Министерство науки и высшего образования Российской Федерации

Санкт-Петербургский Политехнический Университет Петра Великого

—

**Институт Компьютерных Наук и Кибербезопасности**

**Высшая школа Кибербезопасности и Защиты Информации**

**КУРСОВАЯ РАБОТА**

**Анонимизация направленных графов**

по дисциплине «Основы построения защищенных баз данных»

Выполнила

студентка гр. 5151003/90801 Кулеева А.Г.

Руководитель:

Профессор Полтавцева М.А.

«\_\_\_» \_\_\_\_\_\_\_\_\_ 2023 г.

Санкт-Петербург

2023

# СОДЕРЖАНИЕ

|  |  |  |
| --- | --- | --- |
| Введение ………………………………………………………………….. | | 4 |
| 1. | Теоретическая часть ……….………………………………….. | 8 |
| 1.1. | Общие сведения о графовых данных ………………………… | 8 |
| 1.2. | Обзор публикации данных с сохранением конфиденциальности и ее основных этапов ………………… | 12 |
| 1.2.1. | Этап 1: сбор данных от отдельных лиц ……………………… | 14 |
| 1.2.2. | Этап 2: хранение собранных данных, понимание и подготовка к анонимизации ………………………………….. | 14 |
| 1.2.3 | Этап 3: Анонимизация данных ………………………………. | 15 |
| 1.2.3.1. | Удаление непосредственно идентифицируемой информации из исходных данных …………………………………………... | 16 |
| 1.2.3.2. | Выбор метода анонимизации ………………………………… | 16 |
| 1.2.3.3. | Выбор операции анонимизации ……………………………… | 17 |
| 1.2.3.4. | Соблюдение ограничений (если таковые имеются) при операции анонимизации ……………………………………… | 17 |
| 1.2.4. | Этап 4: Анонимное раскрытие/публикация данных ………... | 18 |
| 1.2.5. | Этап 5: анализ опубликованных данных для извлечения встроенных знаний ……………………………………………. | 19 |
| 1.3. | Методы структурной анонимизации, используемые для анонимизации данных в социальных сетях …………………. | 19 |
| 1.3.1. | Методы модификации графа …………………………………. | 25 |
| 1.3.2. | Кластеризация/генерализация графов ……………………….. | 28 |
| 1.3.3. | Вычисление графов с учетом конфиденциальности и дифференциальная анонимность графов ……………………. | 30 |
| 1.3.4. | Гибридные методы анонимизации …………………………… | 31 |
| 1.3.5. | Сравнение методов ……………………………………………. | 32 |
| 1.4. | Методы деанонимизации ……………………………………... | 33 |
| 1.5. | Проблемы анонимизации данных в социальных сетях …….. | 37 |
| 1.6. | Перспективные открытые направления исследований ……... | 38 |
| 1.7. | Вывод по аналитической части ………………………………. | 43 |
| 2. | Практическая часть …………………………………………… | 44 |
| 2.1. | Описание алгоритма анонимизации направленного графа … | 44 |
| 2.1.1. | Обозначения ………………………………….………………... | 44 |
| 2.1.2. | Модели k-анонимности в орграфах ………………………….. | 44 |
| 2.1.3. | Описание применяемого подхода ……………………………. | 47 |
| 2.1.3.1. | Шаг I: Анонимизация последовательностей степеней ……... | 48 |
| 2.1.3.2. | Шаг II: Модификация графа …………………………………. | 52 |
| 2.2. | Описание алгоритма анонимизации ненаправленного графа | 54 |
| 2.2.1. | Описание применяемого подхода ……………………………. | 56 |
| 2.2.2. | K-анонимность с помощью генетического алгоритма ……… | 60 |
| 2.2.3 | Экспериментальный стенд ……………………………………. | 64 |
| 2.3. | Выводы по практической части ……………………………… | 67 |
| Заключение ……………………………………………………………… | | 68 |
| Список использованной литературы ………………………………….. | | 70 |
| Приложение 1 …………………………………………………………… | | 72 |
| Приложение 2 …………………………………………………………… | | 73 |

# 

# ВВЕДЕНИЕ

Большинство организаций, таких как больницы, банки, страховые компании и супермаркеты, собирают соответствующие данные о клиентах для улучшения качества обслуживания. Помимо этих физических организаций, чрезмерный объем пользовательских данных собирается виртуальными платформами, такими как социальные сети. Благодаря значительному развитию информационно-коммуникационных технологий (ИКТ) социальные сети позволяют людям взаимодействовать со своими друзьями, заводить новых знакомых, искать информацию о соответствующем предмете или работе, а также развлекать себя просмотром цифрового контента. Тем временем социальные сети собирают и хранят соответствующие данные о своих пользователях во время предоставления услуг и во время создания учетной записи. Эти собранные данные часто содержат информацию о деятельности пользователя, семейном положении, финансах, хобби, местоположении, интересах, предпочтениях, политических и религиозных взглядах. Кроме того, большинство пользователей с готовностью публикуют другие ценные данные, включая музыкальные предпочтения, точки зрения и социальные проблемы. Исследования показали, что анализ собранных данных с помощью передовых инструментов интеллектуального анализа может помочь организациям значительно улучшить качество обслуживания. Например, это позволяет им понять социальные тенденции, настроения и поведение людей, а также факторы, вызывающие те или иные события в обществе (заболевания, кризис и т.д.). Соответственно, такая информация может быть использована для многих научных или бизнес целей, включая целевую рекламу, рекомендации по релевантному контенту и эффективное принятие решений. Хотя совместное использование данных приносит инновации и позволяет принимать более эффективные решения, это также может поставить под угрозу конфиденциальность пользователей из-за наличия в данных чувствительной информации. Поэтому обработка исходных данных является нежелательной, требуется сначала подготовить данные, прежде чем передать их исследователям. Одним из этапов предобработки является анонимизация данных, которая преобразует исходные данные путем применения к ним некоторых операций для эффективной защиты конфиденциальности пользователя без ухудшения качества анонимных данных. Анонимизация скрывает наиболее чувствительные атрибуты, которые позволили бы однозначно идентифицировать личность пользователя.

Например, рассмотрим рекомендательные системы. Музыкальные сервисы рекомендуют пользователям контент на основе их предпочтений, которые складываются из ранее прослушанных аудио, из того контента, который пользователь оценил положительно или отрицательно. При этом в данном случае сервису нет необходимости знать персональные данные пользователя, такие как ФИО, уникальный идентификатор (телефон, почта), пароли, данные банковской карты. Соответственно для создания рекомендаций все эти данные должны быть удалены, личность пользователя должна быть анонимизирована. А вот знание о местоположении пользователя наоборот является важной информацией, чтобы рекомендовать контент на соответствующем языке.

Другой пример можно рассмотреть в сфере медицины. При постановке диагноза врач никогда не пишет конкретное название заболевания, а ставит код по МКБ. Иногда диагноз может быть уточнен, например, «заболевание легких + КОД». Это тоже своего рода анонимизация данных, призванная сохранить конфиденциальность пациента, поскольку данные о здоровье являются чувствительными. Таким образом, в базе данных больницы, во всех справках и результатах анализов личность пользователя обычно открыта (всегда указаны ФИО), а вот данные о состоянии здоровья может интерпретировать только медицинский сотрудник. Однако для каких-либо исследований в сфере здравоохранения, наоборот, личность пользователей тщательно скрывают, но передают данные о возрасте, анамнезе, сопутствующих заболеваниях, поскольку в данном случае важнее определить не кому принадлежит заболевание, а причины его возникновения у различных групп населения.

Данные социальных связей в основном представлены в виде графа *G(E, V)*, где *V* представляет собой список узлов (пользователей) и *Е* — набор ребер, моделирующих отношения между пользователями. Он предоставляет беспрецедентные возможности для расширенного анализа данных. Социальная связь между пользователями может быть разных типов, таких как личные отношения (друг, родственник, возлюбленный), рабочие отношения (начальник-подчиненный, коллеги по отделу, коллеги по проекту), отношения по учебе (преподаватель-студент, одногруппники, научный руководитель-подопечный) и т.д.

Два популярных подхода к анонимизации — это наивная и структурная анонимизация. При наивной анонимизации публикуется только структура *G* социальных ссылок путем удаления меток ребер и узлов. Однако наивная анонимизация склонна к раскрытию личности, поскольку структура опубликованного графа может раскрыть личность людей, соответствующих узлам. В отличие от этого, подходы структурной анонимизации изменяют структуру *G* для эффективной защиты конфиденциальности пользователя. Эти подходы добавляют новые ребра, вершины и/или модифицируют существующую структура *G*, отвечающую требованиям конфиденциальности и полезности.

Также важно дать определение такому понятию, как социальная сеть, поскольку очень часто в тексте работы будет встречаться это словосочетание. Под социальной сетью не всегда имеется в виду привычный многим феномен. В контексте анонимизации это может быть какой-либо форум или просто сайт, позволяющий пользователям оставлять комментарии или же делать полноценные публикации. Сайт может и не иметь такого функционала, но собирать данные о пользователях, например, сервис для записи на прием к врачу требует такие данные как ФИО, номер полиса ОМС, дату рождения, телефон и почту; сервис для электронной оплаты квитанций ЖКХ требует ввести точные данные о месте проживания; различные маркетплейсы или службы доставок не только собирают ПДн, но и позволяют пользователям комментировать товары и услуги, а также прикладывать фотографии, что также может раскрыть личность человека. Поэтому стоит рассматривать понятие социальных сетей более широко, чем обычно, данный термин был выбран для удобства обобщения всех вышеприведенных примеров.

**Цель работы** — изучить современные методы анонимизации графов.

**Задачи**:

1. Проанализировать информацию о графовых данных и этапах их публикации;
2. Проанализировать методы структурной анонимизации графов;
3. Проанализировать проблемы анонимизации и деанонимизации графов;
4. Практически реализовать один из методов структурной анонимизации графов.

**Объект исследования** — методы структурной анонимизации направленных графов.

**Предмет исследования** — k-степень анонимности

# Теоретическая часть

## Справочная информация о социальных сетях (графы данных)

Данные пользователя социальных сетей могут быть смоделированы с помощью графа *G*, представленного в виде *G(V, E, A)*. Он состоит из набора пользовательских данных *V*, набора социальных связей E, где *E⊆V×V*, и набора атрибутов пользователей *A*, где *A = {Q, S}* содержит *QI* (квази-идентификатор, КИ) и *SA* (чувствительный атрибут, ЧА) пользователей соответственно. *V, E* и *A* также называются узлами, ребрами и метками (т.е. профилем пользователя) соответственно. Эксплуатационную полезность данных социальных сетей можно разделить на три уровня (Таблица 1) [2].

Таблица 1 — Уровни полезности в зависимости от структуры публикуемых данных

|  |  |  |  |
| --- | --- | --- | --- |
| Уровень | l1 | l2 | l3 |
| Название | Раскрытие только структуры графа | Раскрытие узлов | Раскрытие как структуры графа, так и узлов |
| Описание | На этом уровне раскрытия владельцы данных (например, поставщики услуг социальных сетей) публикуют только структуру графа (т.е. вся информация о профилях/метках удаляется перед публикацией данных). | На этом уровне раскрытия владелец данных публикует узлы (профили пользователей), но скрывает структуру самого графа. Например, данные узла хранятся в таблице или матрице и публикуются для целей аналитики и интеллектуального анализа данных. | На этом уровне раскрытия владелец данных раскрывает как структуру графа, так и значения узлов после внесения некоторых изменений в обе сущности. |
| Что скрыто | Данные о пользователях (метки) | Структура графа | Реальный граф видоизменен |
| Что открыто | Структура графа | Данные пользователей в виде таблицы | Аналог реального графа данных |
| Полезность для анализа | Средняя | Средняя | Высокая |

Хотя публикация данных имеет неоценимое значение для достижения множества исследовательских и бизнес-целей, она также влечет угрозы конфиденциальности. Четыре хорошо известные угрозы конфиденциальности, которые могут возникнуть после публикации измененного графа *G’* кратко изложены ниже.

*Раскрытие личности (т.е. повторная идентификация узла)*: это происходит, когда злоумышленник может точно связать/идентифицировать человека по опубликованному анонимному графу.

*Раскрытие границ (т.е. раскрытие отношений)*: раскрывает отношения между пользователями. Например, отношения между пациентом и врачом. Если известно, что врач является экспертом в области лечения рака, раскрытие взаимосвязи может привести к выводу, что пациент может быть болен раком.

*Раскрытие содержимого (т.е. раскрытие меток вершин и ребер)*: это происходит, когда конфиденциальная метка, связанная с ребром или вершиной, обнаруживается из *G’* и этот открытый ярлык может быть напрямую связан с конкретным человеком на исходном графе *G*.

*Раскрытие аффилированной ссылки (т.е. принадлежит ли лицо v к определенной аффилированной группе h):* это происходит, когда связь между пользователем v и аффилированная группа h раскрываются с уверенностью *≥t* и эта обнаруженная ссылка может быть напрямую связана с *v* [1].

Сохранение конфиденциальности при публикации данных стало активной областью исследований в последние годы. Анонимность пользователей социальных сетей может быть нарушена тремя различными способами, такими как действия пользователя на сайтах, сохраненные данные пользователей на серверах и опубликованные анонимизированные данные (см Рисунок 1). Конфиденциальность в первых двух областях может быть сохранена с помощью методических рекомендаций, шифрования и методов нанесения водяных знаков. Между тем, конфиденциальность в третьей области может быть гарантирована только путем разработки и внедрения новых механизмов анонимизации. В рамках данной работы рассматриваются концепции, методы и решения, касающиеся третьей области конфиденциальности в социальных сетях (т.е. публикации графов с сохранением конфиденциальности).

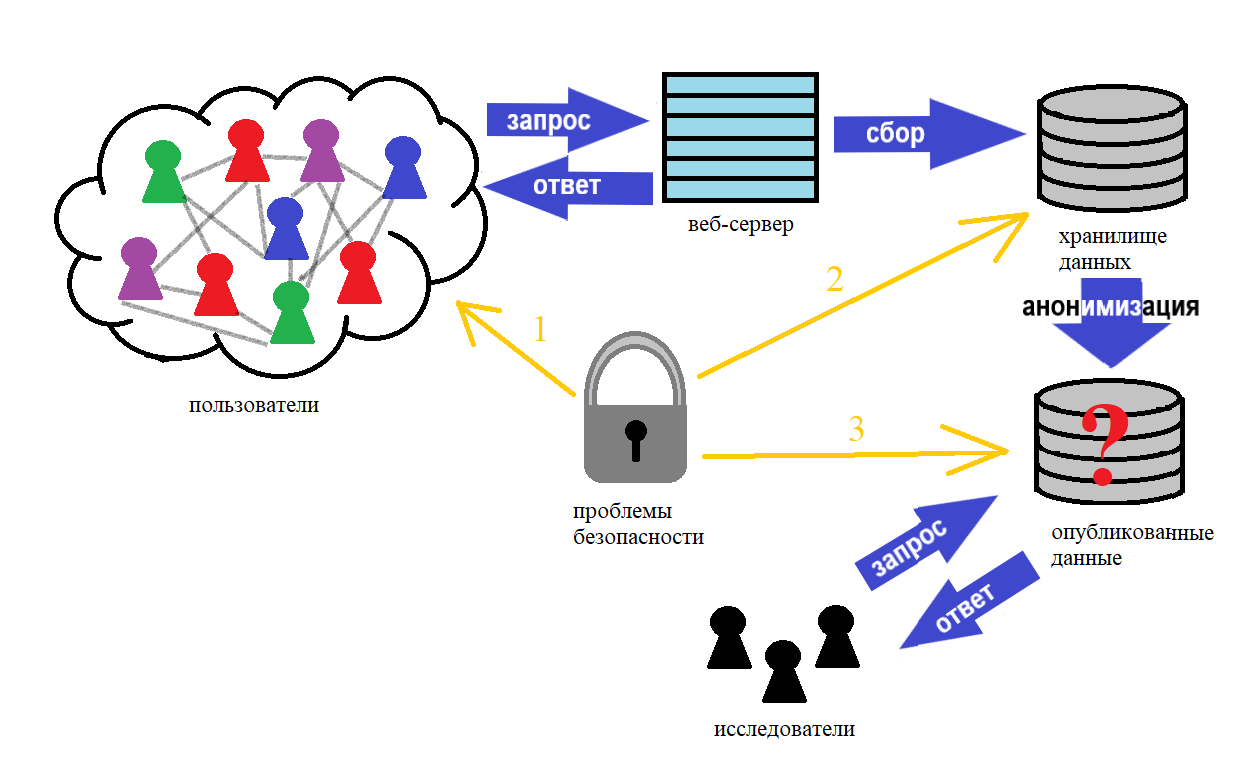


Рисунок 1 — Обзор трех областей конфиденциальности в социальных сетях

В Таблице 2 рассмотрим типы информации, из которой могут состоять пользовательские данные. Все эти источники данных бесценны для детальной аналитики и соответствующего сбора и анализа информации [1].

Таблица 2 — Типы информации о пользователях

| Идентификация пользователя | | Социум (окружение) | | Контент | |
| --- | --- | --- | --- | --- | --- |
| Типы данных | Примеры | Типы данных | Примеры | Типы данных | Примеры |
| Демографические | Пол, возраст, местоположение | Поведенческие | Друзья, знакомые друзей, мероприятия и онлайн-сообщества | Раскрытые | Пользователь сами предоставил эти данные |
| Производные | Национальность (может быть следствием местоположения) | Доверенные | Данные, размещенные пользователем в профиле другого пользователя (т.е. комментарии) |
| Случайные | Это информация, собранная и размещенная другими пользователями в чьем-либо профиле |

## Обзор публикации данных с сохранением конфиденциальности и ее основных этапов

Публикация данных с сохранением конфиденциальности (ПДСК) предоставляет набор инструментов, методов, решений и фреймворков для обмена ценной информацией с аналитиками и исследователями без ущерба для конфиденциальности пользователей. Между тем, даже при анонимном обмене данными злоумышленники все еще могут получить информацию о личности пользователя или личную информацию, используя вспомогательные данные, собранные из внешних источников. Таким образом, многие исследования предполагают защиту конфиденциальности пользователей на всех этапах цикла обработки информации. Типичный процесс ПДСК включает в себя пять основных этапов [1]:

1. сбор данных от отдельных лиц;
2. хранение собранных данных, их понимание и подготовка к анонимизации;
3. анонимизация данных пользователя;
4. публикация анонимных данных;
5. анализ опубликованных данных для извлечения встроенных знаний.

На Рисунке 2 представлена схема ПДСК. У каждого этапа выделены некоторые основные свойства. Ниже представлена краткая информация о каждом этапе.

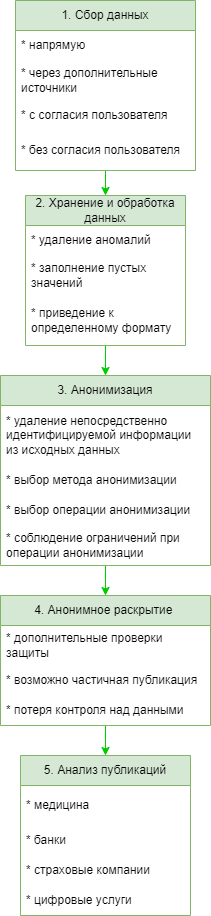


Рисунок 2 — Этапы ПДСК

### Этап 1: Сбор данных от отдельных лиц

На первом этапе ПДСК собираются соответствующие данные от отдельных лиц. В связи со значительным технологическим развитием в последние годы объем данных, генерируемых сенсорными сетями, сайтами социальных сетей, приложениями здравоохранения, Интернетом, онлайн-банкингом, сторонними приложениями, интегрированными в социальные сети, и многими другими офлайн/онлайн компаниями, резко увеличивается. Эти данные собираются от отдельных лиц напрямую или с помощью интеллектуальных устройств (например, мобильных телефонов, ноутбуков, блокнотов и т.д.). Например, если пациент посещает больницу, его/ее личная информация собирается для улучшения лечения. Позже личная информация пациента в сочетании с информацией о заболевании сохраняется в базе данных больницы для вторичного использования. Аналогичным образом, процесс открытия счета в банке требует сбора основной, а также конфиденциальной информации о клиентах.

В последнее время, в связи со значительным развитием ИКТ, большинство поставщиков услуг запустили свои собственные веб-сайты для сбора данных от своих пользователей/заказчиков. Кроме того, многие сайты социальных сетей собирают ценные данные о пользователях без их согласия. Например, они могут собирать информацию об устройстве, временную информацию о потреблении услуг, информацию о местоположении и много другой полезной информации [1].

### Этап 2: Хранение собранных данных, понимание и подготовка к анонимизации

После сбора соответствующих пользовательских данных следующим этапом является хранение собранных данных, их понимание и подготовка к дальнейшим операциям. Компании используют крупномасштабные базы данных для хранения собранных пользовательских данных. Как правило, собранные данные о пользователях могут содержать неправильные значения, выбросы, пропущенные значения для некоторых атрибутов и неполные записи. Поэтому перед процессом анонимизации данных требуется подготовка. Подготовка данных включает в себя удаление имеющихся в данных отклонений, которые не подходят для анализа и могут привести к неточным результатам. Кроме того, из данных пользователя удаляются записи с неизвестными (т.е. отсутствующими) значениями. В некоторых случаях алгоритму обработки данных требуется информация пользователя в определенном формате. Поэтому на этапе подготовки данных выполняется соответствующее форматирование собранных данных и их обогащение, если требуется, для последующих шагов. С помощью предварительной обработки могут быть получены очищенные данные, содержащие полную информацию о каждом пользователе, и они могут быть напрямую загружены в алгоритм анонимизации для обеспечения анонимности [1].

### Этап 3: Анонимизация данных

Анонимизация данных — это практичное и наиболее широко используемое решение для защиты конфиденциальности пользователей при публикации данных. В табличных данных анонимизация данных очищает исходные значения квази-идентификаторов, делая информацию менее специфичной для защиты конфиденциальности и улучшения утилит. Напротив, если данные пользователей приведены в виде графа, анонимизация изменяет его структуру для защиты конфиденциальности пользователей и связанных с ними чувствительных атрибутов без существенного снижения полезности анонимного графа. Кроме того, анонимизация может быть адаптирована с учетом требований владельца данных к конфиденциальности, потребностей законных потребителей информации в полезности и целей публикации данных. Типичный процесс анонимизации включает следующие четыре основных этапа. Все четыре этапа дополняют друг друга и могут быть использованы для создания анонимной таблицы *T’* из оригинальной таблицы *T* или анонимного графа *G’* из оригинального графа *G* [1].

#### Удаление непосредственно идентифицируемой информации из исходных данных

В начале процесса анонимизации из данных удаляется любая информация, которая может идентифицировать кого-либо напрямую однозначно. Например, имя, номер социального страхования, номер паспорта, адрес электронной почты и номер мобильного телефона могут быть связаны с чьей-либо реальной личностью. Следовательно, такая информация удаляется из данных перед их анонимизацией. Это может быть удалено на любом этапе, но более раннее удаление может помочь значительно сэкономить вычислительную мощность.

#### Выбор метода анонимизации

Было предложено и реализовано множество методов анонимизации для различных сценариев, которые зависят от типа представления информации. Например, для графовых данных используется структурная анонимизация, для табличных данных используются реляционные методы. Хотя некоторые исследователи использовали методы реляционной анонимизации для графовых данных. Но из-за существенных различий в терминах информации, содержащейся на графах, реляционные методы не могут быть непосредственно применены к ним. Таким образом, решение о методе анонимизации зависит от исходного представления данных (т.е. графов или таблиц) и целей публикации данных.

#### Выбор операции анонимизации

После выбора подходящего метода анонимизации следующим шагом является применение соответствующей операции анонимизации для искажения исходных значений данных или структуры графа. Выбор операции анонимизации зависит от типа данных, метода анонимизации, а также целей обеспечения конфиденциальности и полезности. Например, при реляционной анонимизации операция обобщения сохраняет больше семантики исходных данных по сравнению с операцией подавления. Напротив, операция подавления в высшей степени подходит для защиты конфиденциальности пользователя по сравнению с обобщением. Кроме того, из-за сложной структуры графа *G*, изменения в небольшой его части может значительно снизить полезность *G’* Таким образом, выбор подходящей операции анонимизации производится для эффективного решения проблемы конфиденциальности и полезности для реальных приложений.

#### Соблюдение ограничений (если таковые имеются) при операции анонимизации

Помимо выбора подходящего метода анонимизации и операции анонимизации, владельцы данных могут применять некоторые ограничения во время анонимизации данных. Такие ограничения могут касаться пороговых значений конфиденциальности и полезности, количества пользователей в классе эквивалентности, распределения личных значений пользователя в классе/кластере и/или количества подключений между пользователями (т.е. степени) в *G’*. Эти ограничения могут быть предложены владельцами данных или выведены из статистики данных. Кроме того, ограничения могут быть применены с учетом возможностей злоумышленников и характера конфиденциальной информации, содержащейся в *T* или *G*.

### Этап 4: Анонимное раскрытие (публикация данных)

Результатом процесса анонимизации является анонимный *G’* или *T’* в зависимости от представления исходных пользовательских данных. Получателями анонимных данных могут быть аналитики, исследователи, сборщики данных, аналитические фирмы и сторонние приложения. Прежде чем сделать анонимные данные общедоступными, владельцы данных выполняют несколько проверок для подтверждения уровня защиты конфиденциальности пользователя и полезности анонимных данных соответственно. После детальных проверок решение о разглашении данных принимается владельцами данных. В некоторых случаях владельцы данных не публикуют полные анонимные данные, вместо этого сначала публикуются некоторые части анонимных данных. Позже публикуется полная анонимная таблица или граф. Кроме того, анонимные данные могут передаваться только соответствующим институтам по электронной почте. Однако анонимные данные, как правило, публикуются через Интернет, так что большое количество людей может получить доступ к опубликованным данным для многофакторной аналитики. После публикации данных владелец данных не имеет никакого контроля над использованием и распространением опубликованных данных. Между тем предполагается, что опубликованные данные будут использоваться только по назначению, и о любой проблеме владельцам данных будет сообщено в явном виде. В некоторых случаях владельцами и издателями данных могут быть две разные стороны. Издатели данных публикуют анонимные данные на своих собственных носителях в соответствии с соглашениями о публикации с фактическими владельцами данных [1].

### Этап 5: Анализ опубликованных данных для извлечения встроенных знаний

Как только анонимные данные были опубликованы, предполагаемые получатели собирают их для анализа. Например, студенты-медики собирают данные о больницах для своих исследований. Они могут выполнить несколько видов тестов, таких как факторы, вызывающие определенное заболевание, распространенное заболевание у людей определенной возрастной группы и симптомы конкретного заболевания. Кроме того, данные банков, содержащие информацию о рейтинге возврата кредита, представляют ценность для страховых компаний. Данные социальных сетей подходят для поставщиков цифровых услуг и маркетинговых фирм. Кроме того, многие сторонние приложения покупают данные пользователя социальных сетей для достижения своих целей. Опубликованные данные предназначены не только для понимания причин некоторых проблем, но и могут помочь в разработке новых правил и шаблонов, которые могут быть полезны в маркетинговых целях. С помощью алгоритмов интеллектуального анализа данных можно анализировать опубликованные данные для получения полезной информации. Кроме того, кластеризация и анализ пользователей полезны для рекомендаций, анализа предпочтений, целевого маркетинга, распространения информации, контроля информации и достоверности информации. Таким образом, обмен данными стал обычным делом для некоторых организаций из-за значительных преимуществ с точки зрения улучшения процесса принятия решений, совершенствования политики, анализа тенденций, прогнозирования и инноваций [1].

## Методы структурной анонимизации, используемые для анонимизации графовых данных

Структурная анонимизация относится к изменению структурных свойств данных графов для защиты от угроз конфиденциальности, возникающих при публикации данных. Матричное представление данных позволяет применять компьютерные инструменты и математические модели для обобщения и извлечения закономерностей, поскольку поиск соответствующих закономерностей из плотных графов чрезвычайно сложен. Например, граф *G* с *n* пользователями может быть смоделирован в виде матрицы смежности *M* размера *n×n*. В матрице смежности отношения между двумя пользователями *i* и *j* может быть представлено значением (*<0, 1>* или *<y, n>*) в ячейке *i, j.* Для представления различных форм пользовательских данных и моделирования структурных свойств социальных сетей необходимо определить тип графа *G*: узлы и ребра могут иметь метки или не иметь их, граф может быть направленным или нет, взвешенным или нет, как показано на Рисунке 3 [1].

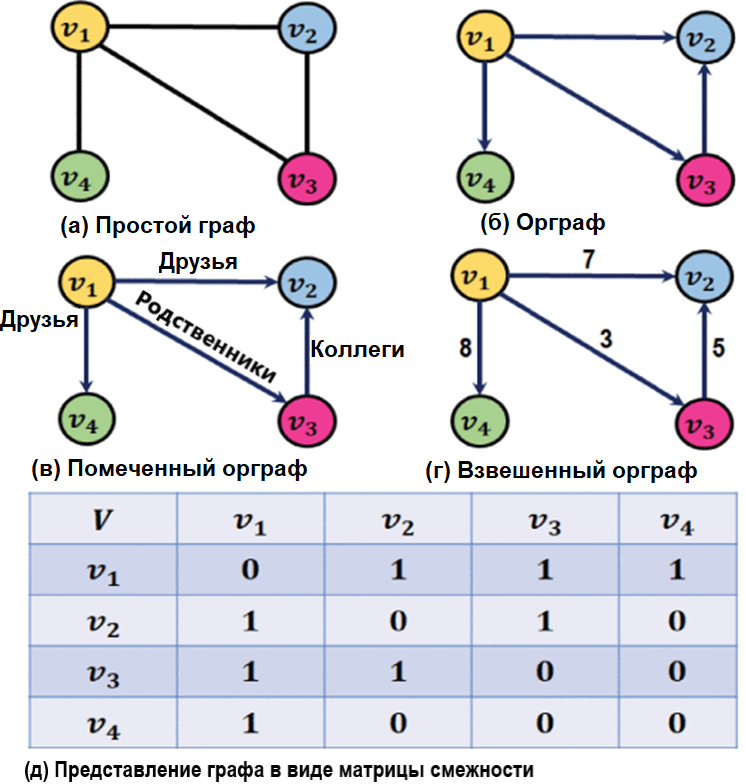


Рисунок 3 — Обзор представления данных с использованием различных форм графов и матрицы

Фрагменты информации, касающиеся конфиденциальности пользователей в данных социального графа, обобщены в Таблице 3. При хранении данных именно в виде графа пользователям требуется защита конфиденциальности большинства фрагментов их личной информации, приведенных в Таблице 3. Напротив, данные в реляционных бд в основном содержат четыре вида информации о пользователях:

1. КИ пользователей,
2. ЧА пользователей,
3. микросреды статистики (т.е. конкретное значение ЧА, разделяемое меньшим количеством людей в наборе данных) о пользователях,
4. макростатистика (т.е. конкретное значение ЧА, разделяемое большим количеством людей в наборе данных) о пользователях.

Кроме того, из-за доступности информации о профилях пользователей и учетных записях в нескольких социальных сетях конфиденциальность пользователей может быть легко нарушена по сравнению с табличными данными [1].

Таблица 3 — Описание фрагментов информации, относящихся к конфиденциальности пользователя в социальных сетях

| № | Тип информации | Описание |
| --- | --- | --- |
| 1 | Узел существует/не существует | Целевой пользователь присутствует/отсутствует в графе данных |
| 2 | Свойства узла | Степень, метка, положение в графе и т.д. |
| 3 | Взаимосвязь ссылок | Связь между узлами (например, пациент -> врач-онколог) |
| 4 | Конфиденциальная метка узла | Например, информация о заболевании |
| 5 | Конфиденциальная метка ребра | Социальная связь между вершинами |
| 6 | Свойства графа | Для анализа соцсетей используются такие метрики,   как промежуточность, центральность, близость, достижимость |
| 7 | Вес связи | Вес ребра (стоимость коммуникации) |
| 8 | Принадлежность узла к некоторой группе | Связь между узлами (т.е. пользователями) и онлайн группами в социальных сетях |

Базовые знания (БЗ) — это факт или информация, известные злоумышленнику об отдельном лице или группе лиц, которые могут быть использованы для вывода ЧА отдельного лица (или нескольких лиц) на основе *G’*. БЗ могут быть получены из различных источников, и степень их серьезности зависит исключительно от возможностей и технических знаний злоумышленников. На практике очень сложно количественно оценить уровень БЗ, которым обладают злоумышленники, и многие существующие алгоритмы принимают определенные фрагменты информации за БЗ, анонимизируя данные пользователя. В Таблице 4 приведены типы БЗ, которые используются злоумышленниками для нарушения конфиденциальности пользователей социальных сетей во время анализа опубликованных графов [3].

Таблица 4 — Типы базовых знаний, используемых злоумышленниками для нарушения конфиденциальности пользователя

| № | Тип БЗ | Пример | Угрозы конфиденциальности, возникающие на основе БЗ |
| --- | --- | --- | --- |
| 1 | Степень вершины | Количество друзей | Раскрытие личности и членства |
| 2 | Атрибуты вершины | Пол, возраст, профессия, диагноз | Раскрытие личности и содержимого |
| 3 | Взаимосвязь ссылок | Канал связи, социальные связи, онлайн группа | Раскрытие содержания и принадлежности |
| 4 | Соседство | Круг друзей, близкие друзья, подписки/подписчики | Раскрытие личности и содержимого (с некоторой вероятностью) |
| 5 | Взаимное соседство | Близкие друзья, взаимные друзья, друзья друзей | Раскрытие личности и содержимого (с высокой вероятностью) |
| 6 | Встроенный подграф | Ребра и вершины | Раскрытие личности и идентификация группы |
| 7 | Граф знаний | Пол, возраст, отношения, подписки, подписчики | Раскрытие личности и содержимого |
| 8 | Аккаунты в нескольких соцсетях | Информация профиля, детали авторизации, имя (никнейм) | Раскрытие личности и содержимого |
| 9 | Свойства графа | Степень, центральность, близость, подграфы | Раскрытие личности, содержимого и идентификация группы |
| 10 | Поведение пользователей | Рекомендации, интересы, паттерны | Раскрытие личности и содержимого |
| 11 | Вспомогательная информация | Графы других соцсетей | Раскрытие личности, содержимого, членства, принадлежности, а также идентификация группы |

После некоторого обобщения литературы можно представить следующую таксономию подходов к публикации графов с сохранением анонимности (Таблица 5). Эти подходы можно в широком смысле разделить на пять категорий, а именно методы модификации графов, методы обобщения/кластеризации графов, методы вычисления графов с учетом конфиденциальности, методы дифференциальной анонимности графов и гибридные методы анонимизации. Краткое описание вместе с соответствующими примерами всех пяти методов обеспечения анонимности приведено в последующих параграфах [4].

Таблица 5 — Таксономия подходов публикации графов с сохранением конфиденциальности

| Методы анонимизации | Подвиды методов | Примеры операций |
| --- | --- | --- |
| Модификация графов | Ограниченная | k-степень анонимности, k-автоморфизм, k-анонимность по соседству |
| Неограниченная | Случайная вставка (удаление), специальная вставка (удаление), переключение |
| Генерализация и кластеризация графов | Супер узлы | Методы кластеризации/обфускации степени вершины |
| Супер ребра | Методы кластеризации ребер |
| Вычисление графов с учетом конфиденциальности | Аудит запросов | Анализ, оценка и отрицание запроса |
| Возмущение выходных данных запроса | Включение фиктивных данных и инъекции шума |
| Дифференциальная анонимность графов | Дифференциальное вычисление ребер | Запросы с вычислением ребер и сокрытие вывода |
| Дифференциальное вычисление вершин | Методы, основанные на степенях и проекциях |
| Гибридные методы | Кластеризация узлов и ребер | k-соседство основанное на нетронутости ребер при k = 1 |
| k-степень и k-соседство | dK1, dK2, dK3 и методы, основанные на clique |

### Методы модификации графа

Эти методы изменяют структуру графа путем добавления/удаления вершин или ребер для сохранения конфиденциальности пользователя. Помимо добавления/удаления, в некоторых случаях ребра и вершины переключаются или переупорядочиваются в кластеры для сохранения конфиденциальности пользователя. Обзор анонимных графов, полученных путем добавления вершин и ребер, показан на Рисунке 5. На Рисунке 5 (б) были созданы два новых ребра (*{v1, v3}* и *{v2, v4}*). Аналогично, на Рисунке 5 (в) показаны два новых узла (*{v7, v8}*) с четырьмя гранями *({v7, v3}, {v7, v2}, {v8, v1}* и *{v8, v2}*) были добавлены в анонимную версию *G* [1].

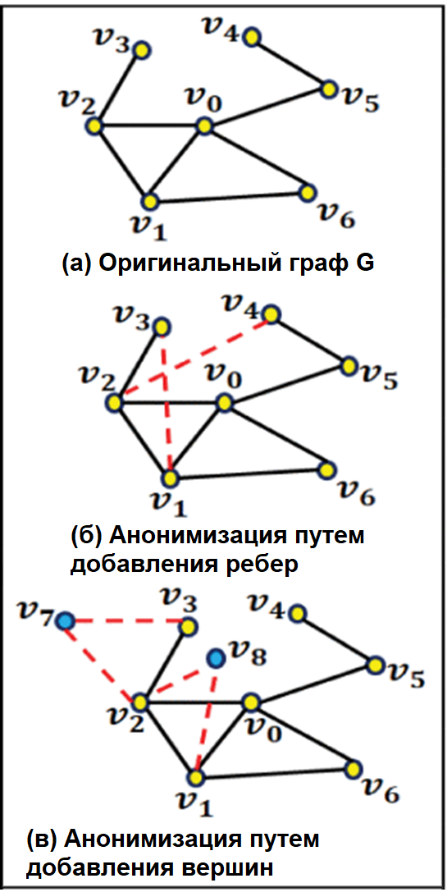


Рисунок 4 — Примеры анонимизации графа путем добавления ребер и вершин

Добавление/удаление узлов и ребер может быть ограниченным или случайным (например, неограниченным) в зависимости от сценария.

Пример анонимизации с помощью методов как ограниченных, так и случайных представлен на Рисунке 6. Оригинальный граф *G* показан на Рисунке 6 (i), (ii). На Рисунке 6 (i)(б) показан пример нарушенной версии сети с помощью операции случайного добавления/удаления. В этом примере два ребра (*{v1, v5}* и *{v2, v3}*) были удалены и добавлены два новых ребра (*{v6, v7}* и *{v8, v9}*) для создания анонимного графа *G’*. Между тем, искаженная версия графа, показанная на Рисунке 6 (i)(в), получена в результате операции случайного переключения. В этом примере два ребра (*{v1, v2}* и *{v4, v5}*) были заменены на (*{v1, v4}* и *{v2, v5}*) для создания анонимного графа *G’*. При случайном возмущении нет жестких ограничений в отношении добавления/удаления/переключения ребер [5].

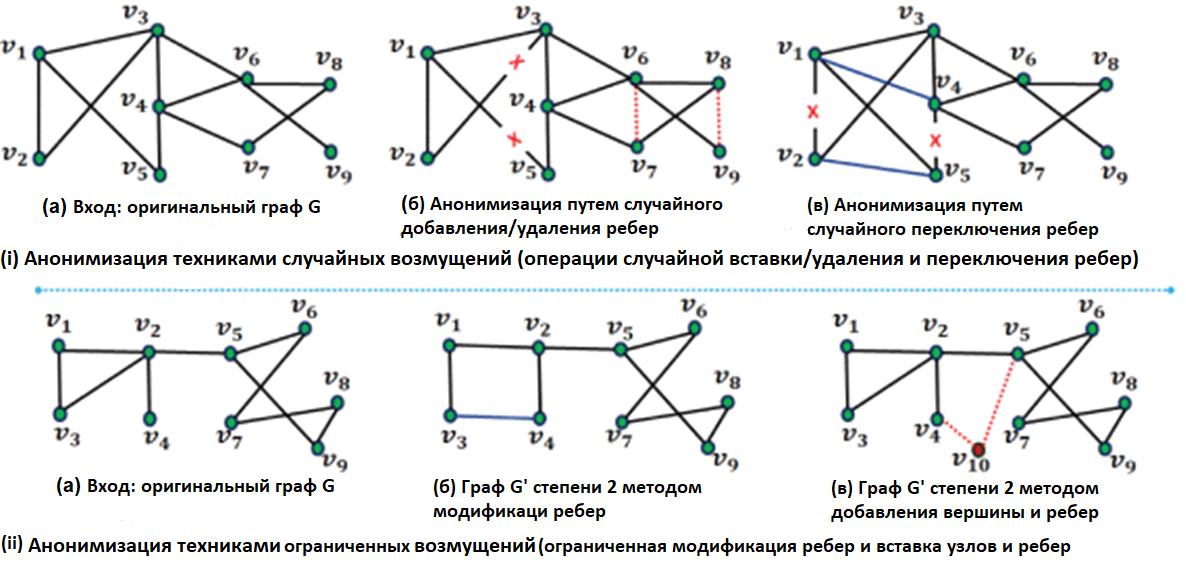


Рисунок 5 — Анонимизация графов с использованием неограниченных и ограниченных методов

В отличие от этого, при ограниченной анонимизации добавление/удаление узлов/ребер ограничено степенью. На Рисунке 6 (ii)(б) показан пример возмущенного графа *G’*, полученного путем применения концепции модификации ребер на исходном графе *G*, приведенном на Рисунке 6 (ii)(a). Возмущенный граф является анонимным *k*-степени, где *k = 2*. Оригинальный граф *G* имеет последовательность степеней *d(G) = {2, 4, 2, 1, 3, 2, 2, 2}*, в то время как модифицированный граф *G’* имеет последовательность степеней *d(G′)={2, 3, 2, 2, 3, 2, 2, 2, 2}*. Количество вершин и ребер в обоих графах одинаковое, а анонимный граф является анонимным 2 степени (т.е. каждая вершина имеет по крайней мере 2 ребра). Единственная модификация в оригинальном графе *G* путем переключения ребра (из *{v3, v2}* в *{v3, v4}*) позволила сделать *G* «2-анонимным». Значение ограничений может быть скорректировано с учетом уровня защиты и структуры графа. Еще один пример анонимности 2-й степени *G’*, полученной путем добавления двух ребер (*{v4, v10}* и *{v5, v10}*), и одной вершины (*v10*), показан на Рисунке 6 (ii)(в). Последовательность степеней *G’* становится *d(G’) = {2, 4, 2, 2, 4, 2, 2, 2, 2, 2},* количество вершин и ребер увеличивается на единицу и две соответственно. При ограниченном возмущении добавление/удаление узлов/ребер соответствует некоторым критериям (т.е. степени, близости, эффективности кластеризации и т.д.), и дальнейшее добавление/удаление узлов/ребер прекращается, как только выполняются определенные критерии [1].

Существует шесть основных методов модификации ребер и вершин для анонимизации данных. Методами модификации являются:

1. добавление ребер,
2. удаление ребер,
3. добавление/удаление ребер,
4. простое переключение ребер,
5. двойное переключение ребер,
6. добавление узлов.

Большинство существующих методов анонимизации данных используют один (или более) из этих шести методов модификации при анонимизации графов. Существует четыре типа графов, которые в основном используются для представления данных пользователей социальных сетей: простой граф, двудольный граф, помеченный граф и неопределенный граф.

### Кластеризация/генерализация графов

Большинство подходов к анонимизации данных социальных сетей основаны на концепциях, которые были предложены для анонимизации табличных данных. Например, k-модель анонимности и ее варианты, такие как k-степень анонимности, k-анонимность изоморфизма, k-анонимность автоморфизма, k-анонимность кандидатов, k-анонимность по соседству и (k, ℓ)-группировка. Подходы, основанные на обобщении/кластеризации, анонимизируют данные социальных сетей путем разделения их на разные кластеры и обобщения кластеров на суперузлы и супер-ребра. Более того, размеры кластеров и степени обобщения определяются таким образом, чтобы в кластеризованной сети сохранялось максимальное количество информации. Концептуальный обзор анонимизации на основе обобщения/кластеризации представлен на Рисунке 7.

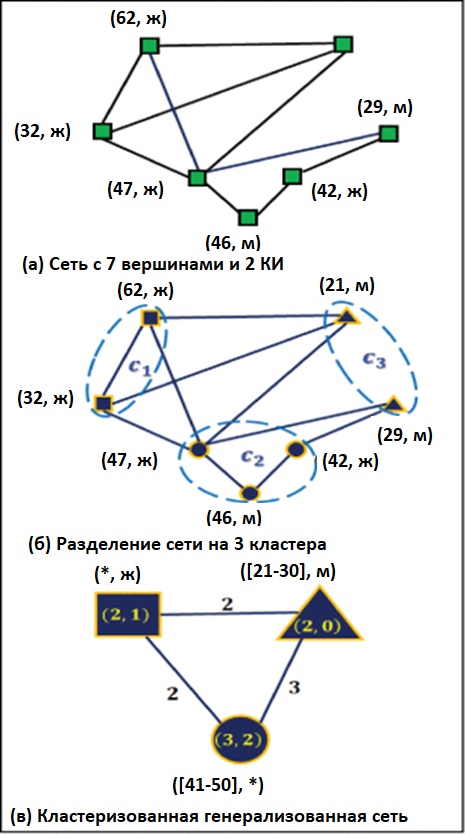


Рисунок 6 — Анонимизация оригинального графа с использованием метода, основанного на кластеризации/обобщении

В качестве входных данных предоставляется сеть (граф) с семью узлами и двумя квази-идентификаторами (возраст, пол) (Рисунок 7 (a)), вся сеть разделена на три кластера *(c1, c2, c3)*, основанных на сходствах КИ (Рисунок 7 (б)), и в качестве выходных данных получается соответствующая обобщенная сеть с тремя суперузлами (Рисунок 7 (в)). Используется три различных символа (квадрат, круг и треугольник) для обозначения пользователей в каждом кластере и суперузлах соответственно. Два числа в каждом суперузле представляют размер кластера (например, количество пользователей) и границы внутри кластера. Например, в кластере *c3* есть два пользователя, но между ними нет границы. Следовательно, значение внутри треугольника (он же суперузел) равно (2, 0). Взвешенные ребра между суперузлами представляют собой межкластерные ребра [6].

### Вычисление графов с учетом конфиденциальности и дифференциальная анонимность графов

Вычисления графов с учетом конфиденциальности и подходы, основанные на дифференциальной анонимности (ДА), не раскрывают всей информации *G’* как и предыдущие два метода. Эти подходы выполняют вычисления на исходных графах *G*, и возвращают выходные данные аналитического вычисления. По сравнению с двумя предыдущими методами, эти подходы допускают ограниченный анализ данных, что может ограничить самый широкий спектр приложений для извлечения знаний и интеллектуального анализа данных. Концепции ДА были адаптированы с учетом структурных свойств для анонимизации графовых данных, и считаются одним из лучших методов вычисления графов с учетом конфиденциальности. Подходы, основанные на ДА, были классифицированы на три категории: ДА на уровне узла, ДА на уровне ребер и ДА на уровне узла и ребер. Эти методы вычисляют полезную статистику из исходного графа таким образом, что сохраняется конфиденциальность пользователя, а данные остаются полезными для аналитических целей [5].

Полезный анализ, предоставляемый методами вычисления графов с учетом конфиденциальности и основанными на ДА подходами, оперирует следующими данными о графе: плотность графа, количество ребер, степень взаимосвязей, распределение степеней, размер сети, централизация, закрытость, количество подграфов, верхние k-пользователи с наивысшей степенью в сети, расстояние/сходство между пользователями, длина пути, коэффициенты кластеризации, обнаружение/извлечение сообщества, гиперграфы, совместное распределение степеней, сокращения, количество пользователей со степенью d, агрегирование, прогнозы, разреженные и плотные сегменты графов — вот лишь некоторые из них. Эти ценные статистические данные могут быть использованы для широкого спектра приложений, включая анализ социальных сетей, маркетинг, анализ предпочтений, совместную фильтрацию, эпидемиологические расследования, распространение информации и т.д.

Помимо этих базовых статистических данных, применение подходов разделения узлов и ребер для публикации разрезов графов и парного расстояния между узлами удобно для приложений, управляемых данными. Кроме того, многие варианты методов ℓ-разнообразия и t-близости были предложены для решения проблем раскрытия контента при публикации данных [1].

### Гибридные методы анонимизации

Гибридные подходы к обеспечению анонимности обычно используют более одного метода обеспечения анонимности для получения анонимизированных данных. Однако сложность гибридных подходов к обеспечению анонимности значительно выше, когда *G* содержит большое количество узлов и ребер. Следовательно, гибридные методы анонимности используются только в конкретных сценариях.

### Сравнение методов

При выборе конкретного метода нужно руководствоваться целями исследования, предметной областью, природой данных и ограничениями конфиденциальности. В вопросе информационной безопасности всегда остро стоит конфликт полезности и безопасности системы. Нужно уметь находить компромисс между полезностью данных и их безопасностью.

Методы модификации графа сохраняют его масштаб, что позволяет анализировать более подробные связи, например, это могут быть связи внутри некоторой социальной группы. При этом слишком высокий уровень анонимизации сделает данные менее понятными и полезными. В основе лежат методы k-анонимизации.

Метод кластеризации больше подходит для анализа нескольких социальных групп. Как было рассмотрено в примере, можно составить кластеры по полу, возрасту или иным КИ. Также этот метод позволяет проводить более точечный анализ при частичной публикации, например, выделить лишь один кластер пользователей, похожих друг на друга, и опубликовать только его. В основе лежат методы кластеризации.

Вычисление графа с учетом анонимности и дифференциальная анонимность вообще не предполагают публикацию графа. В данном случае публикуется лишь некоторая статистическая информация о нем, а также сборщики данных могут отправлять запросы и получать на них ответы, но никогда не получать данные целиком. На данном этапе возникает сложность в том, что нужно понимать, какие запросы отправлять и как обрабатывать полученные ответы. В основе лежат методы статистики.

Гибридный метод позволяет добиться максимальной конфиденциальности, поскольку данные будут обработаны несколько раз, однако это значительно усложняет анализ и вычисление графа.

## Методы деанонимизации

К сожалению, публикация данных может поставить под угрозу конфиденциальность пользователей, поскольку злоумышленники могут получить копию опубликованных данных с целью повторной идентификации людей, используя большое количество вспомогательной информации, полученной из внешних источников. Эта информация может быть собрана из многих источников, включая профили пользователей с сайтов социальных сетей, списки регистрации избирателей, следы передвижения и т.д. Как правило, злоумышленники обладают сильными навыками программирования и знаниями инструментов, поэтому они могут нарушать конфиденциальность пользователя. Благодаря достижениям в области ИКТ масштабы утечек данных растут. В последние годы злоумышленники сосредотачивают внимание на краже информации групп пользователей для достижения намеченных целей. Помимо негативных последствий нарушений конфиденциальности для жизни людей, владельцы данных также теряют доверие своих пользователей в таких обстоятельствах [1].

Интересно, что в некоторых случаях встроенное извлечение знаний позволяет фирмам оставаться конкурентоспособными на рынке в долгосрочной перспективе. В связи с увеличением информационных потоков, доступностью различных социальных сетей, зрелостью средств машинного обучения и интеллектуального анализа данных, развитием вычислительных технологий и возможностями злоумышленников значительно упростился поиск личной информации. Из-за публичного доступа к социальным сетям и недостаточной осведомленности пользователей об анонимности в Интернете защита конфиденциальности на сайтах является очень сложной задачей. Поэтому в последние годы это стало перспективной областью исследований. Злоумышленники могут не только получать информацию о нескольких пользователях, но и повторно идентифицировать людей уникальным образом с помощью множества вспомогательных источников. В Таблице 6 приведены различные подходы к деанонимизации данных, используемые злоумышленниками [7].

Таблица 6 — Описание подходов к деанонимизации

| Подход (используемые функции) | Краткое описание подхода (примеры) |
| --- | --- |
| Сопоставление графов (сетевая структура) | Фокусируется на двух графах одинаковой природы (одной и той же соцсети или возможно двух разных соцсетей). Оба графа используются для сопоставления пользователей между общими анонимизированными узлами (графы знаний, подграфы, встроенные графы). |
| "seed and grow" (растущие ссылки) | Начинается со сложного процесса, называемого посевом, чтобы вырастить вершину, похожую на аккаунт пользователя в графе социальной сети, а затем с помощью ссылок построить другие вершины, основанные на похожести (атрибуты профиля) |
| Сопоставление сходства (отображение графовых функций) | Подсчитывает похожие характеристики между целевым датасетом/графом и дополнительной информацией, чтобы получить корректное сходство (атрибуты пользователя и содержимое) |
| Статистическое сходство (статистика уникальных свойств) | Зависит от использования известных и приобретенных статистик графа. Атака основана на уникальных свойствах пользовательских данных (например, друзья) для повторной идентификации (количество друзей, виртуальные сообщества, принадлежность к группам) |
| Многопоточность (разница в количестве информации) | Фокусируется на последовательных выпусках графов и сопоставлении этих выпусков друг с другом для идентификации добавленных/удаленных пользователей (анализ членства) |
| Классификатор на основе ссылок (дружба, членство в группах) | Анализирует ссылки и информацию о группах пользователей, чтобы сделать вывод о дружбе или членстве в группах для идентификации некоторых личных атрибутов или пользователей целиком (социальные связи) |
| Классификатор на основе графов (членство в группах) | Включает информацию о структуре сообщества, чтобы предсказать наиболее вероятные классы/пользователей для (не)помеченных вершин с целью уникальной повторной идентификации (идентификаторы вершин, профили) |
| Классификатор на основе похожести (активность, поведение) | Подсчитывает похожие паттерны в локальных свойствах, таких как временная активность, текст, география и социальные свойства, с целью повторной идентификации (паттерн использования соцсети, тип соцсети) |
| Пробы (посещенные места, совершенные покупки) | Атака полным перебором для получения личности пользователя или приватной информации из G’ (тип соцсети, активность, имя профиля) |
| Классификатор на основе разреженности (уникальные значения атрибутов и непохожие узлы) | Поиск редких значений, связанных с некоторыми людьми для нарушения конфиденциальности (меньшее количество друзей, локация, уникальная раса/религия) |

Помимо хорошо известных методов, кратко описанных выше, конфиденциальность пользователя также может быть нарушена путем сбора информации об изменении интересов пользователя со временем и прогнозов относительно личной информации пользователя по побочным каналам различных типов (т.е. интересов). В последнее время из-за доступности чрезмерного количества вспомогательной информации и передовых инструментов интеллектуального анализа данных масштабы нарушений конфиденциальности расширяются: от индивидуальной идентификации или раскрытия ЧА до кражи личных данных групп пользователей и выявления сообществ, имеющих общие характеристики или интересы, для получения точных рекомендаций.

Кроме того, когда группа пользователей формирует онлайн-сообщество, поставщики услуг социальных сетей получают доступ к большему количеству информации, включая политические взгляды, предпочтения, семейное или финансовое положение, поскольку пользователь часто с готовностью делится большей информацией о себе в онлайн-сообществе. Соответственно, передача этой золотой жилы данных аналитикам может привести к нарушениям конфиденциальности. Следовательно, совместное использование данных пользователями соцсетей может неожиданным образом поставить под угрозу конфиденциальность [1].

## Проблемы анонимизации данных

Данные социальных сетей обычно представляются в виде графа, и для их очистки перед публикацией применяется подход структурной анонимизации. Анонимизация данных социальных сетей намного сложнее по сравнению с табличными данными из-за сложной структуры и разнообразия информации об объектах (т.е. пользователях), встроенной в графы. Три хорошо известные проблемы, связанные с анонимизацией данных социальных сетей, заключаются в следующем [1]:

1. Моделирование базовых знаний (БЗ) злоумышленников. В социальных сетях соответствующее моделирование БЗ злоумышленников сложнее по сравнению с табличными данными, потому что злоумышленники могут иметь доступ к множеству фрагментов информации о человеке, что позволяет повторно идентифицировать целевого человека из базы данных *G’* используя БЗ.
2. Разработка нового метода анонимизации для данных социальных сетей очень сложна по сравнению с табличными данными из-за структурной зависимости объектов друг от друга. В социальной сети небольшое изменение структуры графа может повлиять на всю сеть. Следовательно, решения adhoc, основанные на подходе «разделяй и властвуй», не могут быть непосредственно применены к данным социальные сети.
3. Количественная оценка полезности анонимного графа. В данных социальных сетей измерить полезность, предлагаемую анонимным графом, непросто. Различия в свойствах *G* и *G’* трудно поддаются количественной оценке. Кроме того, добавление новых ребер и вершин для повышения защиты конфиденциальности часто может приводить к чрезмерной потере информации.

Помимо трех ключевых проблем, описанных выше, структурная анонимизация графов большого масштаба, включающих данные многих объектов, является очень сложной задачей. Кроме того, количество и разнообразие собираемых данных о юридических лицах экспоненциально увеличивается с течением времени. Таким образом, разработка новых структурных подходов к анонимизации для решения этих проблем как с практической, так и с теоретической точек зрения стала насущной необходимостью при одновременном использовании данных пользователей.

## Перспективные открытые направления исследований

Ниже описаны некоторые перспективные открытые направления исследований и проблемы, которые требуют дальнейших исследований и разработок как со стороны академических кругов, так и промышленности.

*Проблемы конфиденциальности групп пользователей*: при анонимизации данных большинство существующих подходов сосредоточены исключительно на сохранении конфиденциальности отдельного человека. Таким образом, они менее устойчивы к сохранению конфиденциальности групп пользователей. Например, k-модель анонимности создает классы эквивалентности с k-пользователями в каждом классе. С одной стороны, это защищает индивидуальную конфиденциальность, скрывая каждого индивида от других k-групп. С другой стороны, она явно раскрывает информацию о группах пользователей.

*Анонимизация несбалансированных наборов данных*: в некоторых случаях реляционный набор данных может быть сильно несбалансированным (т.е. распределение значений ЧА неравномерно), и его анонимизация является очень сложной задачей. В таких наборах данных применяются жесткие ограничения, такие как создание каждого класса ℓ-разнообразия или t-близости на практике невозможно. Следовательно, требуется разработка новых подходов к анонимизации несбалансированных наборов данных для эффективной защиты конфиденциальности пользователей без снижения полезности данных.

*Эффективное решение проблемы компромисса между конфиденциальностью и полезностью*: при анонимизации данных существует серьезный компромисс между конфиденциальностью и полезностью. Согласование анонимизации с целями обеспечения конфиденциальности может отрицательно сказаться на полезности анонимных данных, и наоборот. Эта давняя проблема заключается в поиске новых решений для поддержки анализа больших данных с сохранением конфиденциальности.

*Персонализированное сохранение конфиденциальности в социальных сетях*: в социальных сетях у каждого пользователя свои требования и опасения по поводу конфиденциальности своей информации, которая называется персонализированной конфиденциальностью. Например, некоторые пользователи могут захотеть скрыть только информацию о семейном положении, в то время как другие могут захотеть скрыть информацию обо всех социальных связях (например, обо всех друзьях). Таким образом, персонализированная конфиденциальность предполагает высокий уровень субъективности, и его очень сложно реализовать. Следовательно, требуются инновационные решения, которые могут учитывать требования персонализированной конфиденциальностью пользователей к анонимизации данных.

*Точное моделирование базовых знаний злоумышленников*: злоумышленники получают большое количество информации по побочным каналам в виде базовых знаний о пользователях социальных сетей. Этот список продолжает расти благодаря доступу к другим общедоступным социальным сетям. В последнее время методы интеллектуального анализа текста и обработки естественного языка (NLP) используют содержимое соцсетей пользователей для сопоставления с их личными данными. Таким образом, точное моделирование БЗ при анонимизации данных социальных сетей является очень сложной задачей, и необходимы дальнейшие исследования того, как моделировать БЗ в процессе анонимизации, чтобы эффективно защитить конфиденциальность пользователей в эпоху больших данных.

*Общие решения для анонимизации социальных графов*: как правило, данные социальных сетей моделируются с помощью графов (также известных как социограммы). Эти графы могут быть разных типов, таких как простые, направленные, взвешенные и помеченные. Механизм анонимизации, предложенный для одного типа графа, не может быть непосредственно применен к другому. Например, k-анонимность степеней не может быть применена к ориентированным графам напрямую, поскольку это требует детального анализа последовательностей входящих и исходящих степеней. Следовательно, разработка универсальных методов анонимизации, которые могут работать с несколькими типами графов, станет интересной областью исследований в будущем.

*Контроль крупномасштабных проблем с идентификацией пользователей путем обхода инструментов интеллектуального анализа данных*: в социальных сетях пользователи устанавливают отношения с единомышленниками или людьми, имеющими схожие интересы. Это приводит к формированию онлайн-сообществ. Все больше исследований посвящается именно их выявлению. С одной стороны, обнаружение сообществ полезно для множества целей, таких как распространение и контроль информации. С другой стороны, публикация данных сообщества может поставить под угрозу анонимность как его самого, так и его пользователей. Следовательно, разработка новых решений, устойчивых к обнаружению сообщества и сопоставлению узлов (пользователей) сообщества при публикации данных, стала более актуальной, чем когда-либо.

*Использование глобальных и локальных особенностей данных для защиты от проблем с согласованием сети:* в последнее время активизировалась идентификация пользователей в социальных сетях с использованием множества методов, таких как сопоставление нескольких графов, сопоставление отображаемых имен, анализ содержимого и действий, учетных записей в разнородных социальных сетях и их комбинаций. Соответственно, подходы к обеспечению конфиденциальности нуждаются в значительном совершенствовании в проблемах согласования сети. В связи с этим подходы, которые обеспечивают анонимизацию данных путем использования локальных (т.е. общих отображенных соседей, количества друзей и общих знакомых и т.д.) и глобальных (т.е. взаимосвязи, централизации, силы связей, информации о соседях с несколькими переходами и т.д.) особенностей социального графа, станут перспективной областью исследований в ближайшем будущем.

*Наборы метрик, а не одна метрика для количественной оценки уровня конфиденциальности в анонимных графах*: как правило, для оценки уровня конфиденциальности и полезности в *G’* используется один тип метрик. Более того, в реальных случаях конфиденциальность, определяемая одним показателем, может быть не монотонной (например, показывать более низкие результаты конфиденциальности для более сильных злоумышленников) или надежной с нескольких точек зрения. Следовательно, объединение нескольких показателей конфиденциальности, которые могут более точно измерять уровень конфиденциальности и могут смягчить недостатки отдельных показателей, является многообещающим направлением исследований в ближайшем будущем.

*Разработка механизмов, обеспечивающих конфиденциальность, для исключительных ситуаций*: в течение 2020 года весь мир столкнулся с непредвиденным вызовом со стороны неизвестного врага, называемого коронавирусной инфекцией. Пандемия COVID-19 затронула все профессии по всему миру, и правительства в значительной степени полагаются на немедикаментозные вмешательства (например, строгие карантины, закрытие городов и учреждений, социальное дистанцирование, анализ мобильности пользователей на основе географического местоположения, обнаружение близости, отслеживание цифровых контактов и т.д.) для сдерживания распространения COVID-19. Кроме того, для проведения эпидемиологических расследований некоторые правительства применяли обширные меры (например, данные кредитной карты, сигналы мобильного телефона, данные Bluetooth и GPS, а также данные видеонаблюдения), чтобы найти зараженных людей и особо опасные очаги распространения. Из-за внедрения цифровых методов огромное количество персональных данных попало в киберпространство, по всему миру постоянно происходят сливы информации. Например, в Италии с января по апрель 2020 года количество нарушений конфиденциальности в секторе здравоохранения, связанных с компаниями и частными лицами, удвоилось. Кроме того, ожидается, что количество нарушений конфиденциальности в эпоху после COVID-19 возрастет, поскольку многие компании собирают многофакторные данные об образе жизни людей. Учитывая необходимость сохранения конфиденциальности пользователей, необходимо тщательно соблюдать этические аспекты во время и после эры COVID-19. Следовательно, в исключительных ситуациях, таких как пандемия, требуются решения, учитывающие конфиденциальность, для защиты данных пациентов и других аспектов, связанных с пандемией (например, отслеживание симптомов с сохранением конфиденциальности и отчетность о них, сбор только релевантной информации от пользователей для защиты конфиденциальности, шифрование конфиденциальной информации, отслеживание контактов, а также децентрализованные решения для вычисления вероятности заражения). Следовательно, анализ и устранение потенциальных рисков конфиденциальности и уязвимостей в приложениях для отслеживания контактов, разработанных многими странами для борьбы с пандемией, является динамично развивающейся областью исследований [1].

## Вывод по аналитической части

В первой главе были изучены основный аспекты анонимизации графов. Были даны определения графу данных, анонимизации данных, рассмотрены виды информации, хранимой в графах, и проблемы её безопасности. Был изучен процесс публикации данных и подробно рассмотрен каждый его этап, что дало общее понимание о графах данных, зачем нужна анонимизация и кто использует данные.

Далее были рассмотрены конкретные подходы к анонимизации данных. Всего было выделено 5 основных подходов, каждый из которых имеет в себе различные алгоритмы. Данные подходы можно применять как изолированно, так и в совокупности, однако важно понимать природу данных, предметную область и цели анонимизации. Каждый подход имеет свои сильные и слабые стороны, поэтому для различных ситуаций могут применяться различные методики. Также стоит помнить, что чем выше конфиденциальность данных, тем ниже их полезность и наоборот. Поэтому необходимо также учитывать, какие данные действительно важны для анализа, а какие можно скрыть.

Несмотря на все вышеописанные техники сохранения конфиденциальности пользователей, тем не менее все ещё существует угроза деанонимизации со стороны злоумышленников. Возможные методы деанонимизации были также описаны в данной главе. К сожалению, полная анонимность невозможна, поскольку даже при соблюдении всех правил владельцами данных, пользователи сами могут публиковать какую-либо информацию о себе в открытый доступ. Это одна из проблем анонимизации наряду с разработкой новых методов анонимизации и оценки ее качества.

Отсюда вытекают некоторые перспективные направления исследований в данной области. Резюмируя все вышесказанное, автор делает вывод, что несмотря на активное развитие данной темы в научном сообществе, все ещё есть большое поле для исследований.

# Практическая часть

## Описание алгоритма анонимизации направленного графа

### Обозначения

Пусть *G = (V, A)* — ориентированный и немаркированный граф (также называемый орграфом), где *V* — множество вершин (или узлов), а *A* — множество дуг (или ребер) в *G*. Определим *n = |V|* для обозначения количества вершин и *m = |A|* для обозначения количества дуг. Используем *(vi, vj) ∈ A* для обозначения направленной дуги от вершины *vi* к *vj*, но не наоборот. Наконец, обозначим через *G = (V, A)* и *G’ = (V’, A’)* исходный и возмущенный граф, созданный в процессе анонимизации, соответственно [8].

### Модели k-анонимности в орграфах

Концепция анонимности k-степени может быть непосредственно сопоставлена с последовательностью степеней.

**Определение 1**: Вектор целых чисел *V* является k-анонимным, если каждое отдельное значение *vi ∈ V* появляется по крайней мере k раз.

**Определение 2**: Неориентированная сеть *G = (V, E)* является k-анонимной, если последовательность степеней *G* k-анонимна.

Пусть *V* и *W* соответствуют последовательностям степеней входного и анонимизированного графа соответственно. Расстояние между двумя векторами целых чисел *V = [v1, . . . , vn]* и *W = [w1, . . . , wn]* определяется уравнением 1:

где *vi ∈ V , wi ∈ W* и *|V | = |W| = n*. Чем меньше значение ∆, тем меньше потеря информации в анонимизированной сети.

Прямые наследники вершины *vi ∈ V*, обозначаемые *Γ+(vi)*, определяются как вершины на расстоянии 1 от *vi*, т.е. все *vj: (vi, vj) ∈ A*. Число преемников определяется как степень выхода вершины, . Аналогично, прямыми наследниками вершины *vi* являются все вершины, из которых *vi* может быть достигнута за один прыжок. То есть , а степень вершины определяется как . Следовательно, ориентированный граф связан с двумя последовательностями степеней: последовательностью входящих степеней и последовательностью исходящих степеней . Поскольку каждая дуга соединяет две вершины, очевидно, что:

Важно отметить, что в процессе анонимизации одинаковое количество дуг должно быть добавлено как к in-degree, так и к out-degree, поскольку каждая добавленная дуга подразумевает добавление значения на единицу к in-degree, а также к out-degree. Таким образом, анонимные внутренняя и внешняя степени должны удовлетворять уравнению 2.

Далее рассмотрим две модели для достижения различных уровней конфиденциальности в соответствии с моделью k-анонимности.

Модель независимой анонимности (ki, ko) степени предполагает, что злоумышленник знает степень входа ИЛИ выхода некоторых целевых вершин, но не знает степень входа И выхода целевых вершин.

**Определение 3:** Орграф *G = (V, A)* является независимым *(ki, ko)*-анонимным по степени, если последовательность входящих степеней *G* является ki-анонимной, а последовательность исходящих степеней — ko-анонимной.

В случае, когда ki = ko = k, называется просто независимой анонимностью k-степени.

**Определение 4**: Орграф *G = (V, A)* является независимым анонимным k-степени, если как входная, так и выходная последовательности *G* являются k-анонимными.

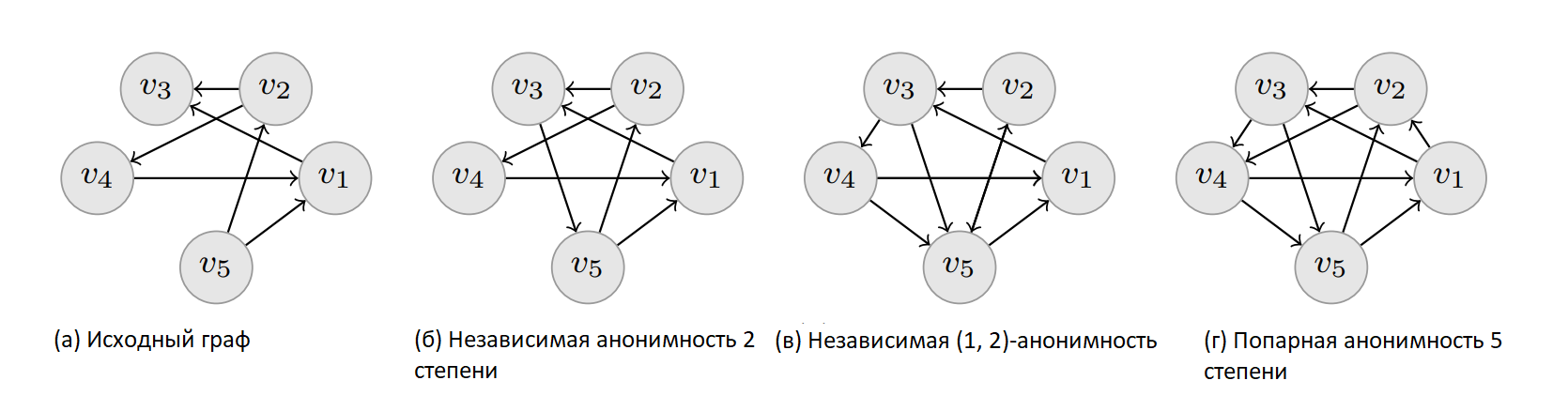


Рисунок 7 — Процесс анонимизации

Наглядный пример независимой анонимности k-степени можно увидеть на Рисунке 7. Исходная сеть, показанная на Рисунке 7 (а), содержит 5 вершин и 6 дуг, а ее последовательности степеней равны *din = {2, 1, 2, 1, 0}* и *dout = {1, 2, 0, 1, 2}*. Таким образом, добавления всего одной дуги от *v3* до *v5* достаточно, чтобы преобразовать эту сеть в независимый граф анонимности 2-й степени. На Рисунке 7 (б) показана анонимная сеть, которая имеет *din = {2, 1, 2, 1, 1}* и *dout = {1, 2, 1, 1, 2}*.

Граф, представленный на Рисунке 7 (б), также является (2, 2)-анонимным в соответствии с определением. Однако, используя эту модель, можно создавать асимметричные уровни конфиденциальности, если учесть, что, например, степень недоступности некоторых целевых вершин может быть основным знанием злоумышленника, и необходимо соответствующим образом защитить данную сеть. На Рисунке 7 (в) показана независимая (1, 2)-анонимная версия графа, где *din = {2, 1, 2, 1, 3}* и *dout = {1, 1, 2, 2, 1}*. Следовательно, можно повторно идентифицировать пользователя, используя информацию о степени, но это невозможно, используя информацию, относящуюся к степени некоторых целевых вершин.

Модель парной анонимности k-степени предполагает, что злоумышленник знает, как внутреннюю, так и внешнюю степень некоторых целевых вершин. Очевидно, что эта модель обеспечивает более высокую степень защиты конфиденциальности, чем вышеупомянутые модели, поскольку она также защищает пользователей от злоумышленника, который знает только степень входа или выхода некоторых целевых вершин. Определим парные степени вершины в виде пары целых чисел, где первое число — это внутренняя степень вершины, а второе — внешняя степень, то есть .

**Определение 5**: Орграф *G = (V, A)* является попарно анонимным степени k, если последовательность парных степеней *G* является k-анонимной, т.е. для каждой пары *(a, b)*, представляющей внутреннюю и внешнюю степени вершины, существует по меньшей мере *k – 1* другие пары с теми же значениями.

Обратите внимание, что парный анонимный граф k-степени всегда является независимым (k, k)-анонимным первой степени, но не наоборот. Таким образом, парная анонимность k-степени сильнее, чем независимая анонимность (k, k)-степени.

На Рисунке 7 (г) представлена парная версия анонимности 5 степени. Для выполнения свойств этой модели необходимо добавить четыре дуги, а ее последовательности степеней — *din = {2, 2, 2, 2, 2}* и *dout = {2, 2, 2, 2, 2}*. Интересно отметить, что эта сеть также является независимой (5, 5)-анонимной. Более того, сеть, изображенная на Рисунке 7 (в), является независимой (2, 2)-анонимной, но не попарно 2-анонимной. На самом деле, она попарно 1-анонимная [8].

### Описание применяемого подхода

В этом разделе рассмотрим алгоритм DGA (Directed Graph Anonymization), разработанный для сохранения конфиденциальности пользователей в непомеченных орграфах в соответствии с предлагаемыми моделями анонимизации. В алгоритме используется концепция анонимности k-степени для анонимизации отношений пользователей, то есть выполняются изменения только в наборе ребер, чтобы сгенерировать новый анонимный граф , где *Gk* — анонимность k-степени и |*Ak*| минимально.

Данный подход к анонимизации ориентированного графа основан на определении 2. Таким образом, преобразуется как входная, так и выходная последовательности *G = (V, A)* путем добавления ребер, чтобы обеспечить анонимность k-степени для ориентированного графа. Подход основан на двух шагах:

1. Анонимизация последовательностей степеней. Строится k-анонимная последовательность из последовательности исходного графа, используя определение 1. Тот же процесс применяется для получения анонимизированной последовательности *doutk*.

2. Добавление поддельных дуг. Второй шаг добавляет поддельные дуги между вершинами, чтобы соответствовать анонимизированной степени входа (*dink*) и степени выхода (*doutk*), получая анонимный ориентированный граф k-степени , где |*Ak*| минимально [7].

#### Шаг I: Анонимизация последовательностей степеней

В первую очередь рассмотрим модель независимой анонимности (ki, ko)-степени. Один и тот же процесс применяется как к входящим (*din*), так и к исходящим последовательностям (*dout*). Цель этого шага заключается в анонимизации последовательности степеней исходной сети. Оптимальная одномерная микроагрегация Хансена и Мукерджи [9] используется для достижения наилучшего распределения групп как для входящих, так и для исходящих последовательностей, а затем вычисляются значения для каждой группы, которые минимизируют расстояние от исходных степенных последовательностей по уравнению 1.

Алгоритм имеет сложность *O(k2n)* в силу его гибкости; изменяя только один параметр, он может вычислить оптимальные k-анонимные последовательности степеней для различных метрик, таких как евклидова, линейная или любая функция узлов в k-группах. Более того, алгоритм был реализован с улучшениями, предложенными в [10], которые значительно сокращают время выполнения.

Представленный подход начинается с применения перестановки *f* к последовательности степеней для изменения порядка элементов. Упорядоченная монотонная последовательность степеней называется неубывающей последовательностью степеней вершины, то есть . Пусть *k* — целое число, такое, что *1 ≤ k < n*, которое является значением анонимности k-степени, т.е. *k*i в случае in-degree и *ko* в противном случае. Как правило, *k* намного меньше *n*. Чтобы применить оптимальную одномерную микроагрегацию согласно [9], необходимо построить новую направленную сеть *Hk,n* и получить оптимальное разбиение, которое представляет собой именно тот набор групп, который соответствует дугам кратчайшего пути от вершины 0 до вершины *n* на этом графе. Обозначим через оптимальное разбиение, где , и каждое из них имеет от *k* до *2k – 1* элементов. Очевидно, что каждый *di ∈ d* принадлежит определенной группе *gj ∈ g*. Поскольку подход основан только на добавлении ребер для изменения структуры графа, необходимо увеличить или сохранить те же значения степени, но не уменьшать ни одно из них, что было бы эквивалентно удалению ребер. Следовательно, оптимальное разбиение соответствует увеличению значения степени каждой вершины до максимального значения ее группы, т.е. Стоимость кратчайшего пути на *Hk,n* обозначает количество добавленных дуг, которое необходимо для достижения значения k-анонимности [7].

Следующим шагом выполняется парная анонимность k-степени. В этой модели нужно одновременно учитывать, как входную, так и выходную степень каждой вершины. Таким образом, каждая пара представляет внутреннюю и внешнюю степени вершины *vi*.

Согласно определению 5, необходимо найти оптимальное разбиение в двумерном пространстве. Было доказано, что задача решения о нахождении парной анонимной последовательности k-степени путем добавления ровно *s* ребер является NP-трудной (см. [11], теорема 23). Следовательно, необходимо использовать многомерную микроагрегацию, чтобы найти квазиоптимальные разбиения за разумное время; в частности, применим алгоритм MDAV [12, 13]. Аналогично вышеупомянутому методу, оптимальное разбиение соответствует увеличению парных значений степени каждой вершины до максимальных парных значений ее группы.

Важно отметить, что к последовательностям внутренней и внешней степеней необходимо добавить одинаковое количество дуг, поскольку каждая новая дуга подразумевает добавление единицы как к внутренней, так и к внешней степенной последовательности. Следовательно, анонимные последовательности должны удовлетворять уравнению 2.

Обозначим как *ηin* количество добавленных дуг во входящей последовательности степеней и через *ηout* количество добавленных дуг в исходящей последовательности для заданной анонимизации k-степени ориентированного графа *G*. Если *ηin ≠ ηout*, то анонимные последовательности степеней не удовлетворяет уравнению 2, которое требуется для ориентированных графов. Следовательно, минимальное количество дуг, которые необходимо добавить к исходному графу, равно, по крайней мере, *max{ηin, ηout}*, если считать, что *ηin*, *ηout* — это количество ребер, необходимых для оптимального микроагрегирования для последовательностей степеней входа/выхода соответственно. Следовательно, если получаем анонимную последовательность k-степени с максимальными ребрами (*ηin*, *ηout*), то она оптимальна.

Пусть *Sin* и *Sout* — оптимальное разбиение последовательности на входную и выходную степени, полученное после применения алгоритмов микроагрегации, где Обратите внимание, что количество разбиений не обязательно должно быть равным (*p ≠ q*). Также важно отметить, что минимальное добавление ребер для выполнения уравнения 2 представлено нахождением минимальных значений для решения:

где и представляют количество добавленных ребер в разбиении *i*, вычисленное по уравнению 4, *αi, βi ≥ 0* и *αi, βi ∈ N*:

где . Для упрощения уравнения и в расчетах будем рассматривать только различные размеры и , которые обозначаются *ai* и *bi* соответственно. Будем обозначать как *R*. Тогда можем получить следующее уравнение из уравнения 3:

где *p' < p* и *q' < q*, поскольку повторяющиеся значения и удаляются. По той же причине значения *αi, βi* в уравнении 5 отличаются от значений в уравнении 3.

Напомним, что при оптимальной микроагрегации для всех *i ≤ max(p, q)*. Следовательно, *k ≤ ai, bi ≤ 2k – 1* для всех *i ≤ max{p', q’}*. Если предположить, что для данного *i0*, то получим уравнение:

Следовательно, решение может быть получено путем решения следующего уравнения:

Теперь, поскольку работа происходит с соответствиями , можно считать, что коэффициенты *αi, βi* меньше что значительно сокращает пространство поиска решений. В худшем случае можно получить решение методом перебора, учитывая все комбинации , что было бы поиском в *O(kk)*, поскольку Более того, на практике можно найти решения уравнения 7 намного быстрее. Во всех рассмотренных последовательностях *αi = 0* для всех *i*. В то время как для некоторых *i1 ≠ i0* и была проверена конгруэнтность поэтому в большинстве случаев достаточно рассмотреть только одну переменную *i1 ≠ i0* [7].

#### Шаг II: Модификация графа

Как упоминалось ранее, алгоритм основан на добавлении поддельных дуг. Другие методы анонимизируют структуру графа путем добавления и удаления дуг, а не только дополнений. В данном подходе рассматривается возможность сохранения дуг исходной сети, поскольку истинные отношения между пользователями могут быть важны для кластеризации или других задач интеллектуального анализа графов. Авторы [14] эмпирически доказали, что добавление ребер является лучшим методом сохранения свойств графа при анонимизации.

После того, как были вычислены k-анонимные входные и выходные последовательности, вычисляется вектор различий между исходной и анонимной последовательностями. То есть и Каждый вектор четко показывает, какие вершины должны увеличить свою степень входа (*δin*) и выхода (*δout*). Для каждого из них используется три процесса модификации ребер, чтобы увеличить степень входа и выхода вершин в *δin* и *δout* соответственно:

1. Добавление ребер: случайным образом выбираем комбинацию вершин, которая удовлетворяет *(vi, vj) ∉ A*, где *vi ∈ δout : δout(vi) > 0* и *vj ∈ δin : δin(vj) > 0*. Степень выхода вершины *vi* и степень входа вершины *vj* увеличиваются, как показано на Рисунке 8 (а).

2. Переключение ребер: происходит между четырьмя вершинами *vi, vj*, *vk, vp* *∈ V*, где *(vi, vj), (vk, vp) ∈ A* и (*vi, vp)* и *(vk, vj*) *∉* A. Он определяется путем удаления дуги *(vk, vp)* и добавления новых дуг (*vi, vp)* и *(vk, vj*), как показано на Рисунке 8 (б). Обратите внимание, что внешняя степень вершины *vi* и внутренняя степень вершины *vj* увеличатся на 1, в то время как степень других вершин останется прежней.

3. Расширение ребра существует между тремя вершинами *vi, vk, vp ∈ V*, где *(vk, vp) ∈ A* и *(vk, vi), (vi, vp)* *∉* A. Дуга *(vk, vp)* удаляется и образуются новые дуги *(vk, vi)* и *(vi, vp)* создаются, как показано на Рисунке 8 (в). Обратите внимание, что степень входа и выхода вершины *vi* увеличивается, в то время как степень вспомогательных вершин остается неизменной.

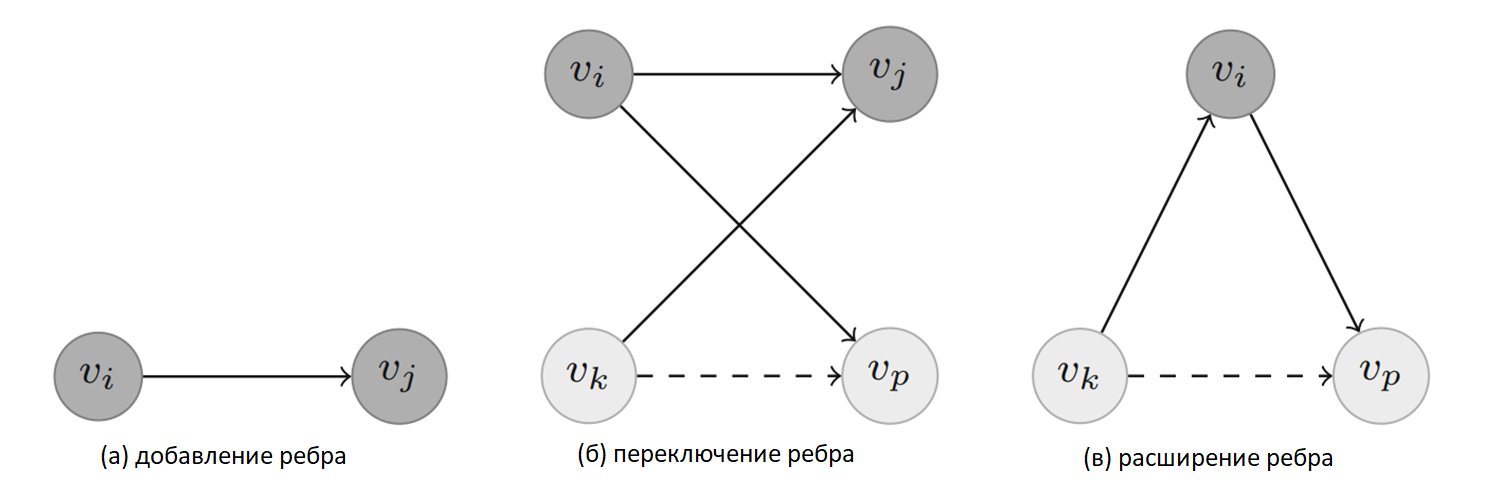


Рисунок 8 — Иллюстрация процессов добавления, переключения и расширения ребер

Процесс описан в алгоритме 1 (см. Приложение 1). Для каждой вершины *vi ∈ δout* алгоритм находит *vi ∈ δin* и добавляет дугу между ними. Из-за разреженности границ реальных сетей этот процесс возможен в нескольких случаях. Однако в некоторых случаях невозможно создать поддельное ребро, как описано ранее. Может случиться так, что порядок добавления ребер может иметь значение, как показано на Рисунке 9. Предположим, что анонимная последовательность такова *δout* = (2, 2, 3) и *δin* = (2, 2, 3), и алгоритм сначала добавил ребра (*v1*, *v4*) и (*v2*, *v5*), как на Рисунке 9 (б). В этом случае будет невозможно применить ни один из трех представленных методов модификации ребер, чтобы добавить единицу к степеням узлов *v3* и *v6*; однако, добавив правильные ребра, последовательность могла бы быть получена с помощью описанных методов модификации ребер, как показано на Рисунке 9 (в). Однако при проведении экспериментов на реальных данных такая ситуация никогда не происходила, возможно, из-за разреженности социальных сетей, а также из-за того, что алгоритмы выбирают добавленные ребра случайным образом — попадая в такую ситуацию, когда никакое возможное ребро не может быть добавлено, потребуется только повторно запустить алгоритм, чтобы избежать этого.

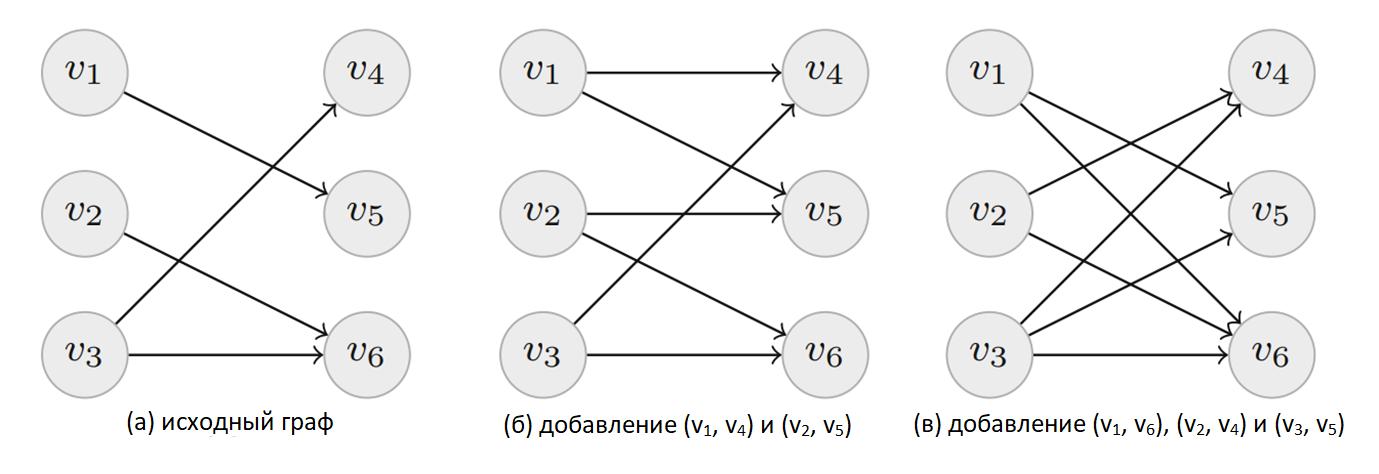


Рисунок 9 — Порядок добавления ребер может иметь значение

## Описание алгоритма анонимизации ненаправленного графа

Пусть *G = (V, E)* — ненаправленный и невзвешенный граф, где *V* — множество вершин (или узлов), а *E* — множество дуг (или ребер) в *G*. В контексте анализа некоторого социума набор *V* представляет отдельных людей, а набор *E* представляет связи между ними. Основная цель этого метода — сделать граф в k-степени анонимным с минимальной потерей информации. Методы анонимности k*-*степени, основанные на модификации графа, гарантируют, что степени всех узлов являются k*-*анонимными. То есть степень каждой вершины в сети, по крайней мере, равна *k* – 1 другим вершинам.

Рассмотрим сценарий анонимизации, представленный на Рисунке 10. Имеется граф с 9 узлами. Хотя граф небольшой, наглядно можно увидеть значительное влияние k-анонимности на структуру графа. Степень анонимности *k* была определена для каждого узла так, чтобы он имел одинаковую степень по крайней мере с *k* – 1 другими узлами графа. Следовательно, на Рисунке 10 (а) требуются два ребра для соединения узлов 1, 4, 6 и 9, чтобы сделать граф анонимным 3 степени. Для достижения анонимности k*-*степени существует три различных способа соединения узлов, проиллюстрированных на Рисунке 10 (б), (в) и (г). Все три варианта приемлемы, но наиболее благоприятным является вариант с минимальными изменениями в исходном графе. Для оценки изменений могут использоваться различные метрики графа, например, центральность, цикличность, близость, длина пути и т.д [15].

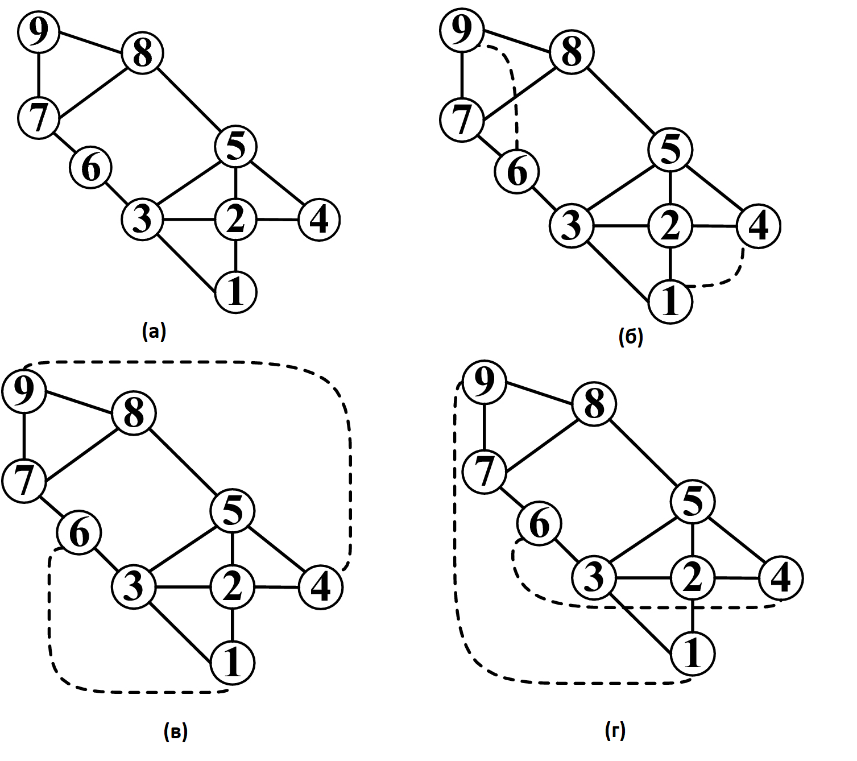


Рисунок 10 — Варианты анонимизации ненавправленного графа

### Описание применяемого подхода

Предлагаемый алгоритм основан на метаэвристическом алгоритме под названием генетическая анонимность k*-*степени с использованием подхода модификации границ (*GKDEM*) и определения структуры сообщества социальных сетей. Полученный граф становится k-анонимным путем модификации исходного графа сети *G*(*V*, *E*). Он становится максимально анонимным из-за природы метаэвристических алгоритмов [16]. Этапы анонимности предлагаемого подхода описаны следующим образом.

1. На первом этапе в качестве входных данных берется последовательности степеней вершин графа в виде набора пар [(*a*, *b*)]. Каждая пара обозначает ребро между вершинами *a* и *b*. Операции на первом шаге следующие:
2. Разбиение вершин исходного графа на основе значения *k* с использованием последовательности степеней вершин графа для вычисления требуемой степени для каждой вершины для достижения k-анонимности.
3. Группирование вершин в похожие группы относительно их ребер. Обнаруженные группы будут использоваться для уменьшения искажений графа.

Результатом является количество ребер, требуемых каждому узлу для достижения k-анонимности в каждой группе. Для выполнения разбиения вершин графа используется алгоритм разбиения графа Ма [17], для обнаружения групп вершин используется алгоритм Лувена [18].

1. Вторым шагом является модификация графа. Результатом предыдущего шага является последовательность, которая показывает количество ребер, необходимых для каждой вершины в каждой группе для достижения k-анонимности. Генетический алгоритм используется для модификации графа путем добавления недостающих ребер к каждому узлу. Выполнение генетического алгоритма для каждой группы анонимизирует ее. Все этапы предлагаемого подхода кратко обобщены в Таблице 7.

Таблица 7 — Описание этапов подхода

|  |  |  |  |
| --- | --- | --- | --- |
| Алгоритм | Цель | Вход | Выход |
| Секционирование графа | Вычислить требуемую степенную последовательности для каждой вершины, чтобы достичь  k-анонимности | Пара вершин, которые соединены ребром | Пары вершин и количество ребер, необходимых для достижения  k-анонимности |
| Поиск групп | Уменьшить искажение графа | Пара вершин, которые соединены ребром | Группы |
| Генетический алгоритм | Добавить необходимые ребра для каждой вершины в группе, чтобы достичь  k-анонимности | Пары вершин и количество ребер, необходимых для достижения k-анонимности в группе | Анонимизированные группы |

В принципе, все алгоритмы, которые выполняют модификацию графа, неизбежно изменяют его структуру из-за добавления ребер или узлов, чтобы сделать граф анонимным. Алгоритм обнаружения групп используется для минимизации изменения структуры графа. Выходные данные алгоритма обнаружения групп представляют вершины в группе, которые используются в генетическом алгоритме для включения новых ребер между ними. Генетический алгоритм идентифицирует вершины, у которых отсутствуют некоторые ребра, и на основе функции пригодности создает допустимые ребра. Достоверность ребра определяется с помощью функции пригодности, из которой более высокое значение означает большую достоверность. В конечном итоге выбирается наиболее подходящий набор вершин, которые принадлежат одной группе, и создаются все возможные ребра. Поскольку все новые ребра создаются внутри каждой группы, обнаруженные группы будут использоваться для уменьшения искажений графа. Блок-схема предлагаемого алгоритма показана на Рисунке 11.

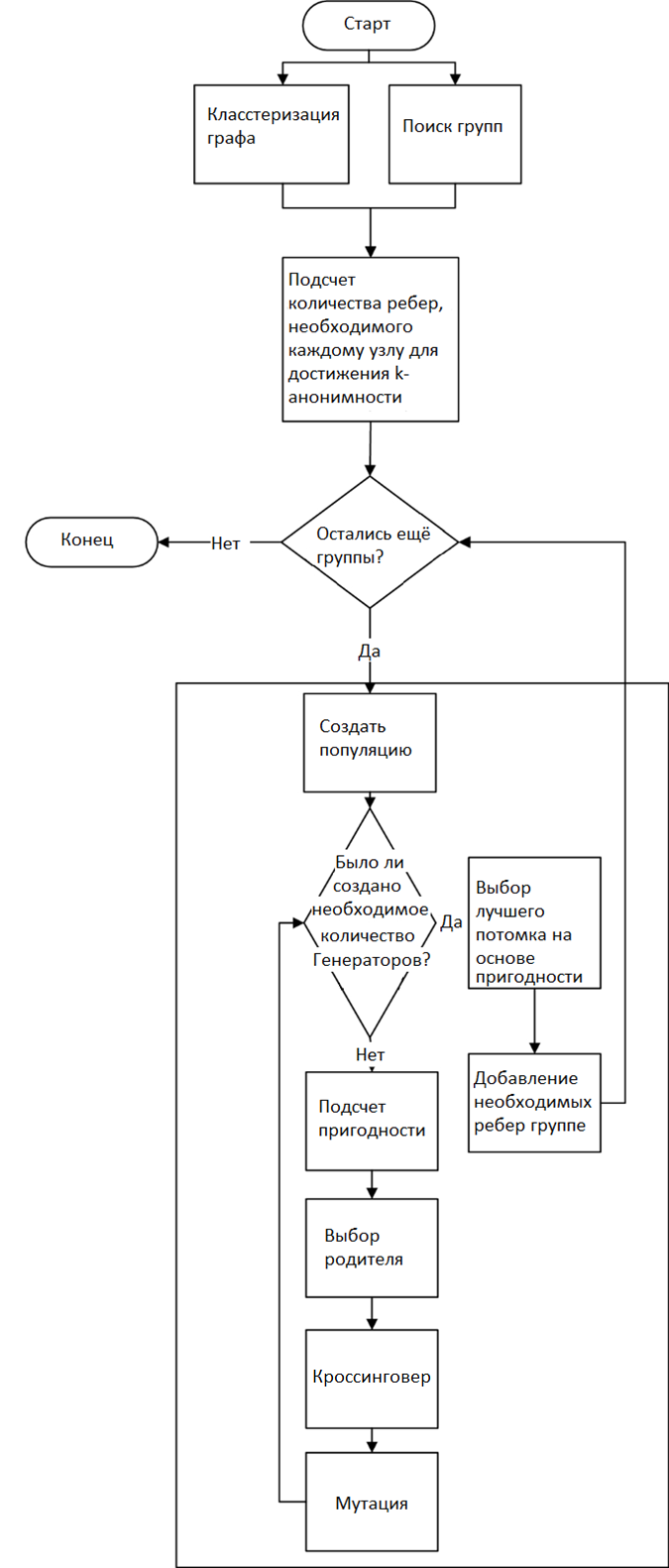


Рисунок 11 — Блок-схема алгоритма

### K-анонимность с помощью генетического алгоритма

K*-*анонимность может быть получена путем обобщения и подавления. Обобщение означает замену атрибута другим соответствующим атрибутом, чтобы сделать результаты менее узнаваемыми. Подавление означает удаление атрибута данных. Получение k-анонимности с использованием обобщения с заявленной целью является NP-трудной задачей. Для решения NP-трудных задач могут использоваться метаэвристические алгоритмы. Генетический алгоритм (ГА), который является одним из метаэвристических алгоритмов, вдохновленных природой, используется для модификации структуры графа. На Рисунке 12 показана блок-схема процесса работы генетического алгоритма.

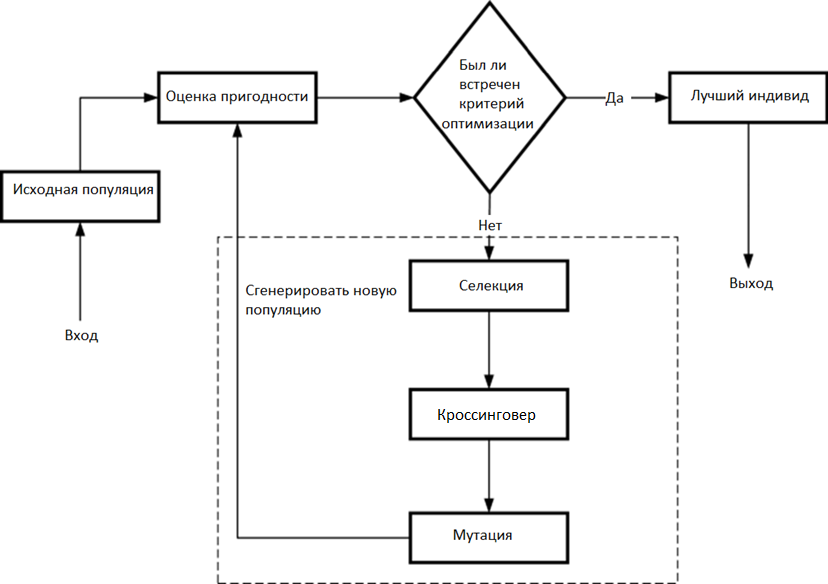


Рисунок 12 — Блок-схема генетического алгоритма

Дж. Холланд разработал генетические алгоритмы для понимания адаптивных процессов природных систем в 1970-х годах. Позже, с 1980-х годов, применение методов ГА было распространено на оптимизацию и машинное обучение, и в настоящее время они популярны в инженерных приложениях.

Рисунок 12 демонстрирует общую концепцию генетического алгоритма. Популяция в ГА представлена хромосомами, которые служат индивидуумами. Начальная популяция обычно создается случайным образом и модифицируется в последующих поколениях. Пригодность каждого индивидуума оценивается для каждой популяции. На этом этапе несколько индивидуумов случайным образом отбираются на основе их пригодности из текущей популяции и модифицируются с помощью скрещивания и мутации для формирования следующего поколения популяции. Новое поколение используется на следующей итерации алгоритма. Популяция развивается в направлении лучших решений на каждой итерации алгоритма. Чтобы остановить итерации, можно использовать либо определенное количество поколений, либо достижение удовлетворительного уровня пригодности. Эффективность и скорость сходимости алгоритма зависят от размера популяции в каждом поколении, количества поколений, операций скрещивания и мутации. Структура генетического алгоритма была выбрана из-за его гибкой формулировки и способности достигать решений с адекватными вычислительными ресурсами. По природе метаэвристических алгоритмов граф должен стать максимально анонимным; следовательно, ГА используется в этой работе для оптимизации модификаций графа.

Популяция ГА состоит из хромосом, представляющих возможные решения. Хромосома в предлагаемом методе состоит из нескольких генов, и каждый ген в хромосоме содержит метку вершины. Для создания хромосомы метки узлов в группе повторяются столько раз, сколько необходимо для достижения k-анонимности. Следовательно, количество хромосомных генов равно сумме требуемых ребер этой группы. Для наглядности приведен пример. Исходными данными является группа в графе с вершинами 1, 2 и 3. Предполагая, что узлу 1 требуется 2 степени, а каждому из узлов 2 и 3 требуется одна степень, чтобы стать анонимным, хромосома будет такой, как показано на Рисунке 13.

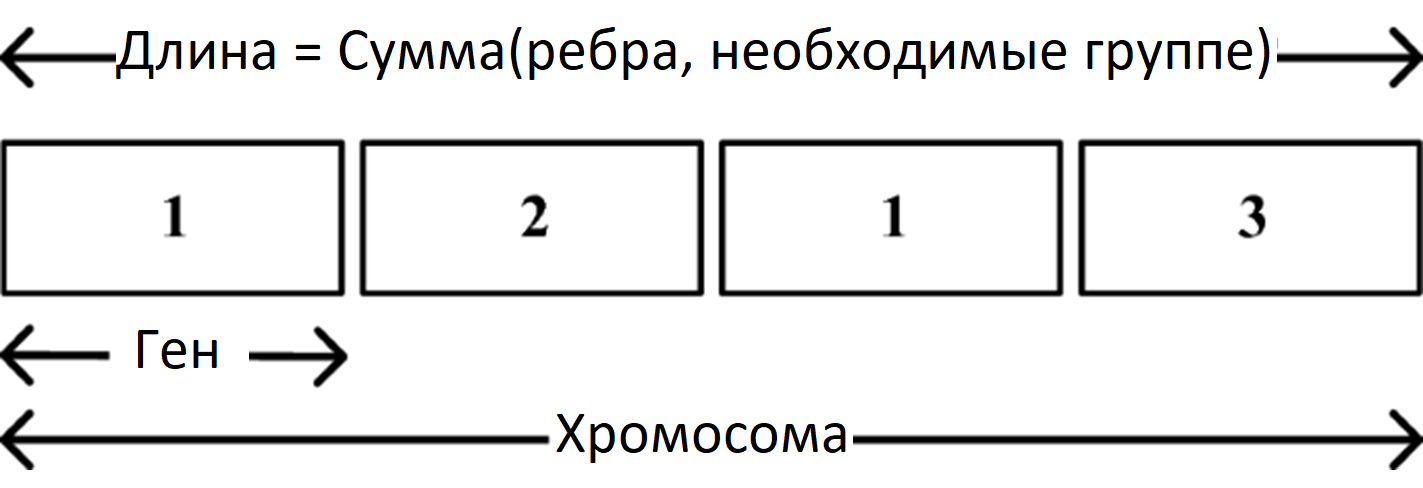


Рисунок 13 — Пример хромосомы

Начальная популяция создается членами сообщества, а условием завершения работы генетического алгоритма в задаче является количество поколений. Начальная популяция, представляющая собой набор хромосом, будет создана для каждого сообщества (группы) графа. Хромосома начальной популяции представляет собой решение проблемы k-анонимности. Хромосома состоит из нескольких генов, и каждый ген в хромосоме содержит метку вершины в группе. Хромосома создается случайным выбором меток узлов. Метки каждого узла повторяются столько раз, сколько необходимо для достижения k-анонимности. Более того, численность популяции определяет количество хромосом в популяции.

Функция пригодности предназначена для прямого или косвенного измерения качества приближения к целевому значению и будет максимальной или минимальной для него. На этом этапе для каждой хромосомы получается одно число, и чем более высокое значение будет получено, тем хромосома лучше. Функция пригодности для задачи максимизации выглядит следующим образом:

1. *A —* Отрицательная оценка дана для предотвращения самозацикливания на графе, поскольку исходный граф не имеет самозацикливаний. Таким образом, любая связь индивида с самим собой бессмысленна. Более того, она отрицательна, потому что любая неправильная хромосома занимает правильную.

2. *B —* Если узел с меткой "*m*" подключен к узлу с меткой "*n*", то очевидно, что "*n*" также подключен к "*m*". Следовательно, для каждого из подключений будет учитываться положительный рейтинг.

3. *C —* Он подсчитывает гены, повторяющиеся в хромосоме больше или меньше, чем это необходимо. За каждый случай дочерний индивид получит одну отрицательную оценку.

4. *D —* Если сгенерированное ребро уже существовало в исходном графе, то учитывается отрицательный рейтинг.

Выбор родителей будет производиться случайным образом из хромосом на основе их пригодности. Вероятность выбора хромосом с более высокой пригодностью больше из-за случайного отбора. Таким образом, даже хромосома с наивысшей пригодностью может быть отброшена в процессе отбора.

Операция скрещивания используется для выбранных хромосом с частотой скрещивания 80%. Это означает, что 20% популяции будут удалены случайным образом, а следующее поколение будет сформировано с помощью скрещивания случайным образом из оставшейся популяции. Данная операция была реализована с использованием метода одноточечного скрещивания. В этом методе объединяются каждые две хромосомы. Половина полученной хромосомы будет принадлежать первому родителю, а вторая половина — другому.

Операция мутации выполняется с вероятностью 10%. Происходит обмен случайно выбранными генами из двух хромосом.

Как только количество желаемых поколений будет сгенерировано, генетические алгоритмы остановятся. Теперь в качестве окончательного ответа следует выбрать лучшего потомка из созданных потомков. На следующем шаге в граф будут добавлены действительные соединения, созданные с помощью генетического алгоритма. Допустимое соединение — это ребро, которого (1) нет в исходном графе, (2) двум соединяющимся узлам требуется одно дополнительное ребро для достижения k-анонимности, и (3) соединение не дублируется [15].

### Экспериментальный стенд

Описанный выше подход был реализован с использованием языка программирования Python (часть кода представлена в Приложении 2). Значение k выбирается в диапазоне {2, 3, 4, 5, ..., 10, 15, 20, 25}. Из-за природы случайного поведения ГА алгоритм выполнялся десять раз для каждого значения k и учитывалось среднее значение результатов. Программа была выполнена в системе с процессором corei3 и 8 ГБ оперативной памяти.

Для более наглядной демонстрации алгоритма был создан граф с 9 вершинами (Рисунок 14), аналогичный примеру на Рисунке 10.

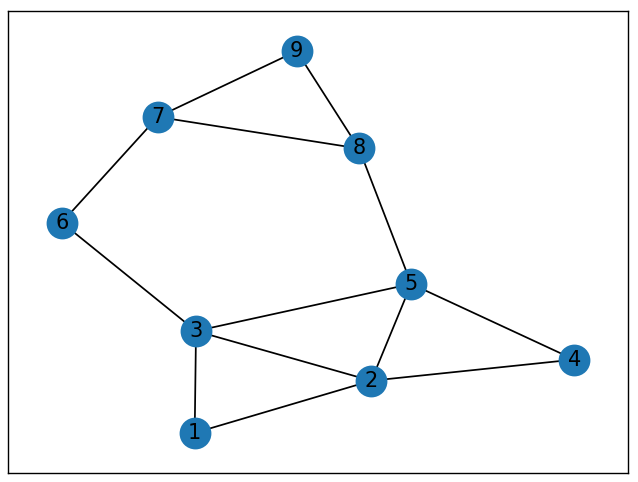


Рисунок 14 — Исходный граф

Далее начинается процесс кластеризации графа и выделения групп. На Рисунке 15 группы обозначены зеленым, оранжевым и красным цветами.

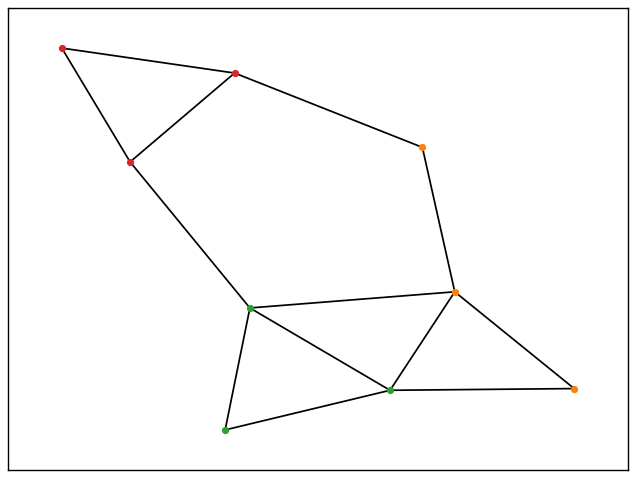


Рисунок 15 — Секционирование и поиск групп

После этого запускается генетический алгоритм. Вывод алгоритма представлен ниже. Результат подобранных ребер на Рисунке 16.

0.0% is running. making ready genetic for: [{1: 1, 2: 0, 3: 0, 4: 1, 5: 0}, 1, 2, 1, 3, 2, 4, 2, 3, 2, 5, 3, 5, 3, 6, 4, 5, 5, 8, 8, 9, 8, 7, 9, 7, 7, 6]

running GA for {1: 1, 2: 0, 3: 0, 4: 1, 5: 0}

0.5% is running. making ready genetic for: [{8: 0, 9: 1, 7: 0, 6: 1}, 1, 2, 1, 3, 2, 4, 2, 3, 2, 5, 3, 5, 3, 6, 4, 5, 5, 8, 8, 9, 8, 7, 9, 7, 7, 6]

running GA for {8: 0, 9: 1, 7: 0, 6: 1}

drawing graph.

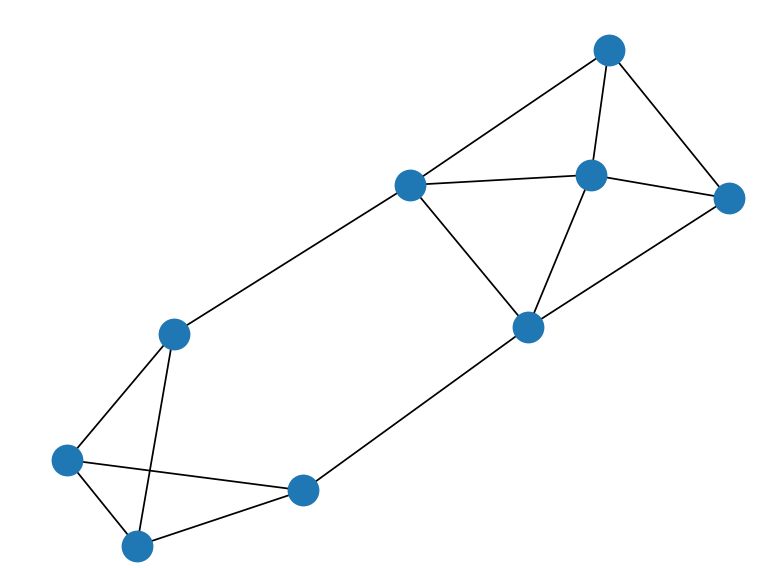


Рисунок 16 — Результат генетического алгоритма

В итоге был получен анонимный граф 3 степени (Рисунок 17).

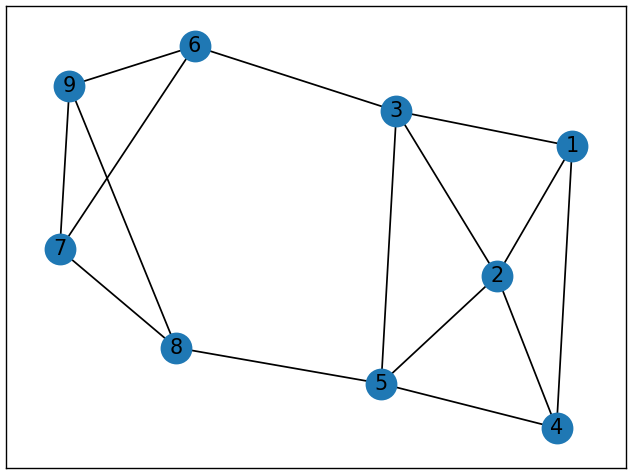


Рисунок 17 — Итоговый анонимизированный граф

Однако исходный граф изначально был дважды анонимным, поэтому при запуске алгоритм срабатывал через раз. Попробуем рассмотреть другой пример, чтоб граф изначально был неанонимным (Рисунок 18). В нём было выделено 2 группы (Рисунок 19).

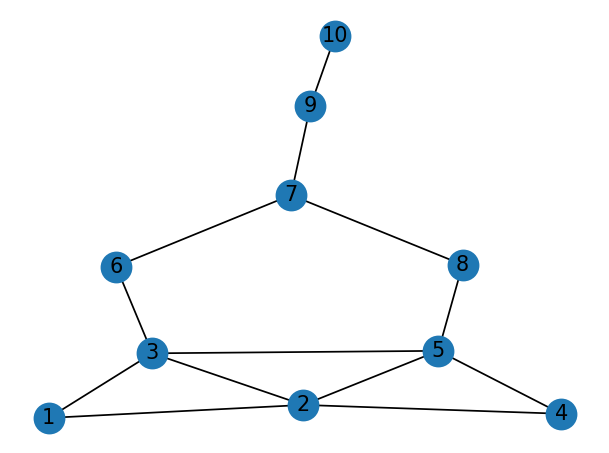


Рисунок 18 — Исходный неанонимный граф

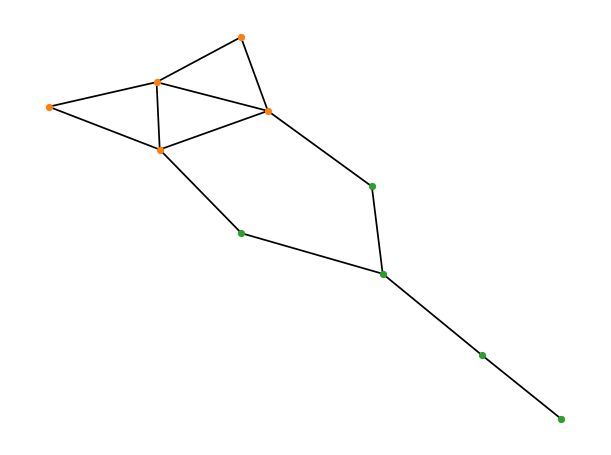


Рисунок 19 — Секционирование и поиск групп

Выберем k = 2, получим результат генетического алгоритма (Рисунок 20) и 2-анонимный граф (Рисунок 21).

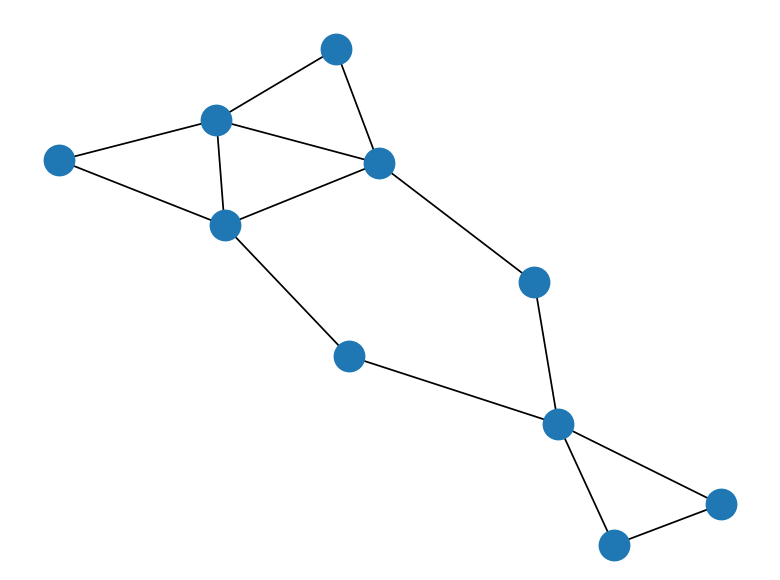


Рисунок 20 — Результат генетического алгоритма для 2-анонимного графа

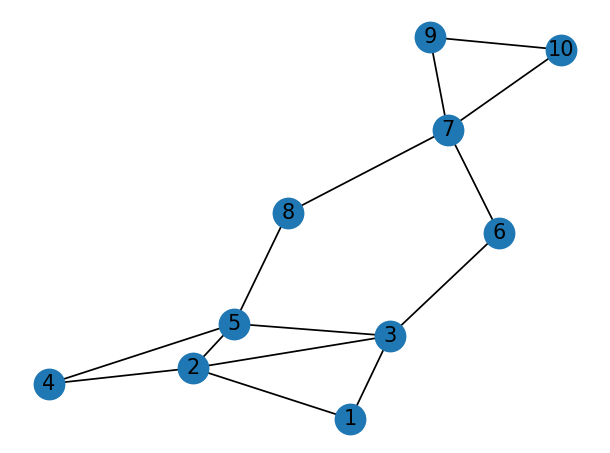


Рисунок 21 — Итоговый 2-анонимный граф

Выберем k = 4, получим результат генетического алгоритма (Рисунок 22) и 2-анонимный граф (Рисунок 23).

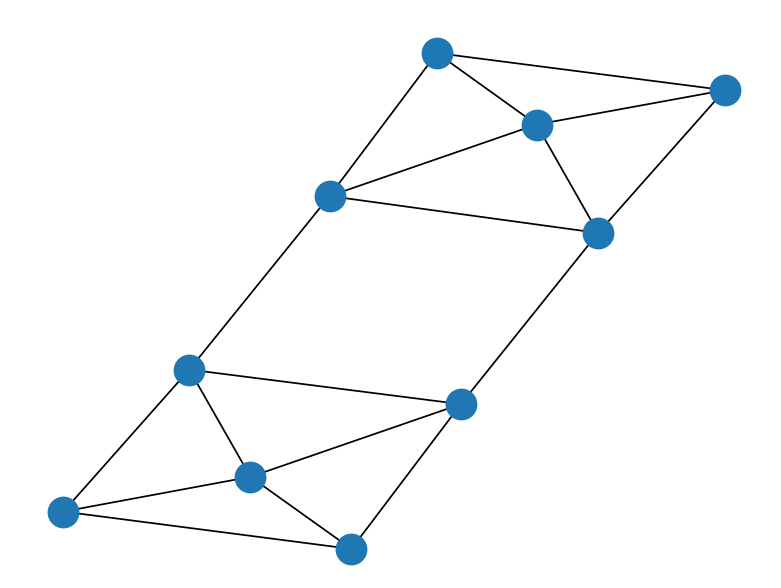


Рисунок 22 — Результат генетического алгоритма для 4-анонимного графа

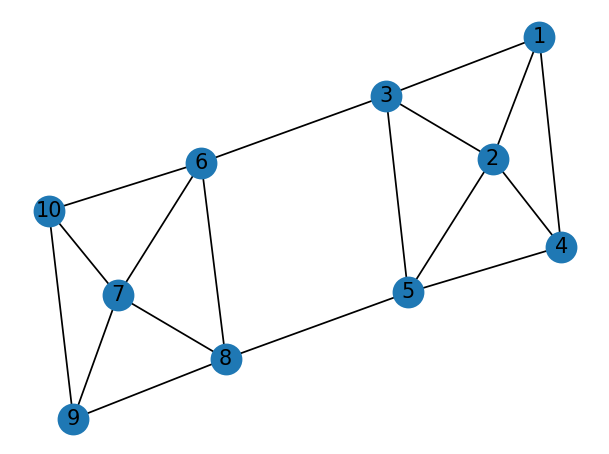


Рисунок 23 — Итоговый 4-анонимный граф

## Выводы по практической части

В практической части были рассмотрены два алгоритма для направленных и ненаправленных графов. Оба алгоритма реализуют подход добавления ребер для достижения k-анонимности. Авторы исследований утверждают, что именно этот подход позволяет максимально сохранить структуру графа, то есть его полезность с точки зрения извлечения информации.

Модификация направленного графа является более сложной задачей, поскольку необходимо учитывать, как входящие, так и исходящие связи, соответственно необходимо проделать в два раза больше работы. Код данного алгоритма была написан на языках R и Java, однако протестировать его не удалось в силу устаревшего синтаксиса.

Модификация ненаправленного графа куда легче, однако и там необходимо учитывать многие факторы, такие как правильный подбор ребер для сохранения максимальной полезности данных. Фреймворк был успешно собран и запущен на целевой машине, алгоритм доказал свою работоспособность и эффективность.

Данные методы нельзя сравнивать, поскольку для различных целей и различных данных могут использоваться как направленные, так и ненаправленные графы. Направленные графы могут использоваться для анализа зависимых связей, то есть когда отношения между людьми не равноценны, например, подписчики и подписки, иерархические отношения (начальник-подчиненный). В случае взаимности таких отношений, стрелка будет в оба конца, но не обязательно. Ненаправленные графы больше подходят для анализа равноценных отношений, например, родственники, друзья, ведь родство и дружба не могут быть односторонними. Таким образом, оба вида графов и оба алгоритма имеют место в представлении, анализе и анонимизации данных.

# ЗАКЛЮЧЕНИЕ

В ходе выполнения данной курсовой работы были изучены современные методы анонимизации графов. Были изучены графовые данные как таковые, преимущества их использования для анализа социальных связей. Был рассмотрен поэтапно процесс публикации данных. Данный процесс включает в себя сбор данных, предобработку, анонимизацию, публикацию и анализ. Анализ данных необходим для достижения научных и бизнес целей, например, для медицинских исследований, анализа причин финансовых кризисов, рекомендательных систем, оценки политической обстановки и многого другого. Однако такой анализ не может проводиться на сырых данных, поскольку в них может содержаться конфиденциальная информация о пользователях. Именно в этот момент и необходима анонимизация. Было рассмотрено пять основных подходов к анонимизации данных, каждый из которых включает в себя множество алгоритмов. Выбор метода анонимизации зависит от многих факторов: целей исследования, предметной области, природы данных и ограничений конфиденциальности. Также важно помнить о конфликте конфиденциальности и полезности и искать компромисс в данном вопросе. Среди изученных методов были выделены методы модификации графов, методы кластеризации, методы дифференциальной анонимности, методы вычисления графов с учетом конфиденциальности и гибридные методы. Каждый метод имеет свои достоинства и недостатки. Однако несмотря на великое разнообразие подходов к решению проблемы все ещё существует угроза деанонимизации данных. Эта угроза и методы ее реализации также были изучены и описаны в аналитической части данной работы. Помимо угрозы деанонимизации также есть общие проблемы и трудности в процессе анонимизации, из которых вытекают перспективные области для исследования.

В практической части были рассмотрены два алгоритма модификации путем добавления псевдо-ребер для направленных и ненаправленных графов. Оба варианта хранения данных и соответствующие им методы анонимизации могут быть применены для заявленных целей в зависимости от конкретной задачи и структуры анализируемых данных. Эффективность алгоритмов имеет под собой доказанную математическую базу. При этом ни один алгоритм не существует изолированно, они все опираются на некоторые другие алгоритмы, что говорит о сложности решения задачи с нуля. Метод для ненаправленных графов, основанный на генетическом алгоритме, был успешно протестирован. Результаты алгоритма удовлетворительные: был получен анонимный граф 3 степени.

Подводя итог всему вышесказанному, автор делает вывод о том, что тема анонимизации графов сейчас активно развивается в научном сообществе. Реализовано уже большое количество методов, однако стремительное развитие ИКТ бросает новые вызовы исследователям данных. Поэтому некоторые проблемы в этой сфере все ещё стоят достаточно остро.

# СПИСОК ИСТОЧНИКОВ

1. A. Majeed and S. Lee, "Anonymization Techniques for Privacy Preserving Data Publishing: A Comprehensive Survey," in IEEE Access, vol. 9, pp. 8512-8545, 2021, doi: 10.1109/ACCESS.2020.3045700.
2. C. Watanabe, T. Amagasa and L. Liu, "Privacy risks and countermeasures in publishing and mining social network data", Proc. 7th Int. Conf. Collaborative Comput. Netw. Appl. Worksharing, pp. 55-66, 2011.
3. B. Zhou, J. Pei and W. Luk, "A brief survey on anonymization techniques for privacy preserving publishing of social network data", ACM SIGKDD Explor. Newslett., vol. 10, no. 2, pp. 12-22, Dec. 2008.
4. J. Vadisala and V. K. Vatsavayi, "Challenges in social network data privacy", Int. J. Comput. Intell. Res., vol. 13, no. 5, pp. 965-979, 2017
5. J. Casas-Roma, J. Herrera-Joancomartí and V. Torra, "A survey of graph-modification techniques for privacy-preserving on networks", Artif. Intell. Rev., vol. 47, no. 3, pp. 341-366, Mar. 2017.
6. T. Tassa and D. J. Cohen, "Anonymization of centralized and distributed social networks by sequential clustering", IEEE Trans. Knowl. Data Eng., vol. 25, no. 2, pp. 311-324, Feb. 2013.
7. D. Al-Azizy, D. Millard, I. Symeonidis, K. O’Hara and N. Shadbolt, "A literature survey and classifications on data deanonymisation", Proc. Int. Conf. Risks Secur. Internet Syst., pp. 36-51, 2015.
8. Casas-Roma, J., Salas, J., Malliaros, F.D. et al. k-Degree anonymity on directed networks. Knowl Inf Syst 61, 1743–1768 (2019). <https://doi.org/10.1007/s10115-018-1251-5>
9. Hansen SL, Mukherjee S (2003) A polynomial algorithm for optimal univariate microaggregation. IEEE Trans Knowl Data Eng 15(4):1043–1044
10. Salas J (2016) Faster univariate microaggregation for power law distributions: k-degree-anonymity for big graphs In: ICDM workshop on privacy and discrimination in data mining. IEEE, Barcelona, Spain
11. Bredereck R, Froese V, Koseler M, Millani M.G, Nichterlein A, Niedermeier R (2018) A parameterized algorithmics framework for digraph degree sequence completion problems. [arXiv:1604.06302v3](http://arxiv.org/abs/1604.06302v3)
12. Domingo-Ferrer J, Torra V (2005) Ordinal, continuous and heterogeneous k-anonymity through microaggregation. Data Min Knowl Discov 11(2):195–212
13. Domingo-Ferrer J, Mateo-Sanz JM (2002) Practical data-oriented microaggregation for statistical disclosure control. IEEE Trans Knowl Data Eng 14(1):189–201
14. Casas-Roma J (2015) An evaluation of edge modification techniques for privacy-preserving on graphs. In: MDAI ’15. Springer, Skövde, Sweden, pp 180–191
15. Rajabzadeh, S., Shahsafi, P. & Khoramnejadi, M. A graph modification approach for k-anonymity in social networks using the genetic algorithm. Soc. Netw. Anal. Min. 10, 38 (2020). <https://doi.org/10.1007/s13278-020-00655-6>
16. Sihag VK (2012) A clustering approach for structural K-anonymity in social networks using genetic algorithm. In: Proceedings of the CUBE international information technology conference, pp 701–706
17. Ma T, Zhang Y, Cao J, Shen J, Tang M, Tian Y, Al-Dhelaan A, Al-Rodhaan M (2015) k-Degree anonymity with vertex and edge modification algorithm. Computing 97(12):1165–1184
18. Blondel VD, Guillaume J-L, Lambiotte R, Lefebvre E (2008) Fast unfolding of communities in large networks. J Stat Mech: Theory Exp 2008(10):P10008

Приложение 1

Алгоритм 1 — Процесс модификации ребер

Function graph\_modification\_process

Input: δin, δout, V and A

Output: Anonymized arc set (A ).

for vi : δout(vi) > 0 do

if ! edge\_addition (vi) then

for v j : δin(v j) > 0 do

if (vi, v j) ∈ A then

edge\_switch (vi, v j)

else

edge\_extension (vi)

end

end

end

end

return A

end

Function edge\_addition(vi)

for v j : δin(v j) > 0 do

if (vi, v j) ∉ A then

create (vi, v j)

return true

end

end

return false

end

Function edge\_switch(vi, v j)

find vk , vp : (vk , vp) ∈ A and (vk , v j), (vi, vp) ∉ A

delete (vk , vp)

create (vi, vp) and (vk , v j)

return true

end

Function edge\_extension(vi)

find vk , vp : (vk , vp) ∈ A and (vk , vi), (vi, vp) ∉ A

delete (vk , vp)

create (vk , vi) and (vi, vp)

return true

end

Приложение 2

Файл main.py — основной код алгоритма k-анонимности

from matplotlib import pyplot as plt  
import pandas as pds  
import numpy as np  
import networkx as nx  
import collections  
import copy  
  
from community\_detection import comunity\_detection\_function, all\_partitions\_all\_levels, find\_community  
import genetic as GA  
from partition import greedy\_partition\_algorithm  
from draw\_graph import draw  
  
def flatten(l):  
 for el in l:  
 if isinstance(el, collections.abc.Iterable) and not isinstance(el, (str, bytes)):  
 yield from flatten(el)  
 else:  
 yield el  
  
if \_\_name\_\_=="\_\_main\_\_":  
 '''  
 initial config  
 '''  
 population\_number=10  
 generation\_number=10  
 k=3  
 show=True  
  
 '''  
 read\_csv  
 '''  
 #read file  
 # datafile = pds.read\_csv("datasets/CA-AstroPh.csv", header=None)  
 # datafile=datafile.values  
 # print(datafile)  
  
 datafile=np.array([(1,2),(1,3),(2,4),(2,3),(4,5),(5,3),(5,8),(8,9),(9,7),(7,6),(6,3),(2,5),(8,7)])  
 #datafile=np.array([(1,2),(1,3),(1,4),(1,5),(1,6),(2,3),(3,4),(2,7)])  
  
 print("reading file finished!")  
  
 '''  
 make graph and plot  
 '''  
 print("drawing graph.")  
 G=nx.Graph()  
 G.add\_edges\_from(datafile)  
 # nx.draw(G)  
 # plt.show()  
 if show:  
 draw(G)  
 print("draw finished.")  
  
 '''  
 run partitining  
 '''  
 print("run partitining")  
 #make degree\_sequence\_list  
 degree\_sequence\_list=np.array([[label, degree] for label, degree in G.degree()])  
 #ordered list, descending  
 degree\_sequence\_list=(degree\_sequence\_list[np.argsort(degree\_sequence\_list[::,-1])][::-1]).tolist()  
 #print("degree\_sequence\_list:", degree\_sequence\_list)  
  
 #\_partitions of need matrix  
 \_partitions=greedy\_partition\_algorithm(degree\_sequence\_list, k, True)  
 #print("\_partitions:", \_partitions)  
  
 #make format of {lable:need}  
 \_=np.array([i for i in flatten(\_partitions)])  
 \_partitions=\_.reshape(int(len(\_)/2),2)  
 need=dict(\_partitions)  
 # print("need:", need)  
  
 '''  
 run comunities detection  
 '''  
 print("run community detection")  
 dendrogram = comunity\_detection\_function(datafile, G, show=show)  
 #dictionary of nodes and comunities of that nodes  
 all\_partitions\_all\_levels\_list=all\_partitions\_all\_levels(dendrogram)  
 #com format: {0: [1, 2, 3, 4, 5], 1: [6, 7, 8, 9]}  
 com=find\_community(all\_partitions\_all\_levels\_list[-1])  
 # print("com:",com)  
  
 '''  
 com & need  
 com={0: [0, 1, 2], 1:[3,4,5]}  
 need={0:1,1:2,2:0,3:4,4:1,5:2}  
  
 com\_need= [[[0, 1], [1, 2], [2, 0]], [[3, 4], [4, 1], [5, 2]]]  
 '''  
 com\_need=[]  
 for item in com.values():  
 com\_need.append([])  
 for i in item:  
 com\_need[-1].append([i,need[i]])  
 # print("com\_need:",com\_need)  
  
 '''  
 genetic init config  
 '''  
 #final edges  
 modified\_edges=[]  
  
 #make template  
 template=[]  
 for item in com\_need:  
 template.append([])  
 template[-1].append(dict(item))  
 template[-1].extend(np.array(G.edges()).flatten())  
  
 \_len\_template=len(template)  
 #run genetic  
 for index,item in enumerate(template):  
 print(str(index/\_len\_template)+"% is running. making ready genetic for:", item)  
 need=item[0]  
 edges=item[1:]  
  
 # #check if needs add node  
 # not\_fit=GA.add\_node\_before\_GA(need)  
 #  
 # #add a node by need = add\_node\_before\_GA  
 # if not\_fit:  
 # need.update({str(index)+"fit":not\_fit})  
 # print("need", need)  
  
 '''  
 run genetic  
 '''  
 if np.sum(np.array(list(need.values()))) != 0:  
 print("running GA for ", need)  
 #print("making init pop")  
 header, init\_pop=GA.make\_initial\_population(need, population\_number)  
 # print("header", header)  
 # print("init\_pop", init\_pop)  
 parents=copy.deepcopy(init\_pop)  
 for i in range(generation\_number):  
 header, parents= GA.selection(header, parents, population\_number, G)  
 # print("selection header", header)  
 # print("selection parents", parents)  
 new\_generation=GA.cross\_over(parents)  
 # print("cross over", new\_generation)  
 new\_generation=GA.mutation(new\_generation, population\_number)  
 # print("mutation", new\_generation)  
 parents=copy.deepcopy(new\_generation)  
 # print(str(i/generation\_number)+"% GA done; of "+str(index/\_len\_template)+"%.")  
 # print("\n")  
  
 #calculate fintness and select best answer  
 fitness\_score=[]  
 for pop in new\_generation:  
 fitness\_score.append(GA.fitness(pop, header, G))  
 # print("fitness\_score", fitness\_score)  
  
 genetic\_output=np.unique(np.sort(np.array([[i,j] for i,j in zip(header, new\_generation[ fitness\_score.index(max(fitness\_score))])]), axis=1), axis=0)  
 #update needs  
 for node1,node2 in genetic\_output:  
 try:  
 node1=int(node1)  
 except:  
 pass  
 try:  
 node2=int(node2)  
 except:  
 pass  
 if need[node1] > 0 and need[node2] > 0 and (not G.has\_edge(node1, node2)) and node2 != node1:  
 need[node1]-=1  
 need[node2]-=1  
 edges.extend([node1,node2])  
  
 #if need to add nodes  
 #connections=GA.add\_node\_after\_GA(need, edges, index)  
 connections=np.array(edges, dtype=str).reshape(int(len(edges)/2),2)  
 # print("connections",connections)  
 modified\_edges.extend(np.array(connections))  
 modified\_edges=np.unique(np.sort(modified\_edges, axis=1), axis=0).tolist()  
  
  
 '''  
 draw output  
 '''  
 print("drawing graph.")  
 G=nx.Graph()  
 G.add\_edges\_from(modified\_edges)  
 nx.draw(G)  
 plt.show()  
 if show:  
 draw(G)  
 print("draw finished.")  
  
print("writing to file...")  
with open("final.txt", "w") as final:  
 for i,j in G.edges():  
 final.write(str(i)+" "+str(j)+"\n")