Anonymizing and Trading Person-specific Data with Trust

Rashid Hussain Khokhar

A Thesis

In

The Concordia Institute

For

Information Systems Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of

Doctor of Philosophy (Information and Systems Engineering) at

Concordia University

Montréal, Québec, Canada

November 2020

CONCORDIA UNIVERSITY

SCHOOL Of GRADUATE STUDIES

This is to certify that the thesis prepared

By: Rashid Hussain Khokhar

Entitled: Anonymizing and Trading Person-specific Data with Trust

and submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy (Information and Systems Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

Dr. Constantinos Constantinides	Chair
Dr. Rozita Dara	External Examiner
Dr. Yan Liu	External to Program
Dr. Amr Youssef	Examiner
Dr. Jia Yuan Yu	Examiner
Dr. Benjamin C. M. Fung	Thesis Supervisor
Dr. Jamal Bentahar	Thesis Co-supervisor

Approved by

Dr. Mohammad Mannan, Graduate Program Director

November 24, 2020

Dr. Mourad Debbabi, Interim Dean
Gina Cody School of Engineering & Computer Science

Abstract

Anonymizing and Trading Person-specific Data with Trust

Rashid Hussain Khokhar, Ph.D.

Concordia University, 2020

In the past decade, data privacy, security, and trustworthiness have gained tremendous attention from research communities, and these are still active areas of research with the proliferation of cloud services and social media applications. The data is growing at a rapid pace. It has become an integral part of almost every industry and business, including commercial and non-profit organizations. It often contains person-specific information and a data custodian who holds it must be responsible for managing its use, disclosure, accuracy and privacy protection. In this thesis, we present three research problems. The first two problems address the concerns of stakeholders on privacy protection, data trustworthiness, and profit distribution in the online market for trading person-specific data. The third problem addresses the health information custodians (HICs) concern on privacy-preserving healthcare network data publishing.

Our first research problem is identified in cloud-based data integration service where data providers collaborate with their trading partners in order to deliver quality data mining services. *Data-as-a-Service* (*DaaS*) enables data integration to serve the demands of data consumers. Data providers face challenges not only to protect private data over the cloud but also to legally adhere to privacy compliance rules when trading person-specific data. We propose a model that allows the collaboration of multiple data providers for integrating their data and derives the contribution of each data provider by valuating the incorporated cost factors. This model serves as a guide for business decision-making, such as estimating the potential privacy risk and finding the sub-optimal value for publishing mashup data. Experiments on real-life data demonstrate that our approach can identify the sub-optimal value in data mashup for different privacy models, including K*-anonymity*, LKC*-privacy*, and ϵ*-differential privacy*, with various anonymization algorithms and privacy parameters.

Second, consumers demand a good quality of data for accurate analysis and effective decision-making while the data providers intend to maximize their profits by competing with peer providers. In addition, the data providers or custodians must conform to privacy policies to avoid potential penalties for privacy breaches. To address these challenges, we propose a two-fold solution: (1) we present the first information entropy-based trust computation algorithm, *IEB_Trust*, that allows a semi-trusted arbitrator to detect the covert behavior of a dishonest data provider and chooses the qualified providers for a data mashup, and (2) we incorporate the *Vickrey-Clarke-Groves (VCG) auction mechanism* for the valuation of data providers' attributes into the data mashup process. Experiments on real-life data demonstrate the robustness of our approach in restricting dishonest providers from participation in the data mashup and improving the efficiency in comparison to provenance-based approaches. Furthermore, we derive the monetary shares for the chosen providers from their information utility and trust scores over the differentially private release of the integrated dataset under their joint privacy requirements.

Finally, we address the concerns of HICs of exchanging healthcare data to provide better and more timely services while mitigating the risk of exposing patients' sensitive information to privacy threats. We first model a complex healthcare dataset using a heterogeneous information network that consists of multi-type entities and their relationships. We then propose *DiffHetNet*, an edge-based differentially private algorithm, to protect the sensitive links of patients from inbound and outbound attacks in the heterogeneous health network. We evaluate the performance of our proposed method in terms of information utility and efficiency on different types of real-life datasets that can be modeled as networks. Experimental results suggest that *DiffHetNet* generally yields less information loss and is significantly more efficient in terms of runtime in comparison with existing network anonymization methods. Furthermore, *DiffHetNet* is scalable to large network datasets.

Acknowledgments

First and foremost, I am deeply thankful to *Allah* (*swt*) for granting me the health, intellectual abilities, and patience to complete this thesis.

I would like to extend my deepest gratitude and appreciation to my supervisor, Dr. Benjamin C. M. Fung. I am deeply indebted to him for all the guidance, constructive criticism, and persistent support through out the period of my doctoral studies and completing this thesis. His kind attitude, patience, and motivation is praiseworthy. I would like to thank my co-supervisor, Dr. Jamal Bentahar, for his valuable time and suggestions in my research. Moreover, I would like to extend my gratitude to my Ph.D. committee members Dr. Amr Youssef, Dr. Jia Yuan Yu, Dr. Yan Liu, Dr. Rozita Dara, and Dr. Constantinos Constantinides for their valuable time, reviewing my thesis, and providing me with insightful comments and suggestions.

This research would not have been possible without the Concordia University International Tuition Fee Remission Award, Concordia Graduate Student Support Program (GSSP) funding, and Zayed University Research Cluster Award Fund and Research Incentive Funds. This support was very important for me to alleviate my financial burdens and to keep my focus on research. I would like to extend my gratitude to Dr. Farkhund Iqbal for his partial financial assistance to support this research.

Finally, this thesis is dedicated to my parents and family. I thank you all for your endless love, continuous support, and prayers for my success. I would also like to extend this dedication to my father-in-law and mother-in-law for their encouragement and support during stressful situations. It is all your limitless love and prayers that I successfully achieved this milestone.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Data is an integral part of almost every industry, such as social media, healthcare, e-commerce, and government. With the advancements of digital technologies and the proliferation of online services, data is growing at a tremendous pace. Data often contains explicit identifying information associated with personal data such as name, social insurance number, birth date, address, phone number, marital status, salary, health record, and so on. A data custodian who holds person-specific information must be responsible for managing the use, disclosure, accuracy and privacy protection of collected data. Privacy is a fundamental human right [97], and for this several privacy legislation and regulations such as *Personal Information Protection and Electronic Documents Act* (*PIPEDA*) by Canada, *Health Insurance Portability and Accountability Act* (*HIPAA*) by the United States, and *General Data Protection Regulation* (*GDPR*) by the European Union, across the globe have been imposed for protecting personal data. These legal and regulatory frameworks enforce companies or businesses who deal with personal data must ensure the protection of individuals by removing identifiable information from the data they own. In this thesis, we present three research problems in perspective to preserving the privacy of individuals in publishing data. The first two problems address the concerns of stakeholders on privacy protection, data trustworthiness, and profit distribution in the online market for trading person-specific data. The third problem addresses the health information custodians (HICs) concern on preserving the privacy of individuals in publishing health-network data. In this chapter, we present some motivations linked to each of the problems studied, followed by our main contributions.

The motivation of first research work is that data consumers demand customers' data over the cloud to identify demographic characteristics of customers from persons-specific data. *Data-as-a-Service* (*DaaS*) is a cloud computing paradigm that provides data on demand to consumers over the Internet [19]. It is becoming popular in commercial setups because it provides flexible and cost-effective collaboration among business enterprises. In the e-market industry, enterprises conduct online market research to collect feedback about their products and services and to identify the demographic characteristics of customers by various means such as surveys, social networks, online purchases, posts, blogs, internet browsing preferences, phone calls, and apps. The primary purpose in collecting personal information is to provide better services, which in turn generate higher revenue.

Business enterprises (or data providers) face four major challenges for trading person-specific information. First, extensive research has shown that simply removing explicit identifying information such as name, social security number, birth date, and telephone number is insufficient for privacy protection. Many organizations believe that enforcing regulatory compliance, such as the Gramm-Leach-Bliley Act (GLBA), which protects the privacy and security of individually identifiable financial information, or simply employing common de-identification methods, such as Health Insurance Portability and Accountability Act (HIPAA) Safe Harbor method, which involves removing 18 types of identifiers from health data, is sufficient for privacy protection. Yet an individual can be re-identified by matching the *quasi-identifiers* QID with an external data source [147]. Second, the data providers collaborate in order to fulfill the demands of a data consumer and to generate more profit by offering better data utility. In addition, they would avoid sharing information other than the final integrated data because the collaborating data providers could be competitors. Third, a cloud service provider may not be a trusted party. The cloud service provider can be a third-party who offers data integration services over the cloud or one of the data providers. Fourth, the data providers want to ensure that the mashup data can facilitate the queries of data consumers.

Second, data trustworthiness is also an important concern to consumer stakeholders in the e-market. According to a recent survey [52], organizations in the U.S. estimate that 33% of their customer data is inaccurate. This skepticism about data elicits the increased risk of non-compliance and regulatory penalties. The study by IBM estimated that $3.1 trillion of the U.S.'s GDP is lost due to poor quality data [138]. Organizations may mitigate these potential risks by taking appropriate

measures regarding the quality of their data, leading to more reliable analysis and decision-making. There is a line of research [28, 93] that focuses on exchanging data between multiple parties from the perspective of ensuring confidentiality and integrity. These works aim to provide prevention from unauthorized use and modification when data are in transit but do not verify data if any party provides false data. Many trust models [18, 148] and frameworks [35, 131] have been proposed to evaluate the security strength of cloud environments, but limited research considers the aspect of data reliability.

Third, with the increasing adoption of digital health platforms through mobile apps and online services, people have greater flexibility connecting with medical practitioners, pharmacists, and laboratories and accessing resources to manage their health-related concerns. Many healthcare institutions are being connected for exchanging healthcare data with the goal of providing better and timely services. Health data contains patients' sensitive information, and it is the obligation of health information custodians (HICs) to ensure the protection of patients' private information in the collection, use, and release of health data as mandated by law [73]. It has been a common practice by many health service providers to obtain the patients' consent in sharing health data. However, HICs have faced increasing privacy breaches of different natures [7, 8, 94] due to negligence of administrative employees, compliance failures, and deployment of weak de-identification methods [23]. Health social networking sites such as MedXCentral, Sermo and PatientsLikeMe have been increasingly adopted by the healthcare professionals and patients for sharing health-related information. This poses risks of privacy breaches on sharing personal health data over these platforms [111, 168].

1.1 Contributions

The main contributions of this thesis are summarized below.

1.1.1 Privacy-preserving data mashup model for trading person-specific data

Data providers adopt cloud-based data integration services to improve collaboration with their trading partners and to deliver quality data mining services. They face challenges not only to protect private data over the cloud but also to legally adhere to privacy compliance rules when

trading person-specific data. They need an effective privacy-preserving data mashup model to deal with the challenges in emerging markets. To address this problem, we contribute with a privacy-preserving data mashup model to quantify and compare the costs and benefits for releasing integrated anonymized data of multiple providers over an individual data provider when trading person-specific information in the e-market. We incorporate relevant factors that are associated with the revenue and costs to determine the net value. We organize these factors into three phases: before data mashup, during data mashup, and after data mashup. Our model helps data providers in making effective-decision by evaluating the benefits of data mashup and impacts of data anonymization based on the choices of privacy models and data mashup anonymization algorithms. The proposed model captures only the relevant factors that are crucial for cost-benefit analysis in our research problem. However, the model provides flexibility for users to include additional factors based on the specific requirements of other scenarios. Experiments on real-life data demonstrate that our approach can identify the sub-optimal value in data mashup for different privacy models, including $\mathtt{k}$-*anonymity*, LKC-*privacy*, and ϵ-*differential privacy*, with various anonymization algorithms and privacy parameters.

1.1.2 Secure trustworthiness assessment and privacy protection in integrating data

Services computing promises on the secure and reliable transport of big data over a cloud to improve business efficiencies. Research communities have investigated the problem of exchanging data between multiple parties from the perspective of ensuring confidentiality and integrity [28, 93]. They have proposed solutions to prevent the unauthorized use and modification of data in transit. However, the existing works do not address the problem of verifying the correctness of private data if any party has provided false data. This is the first work providing a solution to stakeholders to address their concerns on data trustworthiness, privacy protection and profit distribution in integrating data over the cloud. We propose *IEB_Trust*, an information entropy-based trust computation algorithm that allows a semi-trusted arbitrator to detect the covert behavior of a dishonest data provider, evaluates the trustworthiness of the participating data providers by a trust metric, and chooses the qualified providers for data mashup. Compared to the existing work on data trustworthiness [114, 115, 165], our proposed algorithm not only detects fabricated or incorrect data from a dishonest data

provider during the verification process but also preserves the privacy of customers' data owned by a data provider. Furthermore, our method provides better runtime efficiency over provenance-based approaches [40, 114]. We incorporate the Vickrey-Clarke-Groves (VCG) auction mechanism for the valuation of data providers' attributes into the data mashup process. Finally, we derive the monetary shares for the chosen data providers from their contribution in information utility and their attained trust scores over the differentially private release of the integrated dataset under the mutually agreed privacy requirements.

1.1.3 Differentially private release of heterogeneous network for healthcare data

In recent years, heterogeneous information networks (HINs) have gained increasing attention in various application domains such as social media, communications, energy, and health informatics, mainly due to its ubiquitousness and capability of representing rich semantics [151]. Various mining methods have been proposed to tackle the problem of heterogeneity for network analysis, such as ranking-based classification and clustering [81, 157], meta-path-based similarity search [160], relationship prediction and relation strength learning [159, 161], and advanced embedding methods [58, 70, 153]. On the one hand, these mining and embedding methods for heterogeneous networks serve different requirements of data analysis, but on the other hand, the privacy of an individual is at stake unless proper protection measures are deployed. In this thesis, we first model a complex de-identified healthcare dataset using a heterogeneous information network that consists of multi-type entities and their multi-type relationships. Existing solutions [37, 76, 173] consider nodes and edges to each be of a single type and edges to be bidirectional (or undirected). Thus, these solutions cannot maintain important semantics and structural information of the heterogeneous network. This is the first work providing a practical solution to health information custodians (HICs) who wish to release real-life heterogeneous health-network data. We propose *DiffHetNet*, an edge-based differentially private algorithm, to protect the sensitive links of patients from inbound and outbound attacks in a heterogeneous health network. We evaluate the performance of our proposed method in terms of information utility and efficiency on different types of real-life datasets that can be modeled as networks. Experimental results suggest that *DiffHetNet* generally yields less information loss, significantly improved runtime efficiency in comparison with existing network anonymization

methods, and scalability to large network datasets.

1.2 Thesis organization

The rest of the thesis is organized as follows.

- In Chapter 2, we present the background needed to understand the different concepts of our research work. In particular, we first present the formal definitions of widely adopted privacy models. Next, we present a utility measure for classification analysis. Afterwards, we present the principles that are crucial for establishing trust and discuss methods for imputation of missing data. Then, we provide an overview of information networks and discuss widely-adopted graph metrics. Finally, we discuss differential privacy in the context of anonymizing network data, followed by information loss measures.

- In Chapter 3, we summarize the literature to the problems presented in this thesis for the following related areas: monetizing data privacy for business value generation, a trade-off between privacy and utility in data integration, statistical disclosure control methods, policies and regulations with the perspective of data protection, data trustworthiness and auction-based pricing, cryptographic primitives, relational data anonymization in a distributed setup under differential privacy models, network data anonymization under non-differential privacy models and differential privacy models.

- In Chapter 4, we address the problem of developing privacy-preserving data mashup model to quantify the costs and benefits for releasing integrated anonymized data of multiple providers when trading person-specific information in the e-market. First, we provide an introduction followed by the challenges and problem statement, then we present a solution to address the research problem, and finally, we evaluate our proposed model based on the incorporated factors for multiple data providers by conducting extensive experiments on real-life data. The work in this chapter has been published in [95].

- In Chapter 5, we address the problem of data trustworthiness, privacy protection, and profit

distribution in integrating data from multiple data providers for trading person-specific information. First, we provide an introduction followed by an overview of trust mechanism and the problem statement, then we describe our proposed solution to address the problem, and finally, we evaluate the robustness of our proposed approach by conducting extensive experiments on real-life data. The work in this chapter has been published in [96].

- In Chapter 6, we address the problem of publishing heterogeneous network data, particularly, the protection of sensitive links of a patient within health-network data. First, we provide an introduction followed by the formal definition of our problem, then we describe our proposed solution to address the problem, and finally, we evaluate our proposed method by conducting extensive experiments on three real-life datasets that can be modeled as networks. The work in this chapter is under review in a refereed journal.

- Finally, in Chapter 7, we summarize the thesis contributions and shed light on some future research.

Chapter 2

Preliminaries

In this chapter, we present the background needed to understand the different concepts of our research work. In particular, we first present the formal definitions of widely adopted privacy models. Then, we present a utility measure for classification analysis. Afterwards, we present the principles that are crucial for establishing trust and discuss methods for imputation of missing data. Then, we provide an overview of information networks and discuss widely adopted graph metrics. Finally, we discuss differential privacy in the context of anonymizing network data, followed by information-loss measures.

2.1 Privacy models

In this section, we present the formal definitions of widely adopted models from the perspective of a single data custodian, namely k-*anonymity*, LKC-*privacy*, and ϵ-*differential privacy*.

Definition 2.1.1 (k-*anonymity* [147])**.** Let $D(A_1, \ldots, A_m)$ be a data table and QID be its quasi-identifier. D satisfies k-*anonymity* if, and only if, each group of QID appears in at least k records in D . ■

k-anonymity does not provide adequate privacy protection if the sensitive values in an equivalence class (i.e., the group of records matching a QID value) lack diversity, that is, it is subject to attribute linkage attacks. Due to the curse of high dimensionality [10], enforcing k-anonymity on high-dimensional data would result in significant information loss. To overcome this bottleneck,

Mohammed et al. [123] pointed out that, in a real-life privacy attack, it is very difficult for an adversary to acquire the values of all QID attributes of a target victim, leading to the LKC-*privacy* model. In this model, an adversary's background knowledge is bounded by *at most* L values of the QID attributes.

Definition 2.1.2 (LKC-*privacy* [123])**.** Let L be the maximum number of QID attributes acquired by an adversary as prior knowledge about a target victim and $S \subseteq Sens$ be a set of sensitive values. A data table D satisfies LKC-*privacy* if, and only if, for any qid with $0 < |qid| \leq L$,

(1) $|D[qid]| \geq K$, where $K > 0$ is an integer representing the anonymity threshold, *and*

(2) for any $s \in S$, $P(s|qid) \leq C$, where $0 < C \leq 1$ is a real number representing the confidence threshold. ■

Intuitively, LKC-privacy prevents both record and attribute linkage attacks [61] by ensuring that every qid value with maximum length L in D is shared by at least K records and that the confidence of inferring any sensitive values in S is not greater than C, where L, K, C are thresholds and S is a set of sensitive values specified by the data custodian. LKC-privacy bounds the probability of a successful record linkage to be $\leq 1/K$ and the probability of a successful attribute linkage to be $\leq C$, provided that the adversary's background knowledge qid does not exceed L attributes.

Dwork et al. [50] propose *differential privacy* (*DP*) that provides strong privacy guarantees to an individual independently of an adversary's background knowledge and computational power. The intuition of differential privacy is that individual information is not revealed from the output of the analysis in the anonymized data. In other words, it is insensitive whether an individual record is present in the input dataset or not. It is mathematically defined as follows:

Definition 2.1.3 (ϵ-*differential privacy*)**.** [50]. A sanitization mechanism $\mathcal{M}$ provides ϵ-*differential privacy*, if for any neighboring datasets D_1 and D_2 differing by at most one record (i.e., symmetric difference $|D_1 \triangle D_2| \leq 1$), and for any possible sanitized dataset $\hat{D}$,

$$\Pr[\mathcal{M}(D_1) = \hat{D}] \leq e^{\epsilon} \times \Pr[\mathcal{M}(D_2) = \hat{D}],$$

where the probability is taken over the randomness of the $\mathcal{M}$. ■

ϵ is the *privacy budget* that is specified by the data custodian. A smaller value of ϵ results in stronger privacy protection but produces lower data utility. Conversely, a larger value of ϵ results in weaker privacy protection but yields higher data utility.

The *Laplace mechanism* and *exponential mechanism* are the canonical examples of a differentially private mechanism. A standard mechanism to achieve differential privacy is to add random noise to the outcome of the analysis for providing privacy protection. The calibration of noise is done according to the *sensitivity* of the function f.

Definition 2.1.4 (*Sensitivity*)**.** For any function $f : D \rightarrow \mathbb{R}^d$, the sensitivity of f is

$$\Delta f = \max_{D,D'} ||f(D) - f(D')||_1 \tag{1}$$

for all D, D' differing at most by one record. ■

The sensitivity of a function does not depend on the data but instead produces an upper bound to how much noise we must add to the true output to preserve privacy. Suppose function f answers count queries over a dataset D. Then, the Δf is 1 because $f(D)$ can differ at most by 1, due to the addition or removal of a single record.

Laplace mechanism. Dwork et al. [50] proposed the Laplace mechanism. It is appropriate when the output of function f is a real value, and f should perturb its output with a noisy answer to preserve privacy. The noise is calibrated based on the privacy parameter ϵ and the sensitivity of the utility function Δf. Formally, the Laplace mechanism takes as inputs a data set D, the privacy parameter ϵ, and a function f and outputs $f(\hat{D}) = f(D) + \texttt{Lap}(\lambda)$, where $\texttt{Lap}(\lambda)$ is a noise drawn from the Laplace distribution with probability density function $\Pr(x|\lambda) = \frac{1}{2\lambda}\texttt{exp}(-|x|/\lambda)$. The variance of this distribution is $2\lambda^2$, and the mean is 0.

Exponential mechanism. McSherry and Talwar [122] proposed the exponential mechanism. It is appropriate for situations in which it is desirable to choose the best response, because adding noise directly to the count can eradicate its value. Given an arbitrary range $\mathcal{T}$, the exponential mechanism is defined with respect to a utility function $u : (D \times \mathcal{T}) \rightarrow \mathbb{R}$ that assigns a real valued score to every output $t \in \mathcal{T}$, where a higher score means better utility. The exponential mechanism induces a probability distribution over the range $\mathcal{T}$ and then samples an output t. Suppose $\Delta u = \max_{\forall t,D,D'}$

$|u(D,t) - u(D',t)|$ to be the sensitivity of the utility function. The probability associated with each output t is proportional to $\texttt{exp}(\frac{\epsilon u(D,t)}{2\Delta u})$.

Differential privacy is increasingly being accepted as the cornerstone of privacy protection by domain experts due to its robustness and rigorous mathematical definition. In literature, two settings, namely *interactive* and *non-interactive*, are mainly discussed regarding utilization of the privacy budget ϵ. The primary difference is that in the interactive setting [50, 57, 180, 181] the data custodian holds the raw data and a data analyst poses a set of queries in real time for which the data custodian provides differentially private answers. Each query would utilize a fraction of ϵ incrementally to produce a noisy answer. When the entire privacy budget has been depleted, a data analyst would not be able to get the answer by querying the database. On the other hand, in the non-interactive setting, the data custodian first anonymizes its raw data by utilizing the entire privacy budget. Later, the anonymous (ϵ-differentially private version) data releases to the data analyst, who would perform an analysis without any constraints on the data usage. This approach is widely known as *privacy-preserving data publishing* (*PPDP*) [61], which is more appropriate in many real-life data sharing scenarios because of the flexibility for a data analyst to perform an analysis without back and forth querying of the database. In this thesis, we focus on the non-interactive setting for a differentially private release of relational data in a distributed setup.

2.2 Information utility

The information utility is measured depending on the requirements for data analysis. In this thesis we present classification analysis as a utility measure on the consumer's specified service request and analysis task.

Score for classification analysis: We use *information gain*, denoted by $InfoGain(v)$, to measure the *goodness* of a specialization [62]. Our selection criterion, $IGScore(v)$, is to keep the specialization (i.e., replacing a generic value of an attribute with specific value in the domain) $v \rightarrow child(v)$ that has the maximum $InfoGain(v)$:

$$IGScore(v) = InfoGain(v) \tag{2}$$

Let D_x denote the set of records in the data table D generalized to the value x. Let $freq(D_x, cls)$ denote the number of records in D_x having the class cls. Note that $|D_v| = \sum_c |D_c|$, where $c \in child(v)$. The information gain $InfoGain(v)$ and entropy $H(D_x)$ are defined as follows:

$$InfoGain(v) = H(D_v) - \sum_c \frac{|D_c|}{|D_v|} H(D_c) \tag{3}$$

$$H(D_x) = -\sum_{cls} \frac{freq(D_x, cls)}{|D_x|} \times log_2 \frac{freq(D_x, cls)}{|D_x|} \tag{4}$$

where $H(D_x)$ measures the *entropy* of classes for the records in D_x [137], and $InfoGain(v)$ measures the reduction of the entropy by specializing v into $c \in child(v)$. A smaller entropy $H(D_x)$ implies a higher purity of the partition with respect to the class values.

We build a classifier on $2/3$ of the records of the anonymized dataset as the training set and measure the *Classification Error* (CE) on $1/3$ of the records of the anonymized records as the testing set to determine the impact of anonymization on data utility for classification analysis. *Classification Accuracy* (CA) is calculated by $1 - (CE)$. We use the well-known C4.5 classifier [137] for classification analysis.

2.3 Trust aspects

Trust is a critical aspect of decision making in e-commerce. Trust principles are a part of many service-oriented architectures (SOA)-based models where participants in the system interact for service delivery and use [171]. We review the principles that are crucial for trust establishment. First, entities should be identified [88] as they have claimed. In the world of the Internet, where entities are physically isolated, they may use fake identities to show their existences in their interactions. Authentication is a way of validating entities by the use of usernames and passwords, tokens, or digital certificates before granting them access to the resources or applications [29]. Second, it is crucial for trust formation to initialize new entities with trust rates. This process is called *trust bootstrapping*. Third, when one entity trusts another entity's decision there is a risk of an undesirable outcome due to some degree of uncertainty and dependency [102]. The risk is considered to be a

prerequisite before trusting the trustee's behavior. The entities who are involved in an interaction should comply with the norms and rules of trust to avoid penalties for violation. Fourth, trust rates are of two types: local and global [166]. Local trust rating refers to a personalized score in which each trustee would have different scores from the trustors. Global trust rating provides a unique score about the trustee regardless of the trustors involved in the evaluation. Global trust rating often requires the trusted third party (TTP) services to collect feedback from the trustors about trustees and compute the trust rates. Last, security and privacy are the main components for trust establishment. Trust is required when there is uncertainty; it has widely been accepted that perfect security does not exist, even though security measures are necessary to gain trust in many circumstances [26]. Customers who place their orders online and submit private information in the form of their name, address, and credit details necessitate that their private information should not be disclosed or shared by any means with untrusted parties. Building a trust relationship requires protection of customers' privacy in online transactions. We pay attention to some of the aforementioned principles for establishing trust on the data providers in the context of our trust mechanism.

2.4 Methods for imputation of missing data

There are different types of missing data [79], such as Missing at Random (MAR), Missing Completely at Random (MCAR), and Missing Not at Random (MNAR). MAR refers to the probability of missing data of an attribute on other present observations of attributes in the dataset, but not on the attribute's own value. Whereas, MCAR occurs when there is no dependency on the attribute value itself or any other attribute in the dataset. The special case MNAR occurs when the missing data meets neither the condition defined in MAR nor MCAR. In such case, missing values in MNAR cannot be imputed by using other present observations of attributes.

There is extensive research [17, 25, 188, 189] done on machine learning methods such as hot-deck imputation, mean imputation, regression imputation, k-nearest neighbors imputation, and random forest imputation. Hot-deck imputation is a technique for replacing missing values of a non-respondent on one or more attributes with the most similar characteristics to a respondent [17]. This method has been used in practice, but the theory is not as well developed. Mean imputation is a

technique used for replacing missing values of a numerical attribute by the average value, and for a categorical attribute by the mode, i.e., most frequent value. This method is quite simple, but it is not suitable for multivariate analysis. Regression imputation first builds a model from the observed data, then predictions for the incomplete cases are calculated under the fitted model to replace the missing data [189]. The drawback of the regression model is that all predicted values fall directly on the regression line, which decreases variability. Random forest is a type of ensemble learning method [188]. It is used widely for classification and regression tasks. The learning process of a random forest algorithm is based upon the bootstrap aggregation technique, in which a specified number of trees are trained on a given dataset. As the random forest is built upon multiple decision trees, intrinsically it uses the same approach for attribute selection measures such as information gain, gini index, and gain ratio of decision trees. Random forest can deal with missing values with different types of variables. k-nearest neighbors (kNN) imputation is an efficient approach for replacing missing values on some records by computing another value from similar examples in the given dataset [25]. kNN computes the similarity by using a distance metric, such as Euclidean distance. k is a positive integer, when $\mathsf{k} = 1$ the object is simply assigned to the class of that single nearest neighbor. When $\mathsf{k} > 1$ the object is assigned to the class that appears most frequently within the k-subset. kNN generally produces good quality predictions, but the computation cost is high because of computing distances.

2.5 Types of information networks

Generally, an information network is a representation that models the real world, focusing on objects and the interactions between objects [156]. These interactions in the network can be *symmetric* and *asymmetric*. In a symmetric interaction the relationship between objects can be in both directions, whereas asymmetric represents a one-way relationship. Typical examples of information networks are social networks, collaboration networks, health networks, and communication networks.

Definition 2.5.1. *(*Homogeneous information network*)* [156]. Given a network, $G = (V, E)$ with an entity type-mapping function $\varphi : V \rightarrow \mathcal{E}$ and a relation type-mapping function $\psi : E \rightarrow \mathcal{R}$, it is called a *homogeneous information network* if there exists only one type of entities and relations (i.e.,

$|\mathcal{E}| = |\mathcal{R}| = 1$). ■

Definition 2.5.2. *(*Heterogeneous information network*)* [156]. The information network is called a *heterogeneous information network* if the types of entities $|\mathcal{E}| > 1$ or the types of relations $|\mathcal{R}| > 1$. ■

The *network schema* describes the meta structure of a heterogeneous information network, in which type constraints on the set of objects and relationships are specified. Many complex networks are modeled by heterogeneous networks to capture rich semantics. Traditional mining methods [105, 164] are designed for homogeneous networks, which cannot be directly applied to solve the problems of heterogeneity in many real-world networks. Various mining methods have been proposed to tackle the problem of heterogeneity for network analysis, such as ranking-based classification and clustering [81, 157], meta-path-based similarity search [160], relationship prediction and relation strength learning [159, 161], and community evolution [158]. Recently, advanced embedding methods for homogeneous networks [69, 141, 163] and heterogeneous networks [58, 70, 152, 153] have gained increasing attention for large-scale network analysis. On the one hand, these mining and embedding methods for heterogeneous networks serve different requirements of network analysis, but on the other hand, the privacy of an individual is at stake unless proper protection measures are deployed.

2.6 Network measures

Here, we discuss some widely adopted graph metrics, namely betweenness centrality, degree centrality, closeness centrality, and harmonic centrality. These measures [108] contribute to the analysis of the structural properties of a network.

2.6.1 Betweenness centrality

The intuition of this measure is to determine the importance of a node in connecting other nodes. The betweenness of a node v_i in the network is computed by

$$CB(v_i) = \sum_{j \neq i \neq k \in V} \frac{\sigma_{v_j, v_k}(v_i)}{\sigma_{v_j, v_k}} \tag{5}$$

where $|V|$ is the number of nodes in the network, σ_{v_j,v_k} is the total number of shortest paths from node v_j to node v_k, and $\sigma_{v_j,v_k}(v_i)$ is the number of those paths that pass through v_i. To normalize the betweenness centrality, divide the metric in Eq. (5) by $(|V|-1)(|V|-2)$ for directed graphs and by $(|V|-1)(|V|-2)/2$ for undirected graphs.

2.6.2 Degree centrality

A node is in the "central" if it has many direct neighbors. For a directed network, *indegree* is the number of incoming links representing the popularity of a node, whereas *outdegree* is the number of outgoing links representing the sociability of a node. In an undirected network, the *degree* of a node is simply the number of directly connected neighbors ignoring edge directions. The normalized degree centrality CD for a node v_i is computed by

$$CD(v_i) = \frac{d(v_i)}{|V|-1} \tag{6}$$

where $d(v_i)$ is the degree of node v_i.

2.6.3 Closeness centrality

In this measure, a node is in the "central" if it is close to many other nodes, and of which the closeness can be measured by the shortest paths for reaching those nodes. The normalized closeness centrality CC for a node v_i is computed by

$$CC(v_i) = \frac{|V|-1}{\sum_{j\neq i}^{|V|} d(v_j, v_i)} \tag{7}$$

where $d(v_j, v_i)$ is the shortest-path distance between v_j and v_i. If the direction between nodes v_i and v_j is not specified, then the total number of nodes $|V|$ is used in Eq. (7) instead of the path length.

2.6.4 Harmonic centrality

It is a variant of closeness centrality that deals with the scenario of unconnected networks. It is the sum of the reciprocal of the shortest path distances from all other nodes to a given node. The

normalized harmonic centrality CH for a node v_i is computed by

$$CH(v_i) = \frac{1}{(|V|-1)} \times \sum_{j \neq i}^{|V|} \frac{1}{d(v_j, v_i)} \tag{8}$$

If there is no path from v_j to v_i, then $1/d(v_j, v_i)$ becomes 0.

2.7 Differential privacy for network data

Differential privacy [50] is a widely known privacy model with an assumption that all the records in the database are independent of each other. A line of research [85, 101, 179, 182] indicates that differential privacy may not guarantee privacy against adversaries with arbitrary background knowledge when data records are correlated. To tackle this issue, a notion of correlation parameter k is proposed by [37] that provides a similar differential privacy guarantee when releasing network data. In the correlation setting, any record in database D is correlated to *at most* $k-1$ other records. The intuition of their solution is to add extra Laplace noise in the anonymization process to compensate for the effect of correlation.

Definition 2.7.1. *(ϵ-differential privacy under correlation)* [37]. A sanitization mechanism $\mathcal{M}$ provides *ϵ-differential privacy* if for any two datasets D_1 and D_2 with a correlation parameter k that differs on *at most* one record (i.e., symmetric difference $|D_1 \triangle D_2| \leq 1$), and for any possible sanitized dataset $\hat{D}$, we have

$$\Pr[\mathcal{M}(D_1) = \hat{D}] \leq e^{\frac{\epsilon}{k}} \times \Pr[\mathcal{M}(D_2) = \hat{D}],$$

where the probability is taken over the randomness of $\mathcal{M}$. ■

In the literature, *node-differential privacy* [33, 42, 92] and *edge-differential privacy* [37, 76, 173] are the most prevalent formulations for anonymizing network data. In node-DP, two graphs G and G' are neighboring graphs if they differ by *at most* one node and, by extension, all its edges. Whereas in edge-DP, two graphs G and G' are neighboring graphs if they differ by *at most* one edge or an *isolated* node (a node that has no edges). The following definitions define two types of neighboring

graphs under node- and edge-differential privacy, respectively.

Definition 2.7.2. *(*Neighborhood under node-differential privacy*)* [72]. Given graph $G = (V, E)$, where V is a set of nodes and E is a set of edges, two graphs G and G' are neighbors if $|V \oplus V'| = 1$ and $E \oplus E' = \{(u, v)|u \in (V \oplus V') \text{ or } v \in (V \oplus V')\}$. ■

Definition 2.7.3. *(*Neighborhood under edge-differential privacy*)* [72]. Given graph $G = (V, E)$, where V is a set of nodes and E is a set of edges, two graphs G and G' are neighbors if $|V \oplus V'| + |E \oplus E'| = 1$. ■

2.8 Information loss measures

Here, we discuss some generic measures to quantify the information loss when releasing anonymized network G'. The general goal is to minimize information loss. It is the antithesis of data utility, where a decrease in information loss leads to an improvement in data utility.

2.8.1 Mean absolute error

This measures the absolute error by comparing the degree centrality score of a node v_i in the anonymized network G' with respect to the original network G. The mean absolute error (MAE) [174] for all the nodes in the network is computed as follows:

$$MAE(G, G') = \frac{1}{|V|} \times \sum_{i=1}^{|V|} |CD(G', v_i) - CD(G, v_i)| \tag{9}$$

2.8.2 Average relative error

This measures the relative error of a node v_i in the anonymized network G' with respect to the original network G [94]. The average relative error (ARE) for all the nodes in the network is computed as follows:

$$ARE(G, G') = \frac{1}{|V|} \times \sum_{i=1}^{|V|} \frac{|CD(G', v_i) - CD(G, v_i)|}{CD(G, v_i)} \tag{10}$$

2.8.3 Kullback–Leibler divergence

Degree distribution captures the important structural properties of a network. This one computes the frequency count of the occurrence of each degree to differentiate the number of connections between nodes in a network. For a directed network, the frequency counts for the indegree and outdegree of a node are computed based on the type of degree direction. Given the degree distributions of the original network and the anonymized network, $DD(G)$ and $DD(G')$, we measure their difference by *Kullback–Leibler divergence* [99] as follows:

$$KLDiv(DD(G)||DD(G')) = \sum_{i=0}^{|V|-1} DD(G)[i] \cdot \ln\left(\frac{DD(G)[i]}{DD(G')[i]}\right) \tag{11}$$

Chapter 3

Literature Review

In this chapter, we summarize the literature to the problem presented in Chapter 4, Chapter 5, and Chapter 6 for the following related areas: monetizing data privacy for business value generation, trade-off between privacy and utility in data integration, statistical disclosure control methods, policies and regulations with the perspective of data protection, data trustworthiness and auction-based pricing, cryptographic primitives, relational data anonymization under differential privacy models, and network data anonymization under non-differential privacy models and differential privacy models.

3.1 Monetizing data privacy for business value generation

Many organizations are embracing innovations in digital economy to maximize their business value through data. Barbara et al. [176] conducted seven case studies on companies that monetize data by selling information-based products and/or services. They hypothesized that a company whose business model draws upon six sources, such as data, data architecture, data science, domain leadership, commitment to client action, and process mastery, can bring a competitive advantage for information business value. Barbara et al. [175] further identified an approach that they termed "Data Value Assessment" to analyze the costs, benefits, and risks of selling information-based products and services by business enterprises. Li et al. [110] proposed a theoretical framework for private data pricing in an interactive setting. There are three main actors in their proposed architecture:

Data owners contribute their personal data; a *buyer* submits an aggregate query and pays its price to a *market maker*; and a *market maker*, a trusted party to both, answers *buyer* queries on behalf of *data owners* by adding an appropriate noise [50] in response to the query. The *market maker* compensates the *data owners* whenever they suffer from a privacy loss in response to a *buyer's* query. Riederer et al. [142] proposed a mechanism called "transactional privacy" to control the disclosure of personal information in a privacy-preserving system. This mechanism allows end users to release personally identifiable information (PII) by giving them the choice to valuate their personal information. Their system leveraged prior work on auctions and particularly the exponential mechanism [122] to guarantee truthfulness in the bidding process. In this thesis, we follow a distributed approach in a non-interactive setting for data mashup of multiple data providers, which is different from our previous work [94] in which the challenges were to quantify the costs and benefits between privacy and utility from the perspective of a single data custodian. In addition, the business model presented in this thesis can derive the contribution of each data provider in terms of monetary value by computing the information gain on the data mashup.

3.2 Trade-off between privacy and utility in data integration

Arafati et al. [19] proposed a cloud-based framework for a privacy-preserving Data-as-a-Service (DaaS) mashup that enables data providers to integrate their person-specific data on demand depending on a consumer's request for data analysis. In their framework, a data consumer can submit a request with a set of attributes, bid price, and classification accuracy. They introduced a greedy algorithm that can dynamically determine the group of DaaS providers offering the lowest price per attribute. They employed a Privacy-preserving High-dimensional Data Mashup (PHDMashup) algorithm [62] for secure data integration and to preserve the privacy of mashup data using the LKC-privacy model [123]. Mohammed et al. [128] proposed a differentially private algorithm to securely integrate person-specific data from two parties so that integrated data maintains the necessary information to support data utility. They presented a scenario for a distributed setup to integrate the vertically partitioned data, where different attributes for the same set of individuals are held by two parties. No additional information is leaked to any party as a result of integrating data. There

are other works [87, 124] that address the problem of integrating horizontally partitioning data in a distributed manner. This would yield different costs and benefits when quantifying the privacy and utility from the integrated data using horizontal partitioning. In this thesis, the data mashup model employs the approach that was presented in [62] and [128] for vertically partitioned data to satisfy LKC-privacy and ϵ-differential privacy requirements, respectively.

3.3 Statistical disclosure control methods

Many non-perturbative and perturbative anonymization methods, such as global and local recoding [162, 169], suppression and local suppression [118, 169], sampling [154], micro-aggregation [46], noise addition [103], data swapping [41], and post randomization [106] have been adopted in the past with the goal of providing confidentiality and privacy in publishing person-specific data. According to Gehrke [66], the statistical methods used for limiting information disclosure do not formally address how much sensitive information an adversary would glean from the published data. Waal and Willenborg [169] used global recoding and local suppression methods to quantify the information loss in a microdata set. In the case of a global recoding method, specific attribute values are mapped to the same generalized value in all records; in the case of local suppression, the specific value of an attribute in a record changes to a 'missing' value, but the attribute values in other records remain unchanged [170]. Global recoding is the preferable method when there are many unsafe combinations to eliminate in the person-specific data and when one wants to obtain a uniform categorization of attributes [169]. Truta et al. [167] used a microaggregation statistical disclosure control technique to measure the trade-off in disclosure risk and information loss on synthetic data based on the criteria specified by the data owner.

3.4 Policies and regulations for data protection

Currie and Seddon [39] discussed the cross-country approaches to data privacy, regulation, and rules. They did a survey in six countries to collect the views of people on the benefits and risks for adopting cloud computing in a healthcare setup. Generally, healthcare professionals are in favor of adopting cloud computing, but stakeholders involved in the setup have to provide a guarantee for the

protection of personal data subject to the regulations enforced in their jurisdictions. They addressed an important issue of how international governments harmonize an effective legal and regulatory framework for trans-border data flows over the cloud environment. Recent studies [39, 107] showed that more than sixty countries in the world have adopted privacy and data protection laws that regulate trans-border data flows. Hu et al. [77] presented Law-as-a-Service (LaaS) as an emergent technology for cloud service providers to ensure that legal policies are compliant with the laws for users. They presented a conceptual layout of the law-aware semantic policy infrastructure in which a semantic cloud of Trusted Legal Domains (TLDs) are established over the Trusted Virtual Domains (TVDs). Each TLD has a super-peer that provides data integration services for its peers. The super-peer specifies how legal compliance policies are unified and enforced in a domain. Legal policies are composed of OWL-DL ontologies and stratified Datalog rules with negation for a policy's exceptions handling through defeasible reasoning. Description Logic (DL)-based ontologies provide data integration, while Logic Program (LP)-based rules provide data query and protection services.

3.5 Data trustworthiness and auction-based pricing

Different trust models, frameworks, and techniques have been proposed to address the problem of data trustworthiness. Bertino and Lim [27] proposed a framework that consists of two key components. The first component is based on the concept of data provenance in which information relies on the origin of data for computation of trust scores. The second component undertakes the notion of confidence policy in which query results are filtered based on the specified confidence range for use in certain tasks. Dai et al. [40] proposed a provenance-based model in which they evaluated the trustworthiness of data items based on the aspects of data similarity, path similarity, data conflict, and data deduction. Benjelloun et al. [24] introduced databases with uncertainty and lineage in which they combined the concept of *lineage* and *uncertainty* for querying in probabilistic databases.

There are studies related to data trustworthiness in mission-critical applications [115, 165]. Tang et al. [165] proposed trustworthiness analysis for sensor networks in cyber-physical systems to eliminate false alarms that occur due to random noise or defective sensors. They validated events

by using a graph-based filtering approach. However, their method does not deal with coordinated attacks where a fraction of sensing nodes are compromised by malicious attackers. Lim et al. [115] addressed this challenge by adopting a game-theoretic approach based on the Stackelberg competition for defending the network against false data injection. They assessed trust scores for both data items and network nodes using the cyclic framework proposed in [114]. This framework is based on the interdependency property between data items and their associated network nodes in which trust scores are computed using two types of similarity functions. First, *value similarity* is derived from the principle that the more that similar values refer to the same event, the higher the trust scores. Second, *provenance similarity* is based on the principle that the more that different data sources are with similar data values, the higher the trust scores. Mainly, the approaches presented in the above works fall under the category of workflow provenance. In contrast, we are not concerned with the higher level of instrumentation at the data collection phase by data providers because it is not practically efficient to determine the data provenance in the e-market. Furthermore, the above works mainly focus on similarity functions for trust computation but do not consider privacy protection for data trustworthiness. We propose an approach that makes novel use of information entropy to verify the correctness of data in a multiple data providers scenario where a semi-trusted arbitrator cannot derive any customers' private data when evaluating the trustworthiness of the participating data providers.

Karabati et al. [91] studied the challenge of pricing with short-term capacity allocation decisions for multiple products in a single-supplier and multiple-buyers scenario. They proposed an iterative auction mechanism with monotonically increasing prices to maximize the profit of a supplier. Li et al. [113] presented dynamic pricing strategies for resources allocations in cloud workflow systems. Their proposed reverse auction-based mechanism allows resource providers to change the prices during the auction, depending upon their trading situation, to improve the efficiency of resource utilization as well as the competitiveness. Wu et al. [178] employed a *Vickrey-Clarke-Groves* (*VCG*) auction to implement a dynamic pricing scheme for multi-granularity service composition. They considered both coarse-grained and fine-grained services for composition. In their approach, service providers bid for services of different granularities in the composite service, whereas a recipient of the bids decides a composition that minimizes the overall cost while satisfying quality constraints.

They solved the problem of winner determination by an integer programming model. In this thesis, we define the procedure for the valuations of data providers' attributes based on the VCG mechanism.

3.6 Cryptographic primitives

Private set intersection (*PSI*) is a cryptographic primitive that was first formally defined in [55]. The protocols for PSI allow two parties, holding sets A and B, to compute the private intersection without revealing to each other any additional information from their respective sets. At the end of the protocol, either one or both parties may learn the size of the intersection, depending on the application. Since its inception, many variants have been proposed in an attempt to speed up PSI computation, including garbled Bloom filters [48, 78], server-aided computations [47, 89, 90], and computational optimizations [104, 132, 134].

Oblivious Transfer (OT) is one of the fundamental primitives in cryptography and has been extensively used for secure multi-party computation. Particularly, the most efficient OTs were introduced by Pinkas et al. [132] and further strengthened in [104, 133, 134]. Kolesnikov et al. [104] proposed a batched related-key oblivious pseudo-random function (BaRK-OPRF) protocol to improve the performance of semi-honest secure PSI. They achieved a 1-out-of-n OT of random messages for an arbitrarily large n at nearly the same cost as 1-out-of-2 in [80]. The new OPRF construction of Pinkas et al. [134] is similar to Kolesnikov et al. [104] except in handling error correcting code. Kolesnikov et al. [104] demonstrated that their protocol outperforms Pinkas et al. [133] in almost all settings, particularly for the long bit length of input and large values of the input size. In practice, the OT-based protocols are much faster than the random garbled Bloom filter-based protocols for larger set sizes, yet these protocols do not have the lowest communication cost [104]. One desirable property is to achieve the fairness that ensures either all the parties of a group learn the output of the computation or none do [90]. This is not the case with standard approaches to PSI. These approaches are suitable for different motivating applications in private data mining, online recommendation services, and genomic computations.

Our solution to the problem is different from several PSI-based approaches in which the intention is to achieve both privacy and security simultaneously. In our approach, we maintain confidentiality

and integrity by exchanging only an encrypted information gain message and its keyed hash between a data provider and the cloud server based on a random challenge (i.e., attribute request) of the cloud server, instead of exchanging encrypted individual data items. This apparently reduces the overhead of communication. In addition, we do not rely on the cloud server to perform the computation on clients' private data. In the context of privacy, PSI protocols enable parties to privately know the result from their intersection, but the total information is not published for data analysis [183]. However, we intend to securely integrate person-specific data from multiple data providers and to release differentially private data for classification analysis.

3.7 Relational data anonymization under differential privacy models

Here, we discuss related works on differentially private release of relational data in the non-interactive setting for a distributed setup.

The group of works [14, 128] based on distributed approaches are suitable for multiple parties whose prime concern is to integrate their data in a way that no party could learn additional information from other parties' data as a result of data integration. Mohammed et al. [128] proposed an algorithm, called *DistDiffGen*, in which data is vertically partitioned among multiple parties in a distributed setup. It allows two parties to securely integrate their person-specific data while maintaining necessary information to support data utility. Each party in this setup owns a mutually exclusive set of attributes over the same set of records. A similar problem has also been studied by Alhadidi et al. [14] where data is horizontally partitioned among two parties. Each party in this setup owns a disjoint set of records over the same set of attributes. In this thesis, we employ *DistDiffGen* [128] for a distributed setup with an extension for multiple data providers to achieve ϵ-differential privacy. There are existing works that allow data integration for horizontally partitioned databases [87, 124] and vertically partitioned databases [62, 82, 126] under the privacy constraints in a distributed setup. These works are based on syntactic privacy models, which are vulnerable to certain attacks such as minimality attack [177], composition attack [64], and deFinetti attack [98]. Therefore, we adopt differential privacy [50] because it provides strong privacy guarantees against such attacks. Whereas existing work [95] proposed a privacy-preserving data mashup model that allows the collaboration of

multiple data providers for integrating their data and derives the contribution of each data provider by evaluating the incorporated cost factors, our work derives the monetary shares for the chosen data providers from their contribution to information utility over the differentially private integrated data for classification analysis and their trust scores.

3.8 Network data anonymization under non-differential privacy models

A family of works [38, 119, 191, 192] has proposed to preserve the structural information in graph networks. Liu and Terzi [119] proposed an approach to construct an anonymous graph of k-degree anonymity, which requires generation of at least $k-1$ other nodes, for every node v. This notion of anonymity prevents identity disclosure from structural attacks based on adversary knowledge on a certain degree of nodes. Zhou and Pei [191] proposed k-neighborhood anonymization to prevent an adversary's attack with 1-neighborhood background knowledge about the victim. The goal of this approach is to ensure that the identity of an individual may not be revealed with a confidence greater than $1/k$ in the sanitized version of the original graph. Cheng et al. [38] proposed k-isomorphism, a solution that generates k disjoint subgraphs for an input graph G. k-isomorphism prevents an adversary inference on re-identification of nodes and disclosure of edges in the published k-secure graph, denoted by G_k. The complexity of a subgraph isomorphism problem is NP-hard.

Zou et al. [192] developed *K-Match* algorithm, which has the following techniques: graph partitioning, graph alignment, and edge copy to achieve k-automorphism. According to this algorithm, for each node v in the published graph, denoted by G^*, there exist $k-1$ symmetric nodes to resist any structural attacks. They argue that an adversary cannot distinguish v from its other $k-1$ symmetric nodes based on any structural information, and also it cannot identify the target node with a probability higher than $1/k$. Fung et al. [63] presented a method to k-anonymize a social network while preserving frequent-sharing patterns and maximal frequent-sharing patterns. The purpose of the aforementioned graph anonymization algorithms is to defend against graph structural attacks. Zhang et al. [186] argued that these algorithms are not effective in preserving the privacy of an anonymized heterogeneous information network. Generally, all the above works provide prevention against

node re-identification, edge disclosure, or both, based on the assumption that the adversary has access to limited background knowledge about a victim. We propose a solution that does not make any assumptions about the adversary's knowledge of victims by adopting the differential privacy model [50], which provides strong privacy guarantees independently of an adversary's background knowledge.

3.9 Network data anonymization under differential privacy models

In the literature, node-differential privacy [33, 42, 92] and edge-differential privacy [37, 76, 173] are the most prevalent formulations for network data anonymization. Node-DP is too strong to get the desired utility in a sparse network. To overcome this problem Kasiviswanathan et al. [92] developed a customized notion of low-sensitivity based projection operators to preserve certain graph statistics. They employed Laplace and Cauchy distributions for output perturbation. In addition, they devised a generic method to apply any differentially private algorithm for bounded-degree graphs to an arbitrary graph. They assumed that the tail of the degree distribution decreases rapidly, which resembles the characteristics of scale-free networks [86]. A similar problem was also studied by Borgs et al. [33]. They proposed a node-DP algorithm for fitting a high-dimensional statistical model to a sparse network by the use of non-parametric block model approximation. They employed Lipshitz extensions inside the exponential mechanism [122] to control the sensitivity of the score functions. Raskhodnikova et al. [139] proposed some Lipschitz extensions for designing a node-private algorithm to release the degree distribution of a graph. The extensions use convex programming and can be computed in polynomial time. It provides more accurate graph statistics than [92].

Day et al. [42] proposed a graph projection technique to transform an input graph to be θ-degree-bounded for releasing node-private degree distributions. They showed that the sensitivity from the projection is 2θ+1 when releasing a degree histogram, whereas for a cumulative degree histogram the sensitivity is θ+1. Their results indicate a significant improvement over the flow-based approach [139] in releasing node degree distributions. Song et al. [155] proposed a node-private algorithm for online graphs based on the assumption of a bounded maximum degree in the entire

graph sequence. They showed that the sequence of differences in the computed graph statistics has low sensitivity, which can yield better privacy-accuracy trade-off.

The group of works [37, 76, 173], based on edge-DP, prevents disclosure of sensitive relationships among nodes. Sala et al. [146] proposed a partition-based approach to divide the dK-2-series into subseries and then inject the noise proportional to its local maximum degree to generate synthetic graphs. They used large privacy parameters $\epsilon \in [5, 100]$ to evaluate degree-based metrics and node-separation metrics on the resulting DP-synthetic graphs. Under stringent privacy parameters (e.g., $\epsilon \leq 1.0$), the error is large because of the high noise injected by the dK-Perturbation Algorithm (dK-PA) into dK-2, resulting in a significant deviation from the original graph. Wang et al. [173] determined degree correlation parameters from the input graph and then enforced edge-DP on graph-model parameters to generate a perturbed graph. They adopted the concept of smooth sensitivity [130] for calibrating noise magnitude to guarantee privacy.

Chen et al. [37] proposed *DER*, in which a notion of correlation parameter k is introduced to provide a similar differential privacy guarantee when releasing network data with the consideration of data correlation. They formed dense regions from an adjacency matrix of input graph by first identifying a good vertex labeling, then adopting a standard quadtree [54] to explore the dense regions, and finally, making use of the exponential mechanism to reconstruct the leaf nodes of a quadtree. They assumed any record in database D can be correlated to *at most* $k - 1$ other records. It is different from *k-edge differential privacy* [72], where the goal is to protect k edges' collective information but not to conceal the presence of any single edge in the correlated setting. Hu et al. [76] proposed a differentially private method to protect sensitive edges by converting a deterministic graph into an uncertainty form. In this method, they computed the probability for each edge independently of an original structure of the network to inject uncertainty. Lin et al. [116] proposed a DP-graph structural-clustering algorithm, called *DP-SCAN*, in which they define edge-DP of adjacent graphs, and then add the Laplace noise proportional to the global sensitivity of the function. This algorithm partitions an input graph into several clusters, bridge connections, and outliers while preserving sensitive information. The above edge-DP methods focus on preserving privacy in *homogeneous networks*, whereas our proposed edge-DP algorithm protects individuals' sensitive links in *heterogeneous networks*. Existing solutions assume that edges are bidirectional and

that nodes and edges are of a single type, each. In contrast, heterogeneous networks are characterized by having multiple types of nodes and edges. Thus, solutions that are intended for homogeneous networks will not be able to maintain important semantics and structural information if applied to heterogeneous networks. In this thesis, we propose a solution that not only takes into account the types of nodes and edges in a given network, but also considers the direction of edges in the network.

Chapter 4

Privacy-preserving Data Mashup Model for Trading Person-specific Information

4.1 Introduction

Business enterprises have widely adopted web-based mashup technologies for collaboration with their trading partners. A web-based mashup involves the integration of information and services from multiple sources into a single web application. For example, real estate companies mashup their data and other third-party data with Google Maps for comprehensive market analysis. The rapid adoption of mashup technologies by the business enterprises are mainly concerned with revenue and cost factors [150]. *Enterprise Mashup Markup Language* (*EMML*) is a standard proposed by the Open Mashup Alliance to improve collaboration among business enterprises and to reduce the risk and cost of mashup implementation [143]. Several companies including IBM, StrikeIron, Kapow Technologies, and others have been actively involved in leveraging various web-based mashup technologies such as Quick and Easily Done Wiki (QEDWiki), IBM Mashup Center, and *Data-as-a-Service* (*DaaS*). Business enterprises need to focus on a data-oriented perspective along with the initiatives of *Service-Oriented Architecture* (*SOA*). They face challenges not only to protect private data over the cloud but also to legally adhere to privacy compliance rules when trading person-specific data.

Figure 4.1 presents an overview of a privacy-preserving data mashup e-market for trading person-specific information. The process consists of five steps. First, data providers register their available data on the registry hosted by the mashup coordinator, who can be a cloud service provider or one of the data providers. Second, data consumers (or data recipients) submit their data requests to the mashup coordinator. A "data request" can be a simple count query or a complicated data mining request. To provide a concrete scenario in the rest of the chapter, we assume the data request is a data mining request for classification analysis. Third, since a single data provider may not be able to fulfill the data requests from a data consumer, a mashup coordinator dynamically determines the group of data providers whose data, through interconnection, can collectively fulfill the demand of a data consumer. Fourth, the data providers quantify their costs and benefits using joint privacy requirements and integrate their data over the cloud. Finally, the anonymous mashup data is released to the data consumers. The data consumers have the option to perform the data mining operations on the cloud or take the data and perform the data mining operations locally on their own machines.

In the proposed architecture, business enterprises face four major challenges for trading person-specific information. First, extensive research has shown that simply removing explicit identifying information such as name, social security number, birth date, telephone number, and account number is insufficient for privacy protection. Many organizations believe that enforcing regulatory compliance or employing common de-identification methods is sufficient for privacy protection. Indeed, an individual can be reidentified by matching the *quasi-identifiers* QID with an external data source [147]. Second, the data providers collaborate with peer providers in order to fulfil the demands of a data consumer and to generate more profit. In addition, they would avoid sharing information other than the final integrated data because the collaborating data providers could be competitors. Third, a cloud service provider may not be a trusted party. It can be a third-party who offers data integration services over the cloud or one of the data providers. Fourth, the data providers want to ensure that the mashup data can facilitate the queries of data consumers. So, there is a trade-off between data utility and privacy protection in terms of monetary reward. In this chapter, we propose a model that examines the intangible benefits and potential risks of sharing person-specific data for classification analysis. Our model allows the data providers to quantify the costs and benefits in monetary term from trading person-specific information.

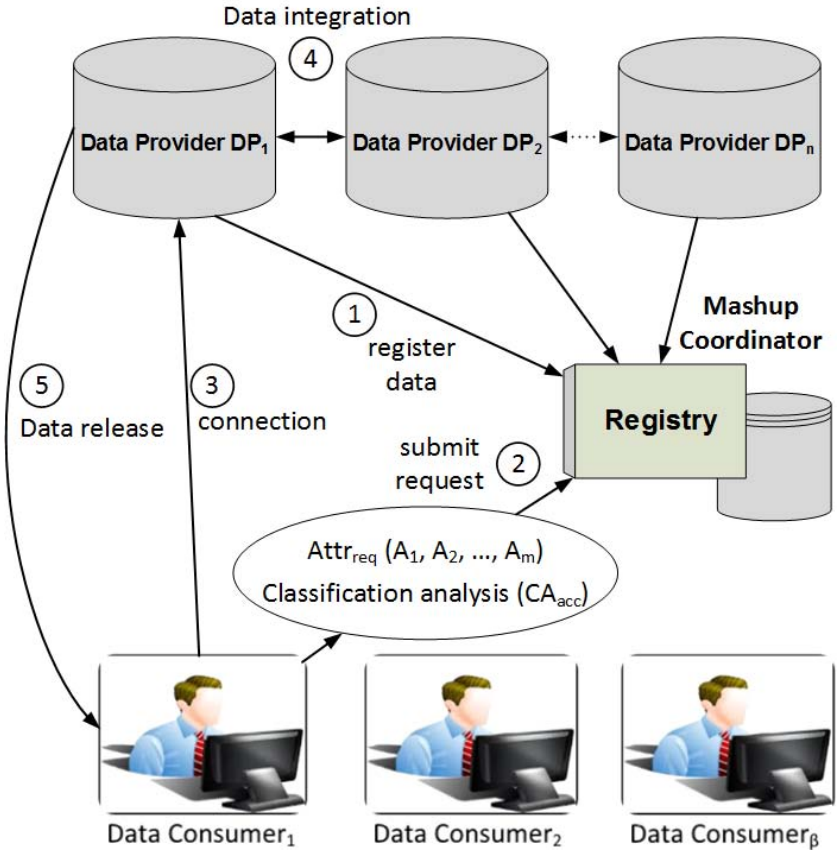


Figure 4.1: Privacy-preserving data mashup architecture for trading person-specific information

Our contributions are summarized as follows: The first three challenges, discussed in the previous paragraph, have already been widely studied in the literature [15, 19, 61, 62, 128, 147]. Here we focus on the fourth challenge that addresses the need of a privacy-preserving data mashup model for trading person-specific information in the e-market. We develop a business model that identifies the data consumers' (e.g., data recipients) requirements and performs the valuation on important parameters associated with revenue and costs for a business. Our business model is suitable for multiple data providers in making decisions where they have the following goals: (a) to find the sub-optimal value on the trade-off between data privacy and data utility and (b) to derive the contribution of each data provider in terms of monetary value. Finally, we show that our proposed approach can effectively achieve both goals through extensive experimental evaluations on real-life, person-specific data. The proposed model captures only the relevant factors that are crucial for cost-benefit analysis in our research problem. However, the model provides flexibility for users to include additional factors

based on the specific requirements of other scenarios.

The rest of the chapter is organized as follows: In Section 4.2, we explain the challenges faced by business enterprises, followed by the problem statement. In Section 4.3, we present our model as a privacy-preserving data mashup solution for e-markets. In Section 4.4, we discuss the limitations of our proposed model. In Section 4.5, we evaluate our proposed model based on the incorporated factors for multiple data providers by conducting extensive experiments on real-life data. Finally, we provide the summary in Section 4.6.

4.2 Challenges and problem statement

In this section, we explain the privacy challenges faced by business enterprises when integrating data from multiple sources, followed by the problem statement.

4.2.1 The challenges

The research problem is identified in [3], where the challenges are to integrate marketing data from multiple sources and to ensure the privacy of the customers. We generalize the problem as follows: Suppose two data providers, DP_1 and DP_2, own raw data tables D_1 and D_2, respectively. Each data provider owns a different set of attributes about the same set of records identified by the common Record IDs, such that DP_1 owns $D_1(RecID, Age, Job)$ and DP_2 owns $D_2(RecID, Sex, Education)$. The data providers want to integrate their data to improve the data utility for classification analysis in order to maximize their profit. The attributes in data tables D_1 and D_2 are classified into four categories for classification analysis: explicit identifier, quasi-identifier (QID), sensitive attribute, and class attribute. An explicit identifier attribute explicitly identifies a person, such as name, social security number (SSN), and account number. A QID attribute, such as date of birth, sex, and education, is a set of predictor attributes whose values are used to predict class attribute. A sensitive attribute, such as disease, salary, and marital status, contains an individual's sensitive information. A class attribute contains the class values for classification analysis. In the following example we discuss the privacy threats that can arise as a result of simply joining the raw data tables of data providers DP_1 and DP_2.

Table 4.1: Raw data table of data providers

	Data Provider DP_1		*Data Provider* DP_2		*Sens*	*Class*
RecID	**Age**	**Job**	**Sex**	**Education**	**Marital-status**	**Loan approval**
1	39	Painter	F	12th	Divorced	N
2	43	Doctor	M	Doctorate	Never-married	Y
3	37	Cleaner	F	12th	Divorced	Y
4	56	Cleaner	M	10th	Never-married	N
5	64	Welder	M	8th	Married-civ-spouse	Y
6	49	Doctor	F	Doctorate	Married-civ-spouse	Y
7	33	Lawyer	F	Masters	Never-married	Y
8	41	Lawyer	F	Doctorate	Married-civ-spouse	N
9	32	Painter	F	12th	Divorced	N
10	52	Cleaner	M	Bachelors	Divorced	Y
11	39	Cleaner	F	11th	Divorced	Y
12	61	Lawyer	M	Doctorate	Married-civ-spouse	Y
13	24	Technician	M	11th	Married-civ-spouse	N
14	44	Technician	F	Bachelors	Divorced	N
15	34	Lawyer	M	Masters	Never-married	Y
16	27	Painter	M	11th	Divorced	N
17	35	Cleaner	F	10th	Divorced	Y
18	41	Cleaner	M	11th	Divorced	Y
19	63	Welder	M	8th	Married-civ-spouse	N

Example 1. Consider a raw data table $D(RecID, A_1, \ldots, A_m, Sens, Class)$ of two data providers DP_1 and DP_2 in Table 4.1 for the predefined generalization hierarchy of the attributes illustrated in Figure 4.2. Both data providers want to release an integrated anonymized dataset D' to the data consumer for joint classification analysis. $RecID$, $Sens$, and $Class$ are shared between data providers DP_1 and DP_2. DP_1 and DP_2 own data tables $D_1(Age, Job)$ and $D_2(Sex, Education)$, respectively. Each record corresponds to the personal information for an individual person. A record in D has the form $\langle v_1, v_2, \ldots, v_m, s, cls\rangle$, where v_i is a value in A_i, s is a sensitive value in $Sens$, and cls is a class value in $Class$. The two data providers want to develop a data mashup service to integrate their data in order to perform classification analysis on the shared $Class$ attribute *Loan approval*, which has two values, Y and N, indicating whether or not the loan is approved.

In a *record linkage* attack [61], an adversary attempts to identify the record of a target victim in the released data table. Assume an adversary knows that the target victim is a female cleaner, denoted by $qid = \langle F, Cleaner\rangle$. The group of records matching qid is denoted by $D[qid]$. If the group size $|D[qid]|$ is small, the adversary may identify the victim's record and his/her sensitive value. The probability of a successful record linkage is $1/|D[qid]|$. In this example, $D[qid] = \{Rec\#3, 11, 17\}$.

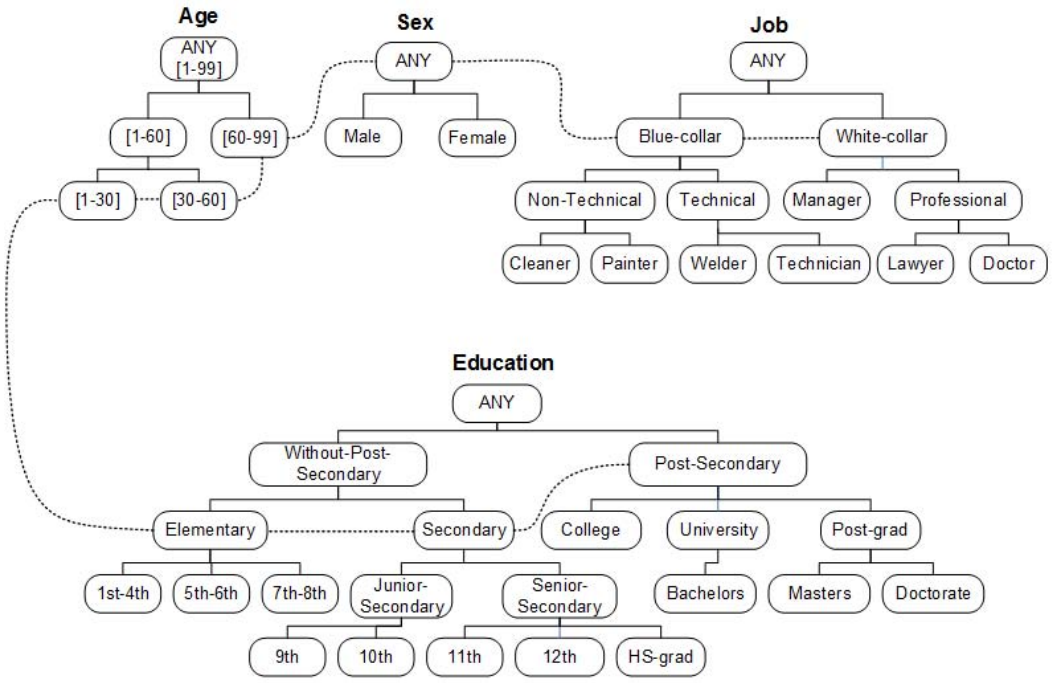


Figure 4.2: Taxonomy trees

In an *attribute linkage* attack [61], an adversary may not be able to accurately identify the record of a target victim but can infer a sensitive value with high confidence if it occurs frequently in the released table. With the prior knowledge qid about a target victim, an adversary can identify a group of records $D[qid]$ and can infer that the victim has sensitive value s with confidence $P(s|qid) = \frac{|D[qid \wedge s]|}{|D[qid]|}$, where $D[qid \wedge s]$ denotes the set of records matching both qid and s. $P(s|qid)$ is the percentage of the records in $D[qid]$ containing s. For example, given $qid = \langle M, Cleaner \rangle$, in Table 4.1, $D[qid \wedge Divorced] = \{Rec\#10, 18\}$, $D[qid] = \{Rec\#4, 10, 18\}$, and $P(Divorced|qid) = 2/3 = 66.67\%$. ■

Many privacy models, such as k-anonymity [147], ℓ-diversity [120], and t-closeness [112] have been proposed to protect against the aforementioned record and attribute linkage attacks in the relational raw data tables. k-*anonymity* prevents record linkage attacks by generalizing the records into equivalence groups of K size with respect to some QID attributes; however, it could suffer from an attribute linkage attack if the sensitive values are not diversified in an equivalence group. The principle of ℓ-*diversity* overcomes this problem by requiring every QID group to contain at least ℓ well-diversified values for the sensitive attribute. This model presents a stronger notion of

privacy against *homogeneity attacks* and *background knowledge attacks*. Mohammed et al. [123] propose a LKC-*privacy* model in which they assume that the adversary's background knowledge is bounded by *at most L* values of the QID attributes. This model provides better data utility in comparison to k-anonymity on high-dimensional data [60]. Dwork et al. [50] propose a *differential privacy* model that ensures the addition or removal of a single database record does not significantly affect the outcome of any computation over a database. It provides strong privacy guarantees to an individual independent of an adversary's background knowledge and computational power.

The aforementioned privacy models are discussed from the perspective of a single data custodian. Another challenge is related to the data mashup of multiple data custodians when consumer data requests cannot be fulfilled by a single data provider. The data mashup is a process over the cloud infrastructure that enables multiple data providers to integrate their data in order to fulfil the demands of data consumers. The cloud service provider may be one of the data providers or a third party, but the mashup scenario for the integration of data from multiple data custodians should not reveal person-specific information of the customers to unauthorized parties. The trust of a customer in an exchange of services with one data provider by sharing person-specific information does not necessarily extend trust to the other data providers. So, there is a need to avoid disclosure of sensitive information during the data mashup process and in the final release of mashup data. There are some known approaches that do not ensure privacy of an individual, such as (1) *mashup-then-generalize* and (2) *generalize-then-mashup*. The first approach integrates the raw data tables from two data providers and then generalizes using single table anonymization methods [59, 109]. This approach fails to preserve privacy because once the mashup coordinator or any other third party holds the integrated raw data it will instantly discover all the private information of both data providers. The second approach generalizes the data providers' tables individually using single-table anonymization methods, then integrates the generalized tables. This approach seems to preserve privacy locally at an individual data provider's end, but it does not guarantee the privacy when there is a quasi-identifier spanning multiple data providers' tables.

To address the above-mentioned privacy issues that arise from the data mashup when data is owned by multiple providers, Fung et al. [62] propose an extended version of the LKC-privacy

model specific to a multiple data providers scenario. The LKC-privacy model is suitable for high-dimensional data, as would normally be the case when integrating data from multiple data providers. This overcomes the problem of high-dimensionality when using k-anonymity. k-anonymity [147] is known to be a special case of LKC-privacy with adversary knowledge $L = |QID|$ and confidence $C = 100\%$, where $|QID|$ is the number of quasi-identifying attributes in the data table [123]. Mohammed et al. [128] have proposed a differentially private data release algorithm for multiple data providers in a distributed setup. Our model employs the approaches presented in [62] and [128] for data mashup of multiple data providers and sets the joint privacy requirements of contributing data providers in order to ensure that no extra information is leaked to any provider as a result of data integration.

4.2.2 Problem statement

Suppose data providers $DP_1, \ldots, DP_n$ own data tables $D_1, \ldots, D_n$, respectively. They want to generate an integrated anonymous dataset D' that fulfils the demands of data consumers and generates more profit in terms of monetary value for the data providers. Our proposed model enables the collaboration between data providers to set their joint privacy requirement for data mashup. It also benefits data providers by quantifying their costs and benefits in trading person-specific information and by determining the contribution of each data provider. Formally, the research problem is stated as follows.

Problem (Data mashup model for valuation of cost factors). Given multiple person-specific raw data tables $D_1, \ldots, D_n$ from data providers $DP_1, \ldots, DP_n$ and a set of requested attributes $Attr_{req}$ for classification analysis from a data consumer, the research problem is to develop a business model that performs the valuation on cost factors to find the sub-optimal value from the anonymized integrated data table D' under the joint privacy requirements of the data providers and to derive the contribution of each data provider $DP_1, \ldots, DP_n$ in terms of monetary value.

4.3 Proposed solution

In this section, we present a privacy-preserving solution for the business enterprises that seek to adopt an appropriate cloud-based data mashup model to manage the challenges of the e-market for trading person-specific information. Section 4.2.1 discusses the challenges of integrating data from multiple data providers, where each data provider owns a different set of attributes. We assume that every data provider intends to maximize the data utility, which in turn maximizes their profits, without violating the mutually agreed-upon privacy requirement. In this chapter, we focus on analyzing the problem of preventing the disclosure of sensitive information during the data mashup process and on the final release of mashup data. We employ anonymization algorithms, namely *Top-Down Specialization*(*TDS*) [62] and *Diff*erentially private anonymization based on *Gen*eralization (*DistDiffGen*) [128], for relational data mashup from multiple data providers. Our model quantifies the costs and benefits of privacy-preserving data publishing for the contributing data providers in terms of monetary value.

In our model, customers, data providers, and data consumers are the main stakeholders. For these stakeholders we identify the most relevant factors, as illustrated in Figure 4.3, to reflect the customers' requirements on data privacy, the data consumers' requirements on data utility, and the data providers' requirements on properly balancing privacy and utility with the goal of releasing the integrated data for profit. One of the limitations of our model is the lack of a standard method to monetize the value of personal data, especially when several parties are involved in collecting person-specific information from the same population. Currently, many companies actively collect personal information by providing monetary rewards to their customers or respondents. There is no standard price for a specific piece of personal information, but some market estimates are available in [4, 65]. It is also pointed out in [4] that there is no commonly accepted methodology for estimating the monetary value of personal data. Person-specific data contains sensitive and non-sensitive information. It is the utmost responsibility of data providers to take preventive measures when dealing with the sensitive information of individuals. Indeed, sensitive data is qualitative by nature. We set the sensitivity level of a dataset on the scale of 1-5 to indicate its significance for privacy protection. Another limitation of our model is the inconsistency of the expected cost of a lawsuit. The expected cost of a lawsuit

depends on the sensitivity of data and can be estimated from the historical cases of privacy breach. An individual may file a lawsuit against a data provider when his or her sensitive information is disclosed to a third party or made public without his or her consent. Although there is no fixed cost related to privacy breach cases, regulatory agencies such as the *Federal Trade Commission* (*FTC*) and the *Securities and Exchange Commission* (*SEC*) have imposed monetary fines and penalties subject to the nature of privacy breaches [144]. According to the revised *HITECH* penalty scheme [44], the penalty for a violation due to *reasonable cause* and not to *willful neglect* is between $\$1,000$ and $\$50,000$ for each violation.

Section 4.3.1 presents the business model for privacy-preserving data mashup. Section 4.3.2 discusses the key business factors for determining the value of integrated data and the factors that contribute to the potential damage cost. Section 4.3.3 discusses privacy-preserving data mashup algorithms. Section 4.3.4 discusses the implicit and explicit risk measures for privacy attacks.

4.3.1 Business model for privacy-preserving data mashup

Our proposed privacy-preserving data mashup business model allows the collaboration of multiple data providers to mashup their data over the cloud and to quantify the costs and benefits of releasing anonymized person-specific information in terms of *monetary value*. Figure 4.3 provides an overview of the proposed model; key factors are organized into three phases: before data mashup, during data mashup, and after data mashup. The left pane of the model depicts the factors held by each data provider, who registers its available data before the data mashup. For example, *Price per attribute*, *Number of attributes*, and *Size of dataset* are the factors that depend on the market value and consumer demand. Data providers can mutually set their key factors. These factors contribute to finding the *Price of a raw dataset* for every data provider. In the presented model, nodes represent different types of factors, and arrows indicate the influences or dependencies between different factors. For example, an arrow pointing from the *Baseline accuracy on raw dataset* to the *Total value of raw dataset* in the model indicates the influence of the *Baseline accuracy on raw dataset* on the *Total value of raw dataset*.

The objective of maximizing the profit can be achieved by balancing the two important factors: maximizing the *Value of integrated data*, and minimizing the *Potential damage cost*. The *Value*

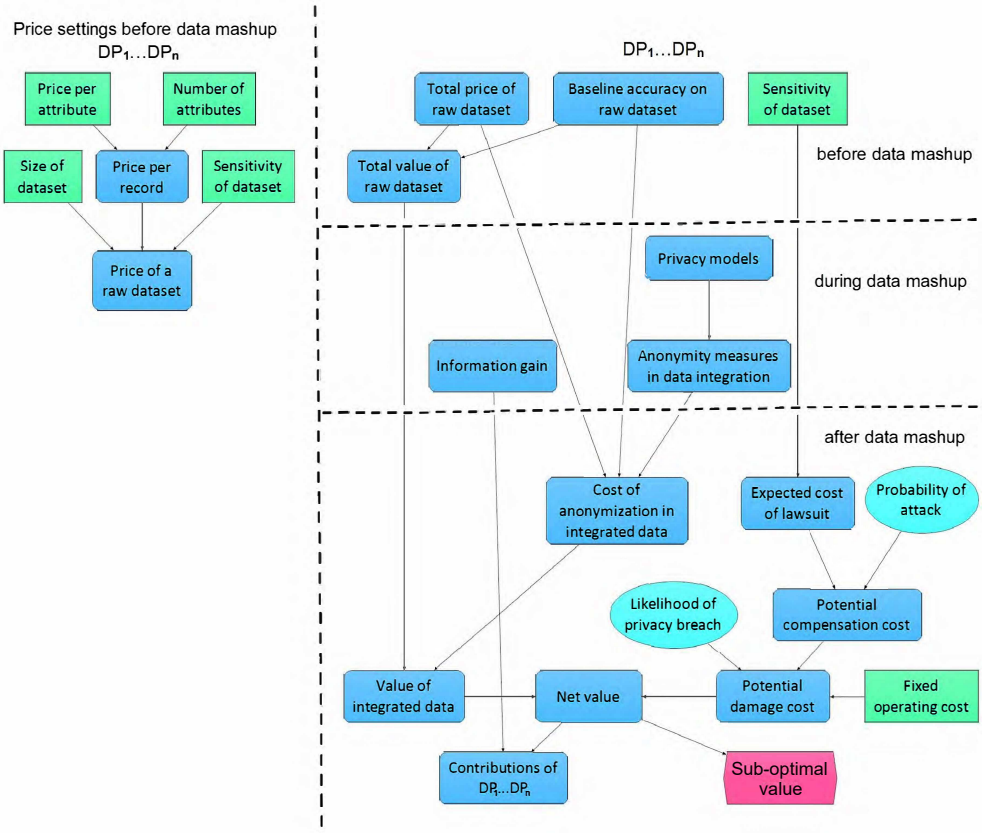


Figure 4.3: Business model for privacy-preserving data mashup

of integrated data depends upon the *Total value of raw dataset* and *Cost of anonymization in integrated data*. The *Cost of anonymization in integrated data* is computed on the data integration of contributing data providers with respect to the classification analysis (data mining) task. Each data provider can compare his or her benefits and costs *before* and *after* participation in the mashup process. For classification analysis, a data provider can estimate the cost of classification analysis on the anonymized data of his or her own data, and then on the integrated data. On the one hand, trading person-specific information has a high value in the e-market, but on the other hand, data providers who collaborate in sharing person-specific information need to be cautious of the risk of privacy breaches and cost of potential damages when integrating data. Our business model allows the participating data providers to: (1) set up their joint privacy requirements during data mashup by choosing the privacy model along with the anonymization algorithm and privacy parameters, and (2) analyze the impact of anonymization on information utility for classification requirement in terms of

monetary value after data mashup. The aforementioned business factors can help the data providers in defining the overall objective of maximizing *Net value*. Furthermore, in the data mashup process the contribution of each data provider is derived fairly from the achieved *Net value* by computing the information gain on the anonymized data. Accordingly, the data provider whose data provides a larger information gain for classification can get a larger share of the monetary net value.

Companies that face similar challenges, and whose business models are primarily based on sharing person-specific information, can be our potential audiences. There are quite a few companies to whom our research problem can be generalized. Some of them are Acxiom, AdAdvisor, AnalyticsIQ, BlueKai, comScore, Datacratic, Dataline, eXelate, Lotame, etc. that aggregate information from various sources for a variety of purposes [6].

4.3.2 Key factors for business model

The selection and valuation of key factors are crucial in developing the cost-benefit business model. We learn and identify key factors from different sources [4, 74]. These factors are broadly classified into two categories: factors that contribute to the *Value of integrated data* and factors that associates with the *Potential damage cost*. We further divide the factors by phases in the data mash up process: before the start of the data mashup process, during the data mashup process, and after the data mashup process.

4.3.2.1 Before data mashup

In this subsection we discuss the factors that are considered as essential prior to performing the cost-benefit analysis. The data providers can set up the market prices on their available data [65] (e.g., set of attributes) before the data mashup process. Let us assume there are n data providers $DP_1, \ldots, DP_n$, and DP_i denotes the identity of the data provider.

4.3.2.1.1 Price per attribute

The price per attribute $Price_{attr_i}$ of a data provider DP_i represents the cost of collecting one successful questionnaire for an attribute. Each DP_i can set a price on their data attributes based upon prior knowledge of market pricing by competing data providers [4]. There is no definite price for

personal identifying attributes, such as name, address, email, birthdate, phone number, etc. But the values can be inferred from cases where personal identity is being sold at a low price, as highlighted in the literature [65]. In our empirical study, we assume the monetary value for $Price_{attr_i}$.

4.3.2.1.2 Number of attributes

The attribute count $Count_{attr_i}$ of a data provider DP_i represents the number of attributes in a single record. Each DP_i owns a different set of attributes.

4.3.2.1.3 Price per record

The price per record $Price_{rec_i}$ of a data provider DP_i represents the unit price of a record. Naturally, it is the product of the price per attribute $Price_{attr_i}$ and the attribute count $Count_{attr_i}$ in a single record. That is,

$$Price_{rec_i} = Price_{attr_i} \times Count_{attr_i} \tag{12}$$

The price of a raw dataset of the data provider DP_i increases as the unit *price per record* increases.

4.3.2.1.4 Size of dataset

The size of a dataset $Size_{ds_i}$ represents the total number of records in the DP_i dataset. Each record has an associated price. As the number of records increases, the overall pricing of a raw dataset also increases.

4.3.2.1.5 Sensitivity of dataset

The sensitivity of a dataset Sen_{ds_i} indicates that a dataset contains sensitive or personally significant information. It is a given qualitative factor and every data provider should consider this factor for privacy risk assessment. The sensitivity level signifies the importance of data privacy for each data provider DP_i. Intuitively, a higher sensitivity level implies a higher price of a raw dataset and a higher impact on the lawsuit and compensation cost.

4.3.2.1.6 Price of a raw dataset

The price of a raw dataset $Price_{rd_i}$ represents the data provider DP_i's selling price of a raw dataset in the e-market. It is the product of the sensitivity of the dataset Sen_{ds_i}, the size of the dataset $Size_{ds_i}$, and the price per record $Price_{rec_i}$, which is formulated as follows:

$$Price_{rd_i} = Sen_{ds_i} \times Size_{ds_i} \times Price_{rec_i} \tag{13}$$

4.3.2.1.7 Total price of raw dataset

The total price of the raw dataset $TPrice_{rd}$ is the sum of the prices of all contributing data providers' raw datasets, which is formulated as follows:

$$TPrice_{rd} = \sum_{i=1}^{n} Price_{rd_i} \tag{14}$$

4.3.2.1.8 Baseline accuracy on raw dataset

Baseline accuracy on raw dataset BA is determined by considering the classification task as the utility function to evaluate the information utility on the raw datasets of contributing data providers. Data providers can compute BA using the secure multiple party classifier [49] without sharing their raw data.

4.3.2.1.9 Total value of raw dataset

The total value of the raw dataset $TValue_{rd}$ represents the monetary value of a raw dataset that the data providers derive from the information utility. It is the product of the total price of the raw dataset $TPrice_{rd}$ and the baseline accuracy of the raw dataset BA, which is formulated as follows:

$$TValue_{rd} = TPrice_{rd} \times BA \tag{15}$$

4.3.2.2 During data mashup

In this subsection, we discuss the factors involved in the data mashup process.

4.3.2.2.1 Privacy models

The participating data providers DP_n can mutually choose the privacy model (refer to Section 2.1 for details), namely k-anonymity, LKC-privacy, and ϵ-differential privacy, prior to integrating their data.

4.3.2.2.2 Anonymity measures in data integration

The participating data providers DP_n can jointly set up the data mashup anonymization algorithm (refer to Section 4.3.3 for details), such as multi-party *TDS* (Algorithm 1) and *DistDiffGen* (Algorithm 2), along with the anonymity thresholds, such as K, L, C, for k-anonymity and LKC-privacy models and ϵ, and h for a ϵ-differential privacy model.

4.3.2.2.3 Information gain

The information gain is employed to determine the usefulness of classification. It computes the reduction of entropy by specializing node v into $c \in child(v)$ as discussed in Section 2.2. Each data provider owns a different set of attributes in the same set of records. Each data provider DP_i computes the information gain or $IGScore(x)$ locally for each candidate and picks the candidate x with the highest value of $IGScore(x)$. Then each data provider DP_i communicates $IGScore(x)$ with the n collaborating data providers for determining the global winner w. The winner w data provider performs specialization $w \in child(w)$ on its own copy locally. The winner w data provider then instructs other n collaborating data providers how to perform specialization (further explanation of this process can be seen in Section 4.3.3). This process is iterative and it runs until no candidate is left in the mark. The information gain $IGScore(x)$ of winner candidate w data provider accumulates under the relevant winner w data provider.

4.3.2.3 After data mashup

In this subsection we discuss the factors that are applied after the data mashup process. These factors help in determining the sub-optimal value and the contribution of each data provider.

4.3.2.3.1 Cost of anonymization in integrated data

To determine the cost of anonymization in integrated data $Cost_{intgdata}$, we make use of the difference between baseline accuracy (BA) and classification accuracy (CA). BA measures the accuracy of classification analysis on raw data while CA measures the accuracy on anonymized integrated data. Therefore, $Cost_{intgdata}$ becomes:

$$Cost_{intgdata} = TPrice_{rd} \times (BA - CA) \tag{16}$$

4.3.2.3.2 Value of integrated data

The value of integrated data $Val_{intgdata}$ is the difference between the total value of raw dataset $TValue_{rd}$ and the cost of anonymization in integrated data $Cost_{intgdata}$. It is the benefit that the data providers can earn from the information utility of classification analysis by trading their integrated data. Formally, $Val_{intgdata}$ is defined as:

$$Val_{intgdata} = TValue_{rd} - Cost_{intgdata} \tag{17}$$

4.3.2.3.3 Probability of attack

The probability of attack $Prob_{atk}$ is employed to determine the implicit weaknesses in privacy protection methods. The data providers can prevent an adversary's attempt to assess the probability of occurrence of a sensitive attribute value in the anonymized integrated dataset using precision and recall measures (refer to Section 4.3.4 for details). The probability of occurrence changes with respect to the chosen privacy model and its level of privacy protection. $Prob_{atk}$ is calculated using F-measure on the sensitive attribute value Sen_{val}. F-measure is a weighted harmonic mean of precision and recall. Formally, $Prob_{atk}$ is defined as:

$$Prob_{atk} = \frac{2 \times (\textit{Precision on } Sen_{val} \times \textit{Recall on } Sen_{val})}{\textit{Precision on } Sen_{val} + \textit{Recall on } Sen_{val}} \tag{18}$$

4.3.2.3.4 Expected cost of lawsuit

The expected cost of lawsuit $Ecost_{lwst}$ is enforced subject to the nature of a privacy breach and the sensitivity of data. It increases as the level of data sensitivity increases. $Ecost_{lwst}$ enables business enterprises to predict the potential cost of privacy breach incident. The monetary cost can be estimated based on the historical trends of privacy breach incidents. The Federal Trade Commission Act (FTCA), Gramm-Leach-Bliley Act (GLBA), Fair Credit Reporting Act (FCRA), and Personal Data Privacy and Security Act regulate the collection, use, and protection of personal information and impose monetary fines and penalties subject to the nature of the data breach [2, 5].

The lawsuit cost is not fixed and it varies with the applied anonymity measures on data mashup. For instance, an adversary may exploit the inherent weakness of the privacy protection method to infer sensitive information about a victim by using the precision and recall measures in the equation of the probability of attack.

4.3.2.3.5 Likelihood of privacy breach

The likelihood of a privacy breach L_{pb} measures an adversary's prowess in inferring the victim's sensitive value. This inference is measured using an attack model (refer to the Section 4.3.4 for details) by exploiting the background knowledge about a victim. We assume that the victim's record is in the integrated published dataset and the adversary knows the victim's QID. Formally, L_{pb} is defined as:

$$L_{pb} = \frac{\textit{Total records count on } Sen_{val}}{\textit{Total records count on class label } Sen_{attr}} \tag{19}$$

where Sen_{val} denotes the value of the sensitive attribute and Sen_{attr} denotes the sensitive attribute in the integrated dataset.

4.3.2.3.6 Potential compensation cost

The potential compensation cost PCC is a factor that can help data providers to determine the approximate cost of compensation prior to sharing the anonymized integrated dataset. It is impacted by the enforcement of privacy policies and privacy protection methods. The potential compensation cost would vary with the risk of sensitive information disclosure of a privacy attack. In general, more

stringent privacy parameters impede the probability of a privacy attack. It is our rational hypothesis that privacy attacks would have an exponential impact on the compensation cost due to the substantial increase in the cost of litigation processes [1]. There is no fixed monetary value for compensation cost in [1], but in the e-market a customer who suffers monetary loss due to the disclosure of his or her sensitive information may claim against data providers (e.g., business enterprises) for compensation. Formally, PCC is defined as:

$$PCC = \exp(Prob_{atk}) \times Ecost_{lwst} \tag{20}$$

4.3.2.3.7 Fixed operating cost

The fixed operating cost F_{OpCost} indicates the fixed monthly cost that business enterprises would have to pay when adopting cloud-services for data integration. Business enterprises would gain more benefits with the adoption of cloud-services comparative to expenditures incurred on hardware and software purchase, setup and installation, licensing and upgrades, maintenance and support, power and utility, and allocation of physical space. F_{OpCost} is a quantitative factor, and its value is independent of the employed anonymity measures in the process of data mashup. It remains the same regardless of the changes in value of integrated data $Val_{intgdata}$.

4.3.2.3.8 Potential damage cost

The potential damage cost *PDC* indicates the cost that the data providers would suffer from data privacy breaches. An adversary may attempt to infer sensitive information about a victim from the anonymized integrated dataset by using an explicit form of a privacy attack as discussed in Section 4.3.2.3.5. In case of a privacy breach, business enterprises (e.g., data providers) would face substantial costs because of the mandatory notification of data breach, handling of regulatory investigations, hiring of external auditors, possibility of class action litigation, and loss of business goodwill and customer relationships [30]. As suggested by existing studies [9, 22, 71], data breaches negatively impact business profitability. We postulate that the likelihood of a privacy breach would have an exponential impact on the potential damage cost because a plaintiff (e.g., customer) seeks redress for alleged harms such as actual monetary loss from the identity theft, emotional distress,

sexual harassment, discrimination, or possible future losses [145]. *PDC* is determined by the likelihood of a privacy breach L_{pb}, the potential compensation cost *PCC*, and the fixed operating cost F_{OpCost}. Formally, PDC is defined as:

$$PDC = \exp(L_{pb}) \times PCC + F_{OpCost} \tag{21}$$

4.3.2.3.9 Net value

The net value NV demonstrates due diligence in evaluating the key business factors on the trade-off between privacy and information utility. It is employed to quantify the difference between the value of integrated data and the potential damage cost on the applied anonymity measures in the mashup process. The net value changes with respect to the chosen privacy model along with the anonymization algorithm and privacy parameters. Formally, NV is calculated as follows.

$$NV = Val_{intgdata} - PDC \tag{22}$$

4.3.2.3.10 Sub-optimal value

The sub-optimal value $Subopt_{val}$ is achieved at the maximum of the net value NV. It changes with the variations of price settings and joint privacy requirements of data providers. NV is realized by the difference between the value of integrated data and the potential damage cost. Formally, $Subopt_{val}$ is defined as:

$$Subopt_{val} = \max(NV) \tag{23}$$

4.3.2.3.11 Contributions of data providers

The contribution of each data provider DP_i is derived from the net value NV by fairly computing first the accumulative information gain of each data provider, denoted by $\widetilde{\mathcal{G}}_{DP_i}$, on the anonymized integrated dataset. Generally, the data provider whose data attributes result in greater information gain can get a proportionally higher share of the monetary net value. Formally, $Cont_{DP_i}$ is defined

as:

$$Cont_{DP_i} = \frac{\widetilde{\mathcal{G}}_{DP_i}}{\sum_{i=1}^{n} \widetilde{\mathcal{G}}_{DP_i}} \times NV \tag{24}$$

4.3.3 Data mashup algorithms

In this section, we discuss our extension on the state-of-the-art anonymization algorithms for data mashup in a multiple data-providers scenario: *Top-Down Specialization*(*TDS*) [62] and *Diff*erentially private anonymization based on *Gen*eralization (*DistDiffGen*) [128].

4.3.3.1 Top-down specialization algorithm for multiple data providers

Algorithm 1 presents an overview of the *Top-Down Specialization* (*TDS*) algorithm to integrate data in a scenario of multiple data providers, which is an extension of Fung et al. [62].

Consider multiple data providers DP_1,...,DP_n, who own private data tables D_1,...,D_n having a common record identifier $RecID$. Initially, every data provider generalizes all of its own attribute values to the topmost value according to the taxonomy trees, as illustrated in Figure 4.2, and maintains a mark $Mark_i$ that contains the topmost value for each attribute A_i in QID. A *taxonomy tree* is specified for each categorical attribute in QID. A leaf node represents a precise value and a parent node represents a generic value. For continuous attributes in QID, taxonomy trees can be grown at runtime, where each node represents an interval, and each non-leaf node has two child nodes representing some optimal binary split of the parent interval [137]. The $\cup Mark_i$ on all attributes represents a generalized table D, denoted by D_g. $\cup Mark_i$ also contains the set of candidates for specialization. A specialization $v \rightarrow child(v)$ is *valid*, written as $IsValid(v)$, if the generalized table D_g still satisfies the privacy requirements stated in Definitions 2.1.1 and 2.1.2 *after* the specialization on v. At each iteration, the TDS multiple data providers mashup (TDSmdpm) algorithm identifies the winner candidate by communicating the $IGScore$ with all the participating data providers (Lines 4-5). The valid candidate that has the highest $IGScore$, among all the candidates, performs the winner specialization and the information gain, denoted by $\widetilde{\mathcal{G}}_{DP_i}$, accumulates $IGScore(x)$ on winner's attribute specializations (Lines 7-13) and updates the $IGScore$ and the $IsValid$ status of the new and existing candidates in the mark (Line 16). The contribution of each data provider is

Algorithm 1 TDS multiple providers data mashup

Initialize every record values in D to the topmost generalized values D_g.
Initialize $\cup Mark_i$ to include only topmost values and update $IsValid(v)$ for every $v \in \cup Mark_i$;
while $\exists v \in \cup Mark_i$ s.t. $IsValid(v)$ **do**
  Find the local winner candidate x of DP_i that has the highest $IGScore(x)$;
  Communicate $IGScore(x)$ with all the other participating data providers to determine the global winner w;
  if the winner w is local **then**
    Specialize w on D_g;
    Instruct all the other data providers to specialize w;
    $\widetilde{\mathcal{G}}_{DP_i} = \widetilde{\mathcal{G}}_{DP_i} + IGScore(x)$;
  else
    Wait for the instruction from the winner data provider;
    Specialize w on D_g using the instruction;
    $\widetilde{\mathcal{G}}_{DP_j} = \widetilde{\mathcal{G}}_{DP_j} + IGScore(x)$;
  end if
  Replace w with $child(w)$ in the local copy of $\cup Mark_i$;
  Update $IGScore(x)$ and $IsValid(x)$ for every candidate $x \in \cup Mark_i$;
end while
Compute the contribution of each data provider according to Eq.(24);
return D_g and $\cup Mark_i$;

computed according to Eq.(24). TDSmdpm terminates when there are no valid candidates in the mark.

Suppose that winner candidate w is local to data provider DP_1 that performs $w \rightarrow child(w)$ on its copy of $\cup Mark_i$ and D_g. This means specializing each record $r \in D_g$ containing w into $r'_1, \ldots, r'_z$; the child values are in $child(w)$. Similarly, all the other data providers $DP_2, \ldots, DP_n$ update their $\cup Mark_i$ and D_g and partition $D_2[r]$ into $D_2[r'_1], \ldots, D_2[r'_z] \ldots D_n[r]$ into $D_n[r'_1], \ldots, D_n[r'_z]$. Since all the other participating data providers do not have w, DP_1 needs to instruct $DP_2, \ldots, DP_n$ on how to partition their records in terms of $RecID$s.

4.3.3.2 DistDiffGen anonymization algorithm for multiple data providers

Algorithm 2 provides an extension of the two-party *Diff*erentially private anonymization based on *Gen*eralization [128] to differentially integrate multiple private data tables D_1,...,D_n sharing a common identifier $RecID$, which is owned by data providers DP_1,...,DP_n for classification analysis. However, the distributed exponential mechanism is limited to two parties. *DistDiffGen* [128] is an

extension of the TDS algorithm [59] to achieve ϵ-differential privacy. The two major extensions over the TDS algorithm include: (1) *DistDiffGen* selects the $Best$ specialization based on the exponential mechanism, and (2) *DistDiffGen* perturbs the generalized contingency table by adding the Laplacian noise to the qid counts. The Laplacian noise is calibrated based on the *sensitivity* of a utility function, which quantifies the maximal impact of adding or deleting a single record on a function. This algorithm provides secure data integration of two parties under the definition of the semi-honest adversary model.

Initially, all values in the predictor attributes $\mathcal{A}^{pr}$ (i.e., attributes used to predict the class attribute) of each data provider are generalized to the topmost value in their taxonomy trees (Line 1), as illustrated in Figure 4.2, and $Mark_i$ contains the topmost value for each attribute A_i^{pr} (Line 2). The predictor attribute $\mathcal{A}^{pr}$ can be either categorical or numerical, but the class attribute is required to be categorical. The value of a categorical attribute is denoted by v_c, whereas the value of a numerical attribute is denoted by v_d. Each data provider keeps a copy of the $\cup Mark_i$ and a generalized data table D_g. The algorithm first determines the split points for all numerical candidates $v_d \in \cup Mark_i$ by using the exponential mechanism (Line 4), then computes the scores for all candidates $v \in \cup Mark_i$ (Line 5). At each iteration the algorithm uses the secure distributed exponential mechanism (DistExp) as presented in [128] (readers may refer to the details of DistExp algorithm) to select a winner candidate $w \in \cup Mark_i$ for specialization (Line 7). Different utility functions (e.g., information gain) can be used to calculate the score. If the winner candidate w is local to DP_i, DP_i specializes w on D_g by splitting its records into child partitions, updates its local copy of $\cup Mark_i$, and instructs all the other participating data providers to specialize and update their local copy of $\cup Mark_i$ (Line 8-11). The information gain, denoted by $\widetilde{\mathcal{G}}_{DP_i}$, accumulates $IGScore(x)$ on winner's attribute specializations (Line 12). DP_i further calculates the scores of the new candidates as a result of the specialization (Line 14). If the winner w is not one of DP_i's candidates, DP_i waits for instructions from the other winner data provider to specialize w and to update its local copy of $\cup Mark_i$ (Lines 16 and 17). This process is iterated until the specified number of the specializations h is reached. The contribution of each data provider is computed according to Eq.(24). Finally, the algorithm perturbs the output by adding the noisy count at each leaf node (Line 22) using the Laplace mechanism.

Algorithm 2 DistDiffGen for multiple data providers

Initialize D_g with one record containing topmost generalized values;
Initialize $Mark_i$ to include the topmost value;
$\epsilon' \leftarrow \frac{\epsilon}{2(|A_n^{pr}|+2h)}$;
Determine the split value for each $v_d \in \cup Mark_i$ with probability $\propto \exp(\frac{\epsilon'}{2\Delta \mathrm{u}}\mathrm{u}(D, v_d))$;
Compute the $IGScore$ for $\forall v \in \cup Mark_i$;
for $iter$ = 1 to h **do**
    Determine the winner candidate w by using the DistExp Algorithm [128];
    if w is local **then**
        Specialize w on D_g;
        Replace w with $child(w)$ in the local copy of $\cup Mark_i$;
        Instruct all the other participating data providers to specialize and update $\cup Mark_i$;
        $\widetilde{\mathcal{G}}_{DP_i} = \widetilde{\mathcal{G}}_{DP_i} + IGScore(x)$;
        Determine the split value for each new $v_d \in \cup Mark_i$ with probability $\propto \exp(\frac{\epsilon'}{2\Delta \mathrm{u}}\mathrm{u}(D, v_d))$;
        Compute the $IGScore$ for each new $v \in \cup Mark_i$;
    else
        Wait for the instruction from the winner data provider;
        Specialize w and update $\cup Mark_i$ using the instruction;
        $\widetilde{\mathcal{G}}_{DP_j} = \widetilde{\mathcal{G}}_{DP_j} + IGScore(x)$;
    end if
end for
Compute the contribution of each data provider according to Eq.(24);
return each leaf node with count $(CT + \mathtt{Lap}(2/\epsilon))$

4.3.4 Risk measurement

In this section, we present an attack model to measure the risk associated with implicit weaknesses of privacy protection methods and the risk caused by explicit knowledge attack.

4.3.4.1 Attack model

Data providers participating in data integration express concern on two types of privacy threats: identity linkage and attribute linkage. Based on background knowledge, adversaries in identity linkage attacks can uniquely identify an individual, whereas adversaries in attribute linkage attacks can infer an individual's sensitive information with relatively high confidence. In this chapter, we employ classification analysis to quantify the potential privacy risks. Specifically, we build a C4.5 classifier by using the sensitive attribute as the class attribute, and we quantify the privacy risks by measuring the accuracy of predicting the sensitive values. There are many types of classification

Table 4.2: Anonymous integrated data ($L = 2$, $K = 2$, $C = 0.5$)

	Data Provider DP_1		*Data Provider* DP_2		*Sensitive*	*Class*
RecID	**Age**	**Job**	**Sex**	**Education**	**Marital-status**	**Loan approval**
1	$[39-99]$	Blue-collar	Any	Secondary	Divorced	N
2	$[39-99]$	White-collar	Any	Post-secondary	Never-married	Y
3	$[33-39]$	Blue-collar	Any	Secondary	Divorced	Y
4	$[39-99]$	Blue-collar	Any	Secondary	Never-married	N
5	$[39-99]$	Blue-collar	Any	Elementary	Married-civ-spouse	Y
6	$[39-99]$	White-collar	Any	Post-secondary	Married-civ-spouse	Y
7	$[33-39]$	White-collar	Any	Post-secondary	Never-married	Y
8	$[39-99]$	White-collar	Any	Post-secondary	Married-civ-spouse	N
9	$[1-33]$	Blue-collar	Any	Secondary	Divorced	N
10	$[39-99]$	Blue-collar	Any	Post-secondary	Divorced	Y
11	$[39-99]$	Blue-collar	Any	Secondary	Divorced	Y
12	$[39-99]$	White-collar	Any	Post-secondary	Married-civ-spouse	Y
13	$[1-33]$	Blue-collar	Any	Secondary	Married-civ-spouse	N
14	$[39-99]$	Blue-collar	Any	Post-secondary	Divorced	N
15	$[33-39]$	White-collar	Any	Post-secondary	Never-married	Y
16	$[1-33]$	Blue-collar	Any	Secondary	Divorced	N
17	$[33-39]$	Blue-collar	Any	Secondary	Divorced	Y
18	$[39-99]$	Blue-collar	Any	Secondary	Divorced	Y
19	$[39-99]$	Blue-collar	Any	Elementary	Married-civ-spouse	N

models, such as naive Bayesian, support vector machines, and so forth, that an adversary can employ to make predictions. Our proposed framework is flexible to adopt other classification methods to quantify the potential privacy risks.

Let D be the raw data, as shown in Table 4.1, and D' be the anonymous integrated data from the mashup process of two data providers, as shown in Table 4.2. Recall that $Marital\text{-}status$ is the sensitive attribute and $Loan\ approval$ is the class attribute. Let us assume that the data providers release their anonymized integrated data table D' to the data consumer (i.e., data recipient) with the classifier. A data recipient (or an adversary) can employ the C4.5 classification algorithm to infer sensitive records of individuals by setting the sensitive attribute $Divorced$ as the class label. This approach is similar to [98] in a way that a data recipient (or an adversary), instead of inferring new records on a class label, can predict the sensitive attribute value of a target victim who is a participant in the anonymized integrated training data.

Table 4.3: Confusion matrix

		Predicted class		
		A	**B**	**C**
Actual class	Divorced **(A)**	4	0	0
	Married-civ-spouse **(B)**	1	0	0
	Never-married **(C)**	0	1	0

4.3.4.1.1 Implicit risk measure

Implicit risk is due to attribute linkage attack [61]: an adversary attempts to infer the sensitive attribute value in the released dataset using a C4.5 classifier. In this type of attack, an adversary can negatively use the precision and recall performance measures to identify a victim's sensitive value. *Precision* indicates the measure of exactness or quality, meaning the number of correctly classified positive elements divided by the total number of elements classified as positive. *Recall* indicates the measure of completeness or quantity, which means the number of correctly classified positive elements divided by the total number of actual positive elements. We measure the adversary's power of inferring sensitive values by calculating the F-measure according to Eq. (18), which is a weighted harmonic mean of precision and recall measures. F-measure represents the probability of attack $Prob_{atk}$. An adversary may use these performance measures to determine the success rate of a privacy attack. We elaborate this by the following example.

Example 2. Consider the anonymous integrated data D' in Table 4.2. Suppose an adversary sets the sensitive attribute *Marital-status* as a class on D'. This results in a new integrated data table T^*. The adversary performs the attack by using the classification model C4.5 on T^* to infer the sensitive attribute value of the victim. Table 4.3 shows the confusion matrix for the classification of three classes. Each instance (e.g., an individual) has an actual class and a predicted class. The rows represent actual classes of the raw records, and the columns represent predictions made by the model. The entries on the diagonal indicate the correct predictions; other entries show the errors. For the sensitive value $Divorced$, true positive $TP = 4$, false negative $FN = 0$, and false positive $FP = 1$. So, the values of performance measures are $Precision = 80\%$, $Recall = 100\%$, and $F\text{-}measure = 88.8\%$. ■

4.3.4.1.2 Explicit risk measure

Explicit risk is due to record linkage attack [61]: an adversary applies his or her background knowledge on the integrated data table T^* to predict the sensitive value of a victim who is part of the anonymized integrated training data. In addition, we assume that an adversary knows that a victim has a record on the table and also has some knowledge about the victim. For example, an adversary knows that the victim is female, age is greater than 35, education level is secondary, and job is cleaning. By applying this external knowledge to the anonymized integrated training data, the adversary finds a total of 3 records on the sensitive value $Divorced$ under the class attribute $Marital\text{-}status$. The likelihood of the privacy breach L_{pb} for this case becomes $3/4$, which is calculated according to Eq. (19). This implies that the adversary has a 75% confidence of inferring the sensitive value of the victim. The likelihood of a privacy breach would increase if the data providers are semihonest [117, 184].

4.4 Limitations

In this section, we discuss some of the limitations of our proposed business model that are inherent problems related to the cost-benefit analysis. Our model provides the basic framework for analyzing the cost-benefit of data mashup. The data providers can add, remove, or adjust the cost factors according to their specific applications and scenarios. The common sources of errors are *omission errors* and *valuation errors*. *Omission error* refers to excluding relevant factors in the process of factor analysis. *Valuation error* refers to making an incorrect estimation of the value of the cost factors, especially in the presence of intangible assets such as person-specific information. These errors do not undermine the value of cost-benefit analysis, and they are expected to decline with the passage of time by the increase in domain knowledge and follow-up of ex-post analysis [31].

The privacy protection, database, and data mining communities have identified many types of potential privacy attacks, such as record linkage attack, attribute linkage attack, table linkage attack, and probabilistic attack. Consequently, many privacy models and anonymization methods [61], such as *MinGen*, *K-Optimize*, *Bottom-Up Generalization*, *Top-Down Specialization*, *Anatomy*, and *ϵ-Differential Additive Noise*, have been proposed to thwart these attacks. The objective of this

chapter is *not* to address all these privacy attacks. Instead, we are presenting a framework with a flexible cost-benefit business model for multiple data providers to achieve sub-mutual benefits given an agreed privacy requirement. Any partition-based anonymization methods that result in equivalent classes with counts are applicable to our framework. To illustrate the effectiveness of our proposed framework and model, in our discussion we adopt two anonymization algorithms, namely *TDSmdpm* and *DistDiffGen*, that can anonymize vertically-partitioned relational data. *TDSmdpm* and *DistDiffGen* were chosen because they can achieve two commonly employed privacy models, LKC-privacy and ϵ-differential privacy, respectively. We would like to emphasize that our model is not limited to these privacy models and anonymization algorithms. They can be replaced, depending on the consent of privacy protection among the data providers. The negotiation process for reaching the consent is beyond the scope of this chapter.

4.5 Empirical study

In this section, we analyze and compare the costs and benefits for each data provider before participation in the data mashup process on their own data and after participation in the data mashup process on the integrated data. We evaluate our business model with the assumption of having 3 data providers who mashup their data using a secure Privacy-preserving High-dimensional Data Mashup (PHDMashup) algorithm [62] in a cloud environment. This model is independent of the cloud platform.

We employ a real-life dataset *Adult*[1] in our experiments, which has been widely used for many empirical studies. It is also known as the *de facto* benchmark for comparing the performance of anonymization algorithms [59, 75, 125]. After removal of records with missing values, the *Adult* dataset contains $45,222$ records with 8 categorical attributes, 6 numerical attributes, and a binary class attribute $Income$ with two levels, $\leq 50K$ or $> 50K$. For a classification analysis task this dataset is split into $30,162$, and $15,060$ records for the training and testing set, respectively. We vertically partition the *Adult* dataset into three partitions P_1, P_2, and P_3 for data providers DP_1, DP_2, and DP_3, respectively. Table 4.4 represents the attributes with their types of each data provider.

[1] Available at: http://archive.ics.uci.edu/ml/datasets/Adult

Table 4.4: Attributes hosted by each data provider

Data Provider DP_1	
Attribute	***Type***
Age	numerical
Hours-per-week	numerical
Workclass	categorical
Capital-gain	numerical
Income	categorical
Marital-status	categorical

Data Provider DP_2	
Attribute	***Type***
Education	categorical
Education-num	numerical
Occupation	categorical
Capital-loss	numerical
Income	categorical
Marital-status	categorical

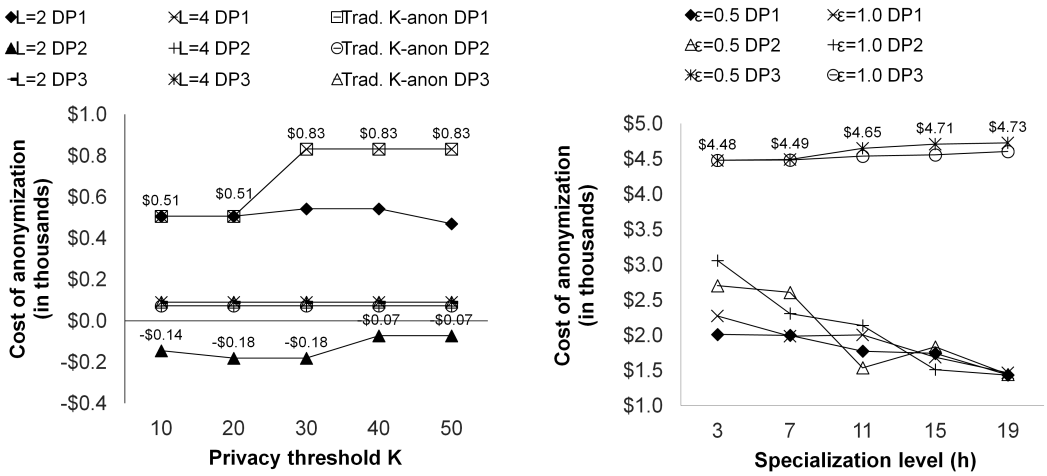
Data Provider DP_3	
Attribute	***Type***
Sex	categorical
Race	categorical
Relationship	categorical
Final-weight	numerical
Native-country	categorical
Income	categorical
Marital-status	categorical

Each data provider computes *Baseline Accuracy (BA)* and *Classification Accuracy (CA)* on its raw dataset and anonymized dataset, respectively, by using a C4.5 classifier. The *BA* is 81.8%, 82.5%, and 75.6% on DP_1, DP_2, and DP_3 datasets, respectively. Whereas, the baseline accuracy (BA) on the integrated data is 85.3% using the secure multiple party classifier [49] without sharing their raw data. We consider $Income$ as the class attribute and *Marital-status* as the sensitive attribute in each data provider's table. The remaining attributes in each data provider's table are the QID attributes. We consider *Married-civ-spouse* and *Divorced* in the attribute *Marital-status* as sensitive. In addition, a common unique ID is included in each table for joining the data provider's tables. All experiments were performed on an Intel Core i3-2350M 2.3GHz PC with 4GB memory.

4.5.1 Cost of anonymization without data mashup

In this section, we analyze the cost of anonymization $Cost_{ad}$ to individual data providers without their participation in the data mashup process. Suppose the sensitivity of the dataset $Sen_{ds_i} = 2$ on the scale of 1-5, the price per attribute $Price_{attr_i} = \$0.1$, the size of dataset $Size_{ds_i} = 45,222$ for the data providers DP_1, DP_2, and DP_3 to fairly quantify and compare the cost of anonymization under different privacy models including k-anonymity, LKC-privacy, and ϵ-differential privacy.

Figure 4.4 depicts the cost of anonymization to each data provider without participating in the data mashup process. Figure 4.4.a depicts the cost of anonymization when privacy models k-anonymity and LKC-privacy are enforced with the anonymity threshold L, K, and C. $Cost_{ad}$ generally increases as K or L increases, but this monotonicity does not maintain for DP_1 and DP_2 with the increase of K. For example, $Cost_{ad}$ decreases by $\$72.35$ for DP_1, when K increases from 40 to 50 when $L = 2$. This is because of the better classification accuracy CA, which is increased



(a) k-anonymity and LKC-privacy (b) ϵ-differential privacy

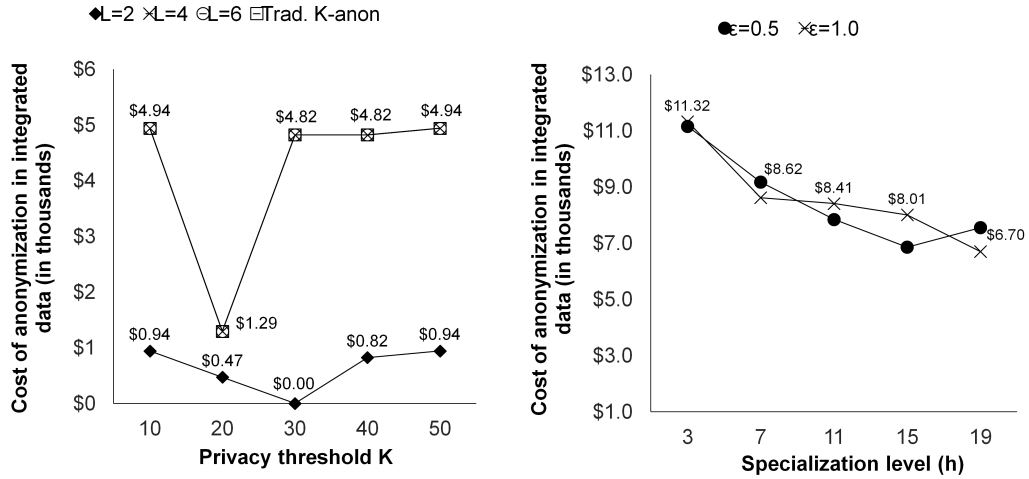
Figure 4.4: Cost of anonymization to individual data provider without data mashup

from 80.3% to 80.5%. This anti-monotonic property of the algorithm helps in finding the sub-optimal anonymization cost. We observe that the DP_1 anonymization cost is higher than DP_2 and DP_3 because DP_1 holds 3 continuous numeric attributes (refer to Table 4.4) that require discretizing into intervals (categorical values) for anonymization. The classification analysis on new data would be less accurate than categorical attributes due to the chance of information loss. The $Cost_{ad}$ of LKC-privacy equals the $Cost_{ad}$ of the traditional k-anonymity when $L = 4$ for each data provider. $Cost_{ad}$ is also insensitive to the change of confidence threshold $10\% \leq C \leq 50\%$.

Figure 4.4.b depicts the cost of anonymization when ϵ-differential privacy is enforced with privacy parameters $\epsilon = 0.5$ and 1.0 and specialization levels $3 \leq h \leq 19$. We observe that $Cost_{ad}$ generally decreases when the specialization level h increases for DP_1 and DP_2 with the setting of a privacy budget to either $\epsilon = 0.5$ or 1.0. But this trend is quite different in relation to DP_3 where $Cost_{ad}$ increases monotonically with the increase in h; the random noise results in lower classification accuracy.

4.5.2 Cost of anonymization in integrated data

In this section, we analyze the cost of anonymization in integrated data $Cost_{intgdata}$ under the joint privacy settings of the three contributing data providers in the data mashup process. Suppose the sensitivity of the dataset $Sen_{ds_i} = 2$ on the scale of 1-5, the price per attribute $Price_{attr_i} = \$0.1$,



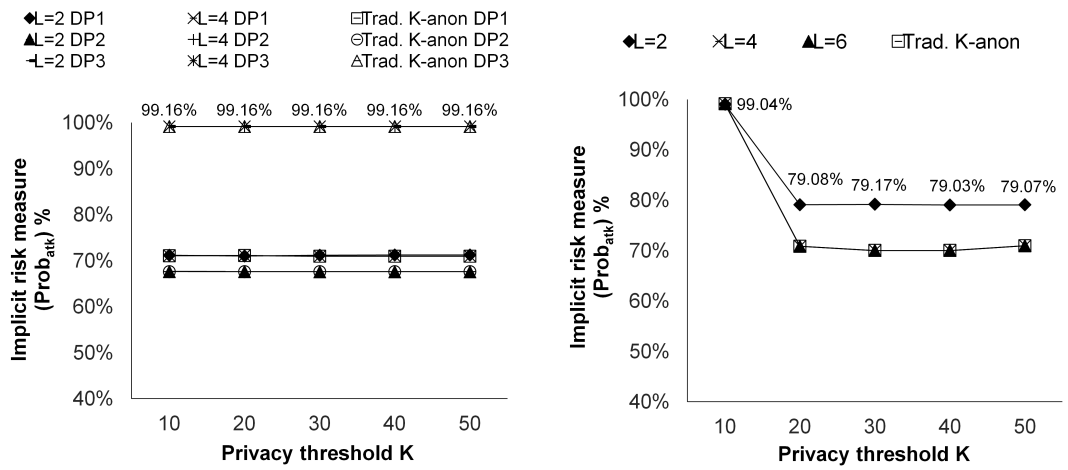
(a) k-anonymity and LKC-privacy (b) ϵ-differential privacy

Figure 4.5: Cost of anonymization in integrated data

the number of attributes $Count_{attr} = 13$ (sum of attributes of DP_1, DP_2, and DP_3) and the size of dataset $Size_{ds_i} = 45,222$ to quantify and compare the cost of anonymization in integrated data under different privacy models, including k-anonymity, LKC-privacy, and ϵ-differential privacy.

Figure 4.5.a depicts the cost of anonymization in integrated data when privacy models k-anonymity and LKC-privacy are enforced with the anonymity threshold $10 \leq K \leq 50$, background knowledge $L = \langle 2, 4, 6 \rangle$, and confidence threshold $C = 50\%$. $Cost_{intgdata}$ generally increases as L increases, but does not exhibit obvious monotonicity with the increase of K. For example, $Cost_{intgdata}$ decreases by $\$3,644.89$ when K increases from 10 to 20 when $L = 4$ and $L = 6$. This is because of improvement in classification accuracy CA, which increases by 3.1%. This helps in finding the sub-optimal anonymization cost. The $Cost_{intgdata}$ of LKC-privacy equals the $Cost_{intgdata}$ of traditional k-anonymity when $L = 4$ and $L = 6$. $Cost_{intgdata}$ is also insensitive to the change of confidence threshold $10\% \leq C \leq 50\%$.

Figure 4.5.b depicts the cost of anonymization in integrated data when ϵ-differential privacy is enforced with privacy parameters $\epsilon = 0.5$ and 1.0 and specialization levels $3 \leq h \leq 19$. We calculate the average accuracy on 10 runs. We observe that $Cost_{intgdata}$ generally decreases as the specialization level h increases, except an increase by $\$693.71$ when privacy budget $\epsilon = 0.5$ and the specialization level h increases from 15 to 19. When ϵ is small, having too many levels makes each specialization less accurate.



(a) without data mashup (b) with data mashup

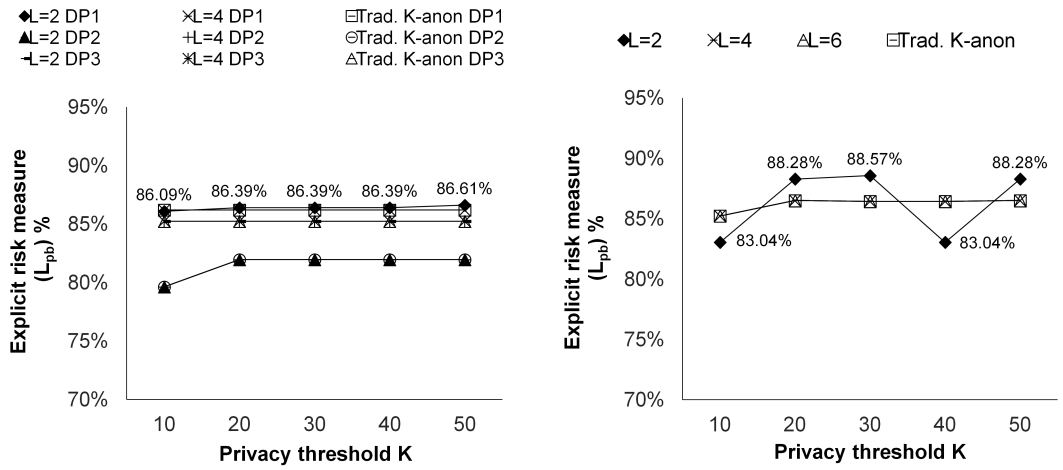
Figure 4.6: Implicit risk measure

4.5.3 Implicit risk measure

In this section, we analyze the implicit risk for each data provider before participation in the data mashup process on their own data and after participation in the data mashup process on the integrated data of the contributing data providers.

Figure 4.6.a depicts the probability of attack $Prob_{atk}$ on the sensitive value *Married-civ-spouse* to the data providers DP_1, DP_2, and DP_3 with privacy threshold $10 \leq K \leq 50$, background knowledge $L = \langle 2, 4 \rangle$, and confidence threshold $C = 50\%$. We observe that the chance of inferring the sensitive attribute value is approximately 71%, 67%, and 99% on the anonymized dataset of DP_1, DP_2, and DP_3, respectively. DP_2 is comparatively better than DP_1 and DP_3 because it has less risk of inferring the sensitive attribute value.

Figure 4.6.b depicts the probability of attack $Prob_{atk}$ on the sensitive value *Married-civ-spouse* in the anonymized integrated dataset of contributing data providers DP_1, DP_2, and DP_3 under the joint privacy settings with the anonymity threshold $10 \leq K \leq 50$, background knowledge $L = \langle 2, 4, 6 \rangle$, and confidence threshold $C = 50\%$. We can observe the trend that $Prob_{atk}$ generally decreases as K or L increases, which also conforms to the theoretical analysis.



(a) without data mashup (b) with data mashup

Figure 4.7: Explicit risk measure

4.5.4 Explicit risk measure

In this section, we analyze the explicit risk for each data provider before participation in the data mashup process on their own data and after participation in the data mashup process on the integrated data of contributing data providers.

Suppose an adversary has prior knowledge about a male victim, that his age is between 40 to 50, his $education$ is $masters$, his $hours\text{-}per\text{-}week$ is > 40., and his $income$ is $\geq 50,000$.

Figure 4.7.a depicts the likelihood of a privacy breach L_{pb} on the sensitive value *Married-civ-spouse* when the aforementioned external knowledge about the victim is linked to the data providers DP_1, DP_2, and DP_3 attributes, where privacy threshold $10 \leq K \leq 50$, background knowledge $L = \langle 2, 4 \rangle$, and confidence threshold $C = 50\%$. We observe that the L_{pb} is approximately 86%, 82%, and 85% on the anonymized dataset of DP_1, DP_2, and DP_3, respectively. DP_2 is comparatively better than DP_1 and DP_3 because it has less risk of a privacy breach.

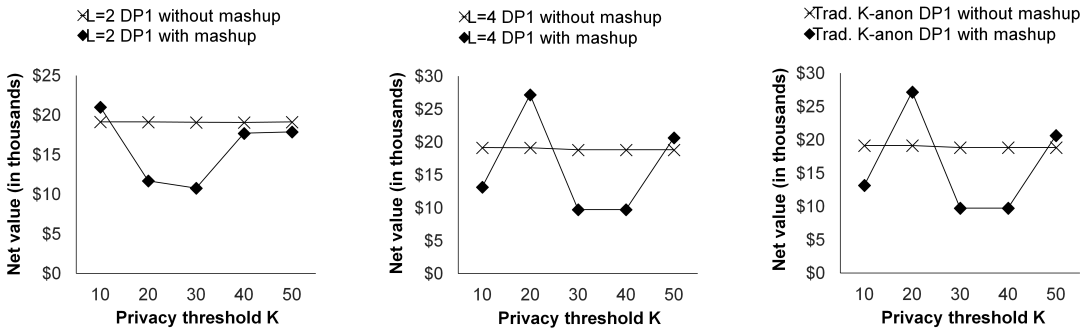
Figure 4.7.b depicts the likelihood of a privacy breach L_{pb} on the sensitive value *Married-civ-spouse* when the aforementioned external knowledge about a victim is linked to the anonymized integrated dataset of contributing data providers DP_1, DP_2, and DP_3 under the joint privacy settings with the anonymity threshold $10 \leq K \leq 50$, background knowledge $L = \langle 2, 4, 6 \rangle$, and confidence threshold $C = 50\%$. Generally, L_{pb} decreases with the increase of L but this trend is not obvious with the increase of K. For example, L_{pb} is 86.44% when $K = 40$ and $L = \langle 4, 6 \rangle$, which is higher

by 3.4% when $L = 2$. This anti-monotonic property of the TDS algorithm helps in identifying the sub-optimal solution. The L_{pb} of LKC-privacy equals the L_{pb} of k-anonymity when $L = 4$ and $L = 6$ because the classification accuracy on the sensitive attribute $Marital\text{-}status$ remains unchanged with the increase of L. Though not shown in the figure, L_{pb} is insensitive to the change of the confidence threshold $10\% \leq C \leq 50\%$.

4.5.5 Impact of privacy requirements on net value

In this section, we analyze the impact of k-anonymity, LKC-privacy, and ϵ-differential privacy requirements on monetary value for each data provider before participation in the data mashup process and after participation on the integrated data of contributing data providers. Suppose the sensitivity of the dataset $Sen_{ds_i} = 2$ on the scale of 1-5, the price per attribute $Price_{attr_i} = \$0.1$, the expected cost of lawsuit $Ecost_{lwst} = \$1000$, the size of dataset $Size_{ds_i} = 45,222$, and the fixed operating cost $F_{OpCost} = \$300$.

Figure 4.8 depicts the impact of k-anonymity and LKC-privacy requirements on DP_1's net value, where privacy threshold $10 \leq K \leq 50$, and confidence threshold $C = 50\%$. Figure 4.8.a depicts the impact on DP_1's net value when the threshold $L = 2$. We observe that DP_1's net value without data mashup (refer to the DP_1's attributes in the Table 4.4) decreases slightly with the increase of K, but it does not maintain monotonicity when $K = 50$. On the other side, DP_1's net value with data mashup drops with the increase of K from 10 to 30, but the net value rises when $K > 30$. This change in trend depends on the information gain for classification analysis of the DP_1's attributes. Figure 4.8.b depicts the impact on DP_1's net value when the threshold $L = 4$. We observe that DP_1's net value without data mashup decreases slightly with the increase of K from 10 to 30, but it is insensitive to change when $K > 30$. On the other side, DP_1's net value with data mashup does not exhibit monotonicity with the increase of K because DP_1's attributes for classification analysis contribute different information gains at different privacy thresholds K on integrated data with collaborating data providers DP_2 and DP_3. Figure 4.8.c depicts the impact on DP_1's net value when the threshold $L = QID$. There are a total of 4 QID attributes in DP_1's dataset. DP_1's net value of traditional k-anonymity is equal to LKC-privacy when $L = 4$. Though not shown in Figure 4.8, net value is insensitive to the change of the confidence threshold $10\% \leq C \leq 50\%$. The maximum net value



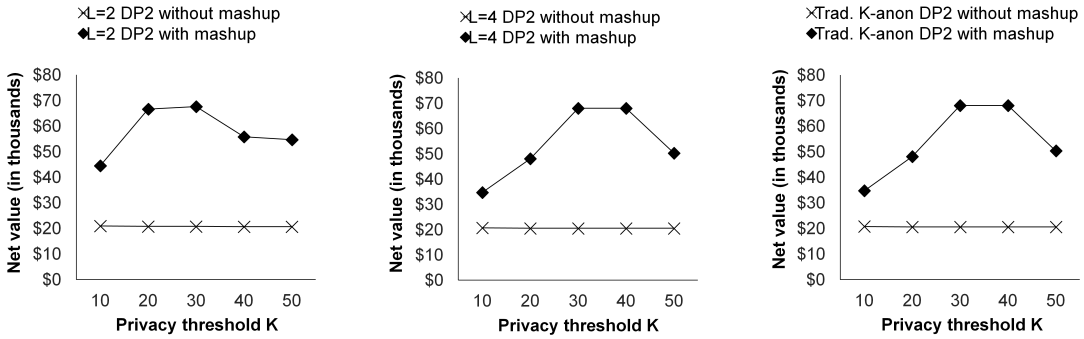
(a) *L=2* (b) *L=4* (c) *L=QID*

Figure 4.8: Impact of k-anonymity and LKC-privacy requirements on DP_1's net value

achieved by the DP_1 is $\$27,190.94$ when $K = 20$ and $L = 4$.

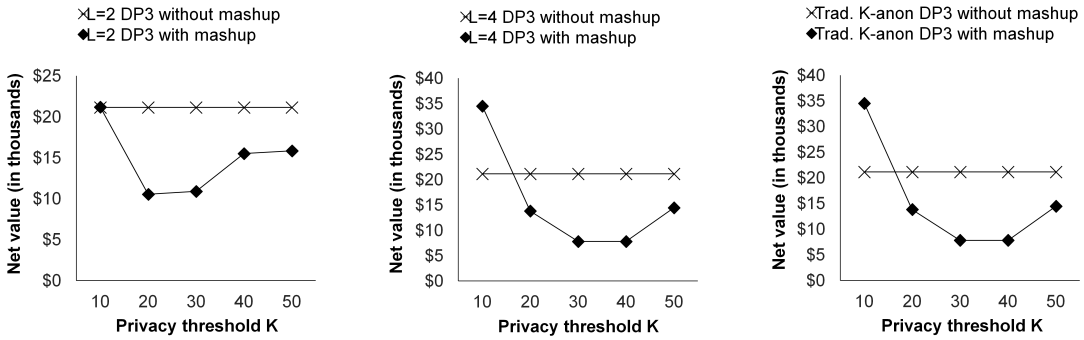
Figure 4.9 depicts the impact of k-anonymity and LKC-privacy requirements on DP_2's net value, where privacy threshold $10 \leq K \leq 50$, and confidence threshold $C = 50\%$. Figure 4.9.a depicts the impact on DP_2's net value when the threshold $L = 2$. We observe that DP_2's net value without data mashup (refer to the DP_2's attributes in the Table 4.4) decreases slightly with the increase of K except when $K = 30$. On the other side, DP_2's net value with data mashup increases with the increase of K from 10 to 30, but the net value drops when $K > 30$. This change in trend depends on the information gain for classification analysis of DP_2's attributes. Figure 4.9.b depicts the impact on DP_2's net value when the threshold $L = 4$. We observe that DP_2's net value without data mashup decreases slightly with the increase of K from 10 to 20, but it is insensitive to change when $K > 20$. On the other side, DP_2's net value with data mashup increases with the increase of K from 10 to 40, but it drops when $K = 50$. This drop in net value is due to the loss of information gain in classification analysis. Figure 4.9.c depicts the impact on DP_2's net value when the threshold $L = QID$. There are a total of 4 QID attributes in DP_2's dataset. DP_2's net value of traditional k-anonymity is equal to LKC-privacy when $L = 4$. Though not shown in Figure 4.9, net value is insensitive to the change in the confidence threshold $10\% \leq C \leq 50\%$. The maximum net value achieved by DP_2 is $\$68,060.37$ when $K = 30$ and $K = 40$, and $L = 4$.

Figure 4.10 depicts the impact of k-anonymity and LKC-privacy requirements on DP_3's net value, where privacy threshold $10 \leq K \leq 50$, and confidence threshold $C = 50\%$. Figure 4.10.a depicts the impact on DP_3's net value when the threshold $L = 2$. We observe that DP_3's net value



(a) *L=2* (b) *L=4* (c) *L=QID*

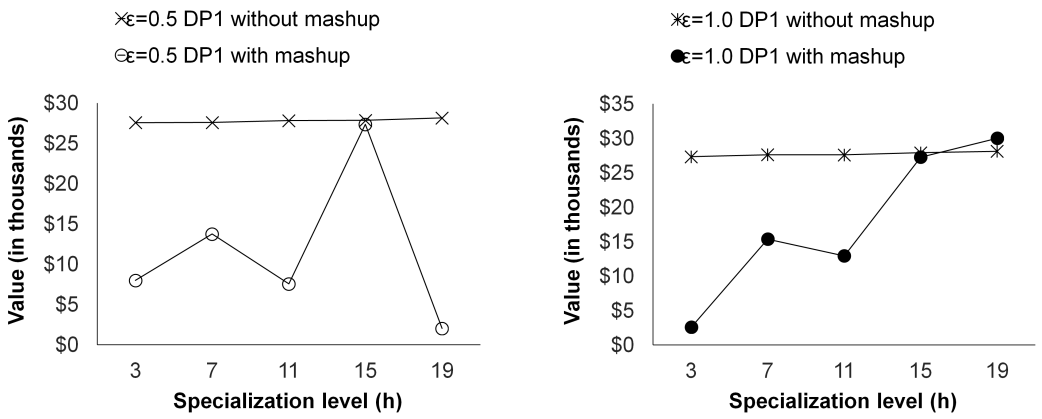
Figure 4.9: Impact of k-anonymity and LKC-privacy requirements on DP_2's net value



(a) *L=2* (b) *L=4* (c) *L=QID*

Figure 4.10: Impact of k-anonymity and LKC-privacy requirements on DP_3's net value

without data mashup (refer to the DP_3's attributes in Table 4.4) is insensitive to change with the increase of K. On the other side, DP_3's net value with data mashup drops with the increase of K from 10 to 20, but the net value gradually rises when $K > 20$. This change in trend depends on the information gain for classification analysis of DP_3's attributes. Figure 4.10.b depicts the impact on DP_3's net value when the threshold $L = 4$. We observe that DP_3's net value without data mashup is insensitive with the increase of K. On the other side, DP_3's net value with data mashup drops with the increase of K except when $K = 50$. This fall in net value is due to the loss of information gain in classification analysis. Figure 4.10.c depicts the impact on DP_3's net value when the threshold $L = QID$. There are a total of 5 QID attributes in DP_3's dataset. DP_3's net value of traditional k-anonymity is equal to LKC-privacy when $L = 4$. Though not shown in Figure 4.10, net value is insensitive to the change of the confidence threshold $10\% \leq C \leq 50\%$. The maximum net value achieved by DP_3 is $\$34,522.01$ when $K = 10$ and $L = 4$.

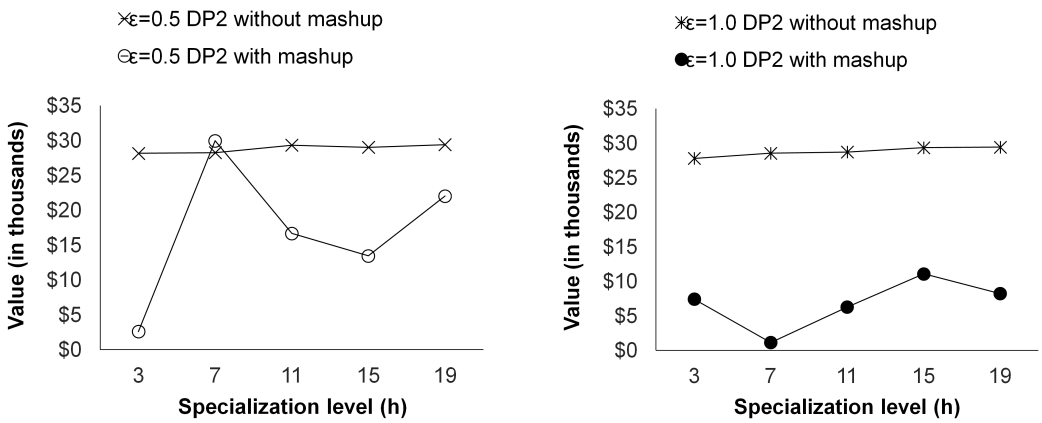


(a) $\epsilon = 0.5$ (b) $\epsilon = 1.0$

Figure 4.11: Impact of ϵ-differential privacy requirements on DP_1's monetary value

Figure 4.11 depicts the impact on DP_1's monetary value when ϵ-differential privacy is enforced with privacy parameters $\epsilon = 0.5$ and 1.0 and specialization levels $3 \leq h \leq 19$. Figure 4.11.a depicts the impact on DP_1's monetary value when the threshold $\epsilon = 0.5$. We observe that DP_1's monetary value without data mashup (refer to the DP_1's attributes in Table 4.4) increases monotonically as the increase in specialization level h. On the other side, DP_1's monetary value with data mashup increases when specialization level h increases from 3 to 7 and 11 to 15, but the value drops due to the loss of data utility when $h = 11$ and $h = 19$. Figure 4.11.b depicts the impact on DP_1's monetary value when the threshold $\epsilon = 1.0$. We observe that DP_1's monetary value without data mashup increases slightly with the increase in the specialization level h except when $h = 11$. DP_1's net value with data mashup generally increases with the increase in h, but it does not maintain monotonicity when $h = 11$ due to the provision of less data utility in classification analysis with collaborating data providers DP_2 and DP_3. The benefits to DP_1 of doing data mashup is higher than going without data mashup by gaining the maximum net value $\$30,0187.37$ when $\epsilon = 1.0$ and $h = 19$.

Figure 4.12 depicts the impact on DP_2's monetary value when ϵ-differential privacy is enforced with privacy parameters $\epsilon = 0.5$ and 1.0 and specialization levels $3 \leq h \leq 19$. Figure 4.12.a depicts the impact on DP_2's monetary value when the threshold $\epsilon = 0.5$. We observe that DP_2's monetary value without data mashup (refer to the DP_2's attributes in Table 4.4) generally increases as the increase in specialization level h except when $h = 15$. DP_2's monetary value with data mashup does not exhibit monotonicity with the increase in the specialization level h due to the loss of data utility

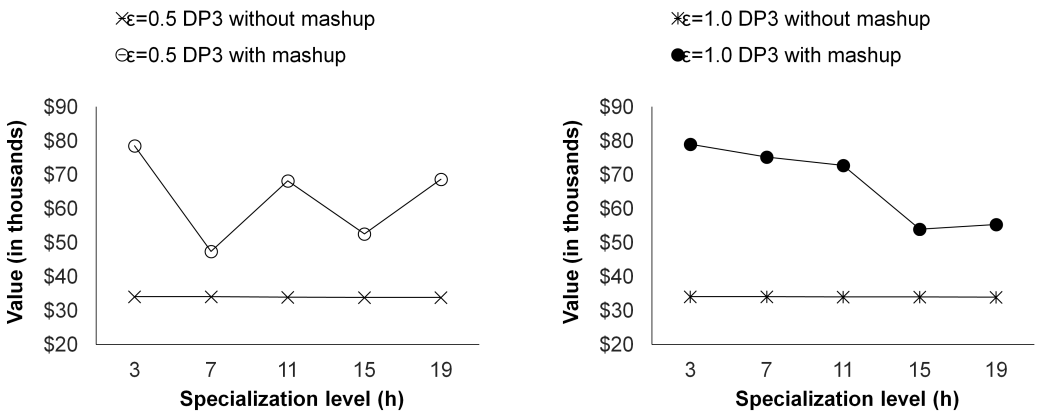


(a) $\epsilon = 0.5$ (b) $\epsilon = 1.0$

Figure 4.12: Impact of ϵ-differential privacy requirements on DP_2's monetary value

in classification analysis when $h = 11$ and the provision of less data utility in comparison to the other collaborating data providers DP_1 and DP_3 when $h = 15$. Figure 4.12.b depicts the impact on DP_2's monetary value when the threshold $\epsilon = 1.0$. We observe that DP_2's monetary value without data mashup increases monotonically with the increase in the specialization level h. On the other side, DP_2's monetary value with data mashup does not exhibit monotonicity with the increase in the specialization level h due to the loss of data utility in classification analysis when $h = 7$ and the provision of less data utility in comparison to other collaborating data providers DP_1 and DP_3 when $h = 19$. The benefits of doing data mashup are higher than doing without data mashup to DP_2 by gaining the maximum net value $\$29,971.26$ when $\epsilon = 0.5$ and $h = 7$.

Figure 4.13 depicts the impact on DP_3's monetary value when ϵ-differential privacy is enforced with privacy parameters $\epsilon = 0.5$ and 1.0 and specialization levels $3 \leq h \leq 19$. Figure 4.13.a depicts the impact on DP_3's monetary value when the threshold $\epsilon = 0.5$. We observe that DP_3's monetary value without data mashup (refer to the DP_3's attributes in the Table 4.4) decreases slightly as the specialization level h increases. DP_3's monetary value with data mashup does not exhibit monotonicity with the increase in the specialization level h, but DP_3's monetary value is greater than DP_1 and DP_2 at specialization levels 3 to 19. Figure 4.13.b depicts the impact on DP_3's monetary value when the threshold $\epsilon = 1.0$. We observe that DP_3's monetary value without data mashup decreases slightly as the specialization level h increases. On the other side, DP_3's monetary value with data mashup decreases with the increase in the specialization level h except when $h = 19$. The



(a) $\epsilon = 0.5$ (b) $\epsilon = 1.0$

Figure 4.13: Impact of ϵ-differential privacy requirements on DP_3's monetary value

benefits of doing data mashup is higher than going without data mashup to DP_3 by gaining the maximum net value $\$78,993.45$ when $\epsilon = 1.0$ and $h = 3$.

4.6 Summary

We have proposed a business model to quantify and compare the costs and benefits for releasing integrated anonymized data of multiple providers over an individual data provider when trading person-specific information in the e-market. Our model enables data providers to set up their joint privacy requirements for classification analysis on mashup data. The data mashup process is implemented fairly that allows data providers to integrate their data subject to the given privacy requirements. During the data mashup process every data provider competes with the other participating data providers to generate more profit from their own data. The data provider whose data provides more information gain will get a proportionally higher share in terms of monetary value from the distribution of the achieved net value. We have incorporated relevant factors that are associated with the revenue and costs to determine the net value. Our model helps data providers in finding the sub-optimal value by evaluating the benefits of data mashup and impacts of data anonymization based on the choices of privacy models and data mashup anonymization algorithms.

Chapter 5

Enabling Secure Trustworthiness Assessment and Privacy Protection in Integrating Data for Trading Person-specific Information

5.1 Introduction

Data are the fuel of today's digital economy. Yet, data coming from a single source often fail to provide a complete picture for big data analytics. To answer complex queries, companies usually have to seek additional data from multiple sources. The emerging cloud paradigm *Data-as-a-Service* (*DaaS*) provides an ideal platform for data integration in order to serve data consumers' demands. However, business data often contain person-specific information. Mashing up personal data from different sources raises concerns on security, privacy, and data reliability. In the past decade, research communities have proposed various trust models [18, 148] and frameworks [35, 131] to evaluate and measure the security strength of cloud environments, but limited research considers the aspect of data reliability. In this chapter, we propose a cloud-based data integration solution that considers privacy protection, data trustworthiness, and fairness of profit distribution among data providers.

Recent studies [52, 138] report the financial impact of poor quality of data on businesses. For effective decision-making, organizations must have to take appropriate measures regarding the quality of their data. Existing works [28, 93] aim to provide prevention from unauthorized use and modification when data is in transit but do not verify data if any party provides incorrect or fabricated data. Our research perspective is to determine the trustfulness of private data held by dishonest data providers who may arbitrarily attempt to provide false data when trading person-specific information in the e-market for monetary benefits. Our proposed method can detect such behavior from dishonest data providers, who resemble adversaries under the covert security model [21]. In literature [11, 43, 55] two protocols are discussed, namely *Private Set Intersection* (*PSI*) and *Private Set Intersection Cardinality* (*PSI-CA*) for privacy and data quality assessment. Freudiger et al. [56] claimed that these protocols are incurred from computational overhead and thus are not applicable to real-world scenarios. They proposed some protocols that operate on reduced dimensionality descriptions and so can be scalable to large datasets. It is a challenging problem to evaluate the trustfulness of private data held by untrusted data providers. In this chapter, we study the problem of untrusted data providers holding overlapping attributes on a person-specific dataset. We illustrate the problem in the following example.

Example 3. Suppose there is a cloud-based data market, where data consumers can place their data mining requests and data providers compete with each other to contribute their data with the goal of fulfilling the requests for monetary reward. Consider the 12 raw data records in Table 5.1, where each record corresponds to the personal information of an individual. The three data providers own different yet overlapping sets of attributes over the 12 records.

Since the data providers collect data from different channels, it is quite possible that their data conflict with each other as illustrated in Table 5.1. According to the predefined generalization hierarchy of the attributes in Fig. 5.1, the individuals in the table can be generalized to two groups: *Non-Technical* and *Technical*. Suppose a data consumer wants to perform a data analysis that depends on the *Non-Technical* and *Technical* groups. Yet, the inconsistent, conflicting, or even inaccurate data may mislead the analysis result. For example, DP_1 and DP_3 state that the individuals in $\{Rec\#3, 5\}$ are $Cleaner$, while DP_2 states that they are $Technician$. A similar conflict can be seen in the

Table 5.1: Raw data owned by three data providers

	Data Provider DP_1			*Data Provider* DP_2			*Data Provider* DP_3		
RecID	**Age**	**Sex**	**Job**	**Sex**	**Education**	**Job**	**Age**	**Education**	**Job**
1	39	M	Lawyer	M	Bachelors	Lawyer	45	Doctorate	Lawyer
2	50	M	Lawyer	M	Masters	Lawyer	50	Doctorate	Lawyer
3	38	M	Cleaner	M	12th	Technician	35	12th	Cleaner
4	53	M	Lawyer	M	Doctorate	Doctor	57	Masters	Lawyer
5	28	F	Cleaner	F	11th	Technician	28	11th	Cleaner
6	37	F	Welder	F	12th	Welder	37	11th	Welder
7	49	F	Painter	F	12th	Cleaner	49	12th	Painter
8	59	M	Doctor	F	Doctorate	Doctor	66	Doctorate	Doctor
9	31	F	Painter	M	12th	Welder	27	12th	Painter
10	42	M	Technician	M	Bachelors	Technician	42	Bachelors	Technician
11	37	M	Lawyer	M	Masters	Lawyer	38	Masters	Lawyer
12	30	M	Lawyer	M	Masters	Lawyer	28	Bachelors	Lawyer

$Rec\#9$, where DP_1 and DP_3 provide the Job as $Painter$, and DP_2 provides the Job as $Welder$. In this example, the Job attribute on $\{Rec\#3, 5, 9\}$ has two different values that are categorized as *Non-Technical* and *Technical*, respectively. These inconsistencies significantly impact the quality of data analysis. ■

Presumably the data providers would have missing values on some attributes, although the same set of records is identified by executing the secure set intersection protocol [11] on the globally unique identifiers [126, 128]. Instead of avoiding participating in the data mashup process, they would prefer to impute missing values by using the machine learning methods appropriate for their datasets. The properties of a dataset such as low dimensional or high dimensional data, single-type or mixed-type data, or linearly separable or non-linearly separable data are a crucial factor before choosing the imputation method. The data providers' decision whether to use a single imputation method or multiple imputation methods is conditional on their missing data. We evaluate the robustness of our approach when an acquisitive data provider employs a machine learning method for imputation of missing data.

In the context of quantifying monetary value through sharing person-specific data, the data providers first must do the valuation of personal data, but there is no determined market price [129, 135] for person-specific data that can be taken as a proxy for the valuation. It is also well-acknowledged from the existing literature [4, 53] that there is no commonly agreed methodology for valuing personal data. However, in the e-market, many companies actively collect personal

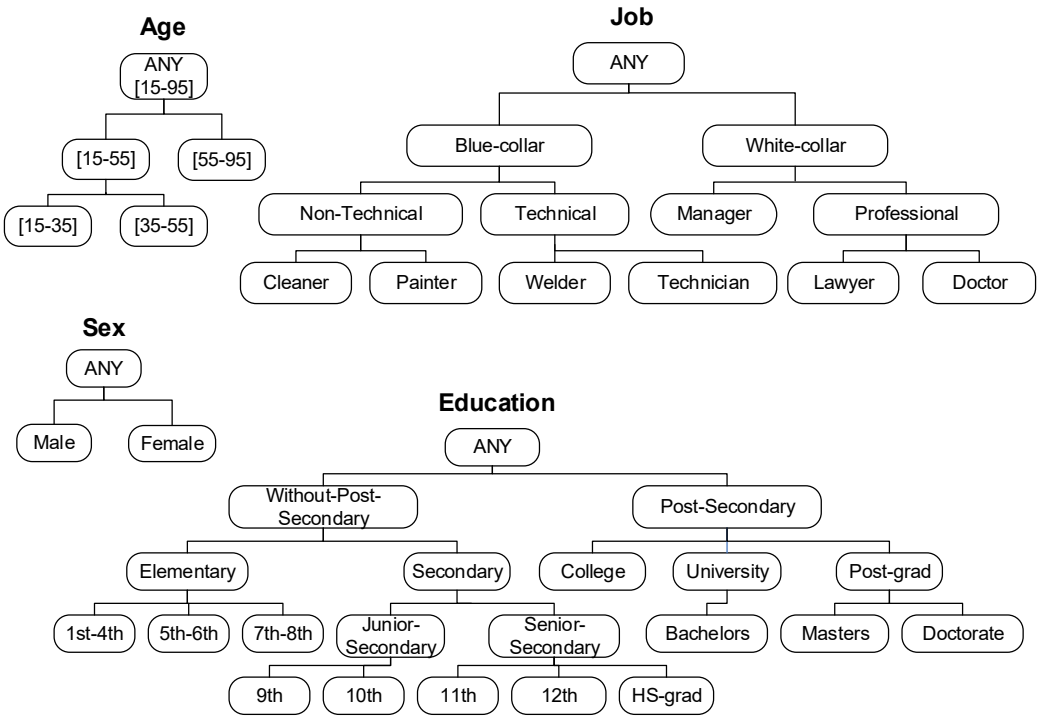


Figure 5.1: Taxonomy trees

information by providing monetary rewards to their customers. In this chapter, we incorporate the Vickrey-Clarke-Groves (VCG) auction mechanism for the valuation of data providers' attributes. We reason that it is a dominant strategy, where no data provider has an incentive to lie about his true valuations. In addition, private data often encode privacy-sensitive information related to individuals that need to be protected when integrating data from the competing data providers. In this chapter we adopt differential privacy [50] because it provides strong privacy guarantees to an individual independently of an adversary's background knowledge, in contrast to underlying assumptions in syntactic privacy models [112, 120, 147] about an adversary's knowledge.

Contributions. We propose a novel solution to address the critical issues of data trustworthiness, privacy protection, and profit distribution for cloud-based data integration services. The data trustworthiness problem has been studied in [114, 115, 165] applications of sensor networks. The provenance-based approach has been used in [40, 114] to evaluate the trustworthiness of network nodes and data items. This approach is primarily used to collect evidence about where the data originates and how the data generates. In this chapter we are not concerned about the high degree of

the instrumentation of customers' private data, which is collected by data providers. However, our proposed approach makes novel use of information entropy to verify the correctness of data from untrusted data providers and also to preserve the privacy of customers' data held by data providers when evaluating the trustworthiness of the providers. We summarize our contributions as follows:

- Our proposed method, *IEB_Trust*, is the first entropy-based trust computation method that enables secure trustworthiness assessment and incorporates fairness in the verification process to restrict dishonest data providers from participation in the next phase for integrating data.
- We compare our proposed method with a closely related method. Results suggest that our entropy-based trust computation algorithm is capable of significantly improving runtime efficiency.
- We evaluate the robustness of our method when an acquisitive data provider adopts machine learning techniques to substitute missing values on their own data and claim them as original data collected from customers to compete with the other participating data providers.
- We define the procedure for setting the price on person-specific attributes in trading personal information from data providers based on the VCG mechanism.
- We integrate data from chosen data providers using *Differentially private anonymization based on Generalization* (*DistDiffGen*) [128] and analyze the impacts of privacy protections and trust scores on data providers' monetary value.

The rest of the chapter is organized as follows: In Section 5.2, we provide an overview of the trust mechanism and the problem statement. In Section 5.3, we present our proposed solution. In Section 5.5, we compare our proposed method and provide empirical study to analyze the trustworthiness of each data provider and further analyze its impact along with the ϵ-*differential privacy* protection on a data provider's monetary value. Finally, we provide the summary in Section 5.6.

5.2 Trust mechanism

In this section, we first provide an overview of our trust mechanism and then formally define the research problem.

5.2.1 Overview of trust mechanism

Fig. 5.2 provides an overview of our trust mechanism in which data providers, data consumers, and cloud service providers are the main entities. Data providers collect person-specific information from customers and intend to participate in the data mashup for generating more profit by competing with peer data providers, data consumers perform data analysis on the received data, and the cloud service provider (CSP) is a semi-trusted arbitrator between data providers and data consumers. The CSP manages three key services: authentication, mashup coordination, and data verification. These services are run on a cloud server by the CSP. First, each data provider has to pass the authentication phase to prove their identity. Second, data consumers submit their data requests to the CSP. In this chapter, we assume that a data consumer runs a classification analysis on its requested attributes by a supervised machine learning method. A resource queue is built by the mashup service to manage data requests from a data consumer, which is accessible only to authenticated data providers. Third, data providers register their available data attributes on the registry hosted by the mashup service; each data attribute is assigned a sequence number based on its arrival. Fourth, the verification process is run to detect false or incorrect data and to determine the trustworthiness of each data provider. Fifth, this process results in determining the accepted data providers. Sixth, the CSP connects the group of accepted data providers with the data consumer to serve its demand. This is done by the mashup service that determines the group of data providers whose data can collectively fulfill the demand of a data consumer. Seventh, the data providers quantify their costs and benefits using joint privacy requirements and integrate their data over the cloud. Finally, the anonymous integrated data is released to the data consumer.

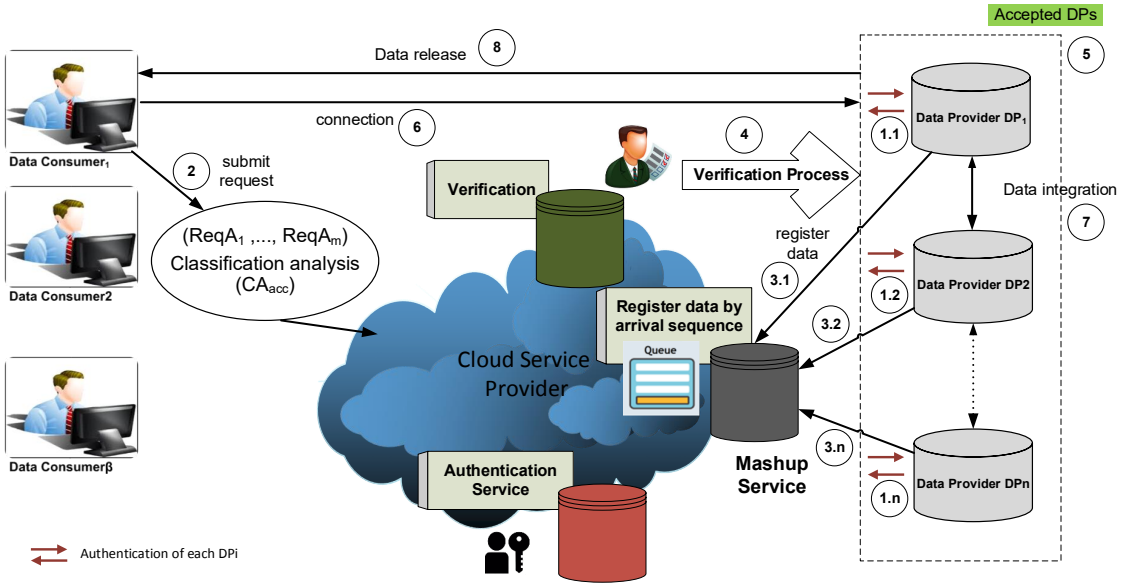


Figure 5.2: Trust mechanism

5.2.2 Problem statement

We describe our problem as follows. There are three main entities discussed in our trust mechanism: data providers, data consumers, and a cloud service provider (CSP). *Data verification* service runs on a cloud server CS, which is managed by the CSP. The purpose of this service is to verify the correctness of data. The CSP is a semi-trusted arbitrator who would not have access to customers' private data, which is held by the data providers. Data providers are considered to be dishonest, meaning that they may arbitrarily attempt to provide false data because they are acquisitive in competing with others in the e-market. The behavior of such data providers is similar to adversaries in the covert security model.

Suppose data providers DP_1,...,DP_n own private data tables $D_1, \ldots, D_n$, respectively. Each record in the data table belongs to a unique individual. All explicit identifiers of an individual, such as name, social security number (SSN), and account number, have been removed. Each D_i is defined over a set of attributes $\mathcal{PA}_i = \{A_1, \ldots, A_d\}$. We assume that the data providers hold overlapping attributes for the same set of records identified by executing the secure set intersection protocol [11, 126] on the globally unique identifiers $RecID$. We require $\forall \mathcal{PA}_i \; \exists \mathcal{PA}_j$ such that

$\mathcal{PA}_i \cap \mathcal{PA}_j \neq \emptyset$, where $i \neq j$, and $\mathcal{PA} = \{\mathcal{PA}_1, \ldots, \mathcal{PA}_n\}$. In addition, each D_i contains a A^{cls} attribute for classification analysis, which is shared among all the data providers. Each $A_{\mathcal{J}}$ is either a categorical or a numerical attribute, but A^{cls} is required to be categorical. A data consumer submits a data request $ReqA = \{ReqA_1, \ldots, ReqA_m\}$ for classification analysis. We assume that each data provider has $\mathcal{PA}_i \subseteq ReqA$ to serve the demand of a data consumer. The goal of this trust computation is to restrict dishonest data providers from participation in the data mashup process when their trust scores drop below a certain threshold.

Problem 1 (Trust computation). Given multiple person-specific raw data tables $D_1, \ldots, D_n$ from data providers $DP_1, \ldots, DP_n$ and a set of requested attributes $ReqA = \{ReqA_1, \ldots, ReqA_m\}$ for classification analysis from a data consumer, the research problem is to verify the correctness of data on the submissions of the overlapping set of attributes $\mathcal{PA}_i = \{A_1, \ldots, A_d\}$ on the same set of records from each data provider DP_i, where $\mathcal{PA}_i \cap \mathcal{PA}_j \neq \emptyset \; \forall \mathcal{PA}_i \; \exists \mathcal{PA}_j$ and $i \neq j$ and to compute the trust score TS_{DP_i} of each data provider.

In the context of data privacy, the data providers want to integrate their data in a way such that no data provider should learn any additional information about the others as a result of data integration. After the completion of trust computation, the data providers $DP_1, \ldots, DP_n$ attain a mutually exclusive set of attributes $\mathcal{PA}_i = \{A_1, \ldots, A_d\}$ over the same set of records for data integration. That is, $\mathcal{PA}_i \cap \mathcal{PA}_j = \emptyset$ for any $1 \leq i, j \leq n$. We assume that for each attribute $A_{\mathcal{J}} \in \mathcal{PA}_i$, a taxonomy tree is provided that defines the hierarchy of values in $\Omega(A_{\mathcal{J}})$, where $\Omega(A_{\mathcal{J}})$ represents the domain of $A_{\mathcal{J}}$. Data providers require doing their attributes' valuations for price setting and jointly setting up the privacy requirements, such as privacy budget ϵ and specialization level h for a *ϵ-differential privacy* model, before data integration. They wish to derive their monetary shares from the information utility of anonymous integrated data $\hat{D}$ for classification analysis and their trust scores.

Problem 2 (Monetary share under ϵ-differential privacy mechanism). Given multiple raw data tables $D_1, \ldots, D_n$ containing mutually exclusive sets of attributes $\mathcal{PA}_i = \{A_1, \ldots, A_d\}$, i.e., $\mathcal{PA}_i \cap \mathcal{PA}_j = \emptyset$ for any $1 \leq i, j \leq n$ over the same set of records, and a data request $ReqA = \{ReqA_1, \ldots, ReqA_m\}$ from a data consumer for classification analysis, the research problem is to derive the monetary share of each DP_i from their information utility and trust scores

over the differentially private release of integrated dataset $\hat{D}$ under the joint privacy requirements and attributes' valuations.

Several companies, such as Acxiom, AnalyticsIQ, Dataline, and Expedia, collect user data including demographic, financial, retail, social, and travel information from multiple sources with the goal of serving different market needs [6]. Our research problem can be generalized to other similar companies who face trustworthy or quality data issues [52] and whose business models are primarily based on sharing person-specific information.

5.3 Proposed solution

In this section, we provide a solution to address the concerns of stakeholders on data trustworthiness, privacy protection, and profit distribution in the online market for trading person-specific data. Section 5.3.1 presents our proposed *IEB_Trust*, an information entropy-based trust computation algorithm to restrict dishonest data providers from participation in the data mashup process and to assess the trustworthiness of each data provider. Section 5.3.2 discusses security properties. Section 5.3.3 provides an analysis of *IEB_Trust* algorithm. Section 5.3.4 provides an evaluation of learner models. Section 5.3.5 provides an auction mechanism for price-setting among data providers who own multiple attributes. Section 5.3.6 presents an algorithm for privacy protection by which data providers can determine the impact of anonymization on data utility for classification analysis. Section 5.3.7 discusses how the chosen data providers can quantify their monetary value.

5.3.1 Trust computation

In Section 5.2.2, we state the problem where the challenge is to verify the correctness of data from untrusted multiple data providers who own overlapping attributes for the same set of records. We assume that the data providers are competitors who intend to maximize their profits. The data providers consider as dishonest anyone who may arbitrarily attempt to provide false data to get a larger monetary share from their participation. To address this problem, we propose a novel algorithm that adopts information entropy for secure trustworthiness assessment of acquisitive data providers. Information entropy has been widely used in machine learning tools and decision-making systems.

Compared to the existing work on data trustworthiness [114, 115, 165], our proposed algorithm not only detects false or incorrect data from a dishonest data provider during the verification process, but also preserves the privacy of customers' data owned by a data provider. Furthermore, our method provides better runtime efficiency over provenance-based approaches [40, 114].

Algorithm 3 presents our approach in more detail. A cloud service provider (CSP) runs this algorithm on a cloud server(CS). Consider multiple data providers $DP_1, \ldots, DP_n$, who own private data tables $D_1, \ldots, D_n$ having overlapping attributes for the same set of records identified by the common record identifier $RecID$ [11, 126]. First, the CS and each DP_i mutually authenticate each other and derive ks_i symmetric keys for all $i \in I$ by the mutual authentication protocol [45] for the secure exchange of messages. Each DP_i has its own ks_i to answer the CS's queries. Second, a data consumer submits a data request $ReqA = \{ReqA_1, \ldots, ReqA_m\}$ to the CS. Third, each data provider DP_i submits an available set of attributes $\mathcal{PA}_i = \{A_1, \ldots, A_d\}$, where $\mathcal{PA}_i \subseteq ReqA$, to the CS. We assume that initially all the participating data providers have "zero" in their trust scores (Line 3). ϵ' is the allocated privacy budget to consume for each requested attribute. A resource queue is created by the mashup service for m requested attributes, where each attribute $A_{\mathcal{J}} \in \mathcal{PA}_i$ of a corresponding data provider is registered with its arrival sequence (Line 9).

Fourth, the verification process is run to determine the trustworthiness of each data provider. In the first round, CS successively selects one attribute $ReqA_x{}'$ uniformly at random without replacement over a domain of m requested attributes and sends an encrypted challenge $E(ks_i, ReqA_x{}')$ to the corresponding data providers $DP_1, \ldots, DP_n$, who own common attribute $A_{\mathcal{J}}$. Prior to responding to this challenge, each DP_i decrypts to retrieve $ReqA_x{}'$, computes information gain on the challenge attribute in Line 16, denoted by $\mathcal{G}^{(1)}_{A_{\mathcal{J}}}$ (refer to Section 5.3.1.1 for details), according to Eq. (27) [136] and then adds noise to a true output. Then DP_i encrypts the message $\psi^{(1)} \leftarrow E(ks_i, \mathcal{G}^{(1)'}_{A_{\mathcal{J}}})$ and computes tags $\Upsilon^{(1)} \leftarrow \mathcal{S}(k_h, \psi^{(1)})$ by using *keyed hash-based message authentication code* (*HMAC*) in Line 17. CS receives the concatenated message, tag, and identity $\psi^{(1)} \| \Upsilon^{(1)} \| DP_i$ on his challenge from each data provider. Then CS computes the comparison to determine the majority candidates by invoking procedure findMajCand($\psi^{(1)} \| \Upsilon^{(1)}$, $size$) in Line 19, where $size$ indicates the number of data providers who own the requested attribute. This procedure returns majority candidate $Maj^{R(1)}_{Cand}$. In the second round, CS generates $\mathcal{K}$ random IDs for the requested challenge $ReqA_x{}'$, i.e., picked in

the first round, from $|D_i|$ records, then generates $\mathcal{P}$ pairs of values for $ReqA_x{}'$ and A^{cls} attributes. CS sends another challenge to each DP_i by concatenating the encrypted $\mathcal{K}$ random IDs and $\mathcal{P}$ pairs of values as $E(ks_i, \mathcal{K}, ReqA_x{}') \| E(ks_i, v_{x'}, v_{cls})$. DP_i decrypts to retrieve $\mathcal{K}$ record IDs and $\mathcal{P}$ pairs of values. DP_i concatenates $\mathcal{K}$ records and $\mathcal{P}$ pairs of values received from the CS. DP_i computes $\mathcal{G}^{(2)}_{A_\mathcal{J}}$ on the concatenated version and then adds noise to a true output, encrypts it as $\psi^{(2)} \leftarrow E(ks_i, \mathcal{G}^{(2)'}_{A_\mathcal{J}})$, and computes the tag as $\Upsilon^{(2)} \leftarrow \mathcal{S}(k_h, \psi^{(2)})$. CS receives $\psi^{(2)} \| \Upsilon^{(2)} \| DP_i$ on the second round challenge from the corresponding data providers in Line 28. CS again invokes procedure findMajCand($\psi^{(2)} \| \Upsilon^{(2)}$, $size$) to determine the majority candidates in Line 29. This process repeats α times. In Line 33 an intersection of both the rounds is computed to determine Maj_{Cand}.

Candidates whose scores match on the majority are considered as Qualified, denoted by $Qual_{DP_i}$, who gain a positive weight γ in their trust scores TS_{DP_i}. Alternatively, candidates whose scores do not match are considered as Non-Qualified, denoted by $UnQual_{DP_i}$. Subsequently, $UnQual_{DP_i}$ is penalized with a negative weight -γ in their trust scores TS_{DP_i}. For example, when two data providers own a common attribute, but their scores do not match, they both will be penalized with a negative trust score. When only a single data provider responds to the CS challenge of $ReqA_x{}'$, it is accepted based on his existing trust score $TS_{DP_i} \geq 0$. However, in this case, the trust score does not increase for that data provider. When a data consumer request for an attribute, which is not fulfilled by the participating data providers, then that attribute is excluded from the verification process, and the data providers gain no monetary value from it. The comparison is performed (Line 45) to select one candidate (or data provider) on each attribute from the qualified data providers $Qual_{DP_n}$ based on their arrival sequences (using first-come first-served (FCFS) rule). If the final aggregated trust score of any data provider becomes < 0 that data provider drops from the final selection for the data mashup and the attributes initially belonging to him are subsequently reassigned to other qualified data providers that appear next in the arrival sequences. The algorithm terminates when there is no more attribute for verification.

5.3.1.1 Computation of information gain

We use information gain as a criterion for splitting attributes [136] based on the concept introduced by Claude Shannon on information theory [149]. We compute information gain on an individual attribute $A_{\mathcal{J}} \in \mathcal{PA}_i$ of each data provider in the presence of a shared class attribute A^{cls} on raw data. Let $D^{\tau} \subseteq D_i$ denote a subset of the data table D_i. Suppose the attribute A^{cls} has $\mathcal{C}$ distinct values. Let $A^{cls}_{i,D^{\tau}}$ be the set of records of class A^{cls}_i in D^{τ}. Let $|D^{\tau}|$ and $|A^{cls}_{i,D^{\tau}}|$ denote the number of records in D^{τ} and $A^{cls}_{i,D^{\tau}}$, respectively. The entropy on the data table D^{τ} is computed as follows.

$$E(D^{\tau}) = -\sum_{i=1}^{\mathcal{C}} \Pr_i \times log_2 \Pr_i \tag{25}$$

where $\Pr_i$ is the probability that an arbitrary record in D^{τ} belongs to class A^{cls}_i. It is estimated by $\frac{|A^{cls}_{i,D^{\tau}}|}{|D^{\tau}|}$.

We can further partition the records in D^{τ} on the attribute $A_{\mathcal{J}}$. If $A_{\mathcal{J}}$ is discrete-valued, then one branch is grown for each known value of $A_{\mathcal{J}}$. On the other side, if $A_{\mathcal{J}}$ is continuous-valued, then two branches are grown, corresponding to $A_{\mathcal{J}} \leq splitpoint$ and $A_{\mathcal{J}} > splitpoint$. It is calculated by the following equation.

$$E_{A_{\mathcal{J}}}(D^{\tau}) = \sum_{j=1}^{\mathcal{V}} \frac{|D^{\tau}_j|}{|D^{\tau}|} \times E(D^{\tau}_j) \tag{26}$$

Finally, we can compute the information gain $\mathcal{G}_{A_{\mathcal{J}}}$ on the chosen attribute $A_{\mathcal{J}}$ of each data provider DP_i as follows.

$$\mathcal{G}_{A_{\mathcal{J}}} = E(D^{\tau}) - E_{A_{\mathcal{J}}}(D^{\tau}) \tag{27}$$

5.3.1.2 Differentially private $\mathcal{G}_{A_{\mathcal{J}}}$

Given a privacy budget ϵ', the sensitivity of the utility function (Δf) is 1, and a true computed $\mathcal{G}_{A_{\mathcal{J}}}$. We add independently generated noise from the Laplace distribution $\texttt{Lap}(1/\epsilon')$ to a true computed $\mathcal{G}_{A_{\mathcal{J}}}$ to have a differentially private version of Eq. (27).

$$\mathcal{G}'_{A_{\mathcal{J}}} = \mathcal{G}_{A_{\mathcal{J}}} + \texttt{Lap}(1/\epsilon') \tag{28}$$

Algorithm 3 *IEB_Trust*

Key Setup: CS and DP_i derive n symmetric keys by mutual authentication protocol
Input: Data consumer attributes request $ReqA_1, \ldots, ReqA_m$, privacy budget ϵ
Input: Data provider DP_n's attributes $A_1, \ldots, A_d$
Output: Accepted DP_n

$DP_1, \ldots, DP_n$ own private data tables $D_1, \ldots, D_n \; \forall i \in I$, where $I = 1, \ldots, n$;
Each DP_i holds set of attributes $\mathcal{PA}_i = \{A_1, \ldots, A_d\}$, over a domain of attributes request $ReqA = \{ReqA_1, \ldots, ReqA_m\}$;
$TS_{DP_i} \leftarrow 0$; /* Initially, trust score is set to 0 for each data provider */
$s_t \leftarrow 0$; /* Initially, arrival sequence is set to 0 for all data providers' attributes */
$\epsilon' \leftarrow \frac{\epsilon}{|ReqA|}$;
while $\exists ReqA_x \in ReqA$ **do**
  for $i \in I$ **do**
    if $\exists ReqA_x \in \mathcal{PA}_i$ **then**
      register arrival sequence s_t on each attribute;
    end if
  end for
end while
Round 1
while $\exists ReqA_x \in ReqA$ **do**
  CS randomly picks $ReqA_x{}'$ over a range of $ReqA_1, \ldots, ReqA_m$ without replacement;
  CS sends challenge $E(ks_i, ReqA_x{}')$ to each DP_i where $\exists ReqA_x{}' \in \mathcal{PA}_i$;
  Each DP_i computes $\mathcal{G}^{(1)}_{A_\mathcal{J}}$ according to Eq. (27) and then adds $\texttt{Lap}(1/\epsilon')$, to have $\mathcal{G}^{(1)'}_{A_\mathcal{J}}$;
  Each DP_i encrypts the message $\psi^{(1)} \leftarrow E(ks_i, \mathcal{G}^{(1)'}_{A_\mathcal{J}})$ and then computes tag $\Upsilon^{(1)} \leftarrow \mathcal{S}(k_h, \psi^{(1)})$;
  CS receives $\psi^{(1)} \| \Upsilon^{(1)} \| DP_i$ on his challenge from the corresponding data providers;
  CS computes comparison to determine $Maj^{R(1)}_{Cand} \leftarrow$ findMajCand($\psi^{(1)} \| \Upsilon^{(1)}$, $size$);
end while
Round 2
while $\exists ReqA_x \in ReqA$ **do**
  for $\ell = 1$ to α **do**
    CS generates $\mathcal{K}$ random IDs for $ReqA_x{}'$ (pick in Round 1) from $|D_i|$ records, where $5 \leq \mathcal{K} \leq 10$;
    CS generates $\mathcal{P}$ pairs of values for $ReqA_x{}'$ and A^{cls} attributes, where $5 \leq \mathcal{P} \leq 10$;
    CS sends challenge $E(ks_i, \mathcal{K}, ReqA_x{}') \| E(ks_i, v_{x'}, v_{cls})$ to each DP_i where $\exists ReqA_x{}' \in \mathcal{PA}_i$;
    Each DP_i computes $\mathcal{G}^{(2)}_{A_\mathcal{J}}$ on the concatenated $\mathcal{K}$ specified records and $\mathcal{P}$ pairs of values and then adds $\texttt{Lap}(1/\epsilon')$, to have $\mathcal{G}^{(2)'}_{A_\mathcal{J}}$;
    Each DP_i encrypts the message $\psi^{(2)} \leftarrow E(ks_i, \mathcal{G}^{(2)'}_{A_\mathcal{J}})$ and then computes tag $\Upsilon^{(2)} \leftarrow \mathcal{S}(k_h, \psi^{(2)})$;
    CS receives $\psi^{(2)} \| \Upsilon^{(2)} \| DP_i$ on his challenge from the corresponding data providers;
    CS computes comparison to determine $Maj^{R(2)}_{Cand_\ell} \leftarrow$ findMajCand($\psi^{(2)} \| \Upsilon^{(2)}$, $size$);
  end for
  CS computes $\bigcap_{\ell=1}^{\alpha} Maj^{R(2)}_{Cand_\ell}$;
end while
CS computes $Maj^{R(1)}_{Cand} \bigcap Maj^{R(2)}_{Cand}$ to determine Maj_{Cand};
for all $Cand \in Maj_{Cand}$ **do**
  set $Cand$ as $Qual_{DP_i}$;
  $TS_{DP_i} = TS_{DP_i} + \gamma$;
end for
for all $Cand \notin Maj_{Cand}$ **do**
  set $Cand$ as $UnQual_{DP_i}$;
  $TS_{DP_i} = TS_{DP_i} - \gamma$;
end for
if $size == 1 \wedge TS_{DP_i} \geq 0$ **then**
  set DP_i as $Qual_{DP_i}$;
end if
Pick one $Cand$ by comparison on the arrival sequences of the $Qual_{DP_n}$ on each attribute;
return Data providers whose final aggregated trust score ≥ 0

5.3.1.3 Discretization

We use *equal-width* method to discretize a continuous-valued attribute $A_{\mathcal{J}}$ into K intervals of equal size. The min_{val} and max_{val} parameters are used for defining the boundaries of the range, whereas arity K is used to determine the number of bins. Each bin is associated with a distinct discrete value. The width of interval is computed by

$$Int_{width} = \frac{max_{val} - min_{val}}{K} \tag{29}$$

Example 4. We continue from Example 3. Consider the example data of numerical type attribute in Table 5.2. In this table *Age* is a numerical attribute, whereas *Loan approval* is an A^{cls} attribute. Data providers DP_1 and DP_3 own raw data tables Table 5.2.(a) and Table 5.2.(b), respectively. DP_3 has somewhat different values on the *Age* attribute in contrast to DP_1 on records $\{ID\#1, 3, 4, 8, 9, 11, 12\}$. They discretize their data on the *Age* attribute, as shown in Table 5.2.(c), according to the parameters of equal width binning. A boundary is defined as $min_{val} = 10.0$ and $max_{val} = 70.0$, whereas arity $K = 5$. Though they have differences in their raw data, the produced discrete version is the same for both since the data values occurred in the specified range. Therefore, the computed information gain 0.34573 is also the same. ■

Table 5.2: Example data of numerical type attribute

Data Provider DP_1		
ID	**Age**	**Loan approval**
1	39	N
2	50	N
3	38	N
4	53	N
5	28	N
6	37	N
7	49	N
8	59	N
9	31	Y
10	42	Y
11	37	Y
12	30	Y

Raw data table (a)

Data Provider DP_3		
ID	**Age**	**Loan approval**
1	45	N
2	50	N
3	35	N
4	57	N
5	28	N
6	37	N
7	49	N
8	66	N
9	27	Y
10	42	Y
11	38	Y
12	28	Y

Raw data table (b)

Discretization		
ID	**Age**	**Loan approval**
1	[34.0 - 46.0]	N
2	[46.0 - 58.0]	N
3	[34.0 - 46.0]	N
4	[46.0 - 58.0]	N
5	[22.0 - 34.0]	N
6	[34.0 - 46.0]	N
7	[46.0 - 58.0]	N
8	[58.0 - 70.0]	N
9	[22.0 - 34.0]	Y
10	[34.0 - 46.0]	Y
11	[34.0 - 46.0]	Y
12	[22.0 - 34.0]	Y

Raw data table (c)

Example 5. We continue from Example 3. Consider the raw data tables of two data providers

who own common attribute, e.g., *Sex* (which has two values, M or F) as shown in the compressed Table 5.3. The class attribute *Loan approval* shared between the data providers has two values, Y or N, indicating whether or not the loan is approved. Both DP_1 and DP_2 have the same number of records and the same count on their records, i.e., $M = 8$, and $F = 4$, but they have different information gain $DP_1 = 0.011580$ and $DP_2 = 0.251629$ on the Sex attribute. Since the data providers are not consistent in providing the same information on the common $RecIDs$, this results in a change in the count for class label values. For instance, DP_1 indicates that there is 1 female whose loan is approved, whereas DP_2 indicates 0 females. ■

Table 5.3: Compressed data table for categorical type attribute

Data Provider DP_1		
Sex	**Loan approval**	**#of Recs.**
M	3Y5N	8
F	1Y3N	4
	Total	12

Raw data table (a)

Data Provider DP_2		
Sex	**Loan approval**	**# of Recs.**
M	4Y4N	8
F	0Y4N	4
	Total	12

Raw data table (b)

5.3.1.4 Computation of trust score

Intuitively, the trust score is a metric for assessing the trustworthiness of each data provider. We compute the trust score TS_{DP_i} locally for each data provider in an iterative manner on each attribute $ReqA_x$ from the CS. γ is a user-defined weight. A data provider qualifying on the majority gains a positive γ weight in the trust score. On the other hand, a disqualified data provider is penalized with a negative -γ weight in the trust score. We aggregate on both positive and negative weights at each iteration to determine the final trust score for each data provider.

$$TS_{DP_i} = \sum_{ReqA_x \in ReqA} \gamma \begin{cases} if(Cand \in Maj_{Cand}) & +\gamma \\ if(Cand \notin Maj_{Cand}) & -\gamma \end{cases} \tag{30}$$

5.3.2 Security properties

In this section, we discuss the security properties of our proposed, *IEB_Trust*, algorithm.

5.3.2.1 Security against covert adversaries

In the context of our problem, a dishonest data provider is a kind of covert adversary who may arbitrarily provide false data on his attribute $A_{\mathcal{J}} \in \mathcal{PA}_i$. The probability of detecting this cheat by our proposed trust computation algorithm is $1 - \xi$ (refer to the Section 5.3.3.1 for details). Each DP_i who has committed to, when registering, the available attributes $\mathcal{PA}_i = \{A_1, \ldots, A_d\}$ is responsible to answer the CS's challenge request, where $\exists ReqA_x{}' \in \mathcal{PA}_i$. When the CS detects a data provider cheating, the provider is penalized with a negative -γ weight in the trust score.

5.3.2.2 Mutual authentication

Before the verification process, each DP_i and the CS mutually authenticate each other by the TLS 1.2 protocol or higher [45, 140]. It is indispensable for the CS to negotiate on the latest stable version of the TLS protocol and stronger cipher suite to prevent against different forms of deception. After successful authentication of each DP_i, they are granted access to the resource queue, where they can register their data attributes.

5.3.2.3 Minimal access for outsourcing verification

The data providers who own customers' private data outsource the verification on their data to the CS. Each DP_i computes locally the information gain function $\mathcal{G}$ on an available attribute $A_{\mathcal{J}} \in \mathcal{PA}_i$, whereas the CS can have access to only an encrypted $\mathcal{G}'_{A_{\mathcal{J}}}$ message, i.e., ψ, and its keyed hash, i.e., Υ for the verification. It benefits the data providers to restrict the CS from accessing the customers' private data. Since encrypted individual data records are not exchanged during the verification, the overhead of computation on the CS is also reduced.

5.3.2.4 Authentication and integrity

HMAC enforces integrity and authenticity. It depends on what underlying hashing function has been used. There are some collision-related vulnerabilities of MD5; however, HMAC-MD5 is not as affected by those vulnerabilities. Regardless, SHA-2 is cryptographically stronger than MD5 and SHA-1. HMAC is constructed by using two nested keys, say k_{in} and k_{out}. These nested keys are not

independent; instead they are derived from a single k_h. Let $\mathbb{M}$ bytes be assumed to be the message blocks for the underlying Merkle-Damgard hash. To derive the keys k_{in} and k_{out}, which are byte strings of length $\mathbb{M}$, we first construct k_h exactly $\mathbb{M}$ bytes long. If the length of $k_h \leq \mathbb{M}$, we pad it out with zero bytes; otherwise, we replace it with $\mathcal{H}(k_h)$ padded with zero bytes. Then we compute

$$k_{in} \leftarrow k_h \oplus ipad$$
$$k_{out} \leftarrow k_h \oplus opad$$

The ipad denotes the *inner* pad and the opad denotes the *outer* pad. These pads are 512 bit constants that never change and are embedded in the implementation of HMAC. HMAC is assumed to be a secure PRF [32]. It provides better protection against length extension attacks. It is built as follows:

$$\mathcal{S}(k_h, \psi) = \mathcal{H}\Big(k_h \oplus opad, \mathcal{H}\Big(k_h \oplus ipad \| \psi\Big)\Big)$$

One of the properties of a cryptographic hash function is that if there is a minor change in an input message, it changes the message digest so extensively that the new message digest appears uncorrelated with the old computed message digest. In our case, we do not apply cryptographic hash functions directly on the input data for data integrity because we allow parties to have minor inaccuracies on numerical attributes for a specified threshold.

5.3.3 Analysis

In this section, we analyze the correctness and security of Algorithm 3.

Proposition 5.3.1. ***(Correctness)*** *Assuming multiple data providers are dishonest, Algorithm 3 correctly computes the trust scores among them, as stated in Problem 1 in Section 5.2.2, to evaluate the trustworthiness of each data provider.*

Proof. Algorithm 3 selects an attribute uniformly at random without replacement from a list $ReqA = \{ReqA_1, \ldots, ReqA_m\}$ of m requested attributes. Each DP_i computes $\mathcal{G}_{A_\mathcal{J}}$ according to Eq. (27) for its matching attribute in the presence of a shared class attribute A^{cls}. For a continuous-valued attribute, each provider follows equal-width method for discretization into intervals of equal size.

Consider $A_{\mathcal{J}}$ is discrete-valued, owned by two providers, where $\Omega(A_{\mathcal{J}}) = \{v_1, v_2\}$ is in its domain of data values. Assume there is a single record between two providers where they have different values. Algorithm 3 computes $\mathcal{G}_{A_{\mathcal{J}}}^{(1)'}$ in the first round for both the data providers and returns different scores. This suggests that they are not the same.

Now, we consider an extended case where two data providers (say DP_1, DP_2) would have different sets of records but the computation of $\mathcal{G}_{A_{\mathcal{J}}}^{(1)'}$ in the first round on the full dataset for both data providers returns the same score, so we have $Maj_{Cand}^{R(1)} = \{DP_1, DP_2\}$. Algorithm 3 verifies further by running the process α times in the second round. During each iteration data providers have to select records over $\mathcal{K}$ random IDs for $A_{\mathcal{J}}$, and they also have to add $\mathcal{P}$ pairs of values $v_{x'}$ and v_{cls} for $A_{\mathcal{J}}$ and a class attribute A^{cls}, respectively, from the CS before computation of $\mathcal{G}_{A_{\mathcal{J}}}^{(2)'}$. Algorithm 3 computes $Maj_{Cand}^{R(1)} \cap (\bigcap_{\ell=1}^{\alpha} Maj_{Cand_\ell}^{R(2)})$ to determine Maj_{Cand}. This determines whether or not the data providers are holding the same data values over the common attribute $A_{\mathcal{J}}$. Data providers are required to match in both the rounds to prove that they have the same score. Since data providers are holding a different set of records, it is not possible for them to match because of the randomness introduced in the second round. □

Proposition 5.3.2. ***(Security)*** *Algorithm 3 is secure against covert adversaries as described in Section 5.3.2.1 by the probabilistic bound of* $1 - \xi$.

Proof. The security of Algorithm 3 depends on the keys derivation in the mutual authentication protocol and the communication of the cloud server CS and data providers DP_n in the verification process.

- A random challenge $E(ks_i, ReqA_x{}')$ is secure because of symmetric keys derivation by [45, 140].
- On a given challenge request, if $\exists ReqA_x{}' \in \mathcal{PA}_i$, each data provider first computes the information gain function on its matching attribute $\mathcal{G}_{A_{\mathcal{J}}} \in \mathcal{PA}_i$, and then perturbs the output by adding noise. This returns a noisy score $\mathcal{G}'_{A_{\mathcal{J}}}$ for which data providers should agree on the scale for digits after the decimal point. It is secured for privacy protection because each DP_i only exchanges an encrypted $\mathcal{G}'_{A_{\mathcal{J}}}$ message, i.e., ψ, and its keyed hash, i.e., Υ, with the CS in

both rounds of the protocol, instead of exchanging encrypted individual data records on their attributes $A_{\mathcal{J}}$.

- Keyed hash-based message authentication code $\mathcal{S}(k_h, \psi)$ is a secure PRF according to [32]. It is computationally infeasible for an adversary to find distinct inputs ψ_1, ψ_2 such that $\mathcal{S}(k_h, \psi_1) = \mathcal{S}(k_h, \psi_2)$.

- Dishonest data providers cannot modify the outputs, i.e., $\psi \| \Upsilon$, of the honest providers in any round of the protocol. They may compute $\mathcal{G}^*_{A_{\mathcal{J}}}$ on their false data and can send their $\psi^* \| \Upsilon^*$ to the CS. The CS computes a comparison and detects cheating from a dishonest data provider with the probability of $1 - \xi$.

□

5.3.3.1 Adversary's inferences

In the following, we estimate the probability of an adversary, i.e., a dishonest data provider, to correctly guess $\mathcal{G}^*_{A_{\mathcal{J}}}$ on a random challenge attribute $ReqA'_x$. An adversary knows $|D^\tau|$, the number of records in D^τ, and $|A^{cls}_{i,D^\tau}|$, the number of records of class A^{cls}_i in D^τ, and computes the entropy of D^τ by Eq. (25). Next, the adversary may try to compute entropy on $A_{\mathcal{J}}$ by the following equation because he knows $|\Omega(A_{\mathcal{J}})|$, the domain size of $A_{\mathcal{J}}$, and $|D^\tau|$, the number of records in D^τ.

$$E^*_{A_{\mathcal{J}}}(D^\tau) = \sum_{j'=1}^{\mathcal{V}'} \frac{|D^\tau_{j'}|}{|D^\tau|} \times - \sum_{i=1}^{\mathcal{C}} \frac{|A^{cls}_{i,D^\tau_{j'}}|}{|D^\tau_{j'}|} \times log_2 \frac{|A^{cls}_{i,D^\tau_{j'}}|}{|D^\tau_{j'}|} \tag{31}$$

There are $|\Omega(A_{\mathcal{J}})|^{|D^\tau|}$ possible arrangements in which an adversary may try to compute $E^*_{A_{\mathcal{J}}}(D^\tau)$. Finally, he computes $\mathcal{G}^*_{A_{\mathcal{J}}}$ having all distinct values by the following equation.

$$\mathcal{G}^*_{A_{\mathcal{J}}} = E(D^\tau) - E^*_{A_{\mathcal{J}}}(D^\tau) \tag{32}$$

This results in ϑ distinct values of $\mathcal{G}^*_{A_{\mathcal{J}}}$, with the lower bound of $\vartheta \approx |D^\tau|$. The probability of

correctly guessing $\mathcal{G}^*_{A_\mathcal{J}}$ for an adversary in our verification process is

$$\xi = \frac{1}{\vartheta} \times (\frac{1}{\vartheta})^{\alpha} \tag{33}$$

5.3.3.2 Detecting cheat against varying dishonest providers

Let n denote the number of participating data providers, and let b denote an upper bound on the number of dishonest data providers who may arbitrarily provide incorrect data in responding to the CS's challenge.

- When $b < n/2$, the verification process guarantees fairness and no honest data providers are negatively affected by their trust levels.

- When $b \leq n - 2$, the verification process guarantees fairness under the arbitrary behavior of dishonest data providers, where the chance of detecting them is $1 - \xi$. It is a type of covert adversarial behavior when the dishonest data providers arbitrarily provide false data on their data inputs, i.e., they neither would be able to appear in the majority nor would be able to undermine the reputations of the honest data providers.

- When $b > n/2$, the verification process does not guarantee fairness on the flip side, i.e., when the behavior of dishonest data providers is not arbitrary. This would be the case when the dishonest data providers not only appear in the majority but also organize in a way to undermine the reputation of the honest data providers. We assume that if a secure set intersection is carried out by using a trusted mediator (e.g., by computing the function on the data providers input) between data providers, then the dishonest providers would not be able to determine the total number of participating data providers in advance. This would restrict them from developing the organized group; still, there is no remedy if they would try by guessing at random.

5.3.4 Evaluation of learner models

We provide an example of a sample data to evaluate the quality of linear regression, k-nearest neighbors (kNN), and random forest learner models.

Example 6. We retrieve the top 1000 records from a real-life *Adult*[1] dataset on attributes *age, education-num, race, sex, income*. The attributes *age, education-num* are of continuous types, whereas *race, sex, income* are of categorical types. We develop learner models in *RapidMiner*[2] to compare the predictive accuracy of linear regression, k-nearest neighbors (kNN), and random forest methods.

For the linear regression model, we set *education-num* as a label, which is considered as a dependent attribute (or variable), and the remaining are considered as independent attributes. We convert non-numeric type attributes to the numeric type. After running 10-fold cross-validation, the *Root Mean Square Error (RMSE)* is found to be 2.438 ± 0.165, which indicates the standard deviation of the residuals. Furthermore, R^2 is found to be 0.127 ± 0.055, which indicates the goodness of fit of this regression model. Its value is close to 0, indicating a weak linear correlation.

For the k-nearest neighbors (kNN) model, we set all attributes as nominal and *education-num* as a label. After running 10-fold cross-validation when $k = 20$, the *accuracy* is found to be $33.90\% \pm 5.59\%$, which indicates the percentage of correct predictions.

For the random forest model, we set the *education-num* attribute as nominal and specify the role as a label. The key parameter 'number of trees' is specified as 10, and the 'gain ratio' is chosen as a criterion for splitting attributes. After running 10-fold cross-validation, the *accuracy* is found to be $32.90\% \pm 0.30\%$, which indicates the percentage of correct predictions. ■

There are no significant performance differences found on running these learner models on the sample dataset. Data providers would use any one or multiple learning methods for missing data imputation.

5.3.5 Price setting using auction mechanism

An auction mechanism can be defined in many different ways depending upon the design requirements. The two variants of 2nd price sealed-bid auctions [51] have widely used, namely Vickrey-Clarke-Groves (VCG) and Generalized Second Price (GSP) mechanisms for multiple items.

The reason for employing the VCG mechanism for determining the pricing on data providers'

[1] Available at: http://archive.ics.uci.edu/ml/datasets/Adult
[2] Available at: https://rapidminer.com/products/studio/

attributes is that truthful bidding is a dominant strategy, and there is no incentive to lie or deviate from reporting true valuations for a data provider. It maximizes the total valuation obtained by data providers. One nice property of the VCG mechanism is that it provides a unique outcome, which is socially optimal, whereas, in the GSP there would be multiple outcomes in terms of Nash equilibrium. One Nash equilibrium would maximize social welfare but not all of them.

We intend to design an auction mechanism for multiple items. It is assumed that the data providers intend to set up a matching market using a 2nd price sealed-bid auction for valuation of their attributes. We formally define the procedure for setting the price as follows:

5.3.5.1 Data providers

Let $DP_1, \ldots, DP_n$ (where $i = 1, \ldots, n$) be the set of data providers who set up a matching market for valuations of their attributes.

5.3.5.2 Positions

Let $P_1, \ldots, P_n$ (where $j = 1, \ldots, n$) be the set of positions for which data providers compete. The higher the position P_j, the more will be its demand rate. The positions should be equal to the number of data providers. If there are more data providers than positions, we simply add fictitious positions of demand rate 0. Similarly, if there are more positions than data providers, we add fictitious data providers of revenue per demand 0.

5.3.5.3 Revenue per demand

Revenue per demand is the expected amount of money that a data provider DP_i receives, denoted by Rev_i, for every demand on its attribute. The monetary values of Rev_i are sorted in descending order.

5.3.5.4 Demand rate

Demand rate is defined as the number of demands requested by a consumer over a period of time, denoted by Q_j. Demand rate varies as per the position P_j. Q_j enumerates in descending order.

5.3.5.5 Data providers' valuations

Data providers' valuations are defined as the data provider DP_i's valuation of the position P_j. It is the product of the revenue per demand Rev_i and the demand rate Q_j, denoted by $Val_{i,j}$. It is computed as follows:

$$Val_{i,j} = Rev_i \times Q_j \tag{34}$$

5.3.5.6 VCG price

VCG price is defined as the harm or externality caused by data provider DP_i to other data providers in terms of reduction of their valuations due to his presence. It is called VCG price, denoted by $ExPrc_{i,j}$, which is paid by data provider DP_i for position P_j. Formally, it is defined by

$$ExPrc_{i,j} = \bigvee_{DP_n - DP_i}^{P_n} - \bigvee_{DP_n - DP_i}^{P_n - P_j} \tag{35}$$

where

- $DP_n - DP_i$ is the set of data providers excluding data provider DP_i;
- $P_n - P_j$ is the set of positions excluding position P_j;
- $\bigvee_{DP_n - DP_i}^{P_n}$ is the sum of data provider values of an optimal matching between sets $DP_n - DP_i$ and P_n; and
- $\bigvee_{DP_n - DP_i}^{P_n - P_j}$ is the sum of data provider values of an optimal matching between sets $DP_n - DP_i$ and $P_n - P_j$.

5.3.5.7 Data providers' valuations after payoff

Data providers' valuations after payoff is defined as the data provider DP_i's valuation on position P_j after paying off harm to other data providers. It is calculated using the following equation.

$$Val_{DP_i} = \max Val_{i,j} - ExPrc_{i,j} \tag{36}$$

5.3.5.8 Valuation of an attribute

Valuation of an attribute can be assessed once a data provider DP_i acquires a certain position P_j. The value of each data provider's attribute per single demand is calculated using the following equation.

$$ValAttr_{DP_i} = \frac{Val_{DP_i}}{Q_j} \tag{37}$$

5.3.5.9 Attribute count

The attribute count $CntAttr_{DP_i}$ of a data provider DP_i represents the number of attributes in a single record. Each DP_i owns a mutually exclusive set of attributes.

5.3.5.10 Price per record

The price per record $PrcRec_{DP_i}$ of a data provider DP_i represents the unit price of a record. Naturally, it is the product of the value per attribute $ValAttr_{DP_i}$ and the attribute count $CntAttr_{DP_i}$ in a single record. That is,

$$PrcRec_{DP_i} = ValAttr_{DP_i} \times CntAttr_{DP_i} \tag{38}$$

5.3.5.11 Size of dataset

The dataset of each data provider DP_i consists of a collection of records, denoted by $|D_i|$. The size of a dataset grows as the number of records in the dataset increases.

5.3.5.12 Price of raw dataset

The price of a raw dataset $PrcRawDS_{DP_i}$ represents the data provider DP_i's selling price of a raw dataset in the e-market. The overall pricing of a raw dataset increases as the number of records or the unit *price per record* increases. It is computed as follows:

$$PrcRawDS_{DP_i} = |D_i| \times PrcRec_{DP_i} \tag{39}$$

5.3.5.13 Total price of raw dataset

The total price of the raw dataset $TPrc_{RawDS}$ is the sum of the pricing of all the contributing data providers' raw datasets. It is computed as follows:

$$TPrc_{RawDS} = \sum_{i=1}^{n} PrcRawDS_{DP_i} \tag{40}$$

5.3.5.14 Total monetary value of raw dataset

First, data providers compute baseline accuracy (BA) for classification analysis using the secure multiple party classifier [49] by maintaining the confidentiality of their raw data. Then they use the information utility of classifying raw data to derive the monetary value of the raw dataset, denoted by $TMValue_{RawDS}$. It is calculated using the following equation:

$$TMValue_{RawDS} = TPrc_{RawDS} \times BA \tag{41}$$

5.3.6 Anonymization method

In this section, we provide an extension of the two-party *Diff*erentially private anonymization in Algorithm 4, which is based on *Gen*eralization [128] to differentially integrate multiple private data tables. This algorithm guarantees ϵ-differential privacy and security definition under the semi-honest adversary model (readers may refer to the detailed analysis in [128], Section 6.3). The two major extensions over the TDS algorithm [59] include: (1) *DistDiffGen* selects the $Best$ specialization based on the exponential mechanism, and (2) *DistDiffGen* perturbs the generalized contingency table by adding the Laplacian noise to the count of each equivalence group.

Generally, there is no incentive for any data provider who executes the algorithm as the purpose is merely to synchronize the anonymization process. We assume a trusted data provider, who attains the highest trust score after running the Algorithm 3, starts the anonymization process. The accepted data providers, as a result of trust computation by Algorithm 3, attain a mutually exclusive set of attributes, i.e., $\mathcal{PA}_i \cap \mathcal{PA}_j = \emptyset$ for any $1 \leq i, j \leq n$ over the same set of records for integrating data.

Initially, all values in the set of attributes $\mathcal{PA}_i = \{A_1, \ldots, A_d\}$ of each data provider are generalized to the topmost value in their taxonomy trees (Line 1), as illustrated in Fig. 5.1, and $Mark_\kappa$ contains the topmost value for each attribute $A_\mathcal{J} \in \mathcal{PA}_i$ (Line 2). Each data provider keeps a copy of the $\cup Mark_\kappa$ and a generalized data table D_g. The attribute $A_\mathcal{J}$ can be either categorical or numerical, but the class attribute is required to be categorical. The split value of a categorical attribute v_c is a generalized value drawn from a pre-defined taxonomy tree of the attribute, whereas the split value of a numerical attribute v_{num} is determined by using the exponential mechanism (Line 4). It partitions the domain range of a numerical attribute into successive intervals $\mathcal{I}_1, \ldots, \mathcal{I}_k$. Line 4 preserves $\epsilon'|A_{num}|$-differential privacy since the cost of each exponential mechanism is ϵ'. In Line 5, a score $IGScore$ is computed for all candidates $v \in \cup Mark_\kappa$. At each iteration, the algorithm uses the secure distributed exponential mechanism (DistExp) as presented in [128] (readers may refer to the details of the DistExp algorithm) to select a winner candidate $w \in \cup Mark_\kappa$ for specialization (Line 7). Different utility functions (e.g., information gain) can be used to calculate the score. If the winner candidate w is local to DP_i, DP_i specializes w on D_g by splitting its records into child partitions, updates its local copy of $\cup Mark_\kappa$, and instructs all the other participating data providers to specialize and update their local copy of $\cup Mark_\kappa$ (Line 8-11). The information gain, denoted by $\widetilde{\mathcal{G}}_{DP_i}$, accumulates $IGScore(x)$ on the winner's attribute specializations (Line 12). DP_i further calculates the scores of the new candidates as a result of the specialization (Line 14). If the winner w is not one of DP_i's candidates, DP_i waits for instructions from the other winner data provider DP_j, where $i \neq j$, to specialize w and to update its local copy of $\cup Mark_\kappa$ (Lines 16 and 17). This process iterates until the specified number of the specializations h is reached. The algorithm perturbs the output by adding the noisy count at each leaf node (Line 21) using the Laplace mechanism. The contribution of each data provider is computed according to Eq. (45). Finally, the monetary share of each data provider is derived according to the Eq. (46).

5.3.7 Quantifying the monetary value

The rationality of quantifying the monetary value is that data providers are the business stakeholders who collaborate in the data integration process to maximize their profits. The profit generated by their collaboration is distributed based on each provider's contribution to information utility and

Algorithm 4 Monetary Shares for Data Providers using DistDiffGen

Input: Data providers' attributes valuations $ValAttr_{DP_n}$
Input: Private data tables $D_1, \ldots, D_n$, privacy budget ϵ, and number of specializations h
Output: Monetary shares $MShare_{DP_n}$

Initialize D_g with one record containing topmost generalized values in each data provider's taxonomy tree;
Initialize $Mark_\kappa$ to include the topmost value;
$\epsilon' \leftarrow \frac{\epsilon}{2(|A_{num}|+2h)}$;
Determine the split value for each $v_{num} \in \cup Mark_\kappa$ with probability $\propto \exp(\frac{\epsilon'}{2\Delta \mathrm{u}} \mathrm{u}(D, v_{num}))$;
Compute the $IGScore$ for $\forall v \in \cup Mark_\kappa$;
for $iter$ = 1 to h **do**
  Determine the winner candidate w by using the DistExp Algorithm [128];
  if w is local **then**
    Specialize w on D_g;
    Replace w with $child(w)$ in the local copy of $\cup Mark_\kappa$;
    Instruct all the other participating data providers to specialize and update $\cup Mark_\kappa$;
    $\widetilde{\mathcal{G}}_{DP_i} = \widetilde{\mathcal{G}}_{DP_i} + IGScore(x)$;
    Determine the split value for each new $v_{num} \in \cup Mark_\kappa$ with probability $\propto \exp(\frac{\epsilon'}{2\Delta \mathrm{u}} \mathrm{u}(D, v_{num}))$;
    Compute the $IGScore$ for each new $v \in \cup Mark_\kappa$;
  else
    Wait for the instruction from the winner data provider;
    Specialize w and update $\cup Mark_\kappa$ using the instruction;
    $\widetilde{\mathcal{G}}_{DP_j} = \widetilde{\mathcal{G}}_{DP_j} + IGScore(x)$;
  end if
end for
Compute count $(CT + \mathrm{Lap}(2/\epsilon))$ for each leaf node;
Compute the contribution of each data provider according to Eq. (45);
Compute monetary share of each data provider according to Eq. (46);
return $MShare_{DP_n}$

its trustworthiness.

5.3.7.1 Cost of anonymization in integrated data

First, the data providers compute classification accuracy (CA) on the anonymized integrated data. Then, they quantify the cost of anonymization in integrated data, denoted by $Cost_{IntDS}$, on the difference between the baseline accuracy (BA) and the classification accuracy (CA). It is computed as follows:

$$Cost_{IntDS} = TPrc_{RawDS} \times (BA - CA) \tag{42}$$

5.3.7.2 Expected value in integrated data

An expected monetary value in integrated data is what the data providers earn from the information utility of classification analysis when trading an anonymized version of integrated data.

The information utility varies with the valuations of data providers' attributes and joint privacy requirements, such as privacy budget ϵ and specialization level h, for a ϵ-*differential privacy* model in a distributed setup, between the data providers. It is calculated on the difference between the total monetary value of the raw dataset $TMValue_{RawDS}$ and the cost of anonymization in integrated data $Cost_{IntDS}$. It is computed as follows:

$$EValue_{IntDS} = TMValue_{RawDS} - Cost_{IntDS} \tag{43}$$

5.3.7.3 Expected value of an individual data provider

The expected monetary value of an individual data provider, denoted by $EValueIndv_{DP_i}$, is determined by the ratio of the number of attributes $CntAttr_{DP_i}$ a data provider owns with the total count of attributes. It is computed as follows:

$$EValueIndv_{DP_i} = EValue_{IntDS} \times \frac{CntAttr_{DP_i}}{\sum_{i=1}^{n} CntAttr_{DP_i}} \tag{44}$$

5.3.7.4 Derivation of monetary share

The derivation of a monetary share depends upon the contribution of each data provider and its trustworthiness. Intuitively, a data provider whose provided data on his attributes result in more information gain, and whose trust level is higher than the other competitors, can get a proportionally larger share of the monetary value. The contribution of each data provider DP_i is derived from the expected monetary value $EValueIndv_{DP_i}$ by fairly computing first the accumulative information gain $\widetilde{\mathcal{G}}_{DP_i}$ of each data provider DP_i on the anonymized integrated dataset. The information gain $IGScore(x)$ of the winner candidate w data provider accumulates under the relevant winner w data provider at each iteration (refer to the Section 5.3.6 for details) for the specified specialization level h. The contribution of each data provider $Contrib_{DP_i}$ is calculated using the following equation:

$$Contrib_{DP_i} = \frac{\widetilde{\mathcal{G}}_{DP_i}}{\sum_{i=1}^{n} \widetilde{\mathcal{G}}_{DP_i}} \times EValueIndv_{DP_i} \tag{45}$$

Finally, the monetary share of each data provider $MShare_{DP_i}$ is derived according to Eq. (30),

i.e., the aggregated trust score of each data provider, and Eq. (45), i.e., the contribution of each data provider. Therefore, $MShare_{DP_i}$ becomes:

$$MShare_{DP_i} = Contrib_{DP_i}(1 + \frac{TS_{DP_i}}{\sum_{i=1}^{n} TS_{DP_i}}) \tag{46}$$

5.4 Limitations

In this section, we discuss how our proposed approach different from secure multi-party computation (SMPC) [190] and the limitations of our work.

SMPC is a generic cryptographic primitive that enables multiple parties to jointly compute the intersection of their private data without revealing any additional information to either side [183]. These approaches are suitable for *privacy-preserving data mining* (*PPDM*) [117], in which multiple data custodians compute a function based on their inputs without sharing their data with others. In this chapter, we focus on *privacy-preserving data publishing* (*PPDP*) [61] in a distributed setting, where the data providers wish to integrate their data with peer data providers over a cloud for better information utility. However, the data integration in Algorithm 4 necessitates that under the specified privacy constraints, no data provider should learn any additional information other than necessary information. One of the limitations of this algorithm is that it guarantees security definition under the semi-honest adversary model [68]. A detailed analysis of the algorithm is presented in [128], Section 6.3. It is reasonable to assume that data providers respect the defined protocol as deviating from the protocol has no direct impact in monetary gain. Besides, if cheating gets detected will lead to a loss of reputation or business for them. To resist against malicious adversaries, all subprotocols of the algorithm should be extended to detect deviations from protocols by data providers.

5.5 Comparative analysis and empirical study

In this section, we first provide a comparison of our approach, followed by an empirical study.

5.5.1 Comparative analysis

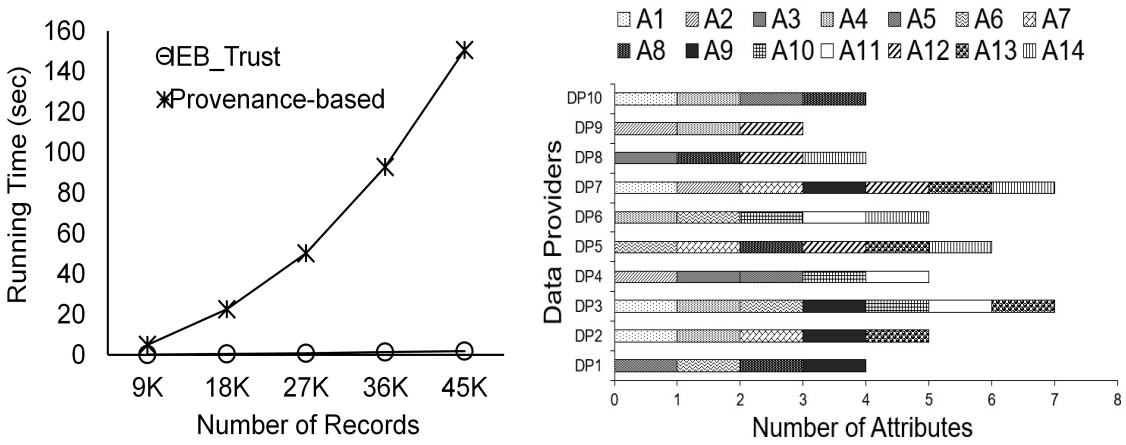
We compare our proposed *IEB_Trust*, an entropy-based trust computation algorithm with the closely related provenance-based trust method [40]. The provenance-based method computes the trust scores for data and data providers using similarity functions, but do not consider privacy protection when evaluating trustworthiness. The fundamental idea of our approach is different. Our method enables secure trustworthiness assessment and preserves the privacy of the customers' data when evaluating the trustworthiness of the participating data providers. For this reason, we are limiting to the runtime comparison in Fig. 5.3a. We evaluate the performance of our proposed method on a real-life *Adult*[3] dataset. It contains $45,222$ records with 8 categorical attributes, 6 numerical attributes, and a binary class attribute $Income$ with two levels, $\leq 50K$ or $> 50K$. The distribution of attributes other than class attribute among 10 data providers is shown in Fig. 5.3b. We generate 10% of data conflicts over randomly chosen attributes. We vary the size of the datasets $|D_i|$ from $10K$ to $50K$ to study the runtime cost. All experiments are conducted on an Intel Core i7 3.4GHz PC with 8GB memory.

The running time includes time elapsed in both the initialization phase and the iteration phase. We observe that the initialization phase of the provenance-based method takes more time to compute data similarity and data conflict. It has worst-case complexity of $\mathcal{O}(n^2)$. While the complexity of our proposed method at the initialization phase is $\mathcal{O}(CntAttr_{DP_i} \cdot |D_i| \log |D_i|)$. Since each data provider computes $\mathcal{G}_{A_\mathcal{J}}$ in a distributed setup, the complexity remains the same in our method. The iteration phase to compute trust is much faster in both the methods. It takes less than one second to complete the trust computation. Fig. 5.3a shows that our method is more efficient in running time over the provenance-based method. Our method is scalable when we need to grow either the number of attributes, the number of data providers, or both on a dataset.

5.5.2 Empirical study

We first analyze the trustworthiness of each data provider and assess the truthfulness of the provided data by a trust score metric. Second, we analyze the impact of ϵ-*differential privacy*

[3] Available at: http://archive.ics.uci.edu/ml/datasets/Adult



(a) Runtime comparison

(b) Distribution of attributes

Figure 5.3: Our method improves the runtime efficiency compared to the provenance-based trust method

requirements along with the aggregated trust score on each data provider's monetary value. We evaluate our proposed method, *IEB_Trust*, with the assumption of having 4 data providers who intend to verify the correctness of their data before participation in the data mashup. This assumption is reasonable because we have a limited number of attributes in the dataset to be shared among data providers.

5.5.2.1 Trust measurement

Our proposed method evaluates the trust of participating data providers based on the following conditions: (1) A data provider is found as honest and gains a positive score; (2) A data provider is found as dishonest and is penalized with a negative score; (3) A single data provider of an attribute that no others own is accepted based on the existing trust score $TS_{DP_i} \geq 0$ without an increase in the trust score; and (4) A data provider who does not register for an attribute has no effects on the trust score.

To demonstrate the effectiveness of our approach, we conduct two cases of experiments that are independent of each other. This means that for each case data providers hold different sets of overlapping attributes with their arrival sequences. In each case, we assume $\gamma = 0.5$, but it does not need to be fixed to a specific weight.

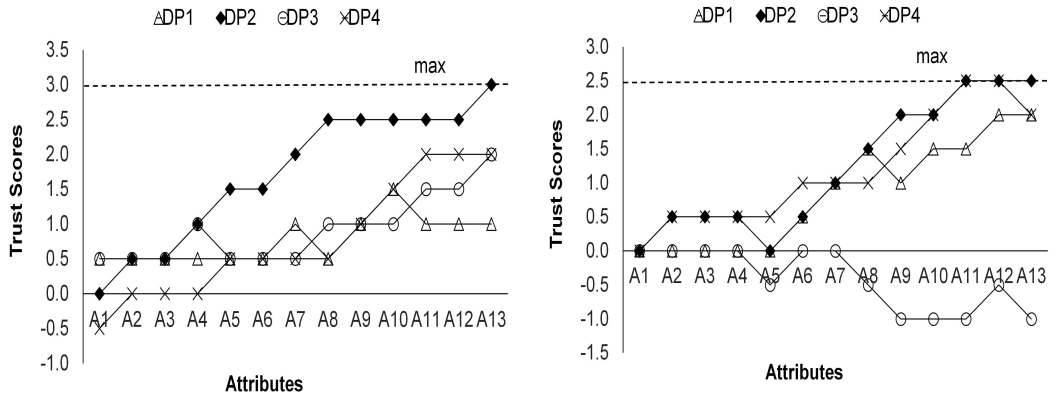
Consider the first case with the participating data providers' attributes and their arrival sequences.

$DP_1 \mapsto A_1{:}s_{t_1}$, $A_7{:}s_{t_1}$, $A_8{:}s_{t_1}$, $A_9{:}s_{t_1}$, $A_{10}{:}s_{t_2}$, $A_{11}{:}s_{t_1}$; $DP_2 \mapsto A_2{:}s_{t_2}$, $A_3{:}s_{t_1}$, $A_4{:}s_{t_1}$, $A_5{:}s_{t_2}$, $A_7{:}s_{t_2}$, $A_8{:}s_{t_3}$, $A_{13}{:}s_{t_1}$; $DP_3 \mapsto A_1{:}s_{t_2}$, $A_4{:}s_{t_2}$, $A_5{:}s_{t_1}$, $A_6{:}s_{t_1}$, $A_8{:}s_{t_2}$, $A_{11}{:}s_{t_2}$, $A_{13}{:}s_{t_2}$; and $DP_4 \mapsto A_1{:}s_{t_3}$, $A_2{:}s_{t_1}$, $A_5{:}s_{t_3}$, $A_9{:}s_{t_2}$, $A_{10}{:}s_{t_1}$, $A_{11}{:}s_{t_3}$, $A_{12}{:}s_{t_1}$. Fig. 5.4a depicts the trust scores analysis for Case 1 based on the demand of a data consumer on attributes $A_1, \ldots, A_{13}$.

It is observed that the DP_2 trust score never drops during the verification process in contrast to the other competing data providers. The flat lines from A_2 to A_6 at trust score level 0.5, and A_9 to A_{12} at trust score level 2.5, indicate that those attributes are not submitted by DP_1 and DP_2, respectively. This is not always the case; for instance, there are flat lines from A_2 to A_3 at trust score level 0.5, A_5 to A_6 at trust score level 0.5, and A_{11} to A_{12} at trust score level 2.0, indicating that DP_2, DP_3, and DP_4 are the single data providers on those attributes. DP_2, DP_3, and DP_4 are accepted because they are maintaining an aggregated trust score ≥ 0 at that point of the verification. However, their trust scores do not increase because they own an attribute that no others own. It is assumed that DP_1 has 5% of missing data on A_8 and A_{11}, DP_3 has 5% of missing data on A_5, and DP_4 has 1% of missing data on A_1. They impute missing data by using the kNN imputation method in order to claim it as original data. Our trust verification approach restricts this dishonest behavior of data providers; for instance, DP_1 at A_8 and A_{11}, DP_3 at A_5, and DP_4 at A_1, by penalizing them with negative weight in their trust scores. Fig. 5.5a depicts the aggregated trust scores for Case 1. DP_2 attains the maximum trust score 3.0 in competing with the other data providers, whereas DP_1 ends up with the minimum trust score 1.0. There is a tie on aggregated trust scores between DP_3 and DP_4.

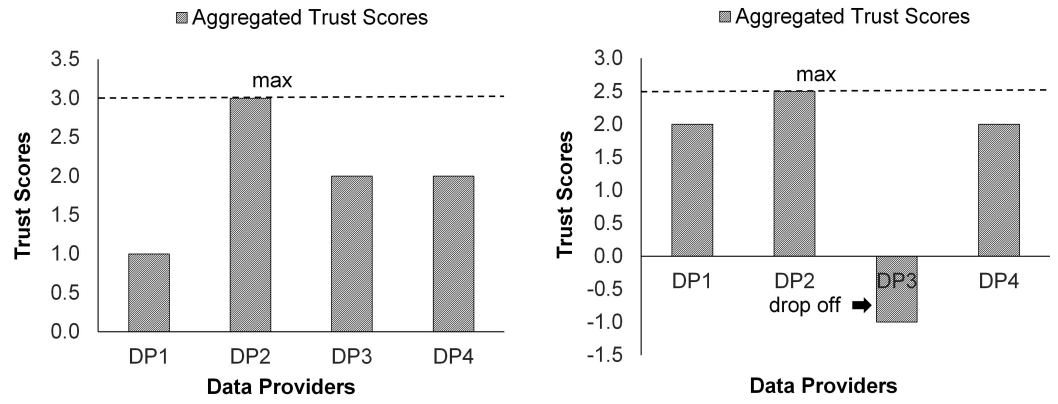
Consider the second case with the participating data providers' attributes and their arrival sequences. $DP_1 \mapsto A_1{:}s_{t_1}$, $A_6{:}s_{t_3}$, $A_7{:}s_{t_1}$, $A_8{:}s_{t_2}$, $A_9{:}s_{t_3}$, $A_{10}{:}s_{t_2}$, $A_{12}{:}s_{t_2}$; $DP_2 \mapsto A_2{:}s_{t_2}$, $A_5{:}s_{t_2}$, $A_6{:}s_{t_4}$, $A_7{:}s_{t_2}$, $A_8{:}s_{t_1}$, $A_9{:}s_{t_2}$, $A_{11}{:}s_{t_1}$; $DP_3 \mapsto A_3{:}s_{t_1}$, $A_5{:}s_{t_1}$, $A_6{:}s_{t_1}$, $A_8{:}s_{t_3}$, $A_9{:}s_{t_1}$, $A_{12}{:}s_{t_1}$, $A_{13}{:}s_{t_2}$; and $DP_4 \mapsto A_2{:}s_{t_1}$, $A_4{:}s_{t_1}$, $A_6{:}s_{t_2}$, $A_9{:}s_{t_4}$, $A_{10}{:}s_{t_1}$, $A_{11}{:}s_{t_2}$, $A_{13}{:}s_{t_1}$. Fig. 5.4b depicts the trust scores analysis for Case 2 based on the demand of a data consumer on attributes $A_1, \ldots, A_{13}$.

It is observed that DP_1, DP_2, and DP_4 maintain their trust scores quite well except for a fall of 0.5 in their trust scores at A_9, A_5, and A_{13}, respectively. The flat lines from A_1 to A_5 at trust score level 0.0, and A_3 to A_5 at trust score level 0.5, indicate that those attributes are not submitted by



(a) *Case 1* (b) *Case 2*

Figure 5.4: Trust scores analysis



(a) *Case 1* (b) *Case 2*

Figure 5.5: Aggregated trust scores

DP_1 and DP_4, except at A_1 and A_4, respectively. Since DP_1 and DP_4 are the single data providers on A_1 and A_4, their trust scores do not increase. However, they are accepted because they maintain an aggregated trust score ≥ 0. We observe that DP_3 is inconsistent in maintaining its trust level throughout the verification process. It is worthwhile to note that our trust verification process restricts the arbitrary behavior of dishonest DP_1 and DP_3 to undermine the trust levels of DP_2 and DP_4. Fig. 5.5b depicts the aggregated trust scores for Case 2. DP_2 attains the maximum trust score 2.5 in competing with the other data providers, whereas DP_3 ends up with a negative trust score of -1.0. This results in the rejection of DP_3 from the final selection in the data mashup.

5.5.2.2 Impact of privacy protection and trust score on DP's monetary value

In this section, we analyze the impact of ϵ-*differential privacy* requirements along with the aggregated trust score on each data provider's monetary value. Recall from Section 5.3.5 that both revenue per demand Rev_i and demand rate Q_j are enumerated in descending order. Suppose $Rev_i = \{\$0.6, \$0.5, \$0.4, \$0.3\}$ and $Q_j = \{9, 8, 7, 6\}$ for data providers DP_1, DP_2, DP_3, and DP_4, respectively. The inputs for Rev_i and Q_j do not need to be fixed to a particular value, it is just assumed here for simplicity.

Case 1 Table 5.4.(a) shows the selection of attributes from each accepted data provider. Baseline accuracy (BA) on the integrated data of accepted data providers is 85.3% using the secure multiple party classifier [49] without disclosing their raw data. We vertically partition the *Adult* dataset into four partitions VP_1, VP_2, VP_3, and VP_4 for data providers DP_1, DP_2, DP_3, and DP_4, respectively. Further, we split the dataset into $30,162$, and $15,060$ records for the training and testing set, respectively. The valuation of each data provider's attribute is \$0.47, \$0.41, \$0.36, and \$0.30, representing $ValAttr_{DP_1}$, $ValAttr_{DP_2}$, $ValAttr_{DP_3}$, and $ValAttr_{DP_4}$ by Eq. (37). The attribute count of each data provider is $CntAttr_{DP_1} = 3$, $CntAttr_{DP_2} = 4$, $CntAttr_{DP_3} = 3$, and $CntAttr_{DP_4} = 3$. The size of the dataset for each data provider $|D_i| = 45,222$.

Table 5.4: Selection of attributes from data providers

DP_1	DP_2	DP_3	DP_4
A_1	A_5	A_8	A_2
A_9	A_4	A_6	A_{12}
A_7	A_{13}	A_{11}	A_{10}
	A_3		

(a) Case 1

DP_1	DP_2	DP_4
A_1	A_8	A_2
A_7	A_9	A_4
A_{12}	A_{11}	A_{10}
		A_6

(b) Case 2

Fig. 5.6 depicts the impact of privacy protection and trust scores on DP_1, DP_2, DP_3, and DP_4's monetary value. ϵ-*differential privacy* is enforced with privacy parameters $\epsilon = 0.2$, 0.4, 0.6, and 0.8 and specialization levels $3 \leq h \leq 19$.

Fig. 5.6a depicts the impact on DP_1, DP_2, DP_3, and DP_4's monetary value when the threshold is $\epsilon = 0.2$. We observe that DP_4 attains the highest monetary share due to more information utility and its aggregated trust score. When specialization level h increases from 3 to 7 and 11 to 15,

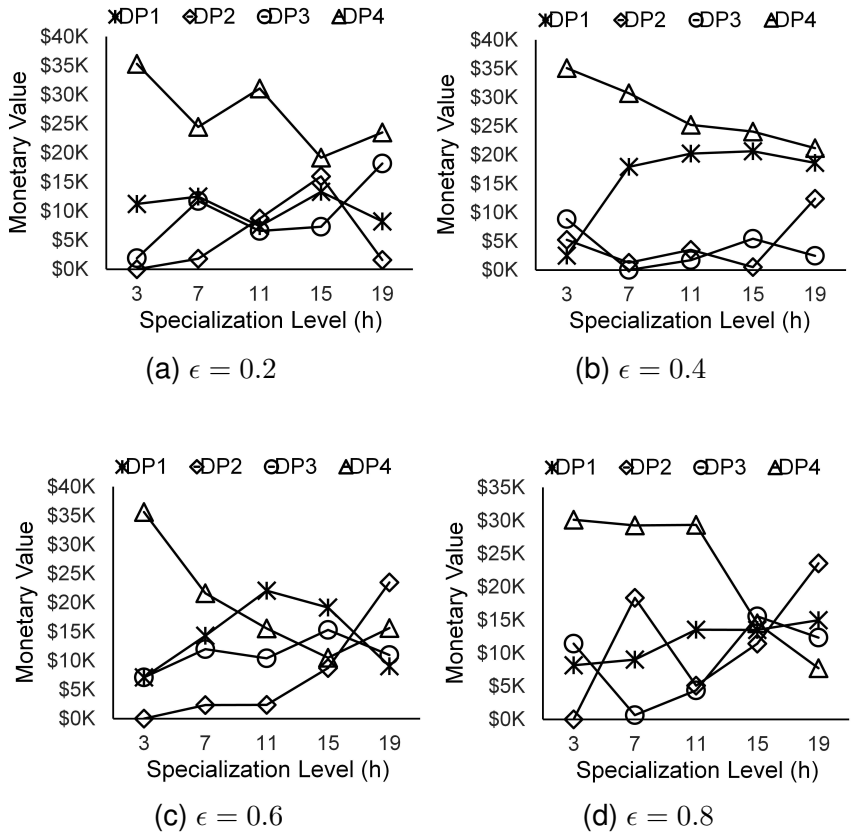


Figure 5.6: Impact of ϵ-*differential privacy* requirements and Trust scores on DP_1, DP_2, DP_3, and DP_4 monetary value (Case 1)

DP_1, DP_2, and DP_3 get increases in their monetary shares, while DP_4's monetary share falls by approximately $\$11K$, though still achieving a higher share than other data providers. Initially, DP_2 has no monetary share when $h = 3$, but it increases with the increase in the specialization level h except when $h = 19$. DP_1, DP_2, and DP_3's monetary shares become closer to each other when $h = 11$.

Fig. 5.6b depicts the impact on DP_1, DP_2, DP_3, and DP_4's monetary value when the threshold is $\epsilon = 0.4$. We observe that DP_4 attains the highest monetary share because of greater information utility and its aggregated trust score. Though DP_1 does not get the highest share, its monetary share becomes closer to DP_4 at $h = 11$, 15, and 19 with the difference of approximately $\$3K$ to $\$5K$. Interestingly, DP_4's monetary share exhibits non-increasing monotonicity with the increase in specialization level h, while DP_1's monetary share increases with the increase in specialization level

h except when $h = 19$. We notice that DP_3 has no monetary share when $h = 7$ because of a lack of information utility for classification analysis. The trust score does not add any monetary value if a data provider fails to contribute to information utility. The trend on DP_2 and DP_3's monetary share is not obvious with the increase in h.

Fig. 5.6c depicts the impact on DP_1, DP_2, DP_3, and DP_4's monetary value when the threshold is $\epsilon = 0.6$. We observe that DP_4 gains the maximum value of monetary share when $h = 3$ and $h = 7$, and DP_1 gains the maximum value of monetary share when $h = 11$ and $h = 15$, whereas DP_2 gains the maximum value of monetary share when $h = 19$. This is because it has greater information utility in competing with the other data providers at the indicated levels of specialization. We observe that DP_2's monetary share increases monotonically as the increase in specialization level h, whereas DP_4's monetary share falls with the increase in specialization level h, except when $h = 19$.

Fig. 5.6d depicts the impact on DP_1, DP_2, DP_3, and DP_4's monetary value when the threshold is $\epsilon = 0.8$. We observe that DP_4 achieves the highest monetary share because of greater information utility and its aggregated trust score. We observe that DP_1's monetary share generally increases as the specialization level h increases, whereas DP_4's monetary share falls with the increase in specialization level h, except when $h = 11$. We notice that when $h = 15$, all data providers' monetary shares become closer, with a difference of approximately $\$4K$.

Case 2 Table 5.4.(b) shows the selection of attributes from each accepted data provider. Baseline accuracy (BA) on the integrated data of accepted data providers is 85.4%, using the secure multiple party classifier [49] without disclosing their raw data. We vertically partition the *Adult* dataset into three partitions VP_1, VP_2, and VP_3 for data providers DP_1, DP_2, and DP_4, respectively. Further, we split the dataset into $30,162$, and $15,060$ records for the training and testing set, respectively. Since DP_3 has dropped from the list of accepted data providers, DP_4 acquires the position of DP_3. Now, the valuation of each data provider's attribute is \$0.47, \$0.41, and \$0.36, representing $ValAttr_{DP_1}$, $ValAttr_{DP_2}$, and $ValAttr_{DP_4}$ by Eq. (37). The attribute count of each data provider is $CntAttr_{DP_1} = 3$, $CntAttr_{DP_2} = 3$, and $CntAttr_{DP_4} = 4$. The size of dataset for each data provider $|D_i| = 45,222$.

Fig. 5.7 depicts the impact of privacy protection and trust scores on DP_1, DP_2, and DP_4's

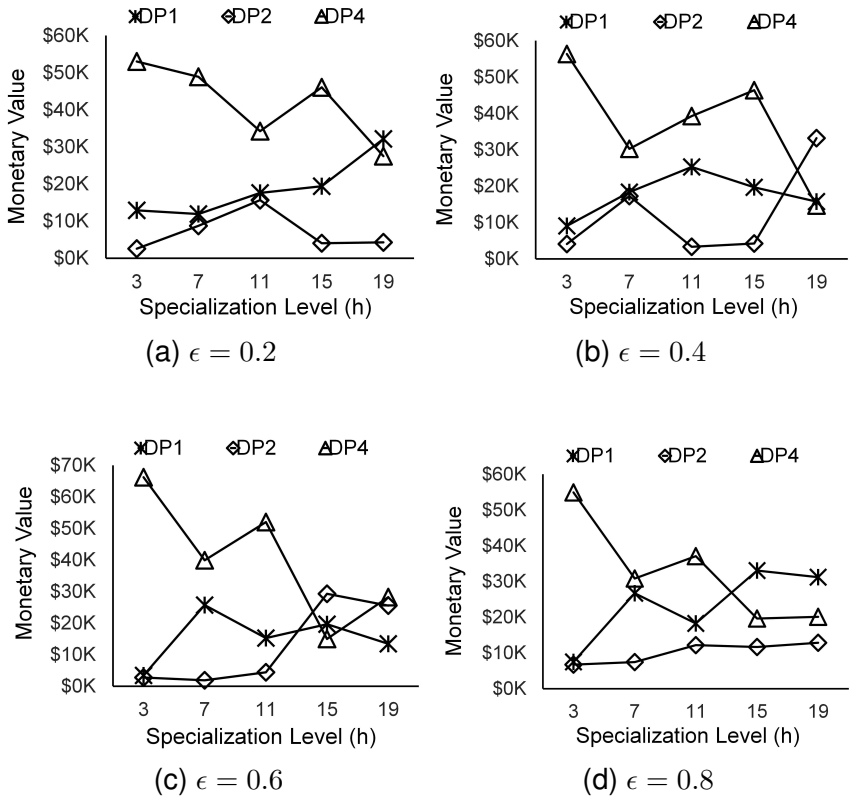


Figure 5.7: Impact of ϵ-*differential privacy* requirements and Trust scores on DP_1, DP_2, and DP_4 monetary value (Case 2)

monetary value. ϵ-*differential privacy* is enforced with privacy parameters $\epsilon = 0.2$, 0.4, 0.6, and 0.8, and specialization levels $3 \leq h \leq 19$.

Fig. 5.7a depicts the impact on DP_1, DP_2, and DP_4's monetary value when the threshold is $\epsilon = 0.2$. We observe that DP_4 attains the highest monetary share because of higher information utility and its trust level, except when $h = 19$. We observe that DP_1's monetary share increases as the specialization level h increases, except when $h = 7$, whereas DP_4's monetary share generally falls with the increase in specialization level h except when $h = 15$. DP_1 gains the maximum value of approximately $\$32K$ of his monetary share when $h = 19$.

Fig. 5.7b depicts the impact on DP_1, DP_2, and DP_4's monetary value when the threshold is $\epsilon = 0.4$. We observe that DP_4 attains the highest monetary share because of higher information utility and its trust level, except when $h = 19$. The trend on DP_1, DP_2, and DP_4's monetary share is not obvious with the increase in specialization level h. DP_2 gains the maximum value of

approximately $\$33K$ of his monetary share when $h = 19$.

Fig. 5.7c depicts the impact on DP_1, DP_2, and DP_4's monetary value when the threshold is $\epsilon = 0.6$. We observe that DP_4 achieves the highest monetary share because of higher information utility and its trust level, except when $h = 15$. DP_4's monetary share drops sharply when h increases from 3 to 7 and 11 to 15, while DP_1 and DP_2 have a significant increase in their monetary shares with this increase in h. DP_2 gains the maximum value of approximately $\$29K$ of monetary share when $h = 15$.

Fig. 5.7d depicts the impact on DP_1, DP_2, and DP_4's monetary value when the threshold is $\epsilon = 0.8$. We observe that DP_4 gains the maximum value of monetary share when $h = 3$, 7, and 11, whereas DP_1 gains the maximum value of monetary share when $h = 15$ and 19. This is because they have more information utility in competing with the other data providers at the indicated levels of specializations. DP_2's monetary share generally increases as the increase in specialization level h, except when $h = 15$. DP_1 and DP_4 do not exhibit monotonicity with the increase in h.

5.6 Summary

In this chapter, we propose a novel entropy-based trust computation algorithm to verify the correctness of data from untrusted multiple data providers who own overlapping attributes over the same set of records. We achieve three main benefits in delegating the verification role to the semi-trusted cloud service provider. First, our method ensures that the cloud service provider cannot derive customers' private data from the information collected during the verification process. Second, the overhead of computation on the cloud server is also reduced because only an encrypted information gain message and its keyed hash are exchanged between a data provider and the cloud server, instead of exchanging encrypted individual data records during the verification process. Third, it also reduces the burden on data consumers to determine which data providers can serve their demands on requested attributes and what their attained trust scores are. Furthermore, we evaluate the robustness of our approach when a data provider employs machine learning method for imputation of missing values on its data. There is no significant difference in perspective to the performance of the imputation method. It is conditional to what proportion of data is missing and whether the data

contains repeated patterns. If the prediction of a missing data happens to be as precise data, then it will be considered as true data. We incorporate the VCG auction mechanism to determine the pricing on data providers' attributes. It maximizes the total valuation obtained by data providers since there is no incentive to lie or deviate from truthful reporting. From the perspective of privacy protection, the accepted data providers as a result of trust computation set up their joint privacy requirements for the data mashup. During the data mashup process, every data provider competes with the other participating data providers to produce more data utility. It is evident from the experiments that an accepted data provider whose data attributes result in more information gain, and whose trust level is higher than the other competitors, can get a proportionally larger share of the monetary value.

Chapter 6

Differentially Private Release of Heterogeneous Network for Healthcare Data

6.1 Introduction

In the past decade, heterogeneous information networks (HINs) have gained increasing attention in various application domains such as social media, communications, energy, and health informatics, mainly due to its ubiquitousness and capability of representing rich semantics [151]. Many complex networks are modeled as graphs, where entities are described by nodes and their relationships are represented by edges. Currently, databases have evolved in order to handle large networks of connected data. In this chapter, we model a complex de-identified healthcare dataset including patients' medical histories, medications, laboratory tests, and demographics, using a heterogeneous information network that consists of multi-type entities and their multi-type relationships. A network schema of a heterogeneous health information network (HHIN) is illustrated in Figure 6.1, which is a graphical representation of real-life health-related data. In the illustrated network schema, *Patient*, *Disease*, *Medication*, and *Lab Test* are entities, whereas *contracts*, *uses*, and *undergoes* represent relationships between entities. We use the terms *network* and *graph* interchangeably.

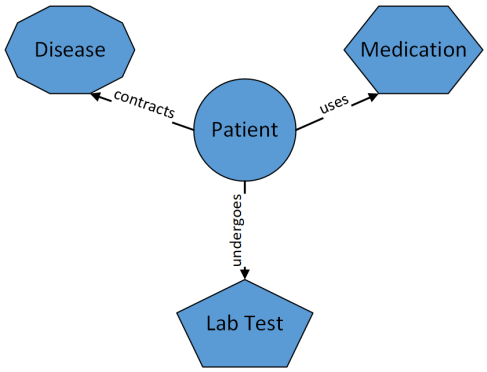


Figure 6.1: Network schema

Figure 6.2 provides an overview of privacy-preserving data publishing of HHIN. In the presented scenario a health information custodian (HIC) collects health-related data from multiple data sources (where a data source is denoted by DS in the figure). The collected data from all sources pertains to the same set of patients and is maintained in a single repository. The fusion of all the collected data results in a typical heterogeneous network. The goal of HIC is to publish the collected data to a data recipient for data analysis without compromising the patients' privacy. To address this real-life problem for health-network data and to bring additive advantages to HIC by properly balancing privacy and utility requirements, we propose a method that converts de-identified health network data into a differentially private version.

It has been a common practice by the HICs to maintain health-related data in central storage to facilitate administrative operations, improve healthcare services, and support medical research [84]. Health data contains sensitive information about patients, and HICs must ensure the protection of patients' private information during the collection, use, and release of health data as mandated by law [73]. Many health-service providers follow the practice of obtaining patients' consent when sharing their health data [94, 127]. However, HICs have faced increasing privacy breaches of different natures [7, 8, 94] due to negligence of administrative employees, compliance failures, and the deployment of weak de-identification methods [23]. Data sharing carries mutual benefits to both the HIC and the data recipient, but it comes with conflicting requirements on data privacy and data utility. To bridge the gap between these two conflicting requirements, several privacy models were proposed in the literature for network or graph anonymization. These models can be apprehended into two types: $syntactic$ and $semantic$ models.

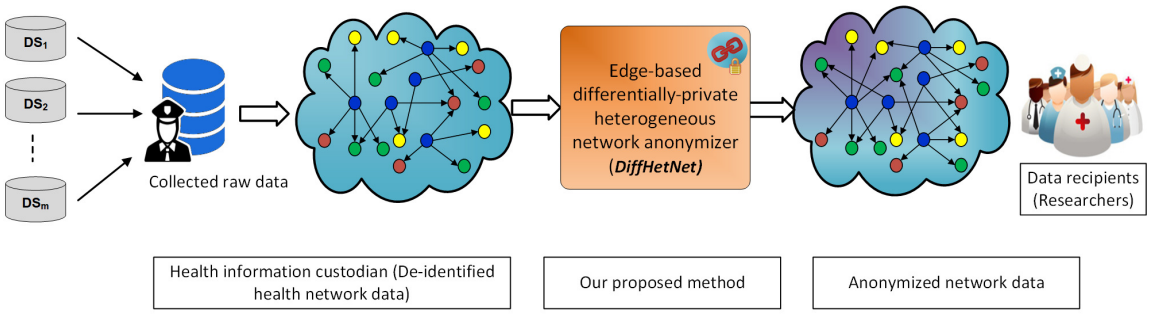


Figure 6.2: Privacy-preserving health network data publishing

There is a line of research [20, 119, 191, 192] based on syntactic privacy models that focuses on preserving structural information in networks. The works in [119, 191] prevent node re-identification, whereas some other works [20, 38, 192] focus on protecting against both node re-identification and edge disclosure in the presence of structural background knowledge of an adversary. Most of these works focus on undirected networks. It is not a good practice to utilize the same methods for anonymizing directed graphs. Generally, if a directed network is anonymized under syntactic-based models without considering the direction of edges it may be prone to re-identification attacks [34], and it also causes a loss of information utility because of the structural properties of the network. Among all privacy models, the works of [20, 38] are relatively better for privacy protection. They both are rooted in k-isomorphism. Chen et al. [37] show that an adversary with moderate background knowledge can identify certain links among nodes on a k-isomorphic graph [38] due to its deterministic nature. The work in [20] provides (k, δ)-privacy to resist against k-core attacks. It is also scalable to massive network data, but its application is limited to *homogeneous networks*, where nodes and edges are to be of a single type.

Another line of research [33, 37, 42, 76, 92, 173] applies *differential privacy* (*DP*) for anonymizing network data. It is a semantic model that provides strong privacy guarantees to an individual independently of an adversary's background knowledge [50]. The two frameworks, namely *interactive* and *non-interactive*, are mainly discussed regarding utilization of the privacy budget ϵ [37, 50, 185]. The primary difference is that in the interactive framework the data custodian holds the raw network data, and a data analyst submits a set of queries in real time, for which the data custodian provides differentially private answers. Each query would utilize a fraction of ϵ to produce a noisy answer. When the entire ϵ has been consumed, a data analyst would not be able to get the

answer by querying the database. On the other hand, in the non-interactive framework, the data custodian first anonymizes its raw network data by utilizing the entire privacy budget. Later, the anonymous data releases to the data analyst, who can perform an analysis without any limitations on the data usage. This approach, widely known as *privacy-preserving data publishing* (*PPDP*) [61], is more appropriate in many real-life network data-sharing scenarios because of the flexibility for a data analyst to perform an analysis without specifying a target analysis. Therefore, in this chapter we focus on the non-interactive framework for network data publishing.

The intuition of differential privacy is that individual information should not be revealed from the output of the analysis in the anonymized data whether or not the individual opted in to be part of the database. *Node-differential privacy* [33, 42, 92] and *edge-differential privacy* [37, 76, 173] are the most common formulations for network data anonymization in the literature. In node-DP, two graphs are neighboring graphs if they differ by *at most* one node and, by extension, all its edges. In edge-DP, two graphs are neighboring graphs if they differ by *at most* one edge. In this chapter we follow the formulation of edge-differential privacy to tackle the problem of protecting sensitive links of a patient in the heterogeneous health network. We focus on preventing the disclosure of sensitive relationships between patient nodes and non-patient nodes from adversarial inbound and outbound privacy attacks.

Compared with existing work on edge-DP [37], our solution to the problem is different in several aspects. First, in contrast to homogeneous network solutions, our solution aims to protect sensitive links of an individual in a *heterogeneous network* that is characterized by having multiple types of nodes and edges. Second, our proposed solution takes the direction of edges into account to maintain the structural properties of the network. Third, our solution extracts the network structural properties without performing vertex labeling [37] (which is required in order to form dense regions for effective anonymization) on an input network; thus, our solution is not sensitive to the density of the input network. Finally, the underlying procedure for anonymization is also different. Our solution comprises two phases, where each phase provides both *indegree* and *outdegree* protection for the input network. The two phases integrate the exponential mechanism that uses the degree-centrality function, which yields a real-valued score. For an input network, the first phase protects vulnerable nodes by picking nodes that are prone to adversarial attacks due to having fewer incoming or outgoing

connections. In the second phase, we preserve information utility by choosing nodes having higher scores and connecting them to the nodes that were picked in the first phase to protect their inbound and outbound connections.

Contributions. This is the first edge-differentially private, non-interactive framework providing a practical solution to health information custodians (HICs) who wish to release real-life heterogeneous health-network data. Our contributions are summarized as follows:

- We model complex, de-identified healthcare data as a heterogeneous information network that consists of multi-type entities along with their directional relationships. Existing solutions [37, 76, 173] consider nodes and edges to each be of a single type and edges to be bidirectional (or undirected). Thus, these solutions cannot maintain important semantics and structural information of the heterogeneous network.

- We propose *DiffHetNet*, a differentially private method to protect patients' sensitive links in a health network. Compared with the anonymization method for undirected networks in [37], our method offers better protection against an adversary's inbound and outbound attacks for learning the existence of a patient's sensitive information. Experimental results suggest that our method generally yields less information loss and is significantly more efficient in terms of runtime when compared with related anonymization methods from the literature. Furthermore, our method effectively extracts the structural properties of an input network, and it is not sensitive to the density of edges in the network. Our experiments demonstrate the density-insensitivity feature of our method.

- We evaluate the performance of our proposed method with respect to information utility and efficiency using different real-life network datasets. In addition, we demonstrate that our approach is scalable to large network datasets.

The rest of this chapter is organized as follows. In Section 6.2, we define the problem. In Section 6.3, we present our proposed differentially private algorithm. In Section 6.4, we compare our proposed method with the existing methods and evaluate the performance in terms of information utility, efficiency, and scalability. Finally, we provide the summary in Section 6.5.

6.2 Problem definition

Suppose a HIC wants to publish collected healthcare-network data in a privacy-preserving manner to a data recipient or a data miner for gaining valuable insights, predicting outbreaks of epidemics, preventing chronic diseases, reducing the cost of healthcare delivery, and improving outcomes for patients, etc. The raw data are fused across multiple data sources, resulting in a typical heterogeneous network, $G = (V, E)$, with a node type-mapping function $\varphi : V \rightarrow \mathcal{E}$ and an edge type-mapping function $\psi : E \rightarrow \mathcal{R}$. Each node $v \in V$ belongs to one particular node type in the node type set $\mathcal{E} : \varphi(v) \in \mathcal{E}$, and each edge $e \in E$ belongs to a particular relation type in the relation type set $\mathcal{R} : \psi(e) \in \mathcal{R}$. If two edges belong to the same relation type, the two edges share the same starting node type as well as the ending node type. Figure 6.1 illustrates the network schema of a heterogeneous health information network (HHIN), where multiple types of nodes $|\mathcal{E}| > 1$ and multiple types of relations $|\mathcal{R}| > 1$ exist in the network. We illustrate the problem in the following example.

Example 7. Consider a heterogeneous directed health network illustrated in Fig. 6.3. In this example, *Patient* (P), *Disease* (D), *Medication* (M), and *Lab Test* (LT) are nodes of different types in the node type set $\mathcal{E}$, whereas *contracts* $(L_{(1)})$, *uses* $(L_{(2)})$, and *undergoes* $(L_{(3)})$ are the types of relationships between nodes in the relation type set $\mathcal{R}$. The number of nodes types $|\mathcal{E}| = 4$, and types of relationships $|\mathcal{R}| = 3$. The total number of nodes $|V| = 14$, and edges $|E| = 26$. Below we discuss potential linkage attacks on a patient's privacy.

In an *indegree linkage* attack, an adversary attempts to link structural background knowledge in the context of incoming connections to a node. For a given two types of nodes $\mathcal{U}$, $\mathcal{V}$ and their relation $L_{(i)}$ in the relation type set $\mathcal{R}$, where $u \in \mathcal{U}$ and $v \in \mathcal{V}$, the set of incoming connections to v_i from u_i with relation-type $L_{(i)}$ are the possible candidates for an indegree linkage. In this example, $P2$ undergoes $LT1$, $LT2$, and $LT3$. It is safe for $P2$ because the patient has had multiple lab tests. However, among the lab tests, $LT3$ is taken only by $P2$, and none are taken by the other patients. Thus, there is a change of indegree linkage attack.

In an *outdegree linkage* attack, an adversary attempts to link structural background knowledge in the context of outgoing connections from a node. For a given two types of nodes $\mathcal{U}$, $\mathcal{V}$ and their

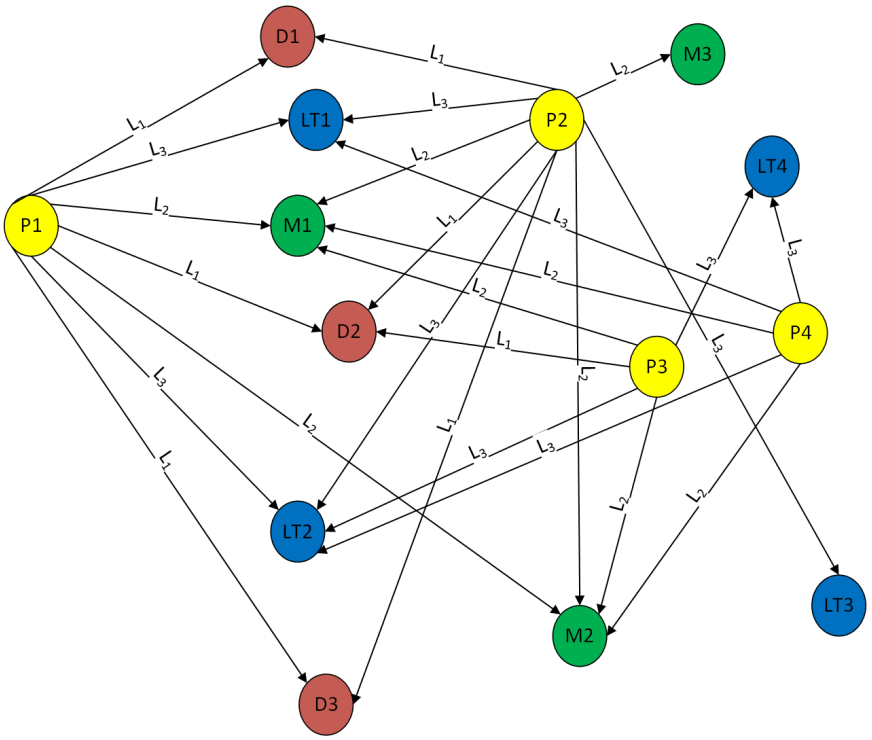


Figure 6.3: An example of original health network

relation $L_{(i)}$ in the relation type set $\mathcal{R}$, where $u \in \mathcal{U}$ and $v \in \mathcal{V}$, the set of outgoing connections from u_i to v_i with relation-type $L_{(i)}$ are the possible candidates for an outdegree linkage. In this example, $P1$, $P2$, and $P3$ contract $D2$. In the context of indegree linkage, $D2$ is safe because multiple patients have contracted it, so an adversary may not be confident in relation to which patient contracted disease $D2$. However, among the patients, $P3$ is only contracted with $D2$ and none of the other diseases. Thus, there is a chance of outdegree linkage attack. ■

In this chapter, we propose a method to achieve edge-differential privacy with the goal of preventing the aforementioned linkage attacks in a heterogeneous network while releasing the data to a third party for research purposes. It is different from the work based on edge-differential privacy under correlation [37] for a homogeneous undirected network as detailed in previous sections. We first present the definition of edge-differential privacy for heterogeneous networks, followed by our problem statement.

Definition 6.2.1. *(*Edge-differential privacy of heterogeneous networks*).* Given a heterogeneous graph $G_1 = (V_1, E_1)$, where V_1 or E_1 are of multiple types (as per Definition 2.5.2), a heterogeneous

graph $G_2 = (V_2, E_2)$ is a neighboring graph to G_1 if the difference between G_1 and G_2 is *at most* one edge (i.e., $|V_1 \oplus V_2| + |E_1 \oplus E_2| = 1$). A sanitization mechanism $\mathcal{M}$ provides *edge-differential privacy* if for any two neighboring heterogeneous graphs, and for any possible sanitized graph $\hat{G}$, we have

$$\Pr[\mathcal{M}(G_1) = \hat{G}] \leq e^{\epsilon} \times \Pr[\mathcal{M}(G_2) = \hat{G}]. \blacksquare$$

The *Laplace mechanism* [50] and *exponential mechanism* [122] are the two most common mechanisms for achieving ϵ-differential privacy. These mechanisms depend on the privacy parameter ϵ and the sensitivity [50] of a function that maps the input database to real values. The *sensitivity* of the function f is defined as follows:

Definition 6.2.2 (*Sensitivity*)**.** For any function $f : G \to \mathbb{R}^d$, the sensitivity of f is

$$\Delta f = \max_{G,G'} ||f(G) - f(G')||_1 \tag{47}$$

for all G, G' differing *at most* by one edge or node (including all its adjacent edges). $\blacksquare$

Laplace mechanism was introduced by Dwork et al. [50]. It is appropriate when the output of function f is a real value, and f should return a noisy answer to preserve privacy. The noise is calibrated based on the privacy parameter ϵ and the sensitivity of the utility function Δf. Formally, the Laplace mechanism takes as inputs a network dataset G, the privacy parameter ϵ, and a function f and outputs $f(\hat{G}) = f(G) + \texttt{Lap}(\lambda)$, where $\texttt{Lap}(\lambda)$ is a noise drawn from the Laplace distribution with probability density function $\Pr(x|\lambda) = \frac{1}{2\lambda}\texttt{exp}(-|x|/\lambda)$, where $\lambda = \frac{\Delta f}{\epsilon}$. The variance of this distribution is $2\lambda^2$, and the mean is 0.

Theorem 1. *[50] For any function $f : G \to \mathbb{R}^d$, the algorithm $\mathcal{M}$ that adds independently generated noise with distribution $\texttt{Lap}(\Delta f/\epsilon)$ to each of the d outputs satisfies ϵ-differential privacy.*

Exponential mechanism was proposed by McSherry and Talwar [122]. It is appropriate when it is desirable to choose the best response, because adding noise directly to the count can destroy its value. Given an arbitrary range $\mathcal{T}$, the exponential mechanism is defined with respect to a utility function $\texttt{u} : (G \times \mathcal{T}) \to \mathbb{R}$ that assigns a real-valued score to every output $t \in \mathcal{T}$, where a higher

score means better utility. The exponential mechanism induces a probability distribution over the range $\mathcal{T}$ and then samples an output t.

Theorem 2. *[122] Given a utility function* $u : (G \times \mathcal{T}) \rightarrow \mathbb{R}$ *with sensitivity* $\Delta u = \max_{\forall t,G,G'} |u(G,t) - u(G',t)|$*, an algorithm* $\mathcal{M}$ *that chooses an output* t *with probability proportional to* $exp(\frac{\epsilon u(G,t)}{2\Delta u})$ *satisfies* ϵ*-differential privacy.*

Sequential composition and *parallel composition* are the two important composition properties of differential privacy [121]. The first property stipulates that if a sequence of differentially private computations take place in isolation on the same input data, then the entire sequence gives the accumulated privacy guarantee. The second property stipulates that if differentially private computations take place on each chunk separately over the split dataset, where chunks are disjoint, then the privacy cost does not accumulate, but it depends only on the worst guarantee of all computations.

Theorem 3 (*Sequential composition* [121])**.** *Let each* $\mathcal{M}_i$ *provide* ϵ_i*-differential privacy. A sequence of* $\mathcal{M}_i(G)$ *over the network* G *provides* $(\sum_i \epsilon_i)$*-differential privacy.*

Theorem 4 (*Parallel composition* [121])**.** *Let each* $\mathcal{M}_i$ *provide* ϵ*-differential privacy. A sequence of* $\mathcal{M}_i(G_i)$ *over a set of disjoint networks* G_i *provides* ϵ*-differential privacy.*

Problem (Edge-differential privacy in HHIN). Given a heterogeneous health information network $G = (V, E)$, where each node $v \in V$ belongs to one particular node type in the node type set $\mathcal{E}$, and each edge $e \in E$ belongs to a particular relation type in the relation type set $\mathcal{R}$, nodes are of multiple types $|\mathcal{E}| > 1$ and relationships are of multiple types $|\mathcal{R}| > 1$, and privacy budget ϵ, the goal is two-fold:

- To publish an anonymized version of network G, denoted by G', that protects patients' privacy by preventing adversarial inference on each incoming and outgoing edge $e \in E$ in accordance with edge-differential privacy.
- To minimize the impact of anonymization on all edges E in G by reducing the errors generated by the mean absolute error, the average relative error, and the Kullback-Leibler divergence, as defined in Eqs. 9, 10, and 11, respectively.

6.3 Proposed solution

In this section, we present an edge-based differentially private solution to protect the sensitive links of a patient from adversarial inbound and outbound attacks in a heterogeneous health network, while minimizing information loss inflicted on edges. Our solution addresses the concern of a health information custodian (HIC) on preserving privacy and the concern of a data recipient on information utility. Section 6.3.1 presents an overview of our proposed *DiffHetNet*, an algorithm based on edge-differential privacy for anonymizing heterogeneous network data. Section 6.3.2 presents the operations for exploring subgraphs favoring lower scores when selecting candidate nodes. Section 6.3.3 presents the operations for generating noisy counts. Section 6.3.4 presents the operations for exploring subgraphs favoring higher scores when selecting candidate nodes. Section 6.3.5 presents the process of edge perturbation in the network. Section 6.3.6 provides privacy analysis, and Section 6.3.7 provides utility analysis of our proposed algorithm.

6.3.1 Overview

We first provide a high-level description of our proposed method in Algorithm 5, followed by detailed discussions of each step.

6.3.1.1 High-level description

We study the problem of protecting patients' privacy when sharing healthcare data. We propose a privacy-preserving solution to this problem. However, our solution is also applicable to other network-data publishing scenarios sharing the same privacy and utility concerns. Our proposed solution is based on edge-differential privacy to anonymize a heterogeneous network. We impose edge-differential privacy on the relationships between patient nodes and non-patient nodes to prevent an adversary's indegree and outdegree linkage attacks, i.e., identifying sensitive relationships. We provide an illustration of privacy attacks in Example 7. Compared with existing works that preserve privacy in homogeneous networks [37, 76, 173], our proposed solution not only considers different types of nodes and edges in a given network, but it also takes into account the direction of edges in the network. Our solution takes a heterogeneous graph G and a privacy budget ϵ as inputs and

outputs a differentially private graph G'.

We observe that nodes with a low number of directed edges are more vulnerable to adversary linkage attacks than nodes with a high number of edges. Our aim is to identify such nodes, i.e., nodes with a low number of directed edges. To do so, we consider the following two cases based on the direction of edges: *indegree* linkage to identify patients, and *outdegree* linkage with respect to the adversary's confidence about a target patient's relationship. In the first case, the identified node is a non-patient node, e.g., a disease or lab test. Consequently, the privacy of patients connected to such a node is at risk because very few patients have a relationship with this node, e.g., contracted a disease or received a lab test. In the second case, the identified node is a patient node. The privacy of the patient node is at risk because the patient's node is connected to only a few other nodes. A patient node with a low number of (outdegree) edges results in a high level of an attacker's confidence with respect to a particular relationship associated with this node, e.g., a patient contracting a disease.

Based on the above observation, we want to protect vulnerable nodes from identification attacks by connecting these nodes to less vulnerable ones. Our proposed solution accomplishes this goal across two phases. The first phase searches the network and identifies nodes with a low number of directed edges. The second phase preserves information utility by choosing nodes with a high number of directed edges, since these nodes are less vulnerable to identification attacks. After that, the second phase connects the nodes picked in this phase to the nodes that were picked in the first phase to protect their inbound and outbound connections.

6.3.1.2 Algorithm

Algorithm 5 presents the anonymization operations, which we split into two phases. Before describing the lines of Algorithm 5, we explain how the input privacy budget ϵ is distributed throughout the algorithm. The input privacy budget ϵ is divided into three portions in Line 1. The first portion, denoted by ϵ_{slo}, is consumed when exploring lower-scoring candidate nodes. The second portion, denoted by ϵ_{nc}, is utilized when generating a noisy count for each candidate node. The third portion, denoted by ϵ_{shi}, is consumed when determining higher-scoring candidate nodes. ϵ_{slo} is allocated to the first phase, and ϵ_{nc} and ϵ_{shi} are allocated to the second phase. We divide ϵ such that the summation of ϵ_{slo} and ϵ_{shi} constitutes the majority of ϵ, and ϵ_{nc} is less than ϵ_{slo} and ϵ_{shi},

Algorithm 5 DiffHetNet Algorithm

Input: Original network $G = (V, E)$, privacy budget ϵ
Output: Anonymous differentially private network G'

1: Allocation of privacy budget $\epsilon \leftarrow \epsilon_{slo} + \epsilon_{nc} + \epsilon_{shi}$; /* for indegree and outdegree of directed network*/
2: Set $\alpha_b = dir$; // input direction
3: Lower-scoring candidates $C_{lo}^{\alpha_b} \leftarrow exploreSGsInOutDegFavLowScores(\alpha_b, \epsilon_{slo}^{\alpha_b}, G)$;
4: **for** $c_i \in C_{lo}^{\alpha_b}$ **do**
5: $\quad \epsilon_{nc}^{\alpha_b\prime} \leftarrow \frac{\epsilon_{nc}^{\alpha_b}}{|C_{lo}^{\alpha_b}|}$;
6: $\quad$ Noisy count $Nc^{\alpha_b} \leftarrow genNoisyCount(c_i, \alpha_b, \epsilon_{nc}^{\alpha_b\prime}, G)$;
7: $\quad$ Higher-scoring candidates $C_{hi}^{\alpha_b} \leftarrow findCandsFavHighScoresProtectInOutDeg(Nc^{\alpha_b}, c_i, \alpha_b, \epsilon_{shi}^{\alpha_b}, G)$;
8: $\quad$ Anonymized sub-network $\tilde{G}^{\alpha_b} \leftarrow edgePerturbation(\forall C_{hi}^{\alpha_b}, Nc^{\alpha_b}, c_i, \alpha_b, G)$;
9: **end for**
10: Generate G' from $\tilde{G}^{\alpha_b}$;
11: **return** G';

respectively. The reason for allocating a larger portion of ϵ to ϵ_{slo} (phase 1) is because Algorithm 5 in Line 3 will attempt to discover vulnerable candidate nodes (due to having fewer incoming or outgoing connections). In order to accurately discover nodes that are more prone to adversarial attacks, differential privacy necessitates allocating a larger portion of privacy budget. Similarly, the reason for allocating a larger portion of the budget to ϵ_{shi} (phase 2) in Line 5 is because we intend to preserve more information utility by choosing candidates that are less vulnerable to identification attacks.

Algorithm 5 in Line 3 explores subgraphs in the input network G and picks candidate nodes having lower scores, denoted by $C_{lo}^{\alpha_b}$. The score for each candidate is computed using the degree-centrality function that yields a real-valued score. We design a procedure that uses the exponential mechanism to favor candidates with lower scores. Next, we generate a noisy count, denoted by Nc^{α_b}, that represents the number of newly-generated edges to be added to each node $c_i \in C_{lo}^{\alpha_b}$ by using the Laplace mechanism in Line 6. Based on the generated noisy count, Line 7 scans the input network G and uses the exponential mechanism to pick nodes favoring higher scores, denoted by $C_{hi}^{\alpha_b}$. Subsequently, we protect the corresponding inbound and outbound connections of each node c_i by adding edges from $C_{hi}^{\alpha_b}$, or removing corresponding edges, to have an anonymized version of sub-network $\tilde{G}^{\alpha_b}$ in Line 8. Finally, the differentially private sub-networks of both indegree and outdegree are combined to form an anonymized network G'.

6.3.2 Selecting candidates favoring lower scores

The rationality of exploring subgraphs in the heterogeneous network G is that nodes having fewer incoming or outgoing connections are more prone to adversarial attacks. Procedure 1 attempts to discover vulnerable candidate nodes in the network. It takes a heterogeneous network G, a privacy budget $\epsilon_{slo}^{\alpha_b}$, and the type of degree direction $\alpha_b = \{in|out\}$ as inputs, and it outputs a list of candidate nodes having lower scores, denoted by $C_{lo}^{\alpha_b}$.

Line 1 allocates a portion of the given privacy budget to each candidate by dividing the given budget from the total number of nodes under a specified direction. Line 4 computes the score for each node v using the normalized degree-centrality metric for a directed graph that yields a real-valued score for each node v under the node type V_{τ^i} in the node type set $\mathcal{E}$ and the corresponding relation-type $L_{(i)}$ in the relation type set $\mathcal{R}$. It is defined as follows:

$$CD(G, v, L_{(i)}^{\alpha_b}) = \frac{d^{\alpha_b}(v)_{v \in V_{\tau^i}, L_{(i)}^{\alpha_b} \in \mathcal{R}}}{|V| - 1} \tag{48}$$

Example 8. We continue from Example 7. Consider the type of degree direction $\alpha_b = \{out\}$, i.e., representing the outgoing connections, the type of node $V_{\tau^i} = \{P\}$, i.e., representing a patient's node label, and the relation-type $L_{(1)} = \{contracts\}$, i.e., representing the relationship to the adjacent node(s) of type $V_{\tau^j} = \{D\}$, i.e., representing a disease's node label, in Fig. 6.3. The normalized degree-centrality scores of nodes $\{P1, P2, P3\} = \{0.23, 0.23, 0.08\}$ by Eq. (48), whereas the number of outgoing connections d^{out} of nodes $\{P1, P2, P3\} = \{3, 3, 1\}$. ■

DiffHetNet makes novel use of the exponential mechanism in Line 5. In this step, the exponential mechanism favors lower scores to choose a candidate node v from a set of candidate nodes under the node type V_{τ^i}. It is presented in Theorem 5. The sensitivity of $\Delta \mathrm{u}$ is 1, because the addition or removal of a single edge in G would change $CD(G, v, L_{(i)}^{\alpha_b})$ by at most 1.

Theorem 5. *Choosing a candidate score from a set of candidate scores satisfies ϵ'-differential privacy.*

Proof. Let $Cand_i$ be the set of candidate scores from which a single score is to be chosen for lower

Procedure 1 *exploreSGsInOutDegFavLowScores* Procedure

Input: Original network $G = (V, E)$
Input: Privacy budget $\epsilon_{slo}^{\alpha_b}$, direction α_b
Output: Lower-scoring candidates $C_{lo}^{\alpha_b}$

1: $\epsilon_{slo}^{\alpha_b\prime} \leftarrow \frac{\epsilon_{slo}^{\alpha_b}}{|V_{\tau^i}^{\alpha_b}|}$;
2: $C_{lo}^{\alpha_b} \leftarrow \emptyset$;
3: **for** each pair of neighboring vertices $v_i, v_j \in V$ **do**
4: Compute the score for every $v \in V_{\tau^i}$ according to Eq. (48);
5: Select $v \in V_{\tau^i}$ with probability $\propto \exp(\frac{\epsilon_{slo}^{\alpha_b\prime}}{2\Delta \mathtt{u}} \cdot \mathtt{u}(G, v, L_{(i)}^{\alpha_b}))$ favoring lower score;
6: Add v to the list $C_{lo}^{\alpha_b}$;
7: **end for**
8: **return** $C_{lo}^{\alpha_b}$;

scores. Our algorithm selects a candidate score $v_i \in Cand_i$ with the following probability:

$$\frac{\exp(\frac{\epsilon_{slo}^{\alpha_b\prime}}{2\Delta \mathtt{u}} \cdot \mathtt{u}(G, v_i, L_{(i)}^{\alpha_b}))}{\sum_{v \in Cand_i} \exp(\frac{\epsilon_{slo}^{\alpha_b\prime}}{2\Delta \mathtt{u}} \cdot \mathtt{u}(G, v, L_{(i)}^{\alpha_b}))} \tag{49}$$

where $\mathtt{u}(G, v_i, L_{(i)}^{\alpha_b})$ is a score computed from a utility function according to Eq. (48), and $\Delta\mathtt{u}$ is the sensitivity of the utility function $\mathtt{u}$. According to Theorem 2, selecting a score with probability proportional to $\exp(\frac{\epsilon'\mathtt{u}(G,t)}{2\Delta \mathtt{u}})$ satisfies ϵ'-differential privacy. □

The scores are inverted for the exponential mechanism to favor lower-scoring candidates. At each iteration, the lower-scoring candidate selected by the exponential mechanism is added to the list $C_{lo}^{\alpha_b}$ in Line 6. This process runs until equilibrium is reached or there are no more lower-scoring candidates in the network. Finally, the list of selected lower-scoring candidate nodes is returned by this procedure.

6.3.3 Generating noisy counts

After obtaining the list of lower-scoring candidate nodes $C_{lo}^{\alpha_b}$, Procedure 2 generates a noisy count for each candidate c_i in the list. A portion of the given budget, denoted by $\epsilon_{nc}^{\alpha_b\prime}$, is allocated to each candidate by dividing it from the total number of lower-scoring candidate nodes. Line 1 generates a noisy count Nc^{α_b} from the Laplace distribution $\mathtt{Lap}(1/\epsilon_{nc}^{\alpha_b\prime})$. It can be a positive or negative value. The noise count Nc^{α_b} of a selected candidate c_i is calibrated according to the potential connecting candidate c_j' of node type V_{τ^j} by considering the set of all possible candidates

Procedure 2 *genNoisyCount* Procedure

Input: Original network $G = (V, E)$
Input: Privacy budget $\epsilon_{nc}^{\alpha_b\prime}$
Input: Selected candidate c_i, direction α_b
Output: Noisy count Nc^{α_b}

1: $Nc^{\alpha_b} \leftarrow \texttt{Lap}(1/\epsilon_{nc}^{\alpha_b\prime})$;
2: **if** $Nc^{\alpha_b} < 0$ **then**
3: $\quad Nc^{\alpha_b} = 0$;
4: **end if**
5: **if** $Nc^{\alpha_b} \geq 1$ **then**
6: $\quad$ **if** $c_i \in V_{\tau^i}$ **then**
7: $\quad\quad Nc^{\alpha_b} = Nc^{\alpha_b} \mod (\ln |\mathbb{U}_{V_{\tau^j}}|)$;
8: $\quad$ **end if**
9: **end if**
10: **return** Nc^{α_b};

that can exist in any network dataset. Formally, it is defined as follows:

$$Nc^{\alpha_b} = Nc^{\alpha_b} \mod (\ln |\mathbb{U}_{V_{\tau^j}}|) \tag{50}$$

where $|\mathbb{U}_{V_{\tau^j}}|$ represents the size of the universal set of all possible nodes under the given node type that can exist in any network data.

6.3.4 Selecting candidates favoring higher scores

The rationality of selecting nodes with a high number of directed edges in the heterogeneous network G is to preserve information utility. These nodes are less vulnerable to identification attacks, and drawing edges from them have a low impact on the overall structure of the network. The composition of a heterogeneous network entails nodes and edges to be of multiple types, so the centrality scores for the influential nodes pose different semantics according to their respective types and the incoming and outgoing directions of their edges.

Procedure 3 takes the network G, a privacy budget $\epsilon_{shi}^{\alpha_b}$, a noisy count Nc^{α_b}, a candidate node c_i, and the type of degree direction α_b as inputs and outputs a list of candidate nodes having higher scores, denoted by $C_{hi}^{\alpha_b}$. Line 1 allocates a portion of the given privacy budget to each candidate by dividing the given budget from the product of the total number of lower-scoring candidate nodes and the noisy count. Line 5 computes the score for each node v using the normalized degree-centrality metric for a directed graph by Eq. (51) that yields a real-valued score for each node v under the node

Procedure 3 *findCandsFavHighScoresProtectInOutDeg* Procedure

Input: Original network $G = (V, E)$
Input: Privacy budget $\epsilon_{shi}^{\alpha_b}$, Noisy count Nc^{α_b}
Input: Selected candidate c_i, direction α_b
Output: Higher-scoring candidates $C_{hi}^{\alpha_b}$

1: $\epsilon_{shi}^{\alpha_b\prime} \leftarrow \frac{\epsilon_{shi}^{\alpha_b}}{|C_{lo}^{\alpha_b}|\cdot|Nc^{\alpha_b}|}$;
2: $C_{hi}^{\alpha_b} \leftarrow \emptyset$;
3: **while** $|c_i| < |Nc^{\alpha_b}|$ **do**
4: **for** each pair of neighboring vertices $v_i, v_j \in V$ **do**
5: Compute the score for every $v \in V_{\tau^j}$ according to Eq. (51);
6: Select $v \in V_{\tau^j}$ with probability $\propto \exp(\frac{\epsilon_{shi}^{\alpha_b\prime}}{2\Delta \mathrm{u}} \cdot \mathrm{u}(G, v, \widetilde{\alpha_b}))$ favoring higher score;
7: Add v to the list $C_{hi}^{\alpha_b}$;
8: **end for**
9: **end while**
10: **return** $C_{hi}^{\alpha_b}$;

type V_{τ^j} in the node type set $\mathcal{E}$. $\widetilde{\alpha_b}$ represents the opposite degree direction. The score is computed as follows:

$$CD(G, v, \widetilde{\alpha_b}) = \frac{d^{\widetilde{\alpha_b}}(v)_{v \in V_{\tau^j}}}{|V| - 1} \tag{51}$$

Example 9. We continue from Example 7. Let us assume that Procedure 1 returns $\{P3 = 0.08\}$ as one of the lower-scoring outdegree candidate nodes having the relation-type $L_{(1)} = \{contracts\}$ with $D2$. To protect its outbound connection we need to find indegree candidate nodes having higher scores based on the exponential mechanism. The type of a potential candidate's degree direction is $\widetilde{\alpha_b} = \{in\}$, i.e., representing the incoming connections, the type of node $V_{\tau^j} = \{D\}$, i.e., representing a disease's node label in Fig. 6.3. The potential candidate $D1$'s centrality score is computed by Eq. (51) is 0.15. ■

DiffHetNet makes novel use of the exponential mechanism in Line 6. In contrast to the presented Theorem 5, this step utilizes the exponential mechanism to choose a candidate node v favoring a higher score from a set of candidate nodes under the node type V_{τ^j}. At each iteration, the higher-scoring candidate selected by the exponential mechanism is added to the list $C_{hi}^{\alpha_b}$ in Line 7. This process repeats until the number of candidate nodes is less than the size of the noisy count. Finally, the list of selected higher-scoring candidate nodes is returned by this procedure.

Procedure 4 *edgePerturbation* Procedure

Input: Original network $G = (V, E)$
Input: Candidates $C_{hi}^{\alpha_b}$, Noisy count Nc^{α_b}
Input: Selected candidate c_i, direction α_b
Output: Anonymized network $\tilde{G}^{\alpha_b}$

1: **if** $Nc^{\alpha_b} == 0$ **then**
2: Remove corresponding edges of c_i from network $\tilde{G}^{\alpha_b}$;
3: **end if**
4: **while** $c'_j \in C_{hi}^{\alpha_b}$ **do**
5: **if** $c_i \in V_{\tau^i}^{\alpha_b}$ **then**
6: Add edge (c'_j, c_i) or vice versa to network $\tilde{G}^{\alpha_b}$;
7: Set the corresponding relation type $L_{(i)}$;
8: **end if**
9: **end while**
10: **return** $\tilde{G}^{\alpha_b}$;

6.3.5 Edge perturbation

This procedure takes the network G, lower-scoring candidate node c_i selected by Procedure 1, a noisy count Nc^{α_b} by Procedure 2, list of higher-scoring candidate nodes $C_{hi}^{\alpha_b}$ by Procedure 3, and the type of degree direction α_b as inputs, and it outputs an anonymized version of sub-network $\tilde{G}^{\alpha_b}$. It protects the corresponding inbound or outbound connections of each candidate node c_i in the list of lower-scoring candidates $C_{lo}^{\alpha_b}$ by either removing the corresponding edges from $\tilde{G}^{\alpha_b}$ or by adding edges from higher-scoring candidate nodes $C_{hi}^{\alpha_b}$.

Line 2 removes the corresponding edge pairs (c_i, c_j) or vice versa of candidate node c_i from $\tilde{G}^{\alpha_b}$ when the noisy count is 0. Line 5 matches the selected candidate's node type $V_{\tau^i}^{\alpha_b}$ along with the degree direction α_b, and then it adds an edge (c'_j, c_i) or vice versa (Line 6) if it does not exist already in the given network G or was added previously in the $\tilde{G}^{\alpha_b}$. Next, the corresponding relationship $L_{(i)}$ is assigned based on the types of source and destination nodes in Line 7. This process repeats for each potential candidate c'_j in the list of higher-scoring candidate nodes $C_{hi}^{\alpha_b}$. Finally, the anonymized version of sub-network $\tilde{G}^{\alpha_b}$ is returned by this procedure.

Example 10. Fig. 6.4 illustrates a possible anonymized version of the example health network. We continue from Example 7. Let us assume that Procedure 3 returns $\{D1 = 0.15\}$ as one of the higher-scoring indegree candidate nodes for the node $P3$ by Procedure 1. The corresponding edge is added between $P3$ and $D1$, and the relationship $L_{(1)}$ is assigned based on the types of source and destination nodes. Now consider that Procedure 1 returns $\{LT4 = 0.15\}$ as one of the lower-scoring

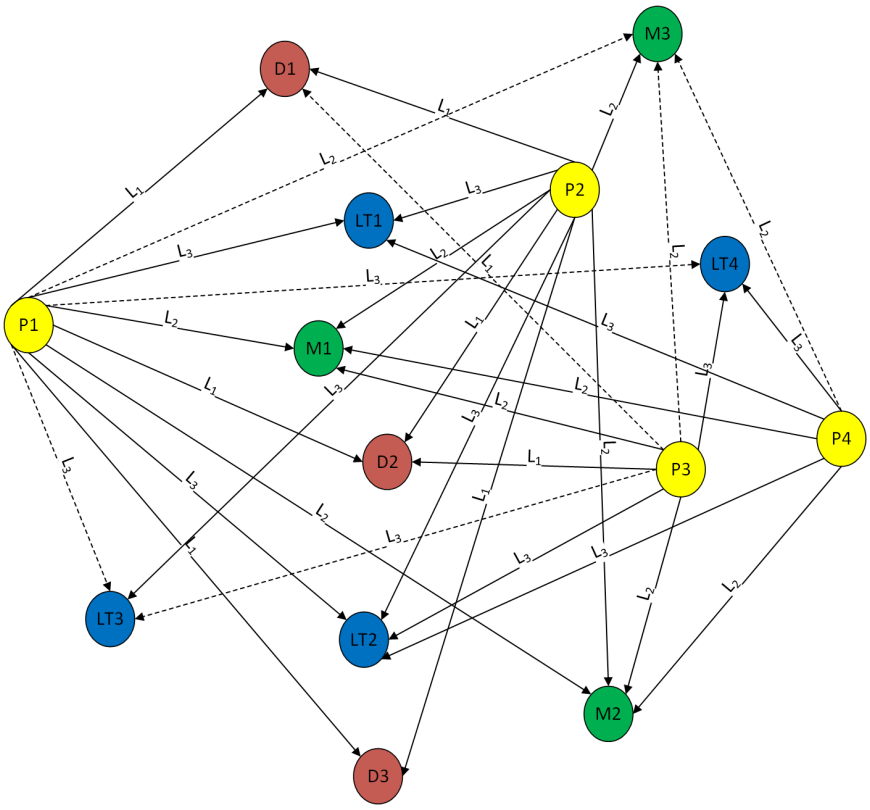


Figure 6.4: Anonymized version of the example health network

indegree candidate nodes having the relation-type $L_{(3)} = \{undergoes\}$ with $P3$ and $P4$. To protect its inbound connection, we need to find higher-scoring outdegree candidate nodes based on the exponential mechanism. Suppose Procedure 3 returns $\{P1 = 0.15\}$ as one of the higher-scoring outdegree candidate nodes for the node $LT4$. The corresponding edge is added between $P1$ and $LT4$, and the relationship $L_{(3)}$ is assigned based on the types of source and destination nodes. ■

6.3.6 Privacy analysis

In this section, we prove that Algorithm 5 satisfies ϵ-differential privacy over heterogeneous network data under the given network schema of Figure 6.1.

Theorem 6. *For a given privacy budget ϵ, Algorithm 5 is ϵ-differentially private over heterogeneous network data.*

Proof. Algorithm 5 picks lower-scoring candidates from a set of candidate nodes by employing the exponential mechanism according to Theorem 5 in Line 3. Each candidate is dedicated with a

privacy budget portion $\epsilon_{slo}^{\alpha_b\prime} = \frac{\epsilon_{slo}^{\alpha_b}}{|V_{\tau^i}^{\alpha_b}|}$ by leveraging *sequential composition* property (Theorem 3). A noisy count is generated for each candidate $c_i \in C_{lo}^{\alpha_b}$ by drawing noise from Laplace distribution $\texttt{Lap}(\frac{\Delta f}{\epsilon})$ (according to Theorem 1) using a privacy budget portion $\epsilon_{nc}^{\alpha_b\prime} = \frac{\epsilon_{nc}^{\alpha_b}}{|C_{lo}^{\alpha_b}|}$ in Line 6. Next, for each candidate c_i, the algorithm picks potential higher-scoring candidate(s) from a set of candidate nodes using the exponential mechanism in Line 7. Each candidate is dedicated with a privacy budget portion $\epsilon_{shi}^{\alpha_b\prime} = \frac{\epsilon_{shi}^{\alpha_b}}{|C_{lo}^{\alpha_b}|\cdot|Nc^{\alpha_b}|}$ by leveraging sequential composition property. Finally, the algorithm post-processes [100] the differentially private inputs $c_i \in C_{lo}^{\alpha_b}$, Nc^{α_b}, and $C_{hi}^{\alpha_b}$ to perturb the network. Hence, Algorithm 5 is ϵ-differentially private because $\epsilon = \epsilon_{slo} + \epsilon_{nc} + \epsilon_{shi}$ by the property of sequential composition (Theorem 3). □

6.3.7 Utility analysis

We measure the utility loss on the anonymized network with respect to the original network by mean absolute error, average relative error, and Kullback–Leibler divergence presented in Section 2.8.

Considering the network schema of Figure 6.1, the goal is to generate a sanitized graph G' so as close to G as possible to minimize the error $\sum_{i=1}^{|V|}|CD(G', v_i) - CD(G, v_i)|$. When G' is identical to G, $\sum_{i=1}^{|V|}|CD(G', v_i) - CD(G, v_i)| = 0$; when G' is totally different from G, $\sum_{i=1}^{|V|}|CD(G', v_i) - CD(G, v_i)| = |V_{\tau^i}^{\alpha_b}| \cdot \sum_{j=1}^{k}|V_{\tau^j}^{\widetilde{\alpha_b}}|$, where $i \neq j$.

Discussion

In Section 6.3.1.2 we discuss the distribution of privacy budget ϵ and its consumption across all the phases of Algorithm 5. The utility guarantee of our proposed algorithm is dependent on the privacy parameter ϵ. Consider s and $\widetilde{s}$ are the scores of a node $v \in V_{\tau^i}^{\alpha_b}$ in G and G', respectively. When $\widetilde{s} < s$, it depends on the following conditions: (1) no new edge is added to a node v, and an existing edge has removed from the node v, and (2) a worst case would be when all existing edges are removed from the node v; when $\widetilde{s} = s$, it depends on the following conditions: (1) no new edge is added to or removed from a node v, and (2) an equal number of edges are added and removed from the node v; when $\widetilde{s} > s$, it depends on the following conditions: (1) a new edge is added to a node v while maintaining all existing edges of the node v, and (2) a worst case would be when newly added edges to the node v are reached to the maximum.

Table 6.1: Statistics of the datasets

Dataset	$\|V\|$	$\|E\|$	Edge Density
ca-GrQc	$5,242$	$28,980$	0.001055
wiki-Vote	$7,115$	$103,689$	0.002049
MIMIC-T1	$13,947$	$103,023$	0.000530
MIMIC-MultiType	$5,786$	$183,795$	0.005491

6.4 Experimental evaluation

In this section, we evaluate the performance of our *DiffHetNet* algorithm in terms of both information utility and efficiency. We compare our method *DiffHetNet* with the *DER* [37] method and its variant *DE* and a random graph [38] (referred to as *Random*). In *DE*, the step of *ArrangeEdge* is simply replaced by randomly inserting edges in each leaf region based on the noisy count. We use three real-life datasets, namely *ca-GrQc*[1], *wiki-Vote*[1], and *MIMIC*[2] from three different types of networks. *ca-GrQc* is an undirected network, extracted from the scientific collaboration network of arXiv GR-QC (General Relativity and Quantum Cosmology) category, where two authors are connected if they co-authored at least one paper. *wiki-Vote* is a directed network extracted from the Wikipedia adminship voting network, where a Wikipedia user is considered for promotion to adminship based on the community votes in favor of or against the promotion. *MIMIC* contains health-related data from a large number of Intensive Care Unit (ICU) patients. It integrates de-identified, comprehensive health data of patients admitted to the Beth Israel Deaconess Medical Center in Boston, Massachusetts. It is accessible to researchers internationally under a data use agreement. *MIMIC-T1* represents a network of a single relation type having nodes that are of different types, i.e., the number of node types $|\mathcal{E}| = 2$, and types of relationships $|\mathcal{R}| = 1$, whereas *MIMIC-MultiType* represents a network of multiple nodes and relations types, i.e., the number of node types $|\mathcal{E}| = 4$, and types of relationships $|\mathcal{R}| = 3$. The statistics of the datasets are shown in Table 6.1. All experiments were performed on a PC with Intel Core i7 2.80GHz and 16GB RAM.

[1]It is publicly available in the Stanford large network dataset collection at: http://snap.stanford.edu/data/index.html

[2]Available at: https://mimic.physionet.org

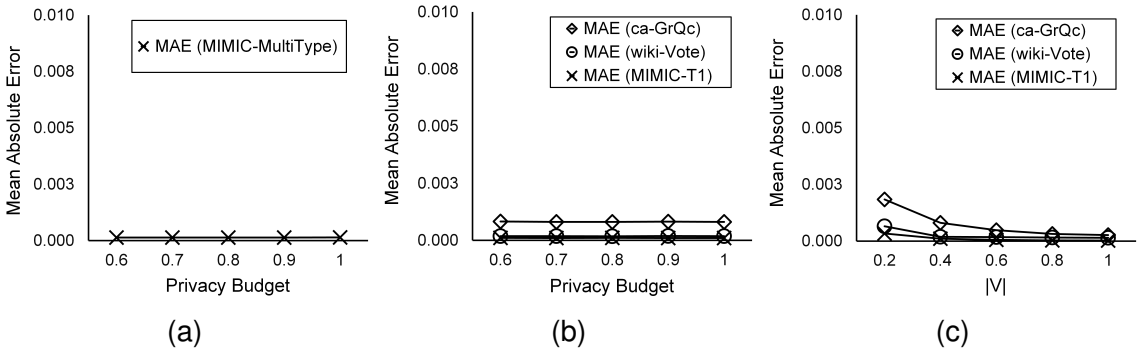


Figure 6.5: Mean absolute error by *DiffHetNet* method under varying ϵ in (a) and (b), and fixed $\epsilon = 1.0$ and varying data size in (c)

6.4.1 Measuring information loss

We measure the information loss on the anonymized network with respect to the original network by mean absolute error, average relative error, and Kullback–Leibler divergence introduced in Section 2.8.

6.4.1.1 Mean absolute error

Fig. 6.5 presents the mean absolute error (MAE) by the *DiffHetNet* method. Fig. 6.5a depicts the MAE under privacy budget ϵ varying from 0.6 to 1.0 on the *MIMIC-MultiType* dataset. It exhibits no change with the increase in ϵ. Fig. 6.5b depicts the MAE under privacy budget varying from 0.6 to 1.0 while fixing the data size to be $0.4 \times |V|$ on the *ca-GrQc*, *wiki-Vote*, and *MIMIC-T1* datasets. The absolute errors on the *ca-GrQc* dataset are slightly greater than the other datasets. However, they remain unchanged with the increase in ϵ and are consistently small on all datasets. Fig. 6.5c depicts the MAE under varying data size while fixing the privacy budget to be $\epsilon = 1.0$ on the *ca-GrQc*, *wiki-Vote*, and *MIMIC-T1* datasets. It generally decreases with the increase in size on all datasets. The results suggest that *DiffHetNet* well preserves the global structure of the anonymized network.

6.4.1.2 Average relative error

Fig. 6.6 presents the average relative error (ARE) by the *DiffHetNet* method. Fig. 6.6a depicts the ARE under privacy budget ϵ varying from 0.6 to 1.0 on the *MIMIC-MultiType* dataset. It generally

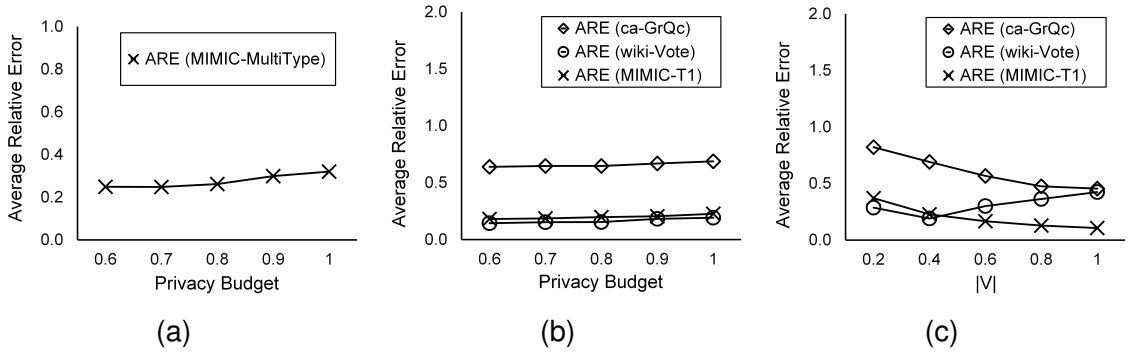


Figure 6.6: Average relative error by *DiffHetNet* method under varying ϵ in (a) and (b), and fixed $\epsilon = 1.0$ and varying data size in (c)

increases monotonically with the increase in ϵ. Fig. 6.6b depicts the ARE under varying privacy budget from 0.6 to 1.0 while fixing the data size to be $0.4 \times |V|$ on the *ca-GrQc*, *wiki-Vote*, and *MIMIC-T1* datasets. It exhibits nondecreasing monotonicity with the increase in ϵ on all datasets. The relative errors on the *ca-GrQc* dataset are higher than the other datasets because more vulnerable candidate nodes are protected in the network. Fig. 6.6c depicts the ARE under varying data size while fixing the privacy budget to be $\epsilon = 1.0$ on the *ca-GrQc*, *wiki-Vote*, and *MIMIC-T1* datasets. The relative errors generally decrease on *ca-GrQc* and *MIMIC-T1* datasets with the increase in data size, while on *wiki-Vote* they first decrease when data size increases from $0.2 \times |V|$ to $0.4 \times |V|$ and later increase with the increase in data size. The reason for this non-monotonicity is that the addition of noise considerably changes the degree-centrality scores for the potentially vulnerable nodes in the anonymized network.

Fig. 6.7 presents the comparison of different methods on average relative error (ARE). Figs. 6.7a and 6.7b depict the ARE of *DiffHetNet*, *DER*, *DE*, and *Random* under varying privacy budget ϵ from 0.6 to 1.0 while fixing the data size to be $0.4 \times |V|$ on *ca-GrQc* and *wiki-Vote* datasets. The relative errors of *DER* and its variant *DE*, when $k = 1$ (static correlation parameter) are smaller on both *ca-GrQc* and *wiki-Vote* datasets. However, their relative errors increase with an increase of k. It is observed that *DiffHetNet* performs better than *DER* when the correlation parameter $k = 20$, and it is closer to *DE* when $k = 1$ on the *wiki-Vote* dataset. The relative errors of *Random* are greater than the other methods in all settings. Our method *DiffHetNet* does not specify static correlation parameter k because of the dynamicity nature of the network. Figs. 6.7c and 6.7d depict the ARE of *DiffHetNet*

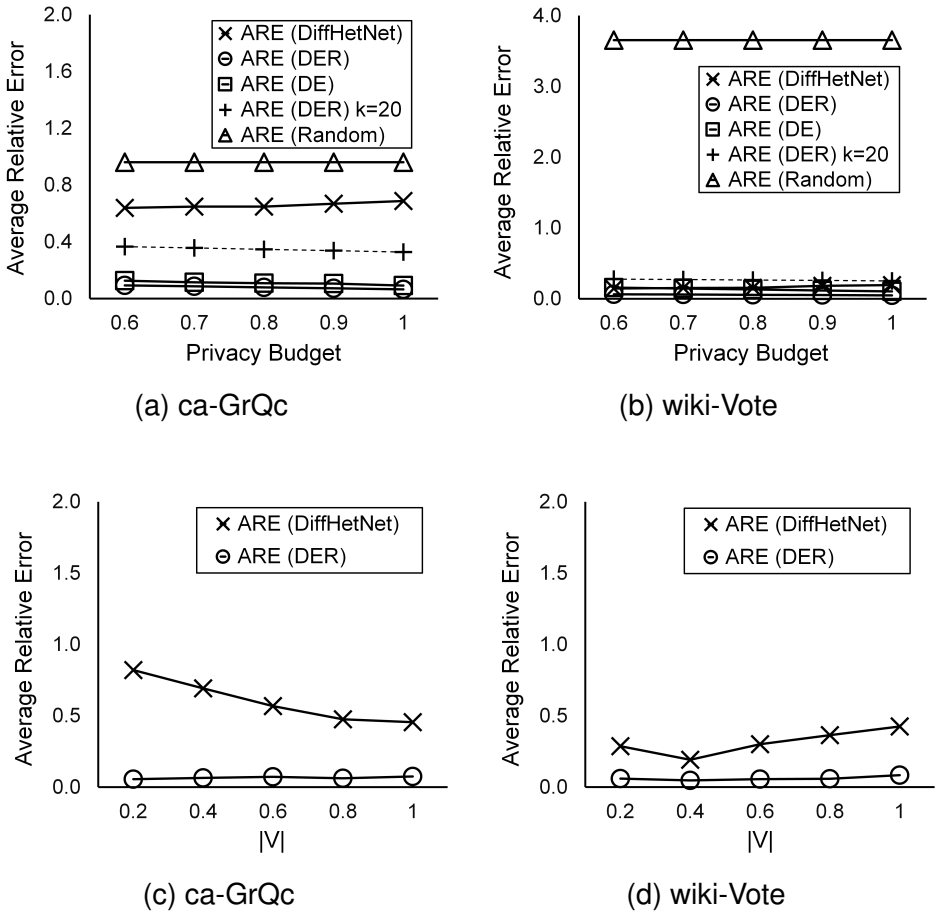


Figure 6.7: Comparison of *DiffHetNet*, *DER*, *DE*, and *Random* methods on average relative error under varying ϵ in (a) and (b), and *DiffHetNet* and *DER* on average relative error under varying data size in (c) and (d)

and *DER* under varying data size, while fixing the privacy budget to be $\epsilon = 1.0$ on *ca-GrQc* and *wiki-Vote* datasets. The relative errors of *DiffHetNet* decrease on *ca-GrQc* when data size increases, while on the *wiki-Vote* dataset they first decrease when data size increases from $0.2 \times |V|$ to $0.4 \times |V|$, and later increase with the increase in data size. The relative errors of the *DER* method on both datasets are small because the correlation parameter is set as low $k = 1$.

6.4.1.3 Kullback–Leibler divergence

Fig. 6.8 presents the KL-divergence by the *DiffHetNet* method. Fig. 6.8a depicts the KL-divergence under varying privacy budget ϵ from 0.6 to 1.0 on the *MIMIC-MultiType* dataset. It

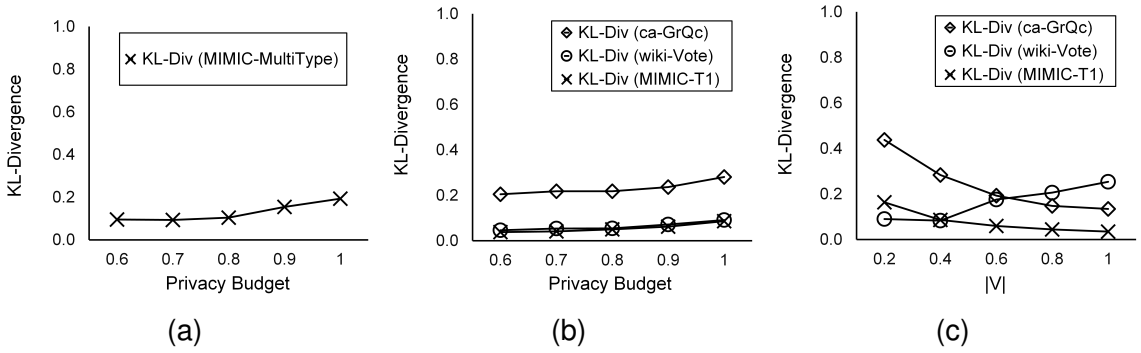


Figure 6.8: KL-Divergence by *DiffHetNet* method under varying ϵ in (a) and (b), and fixed $\epsilon = 1.0$ and varying data size in (c)

generally increases monotonically with the increase in ϵ. When $\epsilon = 1.0$, it reaches 0.19. Fig. 6.8b depicts the KL-divergence under varying privacy budget from 0.6 to 1.0 while fixing the data size to be $0.4 \times |V|$ on the *ca-GrQc*, *wiki-Vote*, and *MIMIC-T1* datasets. It exhibits nondecreasing monotonicity with the increase in ϵ on all datasets. The KL divergences on the *ca-GrQc* dataset are higher than the other datasets. The maximum difference on them is 0.19 when $\epsilon = 1.0$. Fig. 6.8c depicts the KL-divergence under varying data size while fixing the privacy budget to be $\epsilon = 1.0$ on the *ca-GrQc*, *wiki-Vote*, and *MIMIC-T1* datasets. The KL divergences exhibit decreasing monotonicity on *ca-GrQc* and *MIMIC-T1* datasets with the increase in data size, while they are not monotonic on *wiki-Vote* with the increase in data size.

Fig. 6.9 presents the comparison of different methods on KL-divergence. Figs. 6.9a and 6.9b depict the KL divergences of *DiffHetNet*, *DER*, *DE*, and *Random* under varying privacy budget ϵ from 0.6 to 1.0 while fixing the data size to be $0.4 \times |V|$ on *ca-GrQc*, and *wiki-Vote* datasets. In Fig. 6.9a, the KL divergences of *DER* when $k = 1$ (static correlation parameter) are small on the *ca-GrQc* dataset. However, they increase with an increase of k. It is observed that *DiffHetNet* performs better than *DER* when the correlation parameter $k = 20$, and closer to *DE* when $k = 1$ on the *ca-GrQc* dataset. Fig. 6.9b depicts that *DiffHetNet* outperforms all the other methods on the *wiki-Vote* dataset. A significant difference of $\gtrapprox 0.7$ in KL divergences can be observed between *DiffHetNet* and *DER* ($k = 20$) under varying ϵ. The KL divergences of *Random* are greater than the other methods in all settings on both datasets.

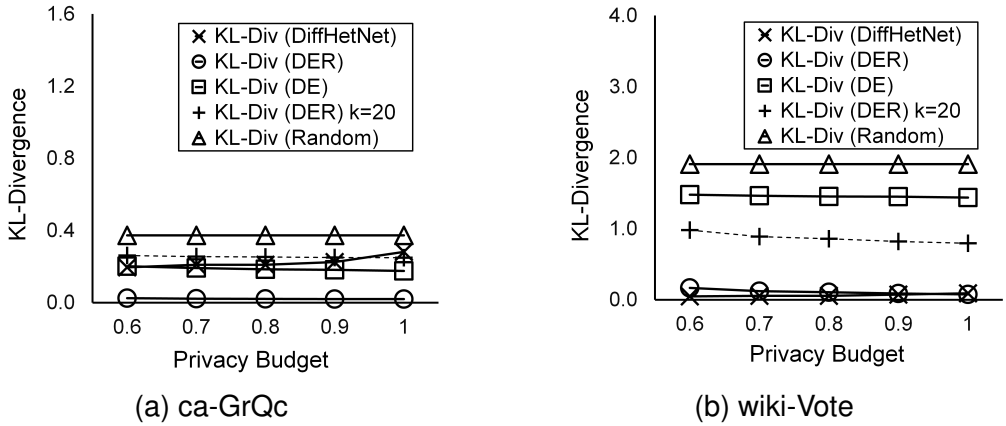
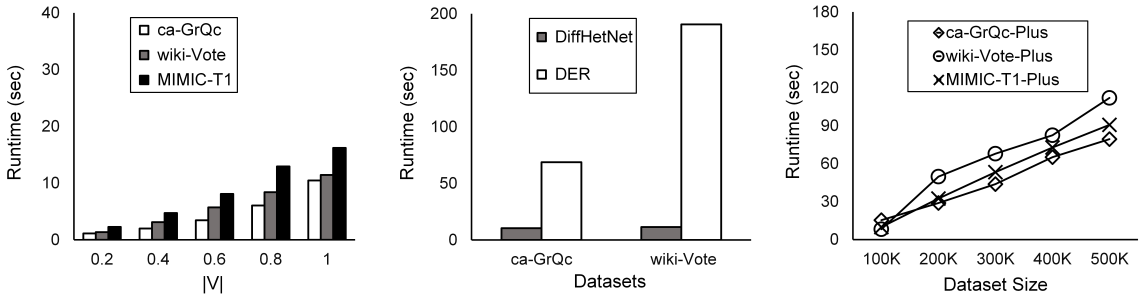


Figure 6.9: Comparison of *DiffHetNet*, *DER*, *DE*, and *Random* methods on KL-Divergence under varying ϵ in (a) and (b)

6.4.2 Efficiency

Fig. 6.10a depicts the runtime of the *DiffHetNet* method under varying data size $|V|$ while fixing the privacy budget to be $\epsilon = 1.0$ on the *ca-GrQc*, *wiki-Vote*, and *MIMIC-T1* datasets. We observe that on all three datasets runtime grows with the increase in data size from 0.2 to 1.0. The runtime to produce anonymization results by *DiffHetNet* on *MIMIC-T1* with $1.0 \times |V|$ data size is approximately 16 s. Fig. 6.10b depicts the comparison of *DiffHetNet* and *DER* methods on runtime when $\epsilon = 1.0$ and data size is $1.0 \times |V|$ on both *ca-GrQc* and *wiki-Vote* datasets. *DiffHetNet* takes approximately 10 s and 11 s on the *ca-GrQc* and *wiki-Vote* datasets, respectively. The results show that our method is more efficient in running time over the *DER* method. In Fig. 6.10c, we fix ϵ to 1.0 and evaluate the scalability of *DiffHetNet* using three datasets: *ca-GrQc-Plus*, *wiki-Vote-Plus*, and *MIMIC-T1-Plus*. The X-axis represents the number of records in thousands, ranging from $100,000$ to $500,000$ records. An edge going from one node to another node represents a single record. We consider no multiple edges (no duplicate records). For each 100K records, we add randomly-generated nodes and edges for *ca-GrQc* and *wiki-Vote* to extend their original size. We name these two extended datasets *ca-GrQc-Plus* and *wiki-Vote-Plus*, respectively. As for *MIMIC-T1-Plus*, this dataset is the result of extracting 500K records from the *MIMIC* data table (MIMIC-III v1.4), which contains $651,047$ records representing ICD (International Classification of Diseases) diagnoses for patients. The runtime of each dataset increases nearly linearly with respect to the increase in the size of the dataset.



(a) Varying data size with fixed $\epsilon = 1.0$

(b) Comparison of *DiffHetNet* and *DER*

(c) Scalability

Figure 6.10: Runtime comparison of *DiffHetNet*

This result suggests that our method is scalable to large network datasets.

6.5 Summary

In this chapter, we propose a practical solution to health information custodians (HICs) for publishing collected healthcare data to data recipients or researchers in a privacy-preserving manner. First, we model a complex de-identified healthcare dataset as a heterogeneous information network that consists of multi-type nodes and their multi-type edges. Then, we propose an edge-based differentially private algorithm to protect the sensitive links of patients from inbound and outbound attacks in the heterogeneous health network. We evaluate the performance of our method in terms of information utility and efficiency on different types of real-life datasets that can be modeled as networks. The experimental results suggest that our method generally yields less information loss and is significantly more efficient in terms of runtime compared to existing network anonymization methods. It is also evident from the experiments that our method is scalable to large network datasets.

Chapter 7

Conclusion and Future Directions

With the advancements in digital technology and the proliferation of online services, data is growing with a dizzying pace. Data has become an integral part of almost every industry such as finance, retail, travel, communications, healthcare, and government. Data often contains person-specific information. A data custodian who holds person-specific information must be responsible for managing the use, disclosure, accuracy and privacy protection of collected data. In this thesis, we presented three research problems. The first two problems addressed the concerns of stakeholders on privacy protection, data trustworthiness, and profit distribution in the e-market for trading person-specific data. The third problem addressed the health information custodians concern on preserving the privacy of health-network data publishing.

First, we propose a novel privacy-preserving data mashup model that allows the collaboration of multiple data providers to mashup their data over the cloud and to quantify and compare the costs and benefits for releasing integrated anonymized over an individual data provider when trading person-specific information in the e-market. To our knowledge, this is the first data mashup model that quantifies the costs and benefits of releasing integrated anonymized data in terms of monetary value. On one side, trading person-specific information comes with a high monetary value, but on the other side data providers who collaborate in sharing person-specific information need to be cautious of the risk of privacy breaches and cost of potential damages when integrating data. Our data mashup model allows the participating data providers to set up their joint privacy requirements during data mashup by choosing the privacy model along with the anonymization algorithm and

privacy parameters, and analyze the impact of anonymization on information utility for classification requirement in terms of monetary value after data mashup. The factors introduced in the model can help the data providers in defining the overall objective of maximizing net value. Furthermore, in the data mashup process the contribution of each data provider is derived from the achieved net value by fairly computing the information gain on the anonymized data. Our model helps data providers in finding the sub-optimal value by evaluating the benefits of data mashup and impacts of data anonymization based on the choices of privacy models and data mashup anonymization algorithms. It is evident from the experiments that the data provider whose data provides more information gain will get a proportionally higher share in terms of monetary value from the distribution of the achieved net value.

Second, we propose a novel solution to address the critical issues of data trustworthiness, privacy protection, and profit distribution for cloud-based data integration services. We present the first information entropy-based trust computation algorithm that allows a semi-trusted arbitrator to detect the covert behavior of a dishonest data provider, evaluates the trustworthiness of the participating data providers by a trust metric, and chooses the qualified providers for data mashup. Compared to the existing work on data trustworthiness [114, 115, 165], our proposed algorithm not only detects fabricated or incorrect data from a dishonest data provider during the verification process but also preserves the privacy of customers' data owned by a data provider. We achieve three main benefits in delegating the verification role to the semi-trusted cloud service provider. First, our method ensures that the cloud service provider cannot derive customers' private data from the information collected during the verification process. Second, the overhead of computation on the cloud server is also reduced because only an encrypted information gain message and its keyed hash are exchanged between a data provider and the cloud server, instead of exchanging encrypted individual data records during the verification process. Third, it also reduces the burden on data consumers to determine which data providers can serve their demands on requested attributes and what their attained trust scores are. Furthermore, we evaluate the robustness of our approach when a data provider employs machine learning method for imputation of missing values on its data. There is no significant difference in perspective to the performance of the imputation method. It is conditional to what proportion of data is missing and whether the data contains repeated patterns. If the prediction of a

missing data happens to be precise data, it will be considered as true data. We incorporate the VCG auction mechanism for the valuation of data providers' attributes into the data mashup process. It maximizes the total valuation obtained by data providers since there is no incentive to lie or deviate from truthful reporting. From the perspective of privacy protection, the accepted data providers as a result of trust computation set up their joint privacy requirements for the data mashup. During the data mashup process, data providers compete among themselves for higher data utility. It is evident from the experiments that an accepted data provider whose data attributes result in more information gain, and whose trust level is higher than the other competitors, can get a proportionally larger share of the monetary value. Furthermore, our method provides better runtime efficiency over provenance-based approaches [40, 114].

Finally, we propose a practical solution to HICs for publishing the healthcare heterogeneous network data to a data miner or recipient in a privacy-preserving manner. We first model a complex de-identified healthcare dataset as a heterogeneous information network that consists of multi-type nodes and their multi-type edges. Then, we propose an edge-based differentially private algorithm to protect the sensitive links of a patient from inbound and outbound attacks in the heterogeneous health network, which to our knowledge has never been addressed before. We evaluate the performance of our method in terms of information utility and efficiency on different types of real-life datasets that can be modeled as networks. It is evident from the experiments that our method generally yields less information loss as well as significant efficiency gain in terms of runtime compared to existing anonymization methods. In addition, our method is scalable to large network datasets.

Broadly, the above-discussed thesis contributions are effective to serve the requirements of commercial and non-profit organizations that are inspired by the practical and real-world needs of the stakeholders. This thesis is a one-step towards solving some interesting research gaps in the literature. However, we find some points that open the doors for further research. We summarize some future directions as follows.

A future data publishing problem could consider other types of data, such as transaction data [36], trajectory data [12, 13, 67], and social network data [16, 83] when addressing the real-world challenges of privacy protection, data trustworthiness, and profit distribution among multiple parties or agents for integrating data.

Another research direction could be to consider anonymizing multiple heterogeneous networks where the challenges are to develop a robust anonymization method that ensures the protection of private information in a way that the individual information held by one data custodian should not be revealed to another custodian who is not authorized to acquire such information at any stage of anonymization. Besides, representation learning techniques [172, 187] can be explored to extract hidden network properties that can lead to developing a robust method in order to counter the underlying privacy threats and mitigate the potential risks.

Bibliography

[1] Review of the Personal Data (Privacy) Ordinance. Office of the Privacy Commissioner for Personal Data, Hong Kong, 2009.

[2] Personal Data Privacy and Security Act. Bill S.1151 - 112th Congress in the Senate of the United States, 2011.

[3] Data Management Platforms Buyer's Guide. Econsultancy Digital Marketing Excellence, 2013.

[4] Exploring the Economics of Personal Data: A Survey of Methodologies for Measuring Monetary Value. *OECD Digital Economy Papers*, (220), 2013.

[5] A Legal Guide to Privacy and Data Security. Minnesota Department of Employment and Economic Development, Gray Plant Mooty, 2014.

[6] Data Partners. Seventh Point, 2014. URL http://www.seventhpoint.com/whitepaper/data-partners/. Last accessed: June 11, 2015.

[7] Cost of a Data Breach Report. Ponemon Institute LLC, 2019. Sponsored by IBM Security.

[8] Karim Abouelmehdi, Abderrahim Beni-Hessane, and Hayat Khaloufi. Big Healthcare Data: Preserving Security and Privacy. *Journal of Big Data*, 5(1):1–18, 2018.

[9] Alessandro Acquisti, Allan Friedman, and Rahul Telang. Is There a Cost to Privacy Breaches? An Event Study. In *Proceedings of the 27th International Conference on Information*, 2006.

[10] Charu C. Aggarwal. On k-Anonymity and the Curse of Dimensionality. In *Proceedings of the 31st International Conference on Very Large Data Bases*, pages 901–909. VLDB Endowment, 2005.

[11] Rakesh Agrawal, Alexandre Evfimievski, and Ramakrishnan Srikant. Information Sharing across Private Databases. In *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data*, pages 86–97. ACM, 2003.

[12] Khalil Al-Hussaeni, Benjamin C. M. Fung, and William K. Cheung. Privacy-preserving Trajectory Stream Publishing. *Data and Knowledge Engineering*, 94:89–109, 2014.

[13] Khalil Al-Hussaeni, Benjamin C. M. Fung, Farkhund Iqbal, Gaby G. Dagher, and Eun G. Park. SafePath: Differentially Private Publishing of Passenger Trajectories in Transportation Systems. *Computer Networks*, 143:126–139, 2018.

[14] Dima Alhadidi, Noman Mohammed, Benjamin C. M. Fung, and Mourad Debbabi. Secure Distributed Framework for Achieving ϵ-Differential Privacy. In *Proceedings of the 12th International Symposium on Privacy Enhancing Technologies*, pages 120–139. Springer, 2012.

[15] Hussain Aljafer, Zaki Malik, Mohammed Alodib, and Abdelmounaam Rezgui. A Brief Overview and an Experimental Evaluation of Data Confidentiality Measures on the Cloud. *Journal of Innovation in Digital Ecosystems*, 1(1–2):1–11, 2014.

[16] Sarah A. Alkhodair, Steven H. H. Ding, Benjamin C. M. Fung, and Junqiang Liu. Detecting Breaking News Rumors of Emerging Topics in Social Media. *Information Processing and Management*, 57(2):102018, 2020.

[17] Rebecca R. Andridge and Roderick J. A. Little. A Review of Hot Deck Imputation for Survey Non-response. *International Statistical Review*, 78(1):40–64, 2010.

[18] Marco Anisetti, Claudio A. Ardagna, and Ernesto Damiani. A Certification-based Trust Model for Autonomic Cloud Computing Systems. In *Proceedings of the 2014 International Conference on Cloud and Autonomic Computing*, pages 212–219, 2014.

[19] Mahtab Arafati, Gaby G. Dagher, Benjamin C. M. Fung, and Patrick C. K. Hung. D-Mash: A Framework for Privacy-preserving Data-as-a-Service Mashups. In *Proceedings of the 7th IEEE International Conference on Cloud Computing*, pages 498–505. IEEE Computer Society, 2014.

[20] Roland Assam, Marwan Hassani, Michael Brysch, and Thomas Seidl. (k, d)-Core Anonymity: Structural Anonymization of Massive Networks. In *Proceedings of the 26th International Conference on Scientific and Statistical Database Management*. ACM, 2014.

[21] Yonatan Aumann and Yehuda Lindell. Security Against Covert Adversaries: Efficient Protocols for Realistic Adversaries. *Journal of Cryptology*, 23(2):281–343, 2010.

[22] Paige Backman and Karen Levin. Privacy Breaches - Impact, Notification and Strategic Plans. Aird and Berlis LLP, 2011.

[23] Lars Backstrom, Cynthia Dwork, and Jon Kleinberg. Wherefore Art Thou R3579x? Anonymized Social Networks, Hidden Patterns, and Structural Steganography. In *Proceedings of the 16th International Conference on World Wide Web*, page 181–190. ACM, 2007.

[24] Omar Benjelloun, Anish D. Sarma, Alon Halevy, Martin Theobald, and Jennifer Widom. Databases with Uncertainty and Lineage. *The VLDB Journal*, 17(2):243–264, 2008.

[25] Lorenzo Beretta and Alessandro Santaniello. Nearest Neighbor Imputation Algorithms: A Critical Evaluation. *BMC Medical Informatics and Decision Making*, 16(3):74, 2016.

[26] Bibi V. D. Berg and Esther Keymolen. Regulating Security on the Internet: Control versus Trust. *International Review of Law, Computers and Technology*, 31(2):188–205, 2017.

[27] Elisa Bertino and Hyo-Sang Lim. Assuring Data Trustworthiness: Concepts and Research Challenges. In *Proceedings of the 7th VLDB Conference on Secure Data Management*, pages 1–12. Springer, 2010.

[28] Elisa Bertino and Ravi Sandhu. Database Security - Concepts, Approaches, and Challenges. *IEEE Transactions on Dependable and Secure Computing*, 2(1):2–19, 2005.

[29] Elisa Bertino, Lorenzo Martino, Federica Paci, and Anna Squicciarini. *Security for Web Services and Service-Oriented Architectures*. Springer, 1st edition, 2009.

[30] Ann Bevitt, Karin Retzer, and Joanna Lopatowska. Dealing with Data Breaches in Europe and Beyond. Practical Law Company, 2012.

[31] Anthony E. Boardman, David H. Greenberg, Aidan R. Vining, and David L. Weimer. *Cost-Benefit Analysis: Concepts and Practice*. Pearson Prentice Hall, 2006.

[32] Dan Boneh and Victor Shoup. *A Graduate Course in Applied Cryptography*. 2017.

[33] Christian Borgs, Jennifer T. Chayes, and Adam Smith. Private Graphon Estimation for Sparse Graphs. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, pages 1369–1377. MIT Press, 2015.

[34] Jordi Casas-Roma, Julian Salas, Fragkiskos D. Malliaros, and Michalis Vazirgiannis. k-Degree Anonymity on Directed Networks. *Knowledge and Information Systems*, 61(3):1743–1768, 2018.

[35] Victor Chang, Yen-Hung Kuo, and Muthu Ramachandran. Cloud Computing Adoption Framework: A Security Framework for Business Clouds. *Future Generation Computer Systems*, 57:24–41, 2016.

[36] Rui Chen, Noman Mohammed, Benjamin C. M. Fung, Bipin C. Desai, and Li Xiong. Publishing Set-Valued Data via Differential Privacy. *The Proceedings of the VLDB Endowment*, 4 (11):1087–1098, 2011.

[37] Rui Chen, Benjamin C. M. Fung, Philip S. Yu, and Bipin C. Desai. Correlated Network Data Publication via Differential Privacy. *The International Journal on Very Large Data Bases*, 23 (4):653–676, 2014.

[38] James Cheng, Ada W. Fu, and Jia Liu. K-isomorphism: Privacy-preserving Network Publication Against Structural Attacks. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 459–470. ACM, 2010.

[39] Wendy L. Currie and Jonathan J. M. Seddon. A Cross-Country Study of Cloud Computing Policy and Regulation in Healthcare. In *Proceedings of the 22nd European Conference on Information Systems*, 2014.

[40] Chenyun Dai, Dan Lin, Elisa Bertino, and Murat Kantarcioglu. An Approach to Evaluate Data Trustworthiness Based on Data Provenance. In *Secure Data Management*, pages 82–98. Springer, 2008.

[41] Tore Dalenius and Steven P. Reiss. Data-Swapping: A Technique for Disclosure Control. *Journal of Statistical Planning and Inference*, 6(1):73–85, 1982.

[42] Wei-Yen Day, Ninghui Li, and Min Lyu. Publishing Graph Degree Distribution with Node Differential Privacy. In *Proceedings of the 2016 International Conference on Management of Data*, page 123–138. ACM, 2016.

[43] Emiliano De-Cristofaro, Paolo Gasti, and Gene Tsudik. Fast and Private Computation of Cardinality of Set Intersection and Union. In *Proceedings of the 11th International Conference on Cryptology and Network Security*, pages 218–231. Springer, 2012.

[44] Department of Health and Human Services. Modifications to the HIPAA Privacy, Security, Enforcement, and Breach Notification Rules Under the HITECH Act and the GINA Act; other Modifications to the HIPAA Rules. (78 FR 5565):5565–5702, 2013.

[45] Tim Dierks and Eric Rescorla. The Transport Layer Security (TLS) Protocol Version 1.2. RFC 5246, 2008.

[46] Josep Domingo-Ferrer and Josep M. Mateo-Sanz. Practical Data-Oriented Microaggregation for Statistical Disclosure Control. *IEEE Transactions on Knowledge and Data Engineering*, 14(1):189–201, 2002.

[47] Changyu Dong, Liqun Chen, Jan Camenisch, and Giovanni Russello. Fair Private Set Intersection with a Semi-Trusted Arbiter. In *Proceedings of the 27th International Conference on Data and Applications Security and Privacy*, pages 128–144. Springer, 2013.

[48] Changyu Dong, Liqun Chen, and Zikai Wen. When Private Set Intersection Meets Big data: An Efficient and Scalable Protocol. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer and Communications Security*, pages 789–800. ACM, 2013.

[49] Wenliang Du and Zhijun Zhan. Building Decision Tree Classifier on Private Data. In *Proceedings of the 2002 IEEE International Conference on Privacy, Security and Data Mining*, volume 14, pages 1–8. Australian Computer Society, Inc., 2002.

[50] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating Noise to Sensitivity in Private Data Analysis. In *Proceedings of the 3rd Conference on Theory of Cryptography*, pages 265–284. Springer, 2006.

[51] David Easley and Jon Kleinberg. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge University Press, 2010.

[52] Experian Information Solutions, Inc. The 2018 State of Data Management: A Public Sector Benchmark Report, 2018.

[53] Claudio Feijóo, José L. Gómez-Barroso, and Peter Voigt. Exploring the Economic Value of Personal Information from Firms' Financial Statements. *International Journal of Information Management*, 34(2):248–256, 2014.

[54] Raphael A. Finkel and Jon L. Bentley. Quad Trees a Data Structure for Retrieval on Composite Keys. *Acta Informatica*, 4(1):1–9, 1974.

[55] Michael J. Freedman, Kobbi Nissim, and Benny Pinkas. Efficient Private Matching and Set Intersection. In *Proceedings of the 2004 International Conference on the Theory and Applications of Cryptographic Techniques, Advances in Cryptology - EUROCRYPT*, pages 1–19. Springer, 2004.

[56] Julien Freudiger, Shantanu Rane, Alejandro E. Brito, and Ersin Uzun. Privacy-preserving Data Quality Assessment for High-fidelity Data Sharing. In *Proceedings of the 2014 ACM Workshop on Information Sharing and Collaborative Security*, pages 21–29. ACM, 2014.

[57] Arik Friedman and Assaf Schuster. Data Mining with Differential Privacy. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 493–502. ACM, 2010.

[58] Tao-yang Fu, Wang-Chien Lee, and Zhen Lei. HIN2Vec: Explore Meta-paths in Heterogeneous Information Networks for Representation Learning. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, page 1797–1806. ACM, 2017.

[59] Benjamin C. M. Fung, Ke Wang, and Philip S. Yu. Anonymizing Classification Data for Privacy Preservation. *IEEE Transactions on Knowledge Data Engineering*, 19(5):711–725, 2007.

[60] Benjamin C. M. Fung, Khalil Al-Hussaeni, and Ming Cao. Preserving RFID Data Privacy. In *Proceedings of the 2009 IEEE International Conference on RFID*, pages 200–207. IEEE Communications Society, 2009.

[61] Benjamin C. M. Fung, Ke Wang, Rui Chen, and Philip S. Yu. Privacy-preserving Data Publishing: A Survey of Recent Developments. *ACM Computing Survey*, 42(4):1–53, 2010.

[62] Benjamin C. M. Fung, Thomas Trojer, Patrick C. K. Hung, Li Xiong, Khalil Al-Hussaeni, and Rachida Dssouli. Service-Oriented Architecture for High-dimensional Private Data Mashup. *IEEE Transactions on Services Computing*, 5(3):373–386, 2012.

[63] Benjamin C. M. Fung, Yan'an Jin, Jiaming Li, and Junqiang Liu. *Recommendation and Search in Social Networks*, chapter Anonymizing Social Network Data for Maximal Frequent-Sharing Pattern Mining, pages 77–100. Springer, 2015.

[64] Srivatsava R. Ganta, Shiva P. Kasiviswanathan, and Adam Smith. Composition Attacks and Auxiliary Information in Data Privacy. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 265–273. ACM, 2008.

[65] Carrie Gates and Peter Matthews. Data Is the New Currency. In *Proceedings of the 2014 New Security Paradigms Workshop*, pages 105–116. ACM, 2014.

[66] Johannes Gehrke. Programming with Differential Privacy: Technical Perspective. *Communications of the ACM*, 53(9), 2010.

[67] Moein Ghasemzadeh, Benjamin C. M. Fung, Rui Chen, and Anjali Awasthi. Anonymizing Trajectory Data for Passenger Flow Analysis. *Transportation Research Part C: Emerging Technologies*, 39:63–79, 2014.

[68] Oded Goldreich. *Foundations of Cryptography: Basic Applications*, volume 2. Cambridge University Press, 2004.

[69] Aditya Grover and Jure Leskovec. Node2vec: Scalable Feature Learning for Networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 855–864. ACM, 2016.

[70] Huan Gui, Jialu Liu, Fangbo Tao, Meng Jiang, Brandon Norick, Lance Kaplan, and Jiawei Han. Embedding Learning with Events in Heterogeneous Information Networks. *IEEE Transactions on Knowledge and Data Engineering*, 29(11):2428–2441, 2017.

[71] Kholekile L. Gwebu, Jing Wang, and Wenjuan Xie. Understanding the Cost Associated with Data Security Breaches. In *Proceedings of the 18th Pacific Asia Conference on Information Systems*, 2014.

[72] Michael Hay, Chao Li, Gerome Miklau, and David Jensen. Accurate Estimation of the Degree Distribution of Private Networks. In *Proceedings of the 9th IEEE International Conference on Data Mining*, pages 169–178. IEEE Computer Society, 2009.

[73] Rebecca Herold and Kevin Beaver. *The Practical Guide to HIPAA Privacy and Security Compliance*. Auerbach, 2nd edition, 2014.

[74] Jack Hirshleifer, Amihai Glazer, and David Hirshleifer. *Price Theory and Applications: Decisions, Markets, and Information*. Cambridge University Pres, 7th edition, 2005.

[75] Bijit Hore, Ravi C. Jammalamadaka, and Sharad Mehrotra. Flexible Anonymization for Privacy-preserving Data Publishing: A Systematic Search Based Approach. In *Proceedings of the 7th SIAM International Conference on Data Mining*, 2007.

[76] Jing Hu, Jun Yan, Zhen-Qiang Wu, Hai Liu, and Yi-Hui Zhou. A Privacy-preserving Approach in Friendly-Correlations of Graph Based on Edge-Differential Privacy. *Journal of Information Science and Engineering*, 35(4):821–837, 2019.

[77] Yuh-Jong Hu, Win-Nan Wu, and Di-Rong Cheng. Towards Law-aware Semantic Cloud Policies with Exceptions for Data Integration and Protection. In *Proceedings of the 2nd International Conference on Web Intelligence, Mining and Semantics*, pages 1–12. ACM, 2012.

[78] Yan Huang, David Evans, and Jonathan Katz. Private Set Intersection: Are Garbled Circuits Better than Custom Protocols? In *Proceedings of the 19th Network and Distributed System Security Symposium*. The Internet Society, 2012.

[79] Lynette A. Hunt. Missing Data Imputation and Its Effect on the Accuracy of Classification. In *Data Science*, pages 3–14. Springer, 2017.

[80] Yuval Ishai, Joe Kilian, Kobbi Nissim, and Erez Petrank. Extending Oblivious Transfers Efficiently. In *Proceedings of the CRYPTO 2003 on Advances in Cryptology*, pages 145–161. Springer, 2003.

[81] Ming Ji, Jiawei Han, and Marina Danilevsky. Ranking-based Classification of Heterogeneous Information Networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 1298–1306. ACM, 2011.

[82] Wei Jiang and Chris Clifton. A Secure Distributed Framework for Achieving k-Anonymity. *The VLDB Journal*, 15(4):316–333, 2006.

[83] Wenjun Jiang, Guojun Wang, and Jie Wu. Generating Trusted Graphs for Trust Evaluation in Online Social Networks. *Future Generation Computer Systems*, 31:48–58, 2014. Special Section: Advances in Computer Supported Collaboration: Systems and Technologies.

[84] Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Li-wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo A. Celi, and Roger G. Mark. MIMIC-III, A Freely Accessible Critical Care Database. *Scientific Data*, 3:160035, 2016.

[85] Zach Jorgensen, Ting Yu, and Graham Cormode. Publishing Attributed Social Graphs with Formal Privacy Guarantees. In *Proceedings of the International Conference on Management of Data*, pages 107–122. ACM, 2016.

[86] Kevin Judd, Michael Small, and Thomas Stemler. What Exactly are the Properties of Scale-Free and Other Networks? *EPL (Europhysics Letters)*, 103(5):58004, 2013.

[87] Pawel Jurczyk and Li Xiong. Distributed Anonymization: Achieving Privacy for Both Data Subjects and Data Providers. In *Proceedings of the 23rd Annual IFIP WG 11.3 Working Conference on Data and Applications Security*, pages 191–207. Springer, 2009.

[88] Audun Jøsang, Roslan Ismail, and Colin Boyd. A Survey of Trust and Reputation Systems for Online Service Provision. *Decision Support Systems*, 43(2):618–644, 2007. Emerging Issues in Collaborative Commerce.

[89] Seny Kamara, Payman Mohassel, and Ben Riva. Salus: A System for Server-aided Secure Function Evaluation. In *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, pages 797–808. ACM, 2012.

[90] Seny Kamara, Payman Mohassel, Mariana Raykova, and Saeed Sadeghian. Scaling Private Set Intersection to Billion-Element Sets. In *Proceedings of the 2014 Financial Cryptography and Data Security*, pages 195–215. Springer, 2014.

[91] Selcuk Karabati and Zehra B. Yalcin. An Auction Mechanism for Pricing and Capacity Allocation with Multiple Products. *Production and Operations Management*, 23(1):81–94, 2014.

[92] Shiva P. Kasiviswanathan, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. Analyzing Graphs with Node Differential Privacy. In *Proceedings of the 10th Theory of Cryptography Conference on Theory of Cryptography*, pages 457–476. Springer, 2013.

[93] Vishal Kher and Yongdae Kim. Securing Distributed Storage: Challenges, Techniques, and Systems. In *Proceedings of the 2005 ACM Workshop on Storage Security and Survivability*, pages 9–25. ACM, 2005.

[94] Rashid H. Khokhar, Rui Chen, Benjamin C. M. Fung, and Siu M. Lui. Quantifying the Costs and Benefits of Privacy-preserving Health Data Publishing. *Journal of Biomedical Informatics*, 50:107–121, 2014. Special Issue on Informatics Methods in Medical Privacy.

[95] Rashid H. Khokhar, Benjamin C. M. Fung, Farkhund Iqbal, Dima Alhadidi, and Jamal Bentahar. Privacy-preserving Data Mashup Model for Trading Person-specific Information. *Electronic Commerce Research and Applications*, 17:19–37, 2016.

[96] Rashid H. Khokhar, Farkhund Iqbal, Benjamin C. M. Fung, and Jamal Bentahar. Enabling Secure Trustworthiness Assessment and Privacy Protection in Integrating Data for Trading Person-specific Information. *IEEE Transactions on Engineering Management*, pages 1–21, 2020.

[97] Ritu Khullar and Vanessa Cosco. Conceptualizing the Right to Privacy in Canada. National Administrative Law, Labour & Employment Law and Privacy & Access Law PD Conference, 2010.

[98] Daniel Kifer. Attacks on Privacy and DeFinetti's Theorem. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data*, pages 127–138. ACM, 2009.

[99] Daniel Kifer and Johannes Gehrke. Injecting Utility into Anonymized Datasets. In *Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data*, page 217–228. ACM, 2006.

[100] Daniel Kifer and Bing-Rong Lin. Towards an Axiomatization of Statistical Privacy and Utility. In *Proceedings of the 29th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, page 147–158. ACM, 2010.

[101] Daniel Kifer and Ashwin Machanavajjhala. No Free Lunch in Data Privacy. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 193–204. ACM, 2011.

[102] Dan J. Kim, Donald L. Ferrin, and Raghav Rao. A Trust-based Consumer Decision-making

Model in Electronic Commerce: The Role of Trust, Perceived Risk, and their Antecedents. *Decision Support Systems*, 44(2):544–564, 2008.

[103] Jay Kim. A Method for Limiting Disclosure in Microdata Based on Random Noise and Transformation. In *Proceedings of the Section on Survey Research Methods*, pages 303–308. American Statistical Association, 1986.

[104] Vladimir Kolesnikov, Ranjit Kumaresan, Mike Rosulek, and Ni Trieu. Efficient Batched Oblivious PRF with Applications to Private Set Intersection. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 818–829. ACM, 2016.

[105] Ioannis Konstas, Vassilios Stathopoulos, and Joemon M. Jose. On Social Networks and Collaborative Recommendation. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 195–202. ACM, 2009.

[106] Peter Kooiman, Leon Willenborg, and Jose Gouweleeuw. *PRAM: A Method for Disclosure Limitation of Microdata*. Number 9705 in Research paper. CBS, 1997.

[107] Christopher Kuner. Regulation of Transborder Data Flows under Data Protection and Privacy Law: Past, Present and Future. *OECD Publishing*, (187), 2011.

[108] Andrea Landherr, Bettina Friedl, and Julia Heidemann. A Critical Review of Centrality Measures in Social Networks. *Business and Information Systems Engineering*, 2(6):371–385, 2010.

[109] Kristen LeFevre, David J. DeWitt, and Raghu Ramakrishnan. Workload-aware Anonymization. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 277–286. ACM, 2006.

[110] Chao Li, Daniel Y. Li, Gerome Miklau, and Dan Suciu. A Theory of Pricing Private Data. *ACM Transactions on Database Systems*, 39(4):1–28, 2014.

[111] Jingquan Li. Privacy Policies for Health Social Networking Sites. *Journal of the American Medical Informatics Association*, 20(4):704–707, 2013.

[112] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. t-Closeness: Privacy Beyond k-Anonymity and ℓ-Diversity. In *Proceedings of the 23rd IEEE International Conference on Data Engineering*, pages 106–115, 2007.

[113] Xuejun Li, Ruimiao Ding, Xiao Liu, Xiangjun Liu, Erzhou Zhu, and Yunxiang Zhong. A Dynamic Pricing Reverse Auction-based Resource Allocation Mechanism in Cloud Workflow Systems. *Scientific Programming*, 2016:1–13, 2016.

[114] Hyo-Sang Lim, Yang-Sae Moon, and Elisa Bertino. Provenance-based Trustworthiness Assessment in Sensor Networks. In *Proceedings of the 7th International Workshop on Data Management for Sensor Networks*, pages 2–7. ACM, 2010.

[115] Hyo-Sang Lim, Gabriel Ghinita, Elisa Bertino, and Murat Kantarcioglu. A Game-Theoretic Approach for High-assurance of Data Trustworthiness in Sensor Networks. In *Proceedings of the 28th IEEE International Conference on Data Engineering*, pages 1192–1203, 2012.

[116] Zijie Lin, Liangliang Gao, Xuexian Hu, Yuxuan Zhang, and Wenfen Liu. Differentially Private Graph Clustering Algorithm Based on Structure Similarity. In *Proceedings of the 2019 the 9th International Conference on Communication and Network Security*, page 63–68. ACM, 2019.

[117] Yehuda Lindell and Benny Pinkas. Secure Multiparty Computation for Privacy-preserving Data Mining. *Journal of Privacy and Confidentiality*, 1(1):59–98, 2009.

[118] Roderick J. A. Little. Statistical Analysis of Masked Data. *Journal of Official Statistics*, 9(2): 407–426, 1993.

[119] Kun Liu and Evimaria Terzi. Towards Identity Anonymization on Graphs. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 93–106. ACM, 2008.

[120] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. ℓ-Diversity: Privacy Beyond k-Anonymity. *ACM Transactions on Knowledge Discovery from Data*, 1(1), 2007.

[121] Frank McSherry. Privacy Integrated Queries: An Extensible Platform for Privacy-Preserving Data Analysis. *Communications of the ACM*, 53(9):89–97, 2010.

[122] Frank McSherry and Kunal Talwar. Mechanism Design via Differential Privacy. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, pages 94–103. IEEE Computer Society, 2007.

[123] Noman Mohammed, Benjamin C. M. Fung, Patrick C. K. Hung, and Cheuk-Kwong Lee. Anonymizing Healthcare Data: A Case Study on the Blood Transfusion Service. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1285–1294. ACM, 2009.

[124] Noman Mohammed, Benjamin C. M. Fung, Patrick C. K. Hung, and Cheuk-Kwong Lee. Centralized and Distributed Anonymization for High-dimensional Healthcare Data. *ACM Transactions on Knowledge Discovery from Data*, 4(4):1–33, 2010.

[125] Noman Mohammed, Rui Chen, Benjamin C. M. Fung, and Philip S. Yu. Differentially Private Data Release for Data Mining. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 493–501. ACM, 2011.

[126] Noman Mohammed, Benjamin C. M. Fung, and Mourad Debbabi. Anonymity Meets Game Theory: Secure Data Integration with Malicious Participants. *The VLDB Journal*, 20(4): 567–588, 2011.

[127] Noman Mohammed, Xiaoqian Jiang, Rui Chen, Benjamin C. M. Fung, and Lucila Ohno-Machado. Privacy-preserving Heterogeneous Health Data Sharing. *Journal of the American Medical Informatics Association*, 20(3):462–469, 2013.

[128] Noman Mohammed, Dima Alhadidi, Benjamin C. M. Fung, and Mourad Debbabi. Secure Two-Party Differentially Private Data Release for Vertically Partitioned Data. *IEEE Transactions on Dependable and Secure Computing*, 11(1):59–71, 2014.

[129] Rachana Nget, Yang Cao, and Masatoshi Yoshikawa. How to Balance Privacy and Money

through Pricing Mechanism in Personal Data Market. In *Proceedings of the SIGIR 2017 Workshop on eCommerce*, volume 2311. CEUR-WS.org, 2017.

[130] Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. Smooth Sensitivity and Sampling in Private Data Analysis. In *Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing*, page 75–84. ACM, 2007.

[131] Talal H. Noor, Quan Z. Sheng, Lina Yao, Schahram Dustdar, and Anne H. H. Ngu. CloudArmor: Supporting Reputation-based Trust Management for Cloud Services. *IEEE Transactions on Parallel and Distributed Systems*, 27(2):367–380, 2016.

[132] Benny Pinkas, Thomas Schneider, and Michael Zohner. Faster Private Set Intersection Based on OT Extension. In *Proceedings of the 23rd USENIX Conference on Security Symposium*, pages 797–812. USENIX Association, 2014.

[133] Benny Pinkas, Thomas Schneider, Gil Segev, and Michael Zohner. Phasing: Private Set Intersection Using Permutation-based Hashing. In *Proceedings of the 24th USENIX Conference on Security Symposium*, pages 515–530. USENIX Association, 2015.

[134] Benny Pinkas, Thomas Schneider, and Michael Zohner. Scalable Private Set Intersection Based on OT Extension. *ACM Transactions on Privacy Security*, 21(2):1–35, 2018.

[135] Yu Pu and Jens Grossklags. Valuating Friends' Privacy: Does Anonymity of Sharing Personal Data Matter? In *Proceedings of the 13th Symposium on Usable Privacy and Security (SOUPS)*, pages 339–355. USENIX Association, 2017.

[136] John R. Quinlan. Induction of Decision Trees. *Machine Learning*, 1(1):81–106, 1986.

[137] John R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.

[138] Dino Quintero, William M. Genovese, KiWaon Kim, Ming J. M. Li, Fabio Martins, Ashish Nainwal, Dusan Smolej, Marcin Tabinowski, and Ashu Tiwary. *IBM Software Defined Environment*. IBM Digital Services Group, Technical Content Services (TCS), 1st edition, 2015.

[139] Sofya Raskhodnikova and Adam Smith. Lipschitz Extensions for Node-Private Graph Statistics and the Generalized Exponential Mechanism. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 495–504, 2016.

[140] Eric Rescorla. The Transport Layer Security (TLS) Protocol Version 1.3. RFC 8446, 2018.

[141] Leonardo F. R. Ribeiro, Pedro H. P. Saverese, and Daniel R. Figueiredo. Struc2vec: Learning Node Representations from Structural Identity. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 385–394. ACM, 2017.

[142] Christopher Riederer, Vijay Erramilli, Augustin Chaintreau, Balachander Krishnamurthy, and Pablo Rodriguez. For Sale : Your Data: By : You. In *Proceedings of the 10th ACM Workshop on Hot Topics in Networks*, pages 1–6. ACM, 2011.

[143] Kevin Roebuck. *Enterprise Mashups: High-impact Strategies - What You Need to Know: Definitions, Adoptions, Impact, Benefits, Maturity, Vendors*. Emereo Publishing, 2012.

[144] Sasha Romanosky and Alessandro Acquisti. Privacy Costs and Personal Data Protection: Economic and Legal Perspectives. *Berkeley Technology Law Journal*, 24(4), 2014.

[145] Sasha Romanosky, David Hoffman, and Alessandro Acquisti. Empirical Analysis of Data Breach Litigation. *Journal of Empirical Legal Studies*, 11(1):74–104, 2014.

[146] Alessandra Sala, Xiaohan Zhao, Christo Wilson, Haitao Zheng, and Ben Y. Zhao. Sharing Graphs using Differentially Private Graph Models. In *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, page 81–98. ACM, 2011.

[147] Pierangela Samarati. Protecting Respondents' Identities in Microdata Release. *IEEE Transactions on Knowledge and Data Engineering*, 13(6):1010–1027, 2001.

[148] Rizwana Shaikh and Sasikumar Mukundan. Trust Model for Measuring Security Strength of Cloud Computing Service. *Procedia Computer Science*, 45:380–389, 2015.

[149] Claude E. Shannon. The Mathematical Theory of Communication. 1949.

[150] Ahmed Shawish and Maria Salama. *Inter-cooperative Collective Intelligence: Techniques and Applications*, chapter Cloud Computing: Paradigms and Technologies, pages 39–67. Springer, 2014.

[151] Chuan Shi, Yitong Li, Jiawei Zhang, Yizhou Sun, and Philip S. Yu. A Survey of Heterogeneous Information Network Analysis. *IEEE Transactions on Knowledge and Data Engineering*, 29 (1):17–37, 2017.

[152] Yu Shi, Huan Gui, Qi Zhu, Lance Kaplan, and Jiawei Han. AspEm: Embedding Learning by Aspects in Heterogeneous Information Networks. In *Proceedings of the 2018 SIAM International Conference on Data Mining*, pages 144–152, 2018.

[153] Yu Shi, Qi Zhu, Fang Guo, Chao Zhang, and Jiawei Han. Easing Embedding Learning by Comprehensive Transcription of Heterogeneous Information Networks. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 2190–2199. ACM, 2018.

[154] Chris Skinner, Catherine Marsh, Stan Openshaw, and Colin Wymer. Disclosure Control for Census Microdata. *Journal of Official Statistics*, 10(1):31–51, 1994.

[155] Shuang Song, Susan Little, Sanjay Mehta, Staal A. Vinterbo, and Kamalika Chaudhuri. Differentially Private Continual Release of Graph Statistics. *CoRR*, abs/1809.02575, 2018.

[156] Yizhou Sun and Jiawei Han. *Mining Heterogeneous Information Networks: Principles and Methodologies*. Morgan & Claypool Publishers, 2012.

[157] Yizhou Sun, Yintao Yu, and Jiawei Han. Ranking-based Clustering of Heterogeneous Information Networks with Star Network Schema. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 797–806. ACM, 2009.

[158] Yizhou Sun, Jie Tang, Jiawei Han, Manish Gupta, and Bo Zhao. Community Evolution Detection in Dynamic Heterogeneous Information Networks. In *Proceedings of the 8th Workshop on Mining and Learning with Graphs*, page 137–146. ACM, 2010.

[159] Yizhou Sun, Rick Barber, Manish Gupta, Charu C. Aggarwal, and Jiawei Han. Co-author Relationship Prediction in Heterogeneous Bibliographic Networks. In *2011 International Conference on Advances in Social Networks Analysis and Mining*, pages 121–128, 2011.

[160] Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S. Yu, and Tianyi Wu. Pathsim: Meta Path-based Top-K Similarity Search in Heterogeneous Information Networks. *Proceedings of the VLDB Endowment*, 4(11):992–1003, 2011.

[161] Yizhou Sun, Charu C. Aggarwal, and Jiawei Han. Relation Strength-Aware Clustering of Heterogeneous Information Networks with Incomplete Attributes. *Proceedings of the VLDB Endowment*, 5(5):394–405, 2012.

[162] Akimichi Takemura. Local Recoding by Maximum Weight Matching for Disclosure Control of Microdata Sets. CIRJE F-Series 40, 1999.

[163] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. LINE: Large-Scale Information Network Embedding. In *Proceedings of the 24th International Conference on World Wide Web*, page 1067–1077. IW3C2, 2015.

[164] Jiliang Tang, Huiji Gao, Xia Hu, and Huan Liu. Exploiting Homophily Effect for Trust Prediction. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, page 53–62. ACM, 2013.

[165] Lu-An Tang, Xiao Yu, Sangkyum Kim, Jiawei Han, Chih-Chieh Hung, and Wen-Chih Peng. Tru-Alarm: Trustworthiness Analysis of Sensor Networks in Cyber-Physical Systems. In *Proceedings of the 2010 IEEE International Conference on Data Mining*, pages 1079–1084, 2010.

[166] Mingdong Tang, Yu Xu, Jianxun Liu, Zibin Zheng, and Xiaoqing Liu. Combining Global and Local Trust for Service Recommendation. In *Proceedings of the 2014 IEEE International Conference on Web Services*, pages 305–312, 2014.

[167] Traian M. Truta, Farshad Fotouhi, and Daniel Barth-Jones. Privacy and Confidentiality Management for the Microaggregation Disclosure Control Method: Disclosure Risk and

Information Loss Measures. In *Proceedings of the 2003 ACM Workshop on Privacy in the Electronic Society*, pages 21–30. ACM, 2003.

[168] Lee Ventola. Social Media and Health Care Professionals: Benefits, Risks, and Best Practices. *Journal of Pharmacy and Therapeutics*, 39(7):491–520, 2014.

[169] AG D. Waal and Leon C. R. J. Willenborg. Optimal Local Suppression in Microdata. *Journal of Official Statistics*, 14(4):421–435, 1998.

[170] Ton D. Waal and Leon Willenborg. Information Loss through Global Recoding and Local Suppression. *Netherlands Official Statistics*, 14:17–20, 1999. Special Issue on Statistical Disclosure Control.

[171] Omar A. Wahab, Jamal Bentahar, Hadi Otrok, and Azzam Mourad. A Survey on Trust and Reputation Models for Web Services: Single, Composite, and Communities. *Decision Support Systems*, 74:121–134, 2015.

[172] Weiqing Wang, Hongzhi Yin, Xingzhong Du, Wen Hua, Yongjun Li, and Quoc V. H. Nguyen. Online User Representation Learning Across Heterogeneous Social Networks. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 545–554. ACM, 2019.

[173] Yue Wang and Xintao Wu. Preserving Differential Privacy in Degree-Correlation Based Graph Generation. *Transactions on Data Privacy*, 6(2):127–145, 2013.

[174] Cort J. Willmott and Kenji Matsuura. Advantages of the Mean Absolute Error (MAE) over the Root Mean Square Error (RMSE) in Assessing Average Model Performance. *Climate Research*, 30(1):79–82, 2005.

[175] Barbara H. Wixom and Lynne Markus. Data Value Assessment: Recognizing Data as an Enterprise Asset. MIT Sloan Center for Information Systems Research, 2015.

[176] Barbara H. Wixom, Anne Buff, and Paul Tallon. Six Sources of Value for Information Businesses. MIT Sloan Center for Information Systems Research and SAS Institute Inc., 2015.

[177] Raymond C. Wong, Ada W. Fu, Ke Wang, and Jian Pei. Minimality Attack in Privacy-preserving Data Publishing. In *Proceedings of the 33rd International Conference on Very Large Data Bases*, pages 543–554. VLDB Endowment, 2007.

[178] Quanwang Wu, MengChu Zhou, Qingsheng Zhu, and Yunni Xia. VCG Auction-based Dynamic Pricing for Multigranularity Service Composition. *IEEE Transactions on Automation Science and Engineering*, 15(2):796–805, 2018.

[179] Xiaotong Wu, Wanchun Dou, and Qiang Ni. Game Theory Based Privacy-preserving Analysis in Correlated Data Publication. In *Proceedings of the Australasian Computer Science Week Multiconference*, pages 1–10. ACM, 2017.

[180] Xiaokui Xiao, Gabriel Bender, Michael Hay, and Johannes Gehrke. iReduct: Differential Privacy with Reduced Relative Errors. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data*, pages 229–240. ACM, 2011.

[181] Yonghui Xiao, Li Xiong, Liyue Fan, Slawomir Goryczka, and Haoran Li. DPCube: Differentially Private Histogram Release through Multidimensional Partitioning. *Transactions on Data Privacy*, 7(3):195–222, 2014.

[182] Bin Yang, Issei Sato, and Hiroshi Nakagawa. Bayesian Differential Privacy on Correlated Data. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 747–762. ACM, 2015.

[183] Xiaoyuan Yang, Xiaoshuang Luo, Xu A. Wang, and Shuaiwei Zhang. Improved Outsourced Private Set Intersection Protocol Based on Polynomial Interpolation. *Concurrency and Computation: Practice and Experience*, 30(1):e4329, 2017.

[184] Andrew C. Yao. Protocols for Secure Computations. In *Proceedings of the 23rd Annual Symposium on Foundations of Computer Science*, pages 160–164. IEEE Computer Society, 1982.

[185] Xiaobo Yin, Shunxiang Zhang, and Hui Xu. Node Attributed Query Access Algorithm Based

on Improved Personalized Differential Privacy Protection in Social Network. *International Journal of Wireless Information Networks*, 26(3):165–173, 2019.

[186] Aston Zhang, Xing Xie, Kevin Chen-chuan, Carl A. Gunter, Jiawei Han, and Xiaofeng Wang. Privacy Risk in Anonymized Heterogeneous Information Networks. In *Proceedings of the 17th International Conference on Extending Database Technology*, pages 595–606, 2014.

[187] Chuxu Zhang, Dongjin Song, Chao Huang, Ananthram Swami, and Nitesh V. Chawla. Heterogeneous Graph Neural Network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 793–803. ACM, 2019.

[188] Le Zhang and Ponnuthurai N. Suganthan. Random Forests with Ensemble of Feature Spaces. *Pattern Recognition*, 47(10):3429–3437, 2014.

[189] Zhongheng Zhang. Missing Data Imputation: Focusing on Single Imputation. *Annals of Translational Medicine*, 4(1), 2016.

[190] Chuan Zhao, Shengnan Zhao, Minghao Zhao, Zhenxiang Chen, Chong-Zhi Gao, Hongwei Li, and Yu-an Tan. Secure Multi-Party Computation: Theory, Practice and Applications. *Information Sciences*, 476:357–372, 2019.

[191] Bin Zhou and Jian Pei. Preserving Privacy in Social Networks Against Neighborhood Attacks. In *Proceedings of the 24th IEEE International Conference on Data Engineering*, pages 506–515. IEEE Computer Society, 2008.

[192] Lei Zou, Lei Chen, and Tamer Özsu. K-Automorphism: A General Framework for Privacy-preserving Network Publication. *Proceedings of the VLDB Endowment*, 2(1):946–957, 2009.