ОБОЗРЕНИЕ
ПРИКЛАДНОЙ И ПРОМЫШЛЕННОЙ
Том 31 МАТЕМАТИКИ Выпуск 1
2024

В. В. Горбань, **Д. А. Зябкин** (Москва, ФСРБИТ; Москва, РТУ–МИРЭА)**. Современные подходы к обезличиванию персональных данных на основе локальной дифференциальной приватности.**

УДК DOI https://doi.org/10.52513/08698325_2024_31_1_1

Резюме: В работе рассматриваются современные подходы к обезличиванию персональных данных на основе локальной дифференциальной приватности.

Ключевые слова: дифференциальная приватность, локальная дифференциальная приватность.

Одним из ключевых аспектов защиты конфиденциальности в цифровом мире является обезличивание персональных данных (ПД). Существует два основных подхода к обезличиванию таких данных: синтактический и статистический. Первый подход подразумевает создание обезличенного массива ПД (далее — МПД) путем преобразования исходного МПД таким образом, чтобы он обладал свойствами k-анонимности, l-разнообразия, t-близости и другими свойствами при заданных пороговых ограничениях. Требуемые свойства и пороговые ограничения выбираются исходя из предполагаемых атак нарушителей. В качестве основной атаки на персональные данные рассматривается атака «сопоставления», которая заключается в обогащении обезличенного МПД информацией из других источников (открытых источников, серых баз данных и т. п.) и восстановлении исходных ПД. В работе [1] показано, что обезличенные МПД уязвимы перед атакой «сопоставления», поскольку синтаксический подход не контролирует объем информации априорных знаний предполагаемого нарушителя и риск раскрытия персональных данных может оказаться неприемлемо высоким даже при высоких пороговых ограничениях. Вследствие этого был предложен статистический подход, который заключается в предоставлении доступа к МПД через случайный механизм, который в ответ на запрос предоставляет агрегированную информацию с зашумлением. Данный подход лег в основу дифференциальной приватности (далее — ДП), которая гарантирует, что добавление или удаление любой записи в исходном МПД не будет оказывать существенного влияния на результаты запросов.

Дифференциальная приватность [1] может быть централизованной (или классической, далее — ЦДП) и локальной (далее — ЛДП) (см. рис. 1) [3]. В ЦДП запросы поступают на сервер, который хранит исходный МПД и вычисляет агрегированные ответы на запрос с применением дифференциально-приватного механизма. ЛДП, в свою очередь, предполагает применение дифференциально-приватных механизмов к ПД локально (например, на устройстве пользователя) и передачу их на сервер, который восстанавливает требуемые статистики для ответа на поступающие запросы.

Недостатком ЦДП является централизованное хранение и обработка данных, что может привести к уязвимостям в случае компрометации сервера, в то время как ЛДП выполняется локально, минимизируя риски утечки конфиденциальности при несанкционированном доступе к данным. Вследствие чего широкое распространение получили методы ЛДП, обзор которых является предметом данной работы.

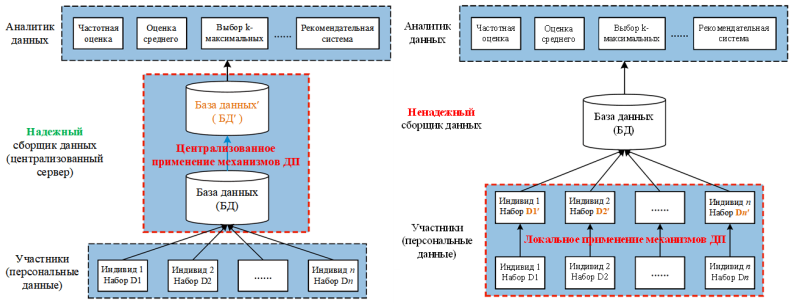

Рис. 1. Схема централизованной (слева) и локальной (справа) дифференциальной приватности

Формально локальный дифференциально-приватный механизм (далее — ЛДПМ) определяется следующим образом. Для $\varepsilon > 0$ рандомизированный механизм M обеспечивает ε-локальную дифференциальную защиту тогда и только тогда, когда для произвольной пары значений v, $v' \in D$, каждого запроса $q \in Q$ и $\forall S \subseteq Y$ выполняется

$$P\big[M(v,q) \in S\big] \leqslant e^{\varepsilon} P\big[M(v',q) \in S\big],$$

где D — множество входных значений механизма M, Y — множество выходных значений механизма M, Q — множество запросов, ε — бюджет приватности, определяющий уровень защиты конфиденциальных данных.

Процесс обработки данных в ЛДП можно представить в виде последовательности четырех этапов, конкретная реализация которых называется *протоколом* [4].

Этап 1: кодирование. На данном этапе происходит сопоставление исходным данным пользователей некоторых кодовых значений (например, целых значений или бинарных строк в виде фильтров Блума и др.).

Этап 2: обезличивание. После кодирования значения искажаются с помощью случайного шума. Например, с заданными вероятностями выполняются случайные инверсии битов в бинарном представлении значений.

Этап 3: агрегирование. Результаты применения ЛДПМ отправляются на сервер, который выполняет агрегирование значений.

Этап 4: вычисление статистик. На заключительном этапе сервер выполняет восстановление требуемых статистик по обезличенным ПД и формирует ответ на запрос.

В табл. 1 приведен список современных ЛДПМ, которые являются представителями классов соответствующих протоколов [4]: GRR, OUE, RAPPOR, OLH, JLRR, HRR.

Таблица 1. Современные ЛДПМ.

Алгоритм	Кодирование	Обезличивание
GRR	$t = v$	$Pr[\hat{t} = v] = \begin{cases} \frac{e^\epsilon}{e^\epsilon+d-1}, & if\ t = v \\ \frac{1}{e^\epsilon+d-1}, & if\ t \neq v \end{cases}$
OUE	$t = [0, \cdots, 1, \cdots, 0]$, where $t[v] = 1$	$Pr[\hat{t}[i] = 1] = \begin{cases} \frac{1}{2}, & if\ t[i] = 1 \\ \frac{1}{e^\epsilon+1}, & if\ t[i] = 0 \end{cases}$
RAPPOR	$r =< \mathcal{H}, t >$; $\mathcal{H} \in \mathbb{H}$; $t = [0, \cdots, 1, \cdots]$ $where\ t[i] = \begin{cases} 1, if\ \mathcal{H}(v) = 1, \\ 0,\ otherwise \end{cases}$	$Pr[\hat{t}[i] = 1] = \begin{cases} 1 - \frac{1}{2}f, & if\ t[i] = 1 \\ \frac{1}{2}f, & if\ t[i] = 0' \end{cases}$ where $f = \frac{2}{e^{\epsilon/2}+1}$
OLH	$r =< \mathcal{H}, t >$; $\mathcal{H} \in \mathbb{H}$; $t = \mathcal{H}(v)$	$Pr[\hat{t} = \mathcal{H}(v)] = \begin{cases} \frac{e^\epsilon}{e^\epsilon+g-1}, & if\ t = \mathcal{H}(v) \\ \frac{1}{e^\epsilon+g-1}, & if\ t \neq \mathcal{H}(v)' \end{cases}$ where $g = e^\epsilon + 1$
JLRR	$\Phi \in \{-\frac{1}{\sqrt{m}}, \frac{1}{\sqrt{m}}\}^{m \times d}$; $r =< i, t >$; $i \in [m]$; $t = \Phi[i, v]$	$\hat{t} = \begin{cases} c_\epsilon dt, & w.p.\ \frac{e^\epsilon}{e^\epsilon+1} \\ -c_\epsilon dt, & w.p.\ \frac{1}{e^\epsilon+1} \end{cases}$, where $c_\epsilon = \frac{e^\epsilon+1}{e^\epsilon-1}$
HRR	$\Phi : 2^d \times 2^d$ Hadamard Matrix, where $\Phi[i, j] = 2^{-d/2}(-1)^{<i,j>}$; $r =< i, t >$; $i \in [2^d]$; $t = \Phi[i, v]$	$Pr[\hat{t} = 1] = \begin{cases} \frac{e^\epsilon}{e^\epsilon+1}, & if\ t = 1 \\ \frac{1}{e^\epsilon+1}, & if\ t = -1 \end{cases}$

В табл. 1 используются следующие обозначения: v — исходное значение, t — кодированное значение, $\widetilde{t}$ — ответ механизма, r — отчет пользователя (обезличенные ПД пользователя, отправляемые на сервер), H — хэш-функция, d — размер битового представления.

Рассмотрим более подробно алгоритм RAPPOR(Randomized Aggregatable Privacy Preserving Ordinal Response) [5], как один из самых распространенных ЛДПМ: на основе RAPPOR были созданы многие специализированные алгоритмы для использования в определенных областях, таких как машинное обучение и IoT.

Этап кодирования в RAPPOR реализован путем представления значений пользователя в виде фильтров Блума: значению v сопоставляется бинарный вектор B_0 длины h. Компоненты вектора B_0 задаются как:

$$B_0[i] = \begin{cases} 1, & \text{если } \exists\ H \in H, \text{что } H(v) = i, \\ 0, & \text{иначе.} \end{cases}$$

где $H = \{H_1, H_2, \ldots, H_m\}$ обозначается как набор из m хэш-функций.

На этапе обезличивания дважды выполняется зашумление фильтра Блума. Сначала реализуется постоянный рандомизированный отклик (создается бинарный вектор B_1) для защиты от атаки среднего. Каждое значение $B_1[i]$ определяется по следующему правилу:

$$B_1[i] = \begin{cases} 1, & \text{с вероятностью } \frac{1}{2}f, \\ 0, & \text{с вероятностью } \frac{1}{2}f, \\ B_0[i] & \text{с вероятностью } 1 - f, \end{cases}$$

где f — настраиваемый пользователем параметр.

Вектор B_1 запоминается и в дальнейшем повторно зашумляется для формирования мгновенного рандомизированного отклика, который передается на сервер в качестве *отчета*. Повторное зашумление производится каждый раз, когда пользователь передает на сервер очередное значение. На основе B_1 создается вектор B_2 (мгновенный рандомизированный отклик) следующим образом: значение $B_2[i]$ устанавливается в 1 согласно условному распределению:

$$P\left(B_2[i] = 1 \vee B_1[i] = 1\right) = p,$$
$$P\left(B_2[i] = 1 \vee B_1[i] = 0\right) = q,$$

где p и q — параметры конфиденциальности.

Для уменьшения влияния коллизий при применении фильтра Блума в RAPPOR пользователи подразделяются на m когорт, в каждой из которых используются свои наборы хэш-функций H.

На последних этапах протокола сервер получает все отчеты (векторы B_2) и вычисляет агрегированную оценку частот. Для этого в RAPPOR применяется линейная регрессия и регрессия лассо [3].

Этапы агрегирования и оценки частот в RAPPOR реализованы следующим образом.

Вначале производится агрегация отчетов по когортам. Затем для каждой когорты j выполняется оценка частот t_{ij} единиц в i-ой позиции суммы всех возможных исходных векторов Блума B_0 (соответствующих когорте):

$$t_{ij} = \frac{c_{ij} - \left(p + \frac{1}{2}fq - \frac{1}{2}fp\right) N_j}{(1-f)\,(q-p)},$$

где c_{ij} — эмпирическая частота единиц в i-й позиции для N_j отчетов (суммы векторов B_2) в когорте j.

Далее решается задача оценки частот исходных значений (восстановление распределения) с помощью регрессионных моделей.

Обучающая выборка строится следующим образом. Задается вектор ответов $Y = (t_{ij}), i \in [1,k], j \in [1,m]$, где m — число когорт, k — длина вектора фильтра Блума. Строится матрица X размера $km \times M$, где M — количество рассматриваемых кандидатов значений: по столбцам в матрице X записаны фильтры Блума B_0^v для каждого возможного исходного значения v, конкатенированные в соответствии с когортами. Стоит отметит, что матрица X является разреженной, так как каждый столбец X содержит не более hm единиц.

Затем строится регрессионная модель Лассо [6] $F_L(X) = Y$ и производится отбор значений, соответствующих ненулевым коэффициентам регрессии: из X создается матрица X' путем удаления столбцов, коэффициенты регрессии Лассо которых равны нулю. Затем строится модель гребневой регрессии $F_R(X') = Y$ и выполняется множественная проверка статистических гипотез на равенство нулю коэффициентов регрессии. Для этого применяется процедура Бенджамини-Хохберга или Холма-Бонферрони. В результате коэффициенты модели F_R, которые статистически значимо отличаются от нуля, являются оценками частот соответствующих исходных значений v и составляют ответ на запрос сервера.

Для RAPPOR доказаны гарантируемые уровни дифференциальной приватности для двух крайних случаев в зависимости от используемых параметров. Мгновенный рандомизированный отклик B_2 обеспечивает ε_1 — дифференциальную приватность (случай перехвата злоумышленником одного отчета пользователя), где

$$\varepsilon_1 = h \log\left(\frac{q^*(1-p^*)}{p^*(1-q^*)}\right),$$

$$q^* = P\left(B_{2_i} = 1 | b_i = 1\right) = f\frac{p+q}{2} + (1-f)\,q,$$

$$p^* = P(B_{2_i} = 1 | b_i = 0) = f\frac{p+q}{2} + (1-f)p.$$

Постоянный рандомизированный отклик B_1 обеспечивает ε_∞ -дифференциальную приватность (случай перехвата злоумышленником всех отчетов пользователя), где

$$\varepsilon_\infty = 2h \ln\left(\frac{1-f/2}{f/2}\right).$$

Таким образом, современные подходы безличивания на основе локальной дифференциальной приватности формализуются в виде протоколов, которые определяют конкретные способы кодирования, обезличивания, агрегирования и восстановления статистических данных. В работе рассмотрен представитель класса протоколов ЛДП, на примере которого продемонстрированы все этапы ЛДП и приведены теоретические оценки уровней конфиденциальности для двух крайних случаев действий злоумышленника.

Согласно концепции ЛДП естественным образом выделяются роли владельца ПД и обработчика обезличенных ПД. Это делает возможным решение задач с использованием обезличенных ПД (например, по анализу телеметрии пользователей, обработке данных геотреков, разработке алгоритмов доверенного искусственного интеллекта и других) ироким кругом организаций, специализация которых — анализ данных и разработки систем искусственного интеллекта. Данные организации будут выступать в роли обработчика обезличенных ПД, для которых гарантируется заданный уровень дифференциальной приватности.

СПИСОК ЛИТЕРАТУРЫ

1. *Dwork C.* Differential Privacy ICALP 2006, Part II, LNCS 4052, p. 1–12, 2006.
2. *Dwork C.*, *McSherry F.*, *Nissim K.*, *Smith A.* Calibrating Noise to Sensitivity in Private Data Analysis TCC 2006, LNCS 3876, p. 265–284, 2006.
3. *Alvim M. et al.* Local differential privacy on metric spaces: optimizing the trade-off with utility. — 2018 IEEE 31st Computer Security Foundations Symposium (CSF). — IEEE, 2018, с. 262–267.
4. *Wang T. et al.* Locally differentially private protocols for frequency estimation. — 26th USENIX Security Symposium (USENIX Security 17), 2017, с. 729–745.
5. *Erlingsson V.*, *Pihur V.*, *Korolova A.* Rappor: Randomized aggregatable privacy-preserving ordinal response. — Proceedings of the 2014 ACM SIGSAC conference on computer and communications security, 2014, с. 1054–1067.
6. *Kairouz P.*, *Bonawitz K.*, *Ramage D.* Discrete distribution estimation under local privacy. — International Conference on Machine Learning. PMLR, 2016, с. 2436–2444.
7. *Tibshirani R.* Regression shrinkage and selection via the lasso Journal of the Royal Statistical Society Series B: Statistical Methodology. 1996, т. 58, № 1, с. 267–288.

Поступила в редакцию
13.XII.2024

UDC DOI https://doi.org/10.52513/08698325_2024_31_1_1

Gorban V. V., ***Zyabkin D. A.*** (Moscow, FSRBIT; Moscow, TVP Laboratory).

Modern approaches to depersonalization of personal data based on local differential privacy

Abstract: The paper examines modern approaches to depersonalization of personal data based on local differential privacy.

Keywords: Differential privacy, local differential privacy.