy-preserving visualization using parallel coordinates. *Proceedings of the IEEE transactions on visualization and computer graphics*, 17(12):2241–2248, 2011.

[12] C. Dwork. Differential privacy. In *Proceedings of the 33rd International Colloquium on Automata, Languages and Programming, part II*, vol. 4052, pp. 1–12. Springer Verlag, 2006.

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

[14] C. Dwork, A. Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

[15] C. Dwork and A. Smith. Differential privacy for statistics: What we know and what we want to learn. *Journal of Privacy and Confidentiality*, 1(2):2, 2010.

[16] J. Feinauer. What it takes to be middle class in each state. http://www.deseretnews.com/top/3184/14/Wyoming-What-it-takes-to-be-middle-class-in-each-state.html.

[17] A. Ghosh, T. Roughgarden, and M. Sundararajan. Universally utility-maximizing privacy mechanisms. *SIAM Journal on Computing*, 41(6):1673–1693, 2012.

[18] Y. Hong, J. Vaidya, H. Lu, P. Karras, and S. Goel. Collaborative search log sanitization: Toward differential privacy and boosted utility. *IEEE Transactions on Dependable and Secure Computing*, 12(5):504–518, 2015.

[19] M. Kay and J. Heer. Beyond weber's law: A second look at ranking visualizations of correlation. *IEEE transactions on visualization and computer graphics*, 22(1):469–478, 2016.

[20] D. Kifer and J. Gehrke. Injecting utility into anonymized datasets. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, pp. 217–228, 2006.

[21] D. Kifer and B.-R. Lin. Towards an axiomatization of statistical privacy and utility. In *Proceedings of the twenty-ninth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pp. 147–158, 2010.

[22] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Incognito: Efficient full-domain k-anonymity. In *Proceedings of the ACM SIGMOD international conference on Management of data*, pp. 49–60, 2005.

[23] N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *Proceedings of the IEEE 23rd International Conference on Data Engineering*, pp. 106–115. IEEE, 2007.

[24] T. Li and N. Li. On the tradeoff between privacy and utility in data publishing. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 517–526, 2009.

[25] G. Loukides and J. Shao. Data utility and privacy protection trade-off in k-anonymisation. In *Proceedings of the international workshop on Privacy and anonymity in information society*, pp. 36–45. ACM, 2008.

[26] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data*, 1(1):3, 2007.

[27] A. Makhdoumi and N. Fawaz. Privacy-utility tradeoff under statistical uncertainty. In *Proceedings of the 51st Annual Allerton Conference on Communication, Control, and Computing*, pp. 1627–1634. IEEE, 2013.

[28] F. McSherry and K. Talwar. Mechanism design via differential privacy. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science.*, pp. 94–103, 2007.

[29] D. McVicar and M. Anyadike-Danes. Predicting successful and unsuccessful transitions from school to work by using sequence methods. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 165(2):317–334, 2002.

[30] J. Oksanen, C. Bergman, J. Sainio, and J. Westerholm. Methods for deriving and calibrating privacy-preserving heat maps from mobile sports tracking application data. *Journal of Transport Geography*, 48:135–144, 2015.

[31] S. R. Rajagopalan, L. Sankar, S. Mohajer, and H. V. Poor. Smart meter privacy: A utility-privacy framework. In *Proceedings of the IEEE International Conference on Smart Grid Communications*, pp. 190–195, 2011.

[32] V. Rastogi, D. Suciu, and S. Hong. The boundary between privacy and utility in data publishing. In *Proceedings of the 33rd international conference on Very large data bases*, pp. 531–542. VLDB Endowment, 2007.

[33] L. Sankar, S. R. Rajagopalan, and H. V. Poor. Utility-privacy tradeoffs in databases: An information-theoretic approach. *IEEE Transactions on Information Forensics and Security*, 8(6):838–852, 2013.

[34] J. Soria-Comas and J. Domingo-Ferrert. Differential privacy via t-closeness in data publishing. In *Proceedings of the 11th Annual International Conference on Privacy, Security and Trust*, pp. 27–35, 2013.

[35] L. Sweeney. Simple demographics often identify people uniquely. *Health (San Francisco)*, 671:1–34, 2000.

[36] L. Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.

[37] T. M. Truta, A. Campan, and P. Meyer. Generating microdata with p-sensitive k-anonymity property. In *Proceedings of the Workshop on Secure Data Management*, pp. 124–141. Springer, 2007.

[38] J. Xu, W. Wang, J. Pei, X. Wang, B. Shi, and A. W.-C. Fu. Utility-based anonymization using local recoding. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 785–790, 2006.